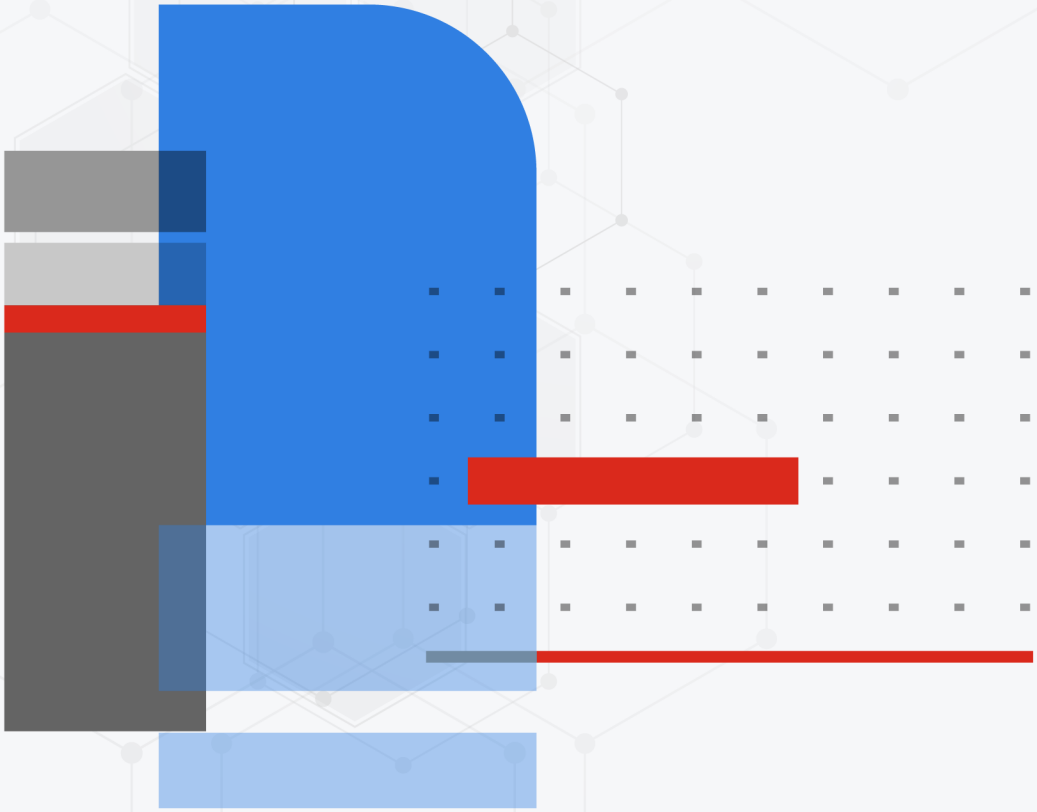
CLI Reference

FortiADC 7.4.2



FORTINET DOCUMENT LIBRARY
https://docs.fortinet.com

FORTINET VIDEO LIBRARY
https://video.fortinet.com

FORTINET BLOG
https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT
https://support.fortinet.com

FORTINET TRAINING & CERTIFICATION PROGRAM
https://www.fortinet.com/training-certification

FORTINET TRAINING INSTITUTE
https://training.fortinet.com

FORTIGUARD LABS
https://www.fortiguard.com

END USER LICENSE AGREEMENT
https://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK
Email: techdoc@fortinet.com

February 7, 2024
FortiADC 7.4.2 CLI Reference
01-600-650988-20200916

TABLE OF CONTENTS

Change Log

Date	Change Description
2020-10-12	FortiADC 6.0.1 CLI Reference initial release. Sensitive language changed (master->primary, slave->secondary, black->block, white->allow).

Introduction

Welcome, and thank you for selecting Fortinet products for your network protection.

Scope

This document describes how to use the command-line interface (CLI) of the FortiADC appliance. It assumes that you have already successfully installed the FortiADC appliance and completed basic setup.

At this stage:

- You have administrative access to the web UI and/or CLI.
- The FortiADC appliance is integrated into your network.

Once that basic installation is complete, you can use this document. This document is a reference for commands you can use to:

- Update the system.
- Configure features and advanced options.
- Diagnose problems.

This document does *not* cover the web UI or first-time setup. For that information, see the *FortiADC Handbook*.

Conventions

This document uses the conventions described in this section.

IP addresses

To avoid IP conflicts that would occur if you used examples in this document with public IP addresses that belong to a real organization, the IP addresses used in this document are fictional. They belong to the private IP address ranges defined by these RFCs.

- RFC 1918: Address Allocation for Private Internets
- RFC 5737: IPv4 Address Blocks Reserved for Documentation
- RFC 3849: IPv6 Address Prefix Reserved for Documentation

For example, even though a real network's Internet-facing IP address would be routable on the public Internet, in this document's examples, the IP address would be shown as a non-Internet-routable IP such as 10.0.0.1, 192.168.0.1, or 172.16.0.1.

Cautions, notes, & tips

This document uses the following guidance and styles for notes, tips and cautions.

Warns you about procedures or feature behaviors that could have unexpected or undesirable results including loss of data or damage to equipment.

Highlights important, possibly unexpected but non-destructive, details about a feature's behavior.

Presents best practices, troubleshooting, performance tips, or alternative methods.

Typographical conventions

Table 1 describes the typographical conventions used in this document.

Typographical conventions

Convention	Example
A GUI element you are instructed to click or select	From Minimum log level, select **Notification**.

CLI input	`config system dns`
	`set primary <address_ipv4>`
	`end`

CLI output	`FortiADC-VM # execute certificate local regenerate self certificate regenerated!`
Emphasis	HTTP connections are *not* secure and can be intercepted by a third party.
File content	`<HTML><HEAD><TITLE>Authentication</TITLE></HEAD><BODY><H4>You must authenticate to use this service.</H4>`
Hyperlink	https://support.fortinet.com
Keyboard entry	Type a name for the configuration such as `virtual_server_1`.
Navigation	Go to System > Maintenance.
Publication	For details, see the *FortiADC Handbook*.

Command syntax

The CLI requires that you use valid syntax, and conform to expected input constraints. It rejects invalid commands.

For command syntax conventions such as braces, brackets, and command constraints such as `<address_ipv4>`, see Notation.

Using the CLI

The command-line interface (CLI) is an alternative to the web UI.

You can use either interface or both to configure the FortiADC appliance. In the web UI, you use buttons, icons, and forms, while, in the CLI, you either type text commands or upload batches of commands from a text file, like a configuration script.

If you are new to Fortinet products, or if you are new to the CLI, this section can help you to become familiar.

Connecting to the CLI

You can access the CLI in two ways:

- Locally — Connect your computer, terminal server, or console directly to the console port.
- Through the network — Connect your computer through any network attached to one of the network ports. To connect using an Secure Shell (SSH) or Telnet client, enable the network interface for Telnet or SSH administrative access. Enable HTTP/HTTPS administrative access to connect using the CLI Console widget in the web UI.

Local access is required in some cases:

- If you are installing your FortiADC appliance for the first time and it is not yet configured to connect to your network, unless you reconfigure your computer's network settings for a peer connection, you might only be able to connect to the CLI using a local console connection. See the *FortiADC Handbook*.
- Restoring the firmware utilizes a boot interrupt. Network access to the CLI is not available until *after* the boot process completes, and therefore local CLI access is the only viable option.

Before you can access the CLI through the network, you usually must enable SSH and/or HTTP/HTTPS and/or Telnet on the network interface through which you will access the CLI.

Connecting to the CLI using a local console

Local console connections to the CLI are formed by directly connecting your management computer or console to the FortiADC appliance, using its DB-9 console port.

Requirements

- A computer with an available serial communications (COM) port
- Console cable (RJ-45-to-DB-9 or null modem cable) included in your FortiADC package
- Terminal emulation software such as PuTTY

The following procedure describes connection using PuTTY software; steps may vary with other terminal emulators.

To connect to the CLI using a local console connection

1. Using the null modem or RJ-45-to-DB-9 cable, connect the FortiADC appliance's console port to the serial communications (COM) port on your management computer.
2. On your management computer, start PuTTY.
3. In the Category tree on the left, go to Connection > Serial and configure the following:

Serial port	COM1 (or, if your computer has multiple serial ports, the name of the connected serial port)
Speed (baud)	9600
Data bits	8
Stop bits	1
Parity	None
Flow control	None

4. In the Category tree on the left, go to Session (not the sub-node, Logging) and from Connection type, select **Serial**.
5. Click **Open**.
6. Press the Enter key to initiate a connection.
   The login prompt appears. When the system first boots up, the admin account is forced to set up a new password.
7. Type a valid administrator account name then press Enter.
8. Type the password for that administrator account and press Enter.

The CLI displays the following text, followed by a command line prompt:

```
Welcome!
```

You can now enter CLI commands, including configuring access to the CLI through SSH or Telnet.

Enabling access to the CLI through the network

SSH, Telnet, or CLI Console widget (via the web UI) access to the CLI requires connecting your computer to the FortiADC appliance using one of its RJ-45 network ports. You can either connect directly, using a peer connection between the two, or through any intermediary network.

If you do not want to use an SSH/Telnet client and you have access to the web UI, you can alternatively access the CLI through the network using the CLI Console widget in the web UI.

You must enable SSH and/or Telnet on the network interface associated with that physical network port. If your computer is not connected directly or through a switch, you must also configure the FortiADC appliance with a static route to a router that can forward packets from the FortiADC appliance to your computer.

You can do this using either:

- a local console connection (see the following procedure)
- the web UI

Requirements

- a computer with an available serial communications (COM) port and RJ-45 port
- terminal emulation software such as PuTTY
- the RJ-45-to-DB-9 or null modem cable included in your FortiADC package
- a crossover Ethernet cable (if connecting directly) or straight-through Ethernet cable (if connecting through a switch or router)

To enable SSH or Telnet access to the CLI using a local console connection

1. Using the network cable, connect the FortiADC appliance's network port either directly to your computer's network port, or to a network through which your computer can reach the FortiADC appliance.
2. Note the number of the physical network port.
3. Using a local console connection, connect and log into the CLI.
4. Enter the following commands:
```
config system interface
   edit <interface_name>
      set allowaccess {http https ping snmp ssh telnet}
end
```
where:

`<interface_name>` is the name of the network interface associated with the physical network port, such as `port1`

{http https ping snmp ssh telnet} is the complete, space-delimited list of permitted administrative access protocols, such as `https ssh telnet`; omit protocols that you do not want to permit

For example, to exclude HTTP, SNMP, and Telnet, and allow only HTTPS, ICMP ECHO (ping), and SSH administrative access on `port1`:
```
config system interface
   edit "port1"
      set allowaccess ping https ssh
   next
end
```

Telnet is not a secure access method. SSH should be used to access the CLI from the Internet or any other untrusted network.

5. To confirm the configuration, enter the command to view the access settings for the interface.
```
show system interface <interface_name>
```
The CLI displays the settings, including the management access settings, for the interface.
6. If you will be connecting indirectly, through one or more routers or firewalls, configure the appliance with at least one static route so that replies from the CLI can reach your client.

Connecting to the CLI using SSH

Once you configure the FortiADC appliance to accept SSH connections, you can use an SSH client on your management computer to connect to the CLI.

Secure Shell (SSH) provides both secure authentication and secure communications to the CLI. Supported SSH protocol versions, ciphers, and bit strengths vary by whether or not you have enabled FIPS-CC mode or are using a low encryption (LENC) version, but generally include SSH version 2 with AES-128, 3DES, Blowfish, and SHA-1.

Requirements

- a computer with an RJ-45 Ethernet port
- a crossover Ethernet cable
- an SSH client such as PuTTY

To connect to the CLI using SSH

1. On your management computer, start PuTTY.
   Initially, the Session category of settings is displayed.
2. In Host Name (or IP Address), type the IP address of a network interface on which you have enabled SSH administrative access.
3. In Port, type `22`.
4. From Connection type, select **SSH**.
5. Click **Open**.
   The SSH client connects to the FortiADC appliance.
   The SSH client may display a warning if this is the first time you are connecting to the FortiADC appliance and its SSH key is not yet recognized by your SSH client, or if you have previously connected to the FortiADC appliance but it used a different IP address or SSH key. If your management computer is directly connected to the FortiADC appliance with no network hosts between them, this is normal.
6. Click **Yes** to verify the fingerprint and accept the FortiADC appliance's SSH key. You will not be able to log in until you have accepted the key.
   The CLI displays a login prompt.
7. Type a valid administrator account name (such as `admin`) and press Enter.
8. Type the password for this administrator account and press Enter.

If three incorrect login or password attempts occur in a row, you will be disconnected. Wait one minute, then reconnect to attempt the login again.

The FortiADC appliance displays a command prompt (its hostname followed by a `#`). You can now enter CLI commands.

Connecting to the CLI using Telnet

Once the FortiADC appliance is configured to accept Telnet connections, you can use a Telnet client on your management computer to connect to the CLI.

Telnet is not a secure access method. SSH should be used to access the CLI from the Internet or any other untrusted network.

Requirements

- a computer with an RJ-45 Ethernet port
- a crossover Ethernet cable

- a FortiADC network interface configured to accept Telnet connections
- terminal emulation software such as PuTTY

To connect to the CLI using Telnet

1. On your management computer, start PuTTY.
2. In Host Name (or IP Address), type the IP address of a network interface on which you have enabled Telnet administrative access.
3. In Port, type `23`.
4. From Connection type, select **Telnet**.
5. Click **Open**.
6. Type a valid administrator account name (such as `admin`) and press Enter.
7. Type the password for this administrator account and press Enter.

If three incorrect login or password attempts occur in a row, you will be disconnected. Wait one minute, then reconnect to attempt the login again.

The CLI displays a command line prompt (by default, its host name followed by a `#`). You can now enter CLI commands.

Command syntax

When entering a command, the CLI requires that you use valid syntax and conform to expected input constraints. It will reject invalid commands.

For example, if you do not type the entire object that will receive the action of a command operator such as `config`, the CLI will return an error message such as:

```
Command fail. CLI parsing error
```

Fortinet documentation uses the following conventions to describe valid command syntax.

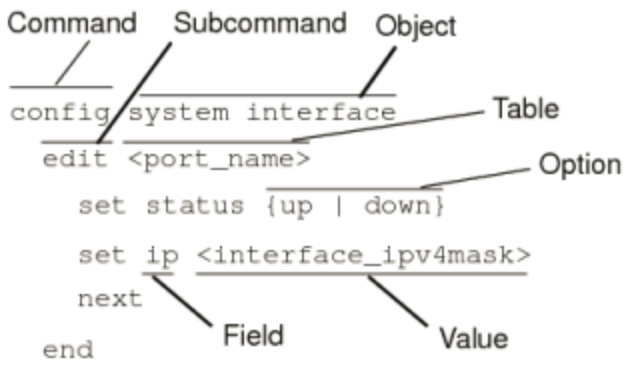
Terminology

Each command line consists of a command word followed by words for the configuration data or other specific item that the command uses or affects, for example:

```
get system admin
```

Fortinet documentation uses the terms in Figure 1 to describe the function of each word in the command line.

Command syntax terminology



The syntax uses the following terms:

- **command** — A word that begins the command line and indicates an action that the FortiADC appliance should perform on a part of the configuration or host on the network, such as `config` or `execute`. Together with other words, such as fields or values, that you terminate by pressing the Enter key, it forms a command line. Exceptions include multi-line command lines, which can be entered using an escape sequence.
  Valid command lines must be unambiguous if abbreviated. Optional words or other command line permutations are indicated by syntax notation.
  If you do not enter a known command, the CLI will return an error message such as:
  ```
  Unknown action 0
  ```
- **subcommand** — A kind of command that is available only when nested within the scope of another command. After entering a command, its applicable subcommands are available to you until you exit the scope of the command, or until you descend an additional level into another subcommand. Indentation is used to indicate levels of nested commands.
  Not all top-level commands have subcommands. Available subcommands vary by their containing scope.
- **object** — A part of the configuration that contains tables and/or fields. Valid command lines must be specific enough to indicate an individual object.
- **table** — A set of fields that is one of possibly multiple similar sets that each have a name or number, such as an administrator account, policy, or network interface. These named or numbered sets are sometimes referenced by other parts of the configuration that use them.
- **field** — The name of a setting, such as `ip` or `hostname`. Fields in some tables must be configured with values. Failure to configure a required field will result in an invalid object configuration error message, and the FortiADC appliance will discard the invalid table.
- **value** — A number, letter, IP address, or other type of input that is usually the configuration setting held by a field. Some commands, however, require multiple input values which may not be named but are simply entered in sequential order in the same command line. Valid input types are indicated by constraint notation.
- **option** — A kind of value that must be one or more words from a fixed set of options.

Indentation

Indentation indicates levels of nested commands, which indicate what other subcommands are available from within the scope.

For example, the `edit` subcommand is available only within a command that affects tables, and the `next` subcommand is available only from within the `edit` subcommand:

```
config system interface
```

```
    edit port1
       set status up
    next
end
```

For information about available subcommands, see Subcommands.

Notation

Brackets, braces, and pipes are used to denote valid permutations of the syntax. Constraint notations, such as `<address_ipv4>`, indicate which data types or string patterns are acceptable value input.

If you do not use the expected data type, the CLI returns an error message such as:

```
object set operator error, -4003 discard the setting
The request URL must start with "/" and without domain name.
```

or:

```
invalid unsigned integer value :-:

value parse error before '-'
Input value is invalid.
```

It might reject or discard your settings instead of saving them when you type `end`.

Command syntax notation

Convention	Description

Square brackets `[ ]`	A non-required (optional) word or words. For example:
	`[verbose {1 | 2 | 3}]`
	indicates that you may either omit or type both the `verbose` word and its accompanying option, such as:
	`verbose 3`

Curly braces `{ }`	A word or series of words that is constrained to a set of options delimited by either vertical bars or spaces. You must enter at least one of the options, unless the set of options is surrounded by square brackets [ ].
Options delimited by vertical bars `|`	Mutually exclusive options. For example: `{enable | disable}` indicates that you must enter either `enable` or `disable`, but must not enter both.

Options delimited by spaces	Non-mutually exclusive options. For example:
	`{http https ping snmp ssh telnet}`
	indicates that you may enter all or a subset of those options, in any order, in a space-delimited list, such as:
	`ping https ssh`

Convention	Description
	Note: To change the options, you must re-type the entire list. For example, to add `snmp` to the previous example, you would type: `ping https snmp ssh` If the option adds to or subtracts from the existing list of options, instead of replacing it, or if the list is comma-delimited, the exception will be noted.
Angle brackets < >	A word constrained by data type. To define acceptable input, the angled brackets contain a descriptive name followed by an underscore (_) and suffix that indicates the valid data type. For example: `<retries_int>` indicates that you should enter a number of retries, such as `5`. Data types include: • `<xxx_name>` — A name referring to another part of the configuration, such as `policy_A`. • `<xxx_index>` — An index number referring to another part of the configuration, such as `0` for the first static route. • `<xxx_pattern>` — A regular expression or word with wild cards that matches possible variations, such as `*@example.com` to match all e-mail addresses ending in `@example.com`. • `<xxx_fqdn>` — A fully qualified domain name (FQDN), such as `mail.example.com`. • `<xxx_email>` — An email address, such as `admin@mail.example.com`. • `<xxx_url>` — A uniform resource locator (URL) and its associated protocol and host name prefix, which together form a uniform resource identifier (URI), such as `http://www.fortinet.com/`. • `<xxx_ipv4>` — An IPv4 address, such as `192.168.1.99`. • `<xxx_v4mask>` — A dotted decimal IPv4 netmask, such as `255.255.255.0`. • `<xxx_ipv4mask>` — A dotted decimal IPv4 address and netmask separated by a space, such as `192.168.1.99 255.255.255.0`. • `<xxx_ipv4/mask>` — A dotted decimal IPv4 address and CIDR-notation netmask separated by a slash, such as such as `192.168.1.99/24`. • `<xxx_ipv6>` — A colon(`:`)-delimited hexadecimal IPv6 address, such as `3f2e:6a8b:78a3:0d82:1725:6a2f:0370:6234`. • `<xxx_v6mask>` — An IPv6 netmask, such as `/96`. • `<xxx_ipv6mask>` — An IPv6 address and netmask separated by a space. • `<xxx_str>` — A string of characters that is *not* another data type, such as `P@ssw0rd`. Strings containing spaces or special characters must be surrounded in quotes or use escape sequences. See Special characters. • `<xxx_int>` — An integer number that is *not* another data type, such as `15` for the number of minutes.

Subcommands

Once you connect to the CLI, you can enter commands.

Each command line consists of a command word that is usually followed by words for the configuration data or other specific item that the command uses or affects, for example:

```
get system admin
```

Subcommands are available from within the scope of some commands.When you enter a subcommand level, the command prompt changes to indicate the name of the current command scope. For example, after entering:

```
config system admin
```

the command prompt becomes:

```
(admin)#
```

Applicable subcommands are available to you until you exit the scope of the command, or until you descend an additional level into another subcommand.

For example, the `edit` subcommand is available only within a command that affects tables; the `next` subcommand is available only from within the `edit` subcommand:

```
config system interface
   edit port1
      set status up
   next
end
```

Available subcommands vary by command.From a command prompt within `config`, two types of subcommands might become available:

- commands that affect fields (see Field commands)
- commands that affect tables (see Table commands)

Subcommand scope is indicated in this CLI Reference by indentation. See Indentation.

Syntax examples for each top-level command in this CLI Reference do not show all available subcommands. However, when nested scope is demonstrated, you should assume that subcommands applicable for that level of scope are available.

Table commands

The following table describes commands used to manage configuration tables that contain sets of members or sets of rules, for example.

Commands for tables

`delete <table_name>`	Remove a table from the current object.

	For example, in `config system admin`, you could delete an administrator account named `newadmin` by typing `delete newadmin` and pressing Enter. This deletes `newadmin` and all its fields, such as `newadmin`'s `first-name` and `email-address`. `delete` is only available within objects containing tables.
`edit <table_name>`	Create or edit a table in the current object. For example, in `config system admin`: • edit the settings for the default `admin` administrator account by typing `edit admin`. • add a new administrator account with the name `newadmin` and edit `newadmin`'s settings by typing `edit newadmin`. `edit` is an interactive subcommand: further subcommands are available from within `edit`. `edit` changes the prompt to reflect the table you are currently editing. `edit` is only available within objects containing tables.
`end`	Save the changes to the current object and exit the `config` command. This returns you to the top-level command prompt.
`get`	List the configuration of the current object or table. In objects, `get` lists the table names (if present), or fields and their values. In a table, `get` lists the fields and their values. For more information on `get` commands, see get.
`purge`	Remove all tables in the current object. For example, in `config user local-user`, you could type `get` to see the list of all local user names, then type `purge` and then `y` to confirm that you want to delete all users. `purge` is only available for objects containing tables. **Caution:** Back up the FortiADC appliance before performing a purge because it cannot be undone. To restore purged tables, the configuration must be restored from a backup. **Caution:** Do not purge `system interface` or `system admin` tables. This can result in being unable to connect or log in, requiring the FortiADC appliance to be formatted and restored.
`show`	Display changes to the default configuration. Changes are listed in the form of configuration commands. For more information on `show` commands, see show.

Example of table commands

From within the `system admin` object, you might enter:

```
edit admin_1
```

The CLI acknowledges the new table, and changes the command prompt to show that you are now within the `admin_1` table:

```
new entry 'admin_1' added
(admin_1)#
```

Field commands

The following table describes commands to manage field settings.

Commands for fields

`abort`	Exit both the `edit` and/or `config` commands without saving the fields.
`end`	Save the changes made to the current table or object fields, and exit the `config` command. (To exit without saving, use `abort` instead.)
`get`	List the configuration of the current object or table. In objects, `get` lists the table names (if present), or fields and their values. In a table, `get` lists the fields and their values.
`next`	Save the changes you have made in the current table's fields, and exit the `edit` command to the object prompt. (To save and exit completely to the root prompt, use `end` instead.) `next` is useful when you want to create or edit several tables in the same object, without leaving and re-entering the `config` command each time. `next` is only available from a table prompt; it is not available from an object prompt.
`set <field_name> <value>`	Set a field's value. For example, in `config system admin`, after typing `edit admin`, you could type `set password newpass` to change the password of the `admin` administrator to `newpass`. **Note:** When using `set` to change a field containing a space-delimited list, type the whole new list. For example, `set <field> <new-value>` will replace the list with the <`new-value`> rather than appending <`new-value`> to the list.
`show`	Display changes to the default configuration. Changes are listed in the form of configuration commands.
`unset <field_name>`	Reset the table or object's fields to default values. For example, in `config system admin`, after typing `edit admin`, typing `unset password` resets the password of the `admin` administrator account to the default (in this case, no password).

Example of field commands

From within the `admin_1` table, you might enter the following command to assign the value `my1stExamplePassword` to the `password` field:

```
set password my1stExamplePassword
```

You might then enter the `next` command to save the changes and edit the next administrator's table.

Permissions

Depending on the account that you use to log in to the FortiADC appliance, you may not have complete access to all CLI commands or areas of the web UI.

Access profiles control which commands and areas an administrator account can access. Access profiles assign either:

- Read (view access)
- Write (change and execute access)
- Both read and write
- No access

Unlike other administrator accounts, the administrator account named `admin` exists by default and cannot be deleted. The `admin` administrator account is similar to a root administrator account. This administrator account always has full permission to view and change all FortiADC configuration options, including viewing and changing all other administrator accounts. Its name and permissions cannot be changed. It is the only administrator account that can reset another administrator's password without being required to enter that administrator's existing password.

For complete access to all commands, you must log in with the administrator account named `admin`.

Tips & tricks

Basic features and characteristics of the CLI environment provide support and ease of use for many CLI tasks.

This section includes:

- Help
- Shortcuts & key commands
- Command abbreviation
- Special characters
- Language support & regular expressions
- Screen paging

Help

- To display brief help during command entry, press the question mark (?) key.
- Press the question mark (?) key at the command prompt to display a list of the commands available and a description of each.
- Press the question mark (?) key after a command keyword to display a list of the objects available with that command and a description of each.
- Type a word or part of a word, then press the question mark (?) key to display a list of valid word completions or subsequent words, and to display a description of each.

Shortcuts & key commands

Shortcuts and key commands

Action	Keys
List valid word completions or subsequent words. If multiple words could complete your entry, display all possible completions with helpful descriptions of each.	?
Complete the word with the next available match. Press the key multiple times to cycle through available matches.	Tab
Recall the previous command. Command memory is limited to the current session.	Up arrow, or Ctrl + P
Recall the next command.	Down arrow, or Ctrl + N
Move the cursor left or right within the command line.	Left or Right arrow
Move the cursor to the beginning of the command line.	Ctrl + A
Move the cursor to the end of the command line.	Ctrl + E
Move the cursor backwards one word.	Ctrl + B
Move the cursor forwards one word.	Ctrl + F
Delete the current character.	Ctrl + D
Abort current interactive commands, such as when entering multiple lines. If you are not currently within an interactive command such as `config` or `edit`, this closes the CLI connection.	Ctrl + C
Continue typing a command on the next line for a multi-line command. For each line that you want to continue, terminate it with a backslash (\ ). To complete the command line, terminate it by pressing the spacebar and then the Enter key, without an immediately preceding backslash.	\ then Enter

Command abbreviation

You can abbreviate words in the command line to their smallest number of non-ambiguous characters. For example, the command `get system status` could be abbreviated to:

g sy st

If you enter an ambiguous command, the CLI returns an error message such as:

ambiguous command before 's'

Value conflicts with system settings.

Special characters

Special characters <, >, (,), #, ', and " are usually not permitted in CLI. If you use them, the CLI will often return an error message such as:

The string contains XSS vulnerability characters

value parse error before '%^@'

Input not as expected.

Some may be enclosed in quotes or preceded with a backslash (\ ) character.

Entering special characters

Character	Key
?	Ctrl + V then ?
Tab	Ctrl + V then Tab
Space (to be interpreted as part of a string value, not to end the string)	Enclose the string in quotation marks: “`Security Administrator`”. Enclose the string in single quotes: `'Security Administrator'`. Precede the space with a backslash: `Security\ Administrator`.
' (to be interpreted as part of a string value, not to end the string)	\'
" (to be interpreted as part of a string value, not to end the string)	\"
\	\\

Language support & regular expressions

Languages currently supported by the CLI interface include:

- English
- Japanese
- Simplified Chinese
- Traditional Chinese

In general, the names of configuration objects should be composed from common characters `A-Z, a-z, 0-9, _, -`.

Characters such as ñ, é, symbols, and ideographs are sometimes acceptable input. Support varies by the nature of the item being configured. CLI commands, objects, field names, and options must use their exact ASCII characters, but some items with arbitrary names or values may be input using your language of choice.

For example, the host name must not contain special characters, and so the web UI and CLI will not accept most symbols and other non-ASCII encoded characters as input when configuring the host name. This means that languages

other than English often are not supported. However, some configuration items, such as names and comments, may be able to use the language of your choice.

To use other languages in those cases, you must use the correct encoding.

The system stores the input using Unicode UTF-8 encoding, but it is not normalized from other encodings into UTF-8 before stored. If your input method encodes some characters differently than in UTF-8, your configured items may not display or operate as expected.

Regular expressions are especially impacted. Matching uses the UTF-8 character values. If you enter a regular expression using another encoding, or if an HTTP client sends a request in an encoding other than UTF-8, matches may not be what you expect.

For example, with Shift-JIS, backslashes (\ ) could be inadvertently interpreted as yen symbols (¥) and vice versa. A regular expression intended to match HTTP requests containing money values with a yen symbol therefore may not work it if the symbol is entered using the wrong encoding.

For best results, follow these guidelines:

- Use UTF-8 encoding, or
- Use only the characters whose numerically encoded values are the same in UTF-8, such as the US-ASCII characters that are also encoded using the same values in ISO 8859-1, Windows code page 1252, Shift-JIS and other encodings, or
- For regular expressions that must match HTTP requests, use the same encoding as your HTTP clients

HTTP clients may send requests in encodings other than UTF-8. Encodings usually vary by the client's operating system or input language. If you cannot predict the client's encoding, you may only be able to match any parts of the request that are in English, because regardless of the encoding, the values for English characters tend to be encoded identically. For example, English words may be legible regardless of interpreting a web page as either ISO 8859-1 or as GB2312, whereas simplified Chinese characters might only be legible if the page is interpreted as GB2312.

To configure the system using other encodings, you might need to switch language settings on your management computer, including for your web browser or Telnet or SSH client. For instructions on how to configure your management computer's operating system language, locale, or input method, see its documentation.

If you choose to configure parts of the system using non-ASCII characters, verify that all systems interacting with the FortiADC appliance also support the same encodings. You should also use the same encoding throughout the configuration if possible in order to avoid needing to switch the language settings of your web browser or Telnet or SSH client while you work.

Similarly to input, your web browser or CLI client should usually interpret display output as encoded using UTF-8. If it does not, your configured items may not display correctly in the web UI or CLI. Exceptions include items such as regular expressions that you may have configured using other encodings in order to match the encoding of HTTP requests that the system receives.

To enter non-ASCII characters in a Telnet or SSH client

1. On your management computer, start your Telnet or SSH client.
2. Configure your Telnet or SSH client to send and receive characters using UTF-8 encoding the encoding.

Support for sending and receiving international characters varies by each Telnet or SSH client. Consult the documentation for your Telnet or SSH client.

3. Log into the FortiADC system.
4. At the command prompt, type your command and press Enter.

You might need to surround words that use encoded characters with single quotes (').

Depending on your Telnet or SSH client's support for your language's input methods and for sending international characters, you may need to interpret them into character codes before pressing Enter.

For example, you might need to enter:

```
edit '\743\601\613\743\601\652'
```

The CLI displays your previous command and its output.

Screen paging

When output spans multiple pages, you can configure the CLI to pause after each page. When the display pauses, the last line displays `--More--`. You can then either:

- Press the spacebar to display the next page.
- Type `Q` to truncate the output and return to the command prompt.

This might be useful when displaying lengthy output, such as the list of possible matching commands for command completion, or a long list of settings. Rather than scrolling through or possibly exceeding the buffer of your terminal emulator, you can simply display one page at a time.

config endpoint-control

This section includes syntax for the following commands:

config endpoint-control fctems

Use this command to configure FortiClient Endpoint Management Server (EMS) connector entries.

It is recommended to configure the FortiClient EMS connector entries from the GUI. For more information, see the FortiADC Handbook on the FortiClient EMS Connector.

The FortiADC Security Fabric device can link to FortiClient Endpoint Management Server (EMS) for endpoint connectors. Up to three EMS servers can be added to the Security Fabric. EMS settings are synchronized between all Fabric members. Once the FortiADC is authorized as a Fabric device in FortiClient EMS, FortiClient EMS automatically synchronizes ZTNA tags, the EMS CA certificate, and FortiClient endpoint information to the FortiADC.

The FortiClient EMS connector is an integral part of the Zero Trust Network Access (ZTNA) functionality. For more information, see the FortiADC Handbook on ZTNA and How device identity and trust context is established with FortiClient EMS.

After you complete the configuration with the `config endpoint-control fctems` command, you must verify the EMS server certificate to authorize the FortiADC as a Fabric Device in FortiClient EMS. To verify the EMS server certificate, use the `execute fctems verify` command. For details, see execute fctems on page 599.

Once the FortiADC is authorized as a Fabric device in FortiClient EMS, FortiClient EMS automatically synchronizes ZTNA tags, the EMS CA certificate, and FortiClient endpoint information to the FortiADC.

Requirements:

- FortiClient EMS running version 7.0.3 or later
- FortiClient running 7.0.1 or later

- FortiADC hardware, VM, or cloud platform that support FortiClient EMS.

FortiClient EMS is supported in most FortiADC platforms but not all of them. The following lists the hardware models, cloud platforms, and VM environments that support FortiClient EMS.

Hardware models:

- FAD-120F, FAD-220F, FAD-300F, FAD-400F, FAD-1200F, FAD-2200F, FAD-4200F, FAD-5000F

Cloud platforms with BYOL (PAYG FortiADC does not support FortiClient EMS):

- AWS (Amazon Web Services), Microsoft Azure, GCP (Google Cloud Platform), OCI (Oracle Cloud Infrastructure), Alibaba Cloud

VM environments:

- VMware, Microsoft Hyper-V, KVM, Citrix Xen, Xen Project Hypervisor
  Note: The most recent certificate embedded license is required. If your license was issued prior to April 2021, please obtain a new certificate embedded license for your VM through Fortinet Customer Service & Support.

- Read-Write access permission for FortiADC Systems settings

Syntax

```
config endpoint-control fctems
   edit <name>
      set server {string}
      set https-port {integer}
      set capabilities {option1}, {option2}, ...
      set call-timeout {integer}
      set preserve-ssl-session {enable|disable}
   next
end
```

server	Server IPv4 address or the domain name of the FortiClient EMS FQDN. For example: 192.0.2.1
https-port	FortiClient EMS HTTPS access port number. Range: 1-65535, default: 443.
capabilities	List of EMS capabilities. **Note**: This option is only available in CLI.
call-timeout	FortiClient EMS call timeout in seconds. Range: 1-180, default: 30. **Note**: This option is only available in CLI.
preserve-ssl-session	Enable/disable preservation of EMS SSL session connection. This is disabled by default. **Note**: This option is only available in CLI. **Warning**: Most users should not touch this setting.

Example

```
config endpoint-control fctems
```

```
   edit "EMS-223"
      set server 10.106.3.223
      set https-port 443
      unset capabilities
      set call-timeout 30
      set preserve-ssl-session disable
   next
end
```

config endpoint-control client

This configuration is automatically created after the FortiClient endpoint information is synchronized to FortiADC from FortiClient EMS.

This is a hidden command. As the records are automatically generated, editing is not recommended. For debug information for FortiClient endpoints registered to FortiClient EMS, use diagnose endpoint-control client list on page 528.

In certain scenarios, you may need to delete a FortiClient endpoint record. For example, when you delete a FortiClient EMS connector from FortiADC and a ZTNA Profile is still using the ZTNA tag that has been synchronized from this FortiClient EMS connector, FortiADC would not delete the related endpoint-control client configuration. In this case, you can delete the ZTNA client record through CLI `config endpoint-control client` or use the purge command to delete the entire table.

Syntax

```
config endpoint-control client
   edit <FCT uid>
      set src-ip <IP>
      set src-mac <MAC>
      set info <EMS>
      config tags
         edit <tag>
         next
      end
   next
end
```

Example

```
config endpoint-control client
   edit "BEEC13BAF89C4EC5BEF1D6DC53012465"
      set src-ip 3858983434
      set vfid 4294967295
      set src-mac 00:50:56:81:63:ea
      set info FCTEMS8822003242
      config tags
         edit "FCTEMS8822003242_Low"
      next
   end
next …
purge
```

```
ADC(client) # purge
This operation will clear all table!
Do you want to continue? (y/n)y
```

config endpoint-control tag

This configuration is automatically created after the ZTNA tags are synchronized to FortiADC from FortiClient EMS.

This is a hidden command. As the records are automatically generated, editing is not recommended. For debug information for ZTNA tags registered to FortiClient EMS, use diagnose endpoint-control tag list on page 528.

In certain scenarios, you may need to delete a ZTNA tag record. For example, when you delete a FortiClient EMS connector from FortiADC and a ZTNA Profile is still using the ZTNA tag that has been synchronized from this FortiClient EMS connector, FortiADC would not delete the related endpoint-control tag configuration. In this case, you can delete the ZTNA tag record through CLI `config endpoint-control tag` or use the purge command to delete the entire table.

Syntax

```
config endpoint-control tags
   edit <$(EMS_SN)_$(TAG_NAME)>
      set uuid <UUID>
      set obj-type TAG
   next
end
```

Example

```
config endpoint-control tags
   edit "FCTEMS8822003242_REvil_IOC_registry_key"
   next
   …
purge
ADC(tags) # purge
This operation will clear all table!
Do you want to continue? (y/n)y
```

config firewall

The `config firewall` commands configure security feature settings.

This chapter is a reference for the following commands:

config firewall connlimit

Use this command to create connection limit security rules for IPv4 addresses.

The firewall connection limit policy allows or denies traffic based on a matching tuple: source address, destination address, and service; and connection count. The purpose is to detect anomalous connection requests.

The limit you specify can be based on the following counts:

- Count of concurrent sessions that match the tuple.
- Count of concurrent sessions from a single host that match the tuple.

The FortiADC system evaluates firewall connection limit policy rules before other rules. It matches traffic against the connection limit table, beginning with the first rule. If no rule matches, the connection is forwarded for further processing. If a rule matches, and the limit has not been reached, the connection is forwarded for further processing. If a rule matches and the limit has been reached, the connection is dropped.

By default, if firewall connection limit rules are not configured, the system does not perform connection limit policy processing. The firewall connection limit can be configured for non-SLB traffic and for Layer 7 SLB traffic, but not Layer 4 SLB traffic.

Note: The purpose of the firewall connection limit is distinct from the virtual server connection limit. The firewall connection limit setting is a security setting; the virtual server connection limit is a capacity setting.

Before you begin:

- You must have a good understanding and knowledge of the capacity of your backend servers.
- You must have created the address configuration objects and service configuration objects that define the matching tuple in your connection limit rules.
- You must have read-write permission for firewall settings.

Syntax

```
config firewall connlimit
   config rule
      edit <name>
         set connection-limit <integer>
         set destination-address <datasource>
         set in-interface <datasource>
         set out-interface <datasource>
         set service <datasource>
         set source-address <datasource>
         set type {host | rule}
         set side {both | destination | source}
      next
   end
end
```

connection-limit	Maximum concurrent sessions. The default is 1,048,576.
destination-address	Destination address object to use to form the matching tuple.
in-interface	Interface that receives traffic.
out-interface	Interface that forwards traffic.
service	Service object to use to form the matching tuple.
source-address	Source address object to use to form the matching tuple.
type	Whether the limit is per rule or per host.
side	When the connection limit is per host, specify whether the connection counter gets incremented when the host IP address appears in: • source—Only increment the counter if the host is the source address. • destination—Only increment the counter if the host is the destination address. • both—Increment the counter if the host is the source or destination address.

Example

```
FortiADC-VM # config firewall connlimit
FortiADC-VM (connlimit) # config rule
FortiADC-VM (rule) # edit dest-rule
Add new entry 'dest-rule' for node 1890

FortiADC-VM (dest-rule) # get
in-interface :
out-interface :
source-address :
destination-address :
service :
type : host
side : both
connection-limit : 1048576

FortiADC-VM (dest-rule) # set in-interface port4
FortiADC-VM (dest-rule) # set out-interface port5
```

```
FortiADC-VM (dest-rule) # set destination-address fw-dest-addr1
FortiADC-VM (dest-rule) # set service fw-http
FortiADC-VM (dest-rule) # set type rule
FortiADC-VM (dest-rule) # end
```

config firewall connlimit6

Use this command to create connection limit security rules for IPv6 addresses.

The firewall connection limit policy allows or denies traffic based on a matching tuple: source address, destination address, and service; and connection count. The purpose is to detect anomalous connection requests.

The limit you specify can be based on the following counts:

- Count of concurrent sessions that match the tuple.
- Count of concurrent sessions from a single host that match the tuple.

The FortiADC system evaluates firewall connection limit policy rules before other rules. It matches traffic against the connection limit table, beginning with the first rule. If no rule matches, the connection is forwarded for further processing. If a rule matches, and the limit has not been reached, the connection is forwarded for further processing. If a rule matches and the limit has been reached, the connection is dropped.

By default, if firewall connection limit rules are not configured, the system does not perform connection limit policy processing. The firewall connection limit can be configured for non-SLB traffic and for Layer 7 SLB traffic, but not Layer 4 SLB traffic.

Note: The purpose of the firewall connection limit is distinct from the virtual server connection limit. The firewall connection limit setting is a security setting; the virtual server connection limit is a capacity setting.

Before you begin:

- You must have a good understanding and knowledge of the capacity of your backend servers.
- You must have created the address configuration objects and service configuration objects that define the matching tuple in your connection limit rules.
- You must have read-write permission for firewall settings.

Syntax

```
config firewall connlimit6
   config rule
      edit <name>
         set connection-limit <integer>
         set destination-address6 <datasource>
         set in-interface <datasource>
         set out-interface <datasource>
         set service <datasource>
         set source-address6 <datasource>
         set type {host | rule}
         set side {both | destination | source}
      next
   end
```

```
end
```

connection-limit	Maximum concurrent sessions. The default is 1,048,576.
destination-address6	Destination address object to use to form the matching tuple.
in-interface	Interface that receives traffic.
out-interface	Interface that forwards traffic.
service	Service object to use to form the matching tuple.
source-address6	Source address object to use to form the matching tuple.
type	Whether the limit is per rule or per host.
side	When the connection limit is per host, specify whether the connection counter gets incremented when the host IP address appears in: • source—Only increment the counter if the host is the source address. • destination—Only increment the counter if the host is the destination address. • both—Increment the counter if the host is the source or destination address.

config firewall global

Use this command to configure the timeout period for the connection tracking sessions for the firewall.

Syntax

```
config firewall global
   edit <name>
      set generic-timeout <integer>
      set tcp-established-timeout <integer>
      set tcp-syn-recv-timeout <integer>
      set tcp-syn-sent-timeout <integer>
      set tcp-close-timeout <integer>
      set tcp-fin-wait-timeout <integer>
      set tcp-last-ack-timeout <integer>
      set udp-timeout <integer>
      set udp-stream-timeout <integer>
   next
end
```

generic-timeout	Specify the timeout of generic connections tracked by the netfilter connection tracking system. It determines how long the kernel will keep track of a connection that is considered idle, such as when it is not sending or receiving any traffic. Once the timeout period has elapsed, the connection tracking entry for that connection will be removed from the system. Range is 1-86400 seconds. Default is 600 seconds.

tcp-established-timeout	Specify the timeout after which an established TCP connection that has not received any traffic will be considered inactive and removed from the connection tracking table. Range is 1-86400 seconds. Default is 3600 seconds.
tcp-syn-recv-timeout	Specify the timeout after which a TCP SYN_RECV state connection entry will be removed from the connection tracking table. Range is 1-86400 seconds. Default is 60 seconds.
tcp-syn-sent-timeout	Specify the timeout after which a TCP SYN_SENT connection entry will be removed from the connection tracking table. Range is 1-86400 seconds. Default is 120 seconds.
tcp-close-timeout	Specify the timeout of TCP connections in CLOSE state. Range is 1-86400 seconds. Default 3 seconds.
tcp-fin-wait-timeout	Specify the timeout for TCP connections in FIN_WAIT state. Range is 1-86400 seconds. Default is 120 seconds.
tcp-last-ack-timeout	Specify the timeout after which a TCP LAST_ACK connection entry will be removed from the connection tracking table. Range is 1-86400 seconds. Default is 30 seconds.
udp-timeout	Specify the timeout of UDP connections tracked by the netfilter connection tracking system. Range is 1-86400 seconds. Default is 30 seconds.
udp-stream-timeout	Specify the timeout of UDP stream connections tracked by the netfilter connection tracking system. Range is 1-86400 seconds. Default is 180 seconds.

Example

```
config firewall global
   edit 1
      set generic-timeout 600
      set tcp-established-timeout 3600
      set tcp-syn-recv-timeout 60
      set tcp-syn-sent-timeout 120
      set tcp-close-timeout 3
      set tcp-fin-wait-timeout 120
      set tcp-last-ack-timeout 30
      set udp-timeout 30
      set udp-stream-timeout 180
   next
end
```

config firewall nat-snat

Use this command to configure source NAT (SNAT) rules.

You use SNAT when clients have IP addresses from private networks. This ensures you do not have multiple sessions from different clients with source IP 192.168.1.1, for example. Or, you can map all client traffic to a single source IP address because a source address from a private network is not meaningful to the FortiADC system or backend servers.

The system maintains this NAT table and performs the inverse translation when it receives the server-to-client traffic. Be sure to configure the backend servers to use the FortiADC address as the default gateway so that server responses are also rewritten by the NAT module.

Note: This SNAT feature is not supported for traffic to virtual servers. Use the virtual server SNAT feature instead.

Before you begin:

- You must have read-write permission for firewall settings.

Syntax

```
config firewall nat-snat
   edit <name>
      set from <ip&netmask>
      set out-interface <datasource>
      set status {enable | disable}
      set to <ip&netmask>
      set traffic-group <datasource>
      set trans-to-type {ip | pool | no-nat}
      set trans-to-ip <class_ip>
      set trans-to-ip-start <class_ip>
      set trans-to-ip-end <class_ip>
   next
end
```

`from`	Address/mask notation to match the source IP address in the packet header. 0.0.0.0/0 matches all IP addresses.
`out-interface`	Interface that forwards traffic.
`status`	Enable or disable SNAT status.
`to`	Address/mask notation to match the destination IP address in the packet header. For example, 192.0.2.0/24.
`traffic-group`	Specify a traffic group configuration object.
`trans-to-type`	• ip—Specify to translate the source IP to a single specified address. • pool—Specify to translate the source IP to the next address in a pool. • no-nat—Specify for no translation.
`trans-to-ip`	Specify an IPv4 address. The source IP address in the packet header will be translated to this address.
`trans-to-ip-start`	First IP address in the SNAT pool.
`trans-to-ip-end`	Last IP address in the SNAT pool.

Example

```
FortiADC-VM # config firewall nat-snat
FortiADC-VM (nat-snat) # edit fw-snat-example
Add new entry 'fw-snat-example' for node 1941

FortiADC-VM (fw-snat-example) # get
from : 0.0.0.0/0
to : 0.0.0.0/0
out-interface :
trans-to-type : ip
trans-to-ip : 0.0.0.0
traffic-group :
status : enable


FortiADC-VM (fw-snat-example) # set to 192.0.2.0/24
FortiADC-VM (fw-snat-example) # set out-interface port5
FortiADC-VM (fw-snat-example) # set trans-to-ip 192.0.2.10

FortiADC-VM (fw-snat-example) # get
from : 0.0.0.0/0
to : 192.0.2.0/24
out-interface : port5
trans-to-type : ip
trans-to-ip : 192.0.2.10
traffic-group :
status : enable

FortiADC-VM (fw-snat-example) # end
```

config firewall policy

Use this command to configure firewall policy rules for IPv4 addresses.

A firewall policy is a filter that allows or denies traffic to be forwarded to the system based on a matching tuple: source address, destination address, and service. By default, firewall policy rules are stateful: if client-to-server traffic is allowed, the session is maintained in a state table, and the response traffic is allowed.

The FortiADC system evaluates firewall policies before other rules. It matches traffic against the firewall policy table, beginning with the first rule. If a rule matches, the specified action is taken. If the session is denied by a firewall policy rule, it is dropped. If the session is accepted, system processing continues.

By default, if firewall rules are not configured, the system does not perform firewall processing; all traffic is processed as if the system were a router, and traffic is forwarded according to routing and other system rules.

Note: You do not need to create firewall rules for routine management traffic associated with the management port or HA ports. The interface “allow access” option enables permitted protocols. The system automatically permits from-self traffic, such as health check traffic, and expected responses.

Before you begin:

- You must have a good understanding and knowledge of firewalls.
- You must have created the address configuration objects and service configuration objects that define the matching tuple in your firewall policy rules.
- You must have read-write permission for firewall settings.

Syntax

```
config firewall policy
   set default-action {deny|accept}
   set stateful {enable|disable}
   config rule
      edit <name>
         set action {deny | accept}
         set deny-log {disable | enable}
         set destination-type {address|addrgrp|external-resource}
         set destination-address <datasource>
         set destination-addrgrp <datasource>
         set destination-external-resource-address <datasource>
         set in-interface <datasource>
         set out-interface <datasource>
         set service <datasource>
         set source-type {address|addrgrp|external-resource}
         set source-address <datasource>
         set source-addrgrp <datasource>
         set source-external-resource-address <datasource>
         set status {enable | disable}
      next
   end
end
```

`default-action`	Action when no rule matches or no rules are configured: • deny—Drop the traffic. • accept—Allow the traffic to pass the firewall.
`stateful`	Enable/disable stateful firewall. When enabled, server response traffic is permitted automatically when the client-to-server rule allows the connection to be established. When disabled, you must create separate rules for client-to-server and server-to-client traffic. Enabled by default.
config rule	
`action`	• deny—Drop the traffic. • accept—Allow the traffic to pass the firewall.
`deny-log`	Enable/disable deny log. When enabled, firewall denied traffic is logged, wherein relevant information for traffic is logged when the traffic triggers the policy action to drop it. The `deny-log` option is only available when `action` is set to `deny`. Disabled by default.
`destination-type`	Select the destination type to use to form the matching tuple. • address • addrgrp

	• external-resource
destination-address	The **destination-address** option is available if the **destination-type** is **address**. Specify the Address object to use as the destination.
destination-addrgrp	The **destination-address-group** option is available if the **destination-type** is **addrgrp**. Specify the Address Group object to use as the destination.
destination-external-resource-address	The **destination-external-resource-address** option is available if the **destination-type** is **external-resource**. Specify the external IP address list imported through the IP Address connector to use as the destination. For details, see config system external-resource on page 427.
in-interface	Interface that receives traffic.
out-interface	Interface that forwards traffic.
service	Service object to use to form the matching tuple.
source-type	Select the source type to use to form the matching tuple. • address • address-group • external-resource
source-address	The **source-address** option is available if the **source-type** is **address**. Specify the Address object to use as the source.
source-addrgrp	The **source-address-group** option is available if the **source-type** is **addrgrp**. Specify the Address Group object to use as the source.
source-external-resource-address	The **source-external-resource-address** option is available if the **source-type** is **external-resource**. Specify the external IP address list imported through the IP Address connector to use as the source. For details, see config system external-resource on page 427.
status	Enable or disable firewall policy rule.

Example

```
FortiADC-VM # config firewall policy
FortiADC-VM (policy) # set default-action deny
FortiADC-VM (policy) # config rule
FortiADC-VM (rule) # edit fw-allow-http
Add new entry 'fw-allow-http' for node 1871

FortiADC-VM (fw-allow-http) # get
in-interface :
out-interface :
source-address :
destination-address :
service :
status : enable
action :

FortiADC-VM (fw-allow-http) # set action accept
```

```
FortiADC-VM (fw-allow-http) # set in-interface port4
FortiADC-VM (fw-allow-http) # set out-interface port5
FortiADC-VM (fw-allow-http) # set source-address fw-source-addr1
FortiADC-VM (fw-allow-http) # set destination-address fw-dest-addr1
FortiADC-VM (fw-allow-http) # set service fw-http

FortiADC-VM (fw-allow-http) # get
in-interface : port4
out-interface : port5
source-address : fw-source-addr1
destination-address : fw-dest-addr1
service : fw-http
status : enable
action : accept

FortiADC-VM (fw-allow-http) # end
```

config firewall policy6

Use this command to configure firewall policy rules for IPv6 addresses.

A firewall policy is a filter that allows or denies traffic to be forwarded to the system based on a matching tuple: source address, destination address, and service. By default, firewall policy rules are stateful: if client-to-server traffic is allowed, the session is maintained in a state table, and the response traffic is allowed.

The FortiADC system evaluates firewall policies before other rules. It matches traffic against the firewall policy table, beginning with the first rule. If a rule matches, the specified action is taken. If the session is denied by a firewall policy rule, it is dropped. If the session is accepted, system processing continues.

By default, if firewall rules are not configured, the system does not perform firewall processing; all traffic is processed as if the system were a router, and traffic is forwarded according to routing and other system rules.

Note: You do not need to create firewall rules for routine management traffic associated with the management port or HA ports. The interface “allow access” option enables permitted protocols. The system automatically permits from-self traffic, such as health check traffic, and expected responses.

Before you begin:

- You must have a good understanding and knowledge of firewalls.
- You must have created the address configuration objects and service configuration objects that define the matching tuple in your firewall policy rules.
- You must have read-write permission for firewall settings.

Syntax

```
config firewall policy6
   set default-action {deny|accept}
   set stateful {enable|disable}
   config rule
      edit <name>
```

```
        set action {deny | accept}
        set deny-log {disable | enable}
        set destination-type {address|addrgrp|external-resource}
        set destination-address6 <datasource>
        set destination-addrgrp6 <datasource>
        set destination-external-resource-address6 <datasource>
        set in-interface <datasource>
        set out-interface <datasource>
        set service <datasource>
        set source-type {address|addrgrp|external-resource}
        set source-address6 <datasource>
        set source-addrgrp6 <datasource>
        set source-external-resource-address6 <datasource>
        set status {enable | disable}
    next
  end
end
```

`default-action`	Action when no rule matches or no rules are configured: • deny—Drop the traffic. • accept—Allow the traffic to pass the firewall.
`stateful`	Enable/disable stateful firewall. When enabled, server response traffic is permitted automatically when the client-to-server rule allows the connection to be established. When disabled, you must create separate rules for client-to-server and server-to-client traffic. Enabled by default.
config rule	
`action`	• deny—Drop the traffic. • accept—Allow the traffic to pass the firewall.
`deny-log`	Enable/disable deny log. When enabled, firewall denied traffic is logged, wherein relevant information for traffic is logged when the traffic triggers the policy action to drop it. The `deny-log` option is only available when `action` is set to `deny`. Disabled by default.
`destination-type`	Select the destination type to use to form the matching tuple. • address • addrgrp • external-resource
`destination-address6`	The **destination-address6** option is available if the **destination-type** is **address**. Specify the Address object to use as the destination.
`destination-addrgrp6`	The **destination-address-group6** option is available if the **destination-type** is **addrgrp**. Specify the Address Group object to use as the destination.
`destination-external-resource-address6`	The **destination-external-resource-address6** option is available if the **destination-type** is **external-resource**. Specify the external IP address list imported through the IP Address connector to use as the destination. For details, see config system external-resource on page 427.

in-interface	Interface that receives traffic.
out-interface	Interface that forwards traffic.
service	Service object to use to form the matching tuple.
source-type	Select the source type to use to form the matching tuple. • address • address-group • external-resource
source-address6	The **source-address6** option is available if the **source-type** is **address**. Specify the Address object to use as the source.
source-addrgrp6	The **source-address-group6** option is available if the **source-type** is **addrgrp**. Specify the Address Group object to use as the source.
source-external-resource-address6	The **source-external-resource-address6** option is available if the **source-type** is **external-resource**. Specify the external IP address list imported through the IP Address connector to use as the source. For details, see config system external-resource on page 427.
status	Enable or disable firewall policy6 rule.

config firewall qos-filter

Use this command to configure QoS rules for IPv4 addresses.

A QoS filter is the policy that assigns traffic to the QoS queue.

Note: The QoS policy feature is not supported for traffic to virtual servers.

Before you begin:

- You must have a good understanding and knowledge of traffic in your network that requires QoS provisioning.
- You must have created the address configuration objects and service configuration objects that define the matching tuple for QoS rules.
- You must have created a QoS queue configuration object.
- You must have read-write permission for firewall settings.

Syntax

```
config firewall qos-filter
   edit <name>
      set destination-address <datasource>
      set in-interface <datasource>
      set out-interface <datasource>
      set queue <datasource>
      set service <datasource>
      set source-address <datasource>
```

```
    set status {enable|disable}
  next
end
```

destination-address	Destination address object to use to form the matching tuple.
in-interface	Interface that receives traffic.
out-interface	Interface that forwards traffic.
queue	QoS queue that will be used for packets that match the filter criteria.
service	Service object to use to form the matching tuple.
source-address	Source address object to use to form the matching tuple.
status	Enable/disable the filter.

Example

```
FortiADC-VM # config firewall qos-filter

FortiADC-VM (qos-filter) # edit qos-premium
Add new entry 'qos-premium' for node 1922

FortiADC-VM (qos-premium) # get
status : enable
in-interface :
out-interface :
source-address :
destination-address :
service :
queue :

FortiADC-VM (qos-premium) # set in-interface port4
FortiADC-VM (qos-premium) # set out-interface port5
FortiADC-VM (qos-premium) # set source-address fw-source-addr1
FortiADC-VM (qos-premium) # set destination-address fw-dest-addr1
FortiADC-VM (qos-premium) # set service fw-http
FortiADC-VM (qos-premium) # set queue lane-1

FortiADC-VM (qos-premium) # get
status : enable
in-interface : port4
out-interface : port5
source-address : fw-source-addr1
destination-address : fw-dest-addr1
service : fw-http
queue : lane-1

FortiADC-VM (qos-premium) # end
```

config firewall qos-filter6

Use this command to configure QoS rules for IPv6 addresses.

A QoS filter is the policy that assigns traffic to the QoS queue.

Note: The QoS policy feature is not supported for traffic to virtual servers.

Before you begin:

- You must have a good understanding and knowledge of traffic in your network that requires QoS provisioning.
- You must have created the address configuration objects and service configuration objects that define the matching tuple for QoS rules.
- You must have created a QoS queue configuration object.
- You must have read-write permission for firewall settings.

Syntax

```
config firewall qos-filter6
   edit <name>
      set destination-address6 <datasource>
      set in-interface <datasource>
      set out-interface <datasource>
      set queue <datasource>
      set service <datasource>
      set source-address6 <datasource>
      set status {enable|disable}
   next
end
```

destination-address6	Destination address object to use to form the matching tuple.
in-interface	Interface that receives traffic.
out-interface	Interface that forwards traffic.
queue	QoS queue that will be used for packets that match the filter criteria.
service	Service object to use to form the matching tuple.
source-address6	Source address object to use to form the matching tuple.
status	Enable/disable the filter.

config firewall qos-queue

Use this command to configure QoS queues.

You can use QoS policies to provision bandwidth for any traffic that matches the rule. You might consider QoS policies for latency- or bandwidth-sensitive services, such as VoIP and ICMP.

The FortiADC system does not provision bandwidth based on the TOS bits (also called differentiated services) in the IP header to control packet queueing. Instead, the system provisions bandwidth based on a source/destination/service matching tuple that you specify.

Note: The QoS policy feature is not supported for traffic to virtual servers.

Basic steps

1. Configure a queue.
2. Configure a QoS filter.

Before you begin:

- You must have read-write permission for firewall settings.

Syntax

```
config firewall qos-queue
   edit <name>
      set bandwidth <digit>[G|M|K]
   next
end
```

`bandwidth`	Maximum bandwidth rate. Specify a number and a unit abbreviation. For example, specify `100K` for 100 Kbps, `10M` for 10 Mbps, and `1G` for 1Gbps. If you do not specify a bandwidth, the default qos-queue is 1G.

Example

The following commands configure a firewall policy rule:

```
FortiADC-VM # config firewall qos-queue

FortiADC-VM (qos-queue) # edit lane-1
Add new entry 'lane-1' for node 1909
FortiADC-VM (lane-1) # end

FortiADC-VM # get firewall qos-queue lane-1
bandwidth : 1G
bandwidth-int : 1073741824
```

config firewall vip

Use this command to configure 1-to-1 NAT rules.

You can use 1-to-1 NAT when you want to publish public or “external” IP addresses for FortiADC resources but want the communication among servers on the internal network to be on a private or “internal” IP address range.

1-to-1 NAT is supported for traffic to virtual servers. The address translation occurs before the ADC has processed its rules, so FortiADC server load balancing policies that match source address (such as content routing and content rewriting rules) should be based on the mapped address space.

The system maintains this NAT table and performs the inverse mapping when it sends traffic from the internal side to the external side.

Before you begin:

- You must have read-write permission for firewall settings.

Syntax

```
config firewall vip
   edit <name>
      set extif <datasource>
      set extip <class_ip>
      set extport <integer>
      set mappedip-min <class_ip>
      set mappedip-max <class_ip>
      set mappedport-min <integer>
      set mappedport-max <integer>
      set portforward {enable | disable}
      set protocol {tcp | udp}
      set status {enable | disable}
      set traffic-group <datasource>
   next
end
```

`extif`	Interface that receives traffic.
`extip`	Specify the first address in the range. The last address is calculated after you enter the mapped IP range.
`extport`	Specify the first port number in the range. The last port number is calculated after you enter the mapped port range.
`mappedip-min`	First address in the range.
`mappedip-max`	Last address in the range.
`mappedport-min`	First port in the range.
`mappedport-max`	Last port in the range.
`portforward`	Enable/disable port forwarding.
`protocol`	TCP or UDP
`status`	Enable or disable static nat status
`traffic-group`	Specify the traffic group name.

Example

```
FortiADC-VM # config firewall vip
FortiADC-VM (vip) # edit 1-to-1-NAT
```

```
Add new entry '1-to-1-NAT' for node 661

FortiADC-VM (1-to-1-NAT) # get
extif :
extip : 0.0.0.0
mappedip-min : 0.0.0.0
mappedip-max : 0.0.0.0
portforward : disable
traffic-group:
status: enable

FortiADC-VM (1-to-1-NAT) # set extif port4
FortiADC-VM (1-to-1-NAT) # set extip 198.51.100.10
FortiADC-VM (1-to-1-NAT) # set mappedip-min 192.0.2.10
FortiADC-VM (1-to-1-NAT) # set mappedip-max 192.0.2.19

FortiADC-VM (1-to-1-NAT) # get
extif : port4
extip : 198.51.100.10
mappedip-min : 192.0.2.10
mappedip-max : 192.0.2.19
traffic-group :
status : enable
portforward : disable
status: enable

FortiADC-VM (1-to-1-NAT) # end
```

config global

The `config global` command is applicable to VDOMs and visible only to super admin users. See Appendix A: Virtual domains for information about special VDOM commands.

This chapter is a reference for the following commands:

config config sync-list

Use this command to push/pull a configuration to/from a target FortiADC system.

Before you begin:

- You must plan for the impact the configuration push/pull has on the target deployment.
- You must have read-write permission for system settings.

Syntax

```
config config sync-list
   edit <name>
      set server-ip <class_ip>
      set password <string>
      set type {global-load-balance link-load-balance load-balance log networking security
           share-resource system user}
      set comment <string>
   next
end
```

`server-ip`	IP address of the remote appliance.
`password`	Password of the remote appliance.
`type`	Space-separated list of configuration types to sync: • global-load-balance — Includes `config global-load-balance` and `config global-dns-server` commands. **Note**: If any GLB configurations reference a shared resource, that shared resource configuration must be synced together with the GLB configuration, otherwise the GLB configuration will not work with the missing referenced object. • link-load-balance — Includes `config link-load-balance` commands.

	• load-balance — Includes `config load-balance` commands.
	• log — Includes `config log` commands and `config system mailserver`.
	• networking — Includes `config router` commands.
	• security — Includes `config security waf` commands.
	• share-resource — Includes the shared resources that are used across different modules.
	These configurations include the following commands:
	`config system health-check`
	`config system health-check-script`
	`config system schedule-group`
	`config system address`
	`config system address6`
	`config system addrgrp`
	`config system addrgrp6`
	`config system isp-addr`
	`config system service`
	`config system servicegrp`
	• system — Includes `config config`, `config system` (except `config system mailserver`), `config user`, and `config vdom` commands.
	• user — Includes `config user` commands

`comment`	A string to describe the purpose of the configuration, to help you and other administrators more easily identify its use. Put phrases in quotes. For example: “SLB and GLB settings to Data Center East”.

Example

```
config config sync-list
   edit 1
      set server-ip 192.168.1.100
      set port 995
      set password xxx
      set type global-load-balance
   next
end
```

See Also

- execute config-sync on page 563

config log setting global_remote

Use this command to configure global remote log settings.

When FortiADC has several VDOMs, the user may not want to configure the syslog server for each VDOM. This command defines syslog configuration at the global context so that each VDOM can be configured in one command.

The syslog server in the global context can be accessed by a root VDOM route.

You can configure and customize the individual VDOM with config log setting remote on page 256.

Syntax

```
config log setting general
set override_global_remote {enable | disable}
next
end
```

override-global-remote	Enable/disable to override global remote.

Example

```
FortiADC-VM (root) # config log setting general
FortiADC-VM (general) # show full-configuration
config log setting general
set override_global_remote disable
end
config log setting remote
edit 1
set status enable
set server 10.106.155.142
set port 514
set proto tcp
set tcp_framing traditional
set loglevel information
set comma-separated-value disable
set facility kern
set event-log-status enable
set event-log-category configuration admin health_check system
set traffic-log-status disable
set attack-log-status disable
next
```

config log setting global_faz

Use this command to configure global FortiAnalyzer log settings.

When FortiADC has several VDOMs, the user may not want to configure the FortiAnalyzer server for each VDOM. This command defines FortiAnalyzer log configuration at the global context so that each VDOM can be configured in one command.

The FortiAnalyzer server in the global context can be accessed by a root VDOM route.

You can configure and customize the individual VDOM with config log setting fortianalyzer on page 259.

Syntax

```
config log setting general
   set override_global_faz {enable | disable}
next
end
```

override-global-faz	Enable/disable to override global FortiAnalyzer.

Example

```
FortiADC-VM (root) # config log setting general
FortiADC-VM (general) # show full-configuration
config log setting general
   set override_global_faz disable
end
config log setting fortianalyzer
   edit 1
      set status enable
      set server 192.8.8.8
      set loglevel information
      set event-log-status enable
      set event-log-category system slb
      set traffic-log-status enable
      set traffic-log-category slb dns
      set attack-log-status enable
      set attack-log-category waf av
   next
end
```

config system global

Use this command to manage system settings.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system global
   set admin-idle-timeout <integer>
```

```
    set advanced-bot-status {enable|disable}
    set config-sync {enable|disable}
    set default-certificate <certname>
    set hardware-ssl {enable|disable}
    set hostname <string>
    set language {english|chinese-simplified}
    set port-http <integer>
    set port-https <integer>
    set port-ssh <integer>
    set port-telnet <integer>
    set share-ip-address {enable|disable}
    set snat-match-local-traffics {enable|disable}
    set ipvs-fullnat-min-port <integer>
    set ipvs-fullnat-max-port <integer>
    set snat-min-port <integer>
    set snat-max-port <integer>
    set socket-min-port <integer>
    set socket-max-port <integer>
    set ssh-cbc-cipher {enable|disable}
    set ssh-hmac-md5 {enable|disable}
    set vdom-admin {enable|disable}
    set vdom-mode {independent-network|share-network}
    set pre-login-banner {enable|disable}
    set sync-slb-statistics {enable|disable}
    set shell-access {enable|disable}
    set shell-username <username>
    set shell-password <password>
    set shell-timeout <integer>
    set threat-analytics {enable|disable}
end
```

`admin-idle-timeout`	Log out an idle administrator session. The default is 30 minutes.
`advanced-bot-status`	Enable/disable the Advanced Bot Protection Fabric Connector. Once enabled and successfully connected, the `config security waf advanced-bot-protection` command becomes available to configure the Advanced Bot Protection policy. See config security waf advanced-bot-protection on page 323. If you want to disable the Advanced Bot Protection connector, the following settings will be impacted: • Advanced Bot Protection policies will not be able to be created or edited, however, but can be deleted via CLI. • Existing ABP policies can be seen and deleted through CLI only. • In the WAF Profile configuration, the Advanced Bot Protection option can only be set through CLI. For more information, see the Handbook topic about Advanced Bot Protection.
`config-sync`	Enable/disable the configuration synchronization feature. This feature is related to the execute config-sync command, not HA synchronization. Disabled by default.
`default-certificate`	The default is Factory.

`hardware-ssl`	Enable/disable hardware SSL acceleration. The setting has no effect on FortiADC-VM.
`hostname`	You can configure a hostname to facilitate system management. If you use SNMP, for example, the SNMP system name is derived from the configured hostname. The hostname can be up to 35 characters in length. It can include US-ASCII letters, numbers, hyphens, and underscores, but not spaces and special characters. The System Information widget and the `get system status` CLI command display the full hostname. If the hostname is longer than 16 characters, the name is truncated and ends with a tilde (~) to indicate that additional characters exist, but are not displayed.
`language`	English or Simplified Chinese.
`port-http`	Specify the port for the HTTP service. Usually, HTTP uses port 80.
`port-https`	Specify the port for the HTTPS service. Usually, HTTPS uses port 443.
`port-ssh`	Specify the port for the SSH service. Usually, SSH uses port 22.
`port-telnet`	Specify the port for the Telnet service. Usually, Telnet uses port 25.
`share-ip-address`	Enable this option to share NAT IP pools/addresses between L4, L7 virtual servers, and SNAT policy. Once enabled, SNAT across the firewall, L4 VS and L7 VS can use the same IP address, but with different port ranges that can be customized.
`snat-match-local-traffics`	If **share-ip-address** is enabled, **snat-match-local-traffics** becomes configurable. Enable/disable the SNAT rule to match with the local traffic.
`ipvs-fullnat-min-port`	If **share-ip-address** is enabled, **ipvs-fullnat-min-port** becomes configurable. Specify the L4 VS FULLNAT port range minimum.
`ipvs-fullnat-max-port`	If **share-ip-address** is enabled, **ipvs-fullnat-max-port** becomes configurable. Specify the L4 VS FULLNAT port range maximum.
`snat-min-port`	If **share-ip-address** is enabled, **snat-min-port** becomes configurable. Specify the SNAT port range minimum.
`snat-max-port`	If **share-ip-address** is enabled, **snat-max-port** becomes configurable. Specify the SNAT port rang maximum.
`socket-min-port`	If **share-ip-address** is enabled, **socket-min-port** becomes configurable. Specify the L7 VS port range minimum.
`socket-max-port`	If **share-ip-address** is enabled, **socket-max-port** becomes configurable. Specify the L7 VS port range maximum.
`ssh-cbc-cipher`	Disabled by default. Enable if you want to use this cipher.
`ssh-hmac-md5`	Disabled by default. Enable if you want to use this cipher.

vdom-admin	Enables the virtual domain feature.
vdom-mode	The **vdom-mode** option becomes available if **vdom-admin** is enabled. Select either of the following virtual domain modes: • independent-network — each VDOM functions independently within its own network, unaffected by activity from other VDOMs on the system. • share-network — VDOMs function as administrative domains (ADOMs), sharing the same network interface and routing between all ADOMs. There are different CLI functions available to administrators of the root ADOM and non-root ADOM. For more information, see Appendix A: Virtual domains on page 620.
pre-login-banner	Enables the pre-login banner feature.
sync-slb-statistics	Enable/disable the statistic data between the SLB and GLB.
shell-access	Enable/disable the shell access. This is disabled by default.
shell-username	Specify the username to login to the shell.
shell-password	Specify the password to access the shell.
shell-timeout	The expire time, in minutes, after the shell access is enabled. (Range: 1-1200 minutes).
threat-analytics	Enable/disable the Threat Analytics connector. If you do not already have a license for the Fortinet AI Threat Analytic service, FortiADC offers a 14-day Evaluation license to evaluate the Fortinet AI Threat Analytics service. During this 14-day trial period, you can disable and re-enable AI Threat Analytics anytime. The 14-day trial period starts from the first time AI Threat Analytics is enabled. For more information about the AI Threat Analytics integration with FortiADC, see the FortiADC Handbook topic on AI Threat Analytics.

Example

```
FortiADC-VM # get system global
                              default-certificate          : Factory
                              hostname                     : FortiADC-VM
                              vdom-admin                   : disable
                              admin-idle-timeout           : 480
                              port-http                    : 80
                              port-https                   : 443
                              port-ssh                     : 22
                              port-telnet                  : 23
                              share-ip-address             : enable
                              snat-match-local-traffics    : enable
                              ipvs-fullnat-min-port        : 5000
                              ipvs-fullnat-max-port        : 21846
                              snat-min-port                : 21847
                              snat-max-port                : 43690
                              socket-min-port              : 43691
                              socket-max-port              : 65535
```

```
language                        : english
hardware-ssl                    : enable
gui-system                      : enable
gui-router                      : enable
gui-log                         : enable
ssh-cbc-cipher                  : disable
ssh-hmac-md5                    : disable
config-sync-enable              : disable
pre-login-banner                : enable
sync-slb-statistics             : enable
shell-access                    : enable
shell-username                  : user
shell-password                  : 123456
shell-expire-time               : 10
threat-analytics                : enable
```

config system vdom-link

Use this command to create VDOM links and configure its link type. By default, VDOM links are created as ethernet links. The link type can't be changed after it has been created. To set a VDOM link type to point-to-point (ppp), it needs to be created in the CLI.

This command will automatically create a VDOM-link pair in the system interface. However, by default, these VDOM links will not be assigned an IP address or `allowaccess` options, so you would not be able to route traffic between the VDOM links until these settings are configured in the system interface.

Inter-VDOM routing is only available for these classic scenarios: static route, PBR, L4 SLB, L7 SLB and NAT. It is currently not supported in IPv6 related configurations.

Syntax

```
config global
   config system vdom-link
      edit <vdom-link-name>
         set type {ethernet|ppp}
      next
   end
end
```

`type`	Specify the VDOM link type from the following: • ppp • ethernet

config global-dns-server

The `config global-dns-server` commands configure the DNS server used in global load balancing.

This chapter is a reference for the following commands:

config global-dns-server address-group

Use this command to configure the source and destination IP addresses that are the matching criteria for DNS policies. The system includes the predefined address groups `any` and `none`.

Before you begin:

- You must have read-write permission for global load balancing settings.

After you have configured an address group, you can specify it in the DNS64 and DNS policy configurations.

Syntax

```
config global-dns-server address-group
   edit <name>
      config member
         edit <No.>
            set action {include|exclude}
            set addr-type {ipv4|ipv6}
            set ip-network <ip&netmask>
            set ip6-network <ip&netmask>
         next
      end
   next
end
```

`action`	• include—The rule logic creates an address object that includes addresses matching the specified address block.

	• exclude—The rule logic creates an address object that excludes addresses matching the specified address block.
addr-type	IPv4 or IPv6
ip-network	Address/mask notation to match the IP address in the packet header. Create objects to match source IPv4 address and different objects to match destination IPv4 address.
ip6-network	Address/mask notation to match the IPv6 address in the packet header. Create objects to match source IPv6 address and different objects to match destination IPv6 address.

Example

```
FortiADC-VM # config global-dns-server address-group
FortiADC-VM (address-group) # edit campus
Add new entry 'campus' for node 2206

FortiADC-VM (campus) # config member
FortiADC-VM (member) # edit 1
Add new entry '1' for node 2209

FortiADC-VM (1) # get
action : include
addr-type : ipv4
ip-network : 0.0.0.0/0

FortiADC-VM (1) # set ip-network 192.0.2.0/24
FortiADC-VM (1) # end
FortiADC-VM (campus) # end

FortiADC-VM # config global-dns-server address-group
FortiADC-VM (address-group) # edit branch
Add new entry 'branch' for node 2206

FortiADC-VM (branch) # config member
FortiADC-VM (member) # edit 1
Add new entry '1' for node 2209
FortiADC-VM (1) # set ip-network 198.51.100.0/24
FortiADC-VM (1) # end
FortiADC-VM (branch) # end

FortiADC-VM # show global-dns-server address-group
config global-dns-server address-group
edit "campus"
config member
edit 1
set ip-network 192.0.2.0/24
next
end
next
edit "branch"
config member
edit 1
```

```
set ip-network 198.51.100.0/24
next
end
next
end
```

config global-dns-server dns64

Use this command to map IPv4 addresses to AAAA queries when there are no AAAA records. This feature is optional. It can be used in network segments that use NAT64 to support IPv6 client communication with IPv4 backend servers.

Before you begin:

- You must have a good understanding of DNS and knowledge of the DNS deployment in your network.
- You must have configured address objects that specify the network segments for which the DNS64 map applies.
- You must have read-write permission for global load balancing settings.

After you have created a DNS64 configuration, you can specify it in a DNS policy configuration.

Syntax

```
config global-dns-server dns64
   edit <name>
      set exclude {any | none | <datasource>}
      set mapped-address {any | none | <datasource>}
      prefix6 <ip&netmask>
      source-address {any | none | <datasource>}
   next
end
```

exclude	Specify a wildcard (any or none) or an address object. Allows specification of a list of IPv6 addresses that can be ignored. Typically, you exclude addresses that do have AAAA records.
mapped-address	Address object that specifies the IPv4 addresses that are to be mapped in the corresponding A RR set.
prefix6	IP address and netmask that specify the DNS64 prefix. Compatible IPv6 prefixes have lengths of 32, 40, 48, 56, 64 and 96 as per RFC 6052. Each DNS64 configuration has one prefix. Multiple configurations can be defined.
source-address	Specify an address object. Only clients that match the source IP use the DNS64 lookup table.

Example

```
FortiADC-VM # config global-dns-server dns64
FortiADC-VM (dns64) # edit 1
Add new entry '1' for node 2289
```

```
FortiADC-VM (1) # get
prefix6 : ::/0
source-address :
mapped-address :
exclude :

FortiADC-VM (1) # set prefix6 64:ff::/96
FortiADC-VM (1) # set source-address any
FortiADC-VM (1) # set mapped-address dns64_mapped_pool
FortiADC-VM (1) # set exclude none

FortiADC-VM (1) # get
prefix6 : 64:ff::/96
source-address : any
mapped-address : dns64_mapped_pool
exclude : none

FortiADC-VM (1) # end
```

config global-dns-server dsset-info-list

Use this command to paste in the content of the DSSET files provided by child domain servers or stub domains.

If you enable DNSSEC, secure communication between the FortiADC DNS server and any child DNS servers is based on keys contained in delegation signer files (DSSET files). In DNSSEC deployments, DSSET files are generated automatically when the zone is signed by DNSSEC.

Note: You use the Global DNS zone configuration to generate the DSSET file for this server. The file generated by the zone configuration editor is the one you give to any parent zone or the registrar of your domain.

Before you begin:

- You must have a good understanding of DNSSEC and knowledge of the DNS deployment in your network.
- You must have used DNSSEC to sign the child domain servers and have downloaded the DSset files to a location you can reach from your management computer.
- You must have read-write permission for global load balancing settings.

Syntax

```
config global-dns-server dsset-info-list
   edit <name>
      set filename <string>
      set content <string>
   next
end
```

`filename`	Specify the filename. The convention is dsset-<domain>, for example, `dsset-example.com`.

`content`	Specify (paste) the DSset file content. The content of DSset files is similar to the following:
	`dns.example.com. IN DS 13447 5 1`
	`A5AD9EFB6840F58CF817F3CC7C24A7ED2DD5559C`

config global-dns-server general

Use this command to configure basic behavior for the DNS server.

The general settings configuration specifies the interfaces that listen for DNS requests. By default, the system listens on the IPv4 and IPv6 addresses of all configured interfaces for DNS requests.

The other settings in the general settings configuration are applied when traffic does not match a Global DNS policy.

From general settings, you can also enable DNS over HTTP/HTTPS (DoH) and DNS over TLS (DoT) to encrypt the DNS query.

Before you begin:

- You must have a good understanding of DNS and knowledge of the DNS deployment in your network.
- You must have read-write permission for global load balancing settings.
- If enabling DNS over HTTPS/TLS, you must have prepared a dedicated DNS server domain and a certificate pair for your DNS over HTTPS/TLS service. For details, see the FortiADC Handbook topic on Configuring DNS over HTTPS and DNS over TLS.

Syntax

```
config global-dns-server general
   set dnssec-validate-status {enable|disable}
   set forward {first|only}
   set forwarders <datasource>
   set gds-status {enable|disable}
   set minimal-responses {enable|disable}
   set ipv4-accessed-status {enable|disable}
   set ipv6-accessed-status {enable|disable}
   set listen-on-all-interface {enable|disable}
   set listen-on-interface <datasource>
   set dns-over-https {enable|disable}
   set dns-over-https-port <integer>
   set dns-over-https-listen-on-interface <datasource>
   set dns-over-http {enable|disable}
   set dns-over-http-port <integer>
   set dns-over-http-listen-on-interface <datasource>
   set dns-over-tls {enable|disable}
   set dns-over-tls-port <integer>
   set dns-over-tls-listen-on-interface <datasource>
   set certificate <datasource>
   set recursion-status {enable|disable}
   set response-rate-limit <datasource>
   set traffic-log {enable|disable}
   set use-system-dns-server {enable|disable}
end
```

dnssec-validate-status	Enable/disable DNSSEC validation.
forward	• first—The DNS server queries the forwarder before doing its own DNS lookup. • only—Only queries the forwarder. Does not perform its own DNS lookups.
forwarders	If the DNS server zone has been configured as a forwarder, specify the remote DNS server to which it forwards requests.
gds-status	Enable/disable the DNS server configuration.
minimal-responses	Enables/disables Minimal Responses to hide the **Authority Section** and **Additional Section** of DNS queries. When the DNS query only shows the minimal response, it can significantly increase the performance of the FortiADC DNS service by increasing the QPS.
ipv4-accessed-status	Enable/disable listening for DNS requests on the interface IPv4 address.
ipv6-accessed-status	Enable/disable listening for DNS requests on the interface IPv6 address.
listen-on-all-interface	Enable listening on all interfaces.
listen-on-interface	The **listen-on-interface** option is available if **listen-on-all-interface** is disabled. If you do not listen on all interfaces, select one or more ports to listen on.
dns-over-https	Enable/disable DNS over HTTPS to encrypt DNS queries using the HTTPS protocol.
dns-over-https-port	The **dns-over-https-port** option is available if **dns-over-https** is enabled. Specify the port to listen on DNS over HTTPS. Default: 443 Range: 1-65535.
dns-over-https-listen-on-interface	The **dns-over-https-listen-on-interface** option is available if **dns-over-https** is enabled. Specify the interface(s) to listen on for DNS over HTTPS.
dns-over-http	Enable/disable DNS over HTTP to encrypt DNS queries using the HTTP protocol.
dns-over-http-port	The **dns-over-http-port** option is available if **dns-over-http** is enabled. Specify the port to listen on DNS over HTTP. Default: 80 Range: 1-65535.
dns-over-http-listen-on-interface	The **dns-over-http-listen-on-interface** option is available if **dns-over-http** is enabled. Specify the interface(s) to listen on for DNS over HTTP.
dns-over-tls	Enable/disable DNS over TLS to encrypt DNS queries using the TLS protocol.
dns-over-tls-port	The **dns-over-tls-port** option is available if **dns-over-tls** is enabled. Specify the port to listen on DNS over TLS. Default: 853 Range: 1-65535.
dns-over-tls-listen-on-interface	The **dns-over-tls-listen-on-interface** option is available if **dns-over-tls** is enabled. Specify the interface(s) to listen on for DNS queries for DNS over TLS.
certificate	The **certificate** option is available if **dns-over-https** or **dns-over-tls** is enabled.

	Specify the certificate object to apply for DNS over HTTPS or DNS over TLS. This certificate must refer to the DNS server domain or IP address. For details, see the FortiADC Handbook topic on Configuring DNS over HTTPS and DNS over TLS.
`recursion-status`	Enable/disable recursion. If enabled, the DNS server attempts to do all the work required to answer the query. If not enabled, the server returns a referral response when it does not already know the answer.
`response-rate-limit`	Specify a rate limit configuration object.
`traffic-log`	Enable/disable logging.
`use-system-dns-server`	Forward DNS requests to the system DNS server instead of the forwarder.

Example

```
FortiADC-VM # config global-dns-server general

FortiADC-VM (general) # get
gds-status : disable
minimal-responses : disable
recursion-status : enable
dnssec-status : disable
dnssec-validate-status : disable
ipv6-accessed-status : enable
ipv4-accessed-status : enable
traffic-log : disable
listen-on-all-interface : enable
forward : first
use-system-dns-server : enable
response-rate-limit :
dns-over-https : enable
dns-over-https-port : 443
dns-over-https-listen-on-interface : port2 port3
dns-over-http : enable
dns-over-http-port : 80
dns-over-http-listen-on-interface : port2 port3
dns-over-tls : enable
dns-over-tls-port : 853
dns-over-tls-listen-on-interface : port2 port3
certificate : dns_fortiadc-qa_com

FortiADC-VM (general) # set gds-status enable
FortiADC-VM (general) # end
```

config global-dns-server policy

Use this command to configure a rulebase that matches traffic to DNS zones.

Traffic that matches both source and destination criteria is served by the policy. Traffic that does not match any policy is served by the DNS “general settings” configuration.

Before you begin:

- You must have a good understanding of DNS and knowledge of the DNS deployment in your network.
- You must have configured address objects, remote servers, DNS zones, and optional configuration objects you want to specify in your policy.
- You must have read-write permission for global load balancing settings.

Syntax

```
config global-dns-server policy
   edit <name>
      set destination-address <datasource>
      set dns64-list {<datasource> ...}
      set dnssec-validate-status {enable|disable}
      set forward {first | only}
      set forwarders <datasource>
      set recursion-status {enable|disable}
      set response-rate-limit <datasource>
      set source-address <datasource>
      set zone-list {<datasource> ...}
   next
end
```

`destination-address`	Address object to specify the destination match criteria.
`dns64-list`	Specify one or more DNS64 configurations to use when resolving IPv6 requests.
`dnssec-validate-status`	Enable/disable DNSSEC validation.
`forward`	• first—The DNS server queries the forwarder before doing its own DNS lookup. • only—Only queries the forwarder. Does not perform its own DNS lookups.
`forwarders`	If the DNS server zone has been configured as a forwarder, specify the remote DNS servers to which it forwards requests.
`recursion-status`	Enable/disable recursion. If enabled, the DNS server attempts to do all the work required to answer the query. If not enabled, the server returns a referral response when it does not already know the answer.
`response-rate-limit`	Specify a rate limit configuration object.
`source-address`	Address object to specify the source match criteria.
`zone-list`	Specify one or more zone configurations to serve DNS requests from matching traffic.

Example

```
FortiADC-VM (policy) # edit lan_policy
Add new entry 'lan_policy' for node 2236
```

```
FortiADC-VM (lan_policy) # get
source-address :
destination-address :
zone-list :
dns64-list :
recursion-status : enable
dnssec-status : disable
dnssec-validate-status: disable
forward : first
forwarders :
response-rate-limit :

FortiADC-VM (lan_policy) # set source-address campus
FortiADC-VM (lan_policy) # set destination-address any
FortiADC-VM (lan_policy) # set zone-list lan-zone
FortiADC-VM (lan_policy) # next

FortiADC-VM (policy) # edit wan_policy
Add new entry 'wan_policy' for node 2236

FortiADC-VM (wan_policy) # set source-address branch
FortiADC-VM (wan_policy) # set destination-address any
FortiADC-VM (wan_policy) # set zone-list wan-zone
FortiADC-VM (wan_policy) # end

FortiADC-VM # get global-dns-server policy lan_policy
source-address : campus
destination-address : any
zone-list : lan-zone
dns64-list :
recursion-status : enable
dnssec-status : disable
dnssec-validate-status: disable
forward : first
forwarders :
response-rate-limit :

FortiADC-VM # get global-dns-server policy wan_policy
source-address : branch
destination-address : any
zone-list : wan-zone
dns64-list :
recursion-status : enable
dnssec-status : disable
dnssec-validate-status: disable
forward : first
forwarders :
response-rate-limit :
```

config global-dns-server remote-dns-server

Use this command to create a list of DNS forwarders.

DNS forwarders are commonly used when you do not want the local DNS server to connect to Internet DNS servers. For example, if the local DNS server is behind a firewall and you do not want to allow DNS through that firewall, you implement DNS forwarding to a remote server that is deployed in a DMZ or similar network region that can contact Internet DNS servers.

Before you begin:

- You must have a good understanding of DNS and knowledge of the remote DNS servers that can be used to communicate with Internet domain servers.
- You must have read-write permission for global load balancing settings.

After you have configured a remote DNS server, you can select it in the DNS zone and DNS policy configurations.

Syntax

```
config global-dns-server remote-dns-server
   edit <name>
      config member
         edit <No.>
            set addr-type {ipv4|ipv6}
            set ip <class_ip>
            set ip6 <class_ip>
            set port <integer>
         next
      end
   next
end
```

addr-type	IPv4 or IPv6
ip	IP address of the remote DNS server.
ip6	IP address of the remote DNS server.
port	Port number the remote server uses for DNS. The default is 53.

Example

```
FortiADC-VM # config global-dns-server remote-dns-server
FortiADC-VM (remote-dns-ser~e) # edit google.com
Add new entry 'google.com' for node 2329

FortiADC-VM (google.com) # config member
FortiADC-VM (member) # edit 1
Add new entry '1' for node 2331

FortiADC-VM (1) # get
addr-type : ipv4
ip : 0.0.0.0
```

```
port : 53

FortiADC-VM (1) # set ip 8.8.8.8

FortiADC-VM (1) # get
addr-type : ipv4
ip : 8.8.8.8
port : 53
FortiADC-VM (1) # end
FortiADC-VM (google.com) # end
```

config global-dns-server response-rate-limit

Use this command to configure response rate limit objects that you specify in the DNS policy and DNS general configurations.

The response rate limit keeps the FortiADC authoritative DNS server from being used in amplifying reflection denial of service (DoS) attacks.

Before you begin:

- You must have a good understanding of DNS.
- You must have read-write permission for global load balancing settings.

After you have created a response rate limit configuration, you can select it in the DNS policy and DNS general settings configurations.

Syntax

```
config global-dns-server response-rate-limit
   edit <name>
      set per-second <integer>
   next
end
```

per-second	Maximum number of responses per second. The valid range is 1-2040. The default is 1000.

Example

```
FortiADC-VM # config global-dns-server response-rate-limit
FortiADC-VM (response-rate-~i) # edit gdns-rl-1
Add new entry 'gdns-rl-1' for node 2313
FortiADC-VM (gdns-rl-1) # end

FortiADC-VM # get global-dns-server response-rate-limit gdns-rl-1
per-second : 1000
```

config global-dns-server trust-anchor-key

Use this command to change the trust anchor key (if necessary).

DNSSEC validation requires that a DNS name server know the trust anchor key for the root DNS domain in order to validate already signed responses. In general, trust anchor keys do not change often, but they do change occasionally, and might change unexpectedly in the event the keys are compromised.

The FortiADC DNS server is preconfigured with a trust anchor key for the root DNS domain. If you are informed that you must update this key, you can use the configuration editor to paste the new content into the DNS server configuration.

Further reading:

http://data.iana.org/root-anchors/draft-icann-dnssec-trust-anchor.html

Before you begin:

- You must have a good understanding of DNSSEC and knowledge of the DNS deployment in your network.
- You must have already obtained the key so that you can copy and paste it into the DNS server configuration.
- You must have read-write permission for global load balancing settings.

Syntax

```
config global-dns-server trust-anchor-key
   edit <name>
      set value <string>
      set description <string>
   next
end
```

value	The key value. The key format is a string with the following format: \"<domainname>\" <num1> <num2> <num3> \"<content>\" The following is an example: \".\" 256 3 5 \"AwEAAbDrWmiIReotvZ6FObgKygZwUxSUJW9z5pjiQMLH0JBGXooHrR16 pdKhI9mNkM8bLUMtwYfgeUOYXIvfagee8rk=\"
description	A description of this configuration.

Example

```
FortiADC-VM # config global-dns-server trust-anchor-key
FortiADC-VM (trust-anchor-key) # edit sss
Add new entry 'sss' for node 2240

FortiADC-VM (sss) # get
value :
description :

FortiADC-VM (sss) # set
*value key value
description key description
```

```
FortiADC-VM (sss) # set value "\".\" 256 3 5
     \"AwEAAbDrWmiIReotvZ6FObgKygZwUxSUJW9z5pjiQMLH0JBGXooHrR16
     pdKhI9mNkM8bLUMtwYfgeUOYXIvfagee8rk=\""
FortiADC-VM (sss) # end
```

config global-dns-server zone

Use this command to configure DNS zone and resource records.

The DNS zone configuration is the key to the global load balancing solution. This configuration contains the key DNS server settings, including:

- Domain name and name server details.
- Type—Whether the server is the primary or a forwarder.
- DNSSEC—Whether to use DNSSEC and the DNSSEC algorithm/key size.
- DNS RR records—The zone configuration contains resource records (RR) used to resolve DNS queries delegated to the domain by the parent zone.

You can specify different DNS server settings for each zone you create. For example, the DNS server can be a primary for one zone and a forwarder for another zone.

Before you begin:

- You must have a good understanding of DNS and knowledge of the DNS deployment in your network.
- You must have authority to create authoritative DNS zone records for your network.
- You must have read-write permission for global load balancing settings.

After you have configured a DNS zone, you can select it in the DNS policy configuration.

Syntax

```
config global-dns-server zone
   edit <name>
      set type {forward|fqdn-generate|primary}
      set domain-name <string>
      set negative-ttl <integer>
      set primary-server-ip <class_ip>
      set primary-server-ip6 <class_ip>
      set primary-server-name <string>
      set responsible-mail <string>
      set ttl <integer>
      set forward {first | only}
      set forwarders <datasource>
      set dnssec-status {enable|disable}
      set dnssec-algorithm
           {ECDSAP256SHA256|ECDSAP384SHA384|NSEC3RSASHA1|RSASHA1|RSASHA256|RSASHA512}
      set dnssec-keysize {1024|2048|4096}
      set dsset-info <string>
      set dssetinfo-filename <string>
```

```
set dsset-info-list <datasource>
set KSK <string>
set KSK-Filename <string>
set ZSK <string>
set ZSK-Filename <string>
config a-aaaa-record
   edit <No.>
      set hostname <string>
      set source-type {ipv4 | ipv6}
      set ip <class_ip>
      set ip6 <class_ip>
      set method wrr
      set weight <integer>
   next
end
config cname-record
   edit <No.>
      set alias <string>
      set target <string>
   next
end
config mx-record
   edit <No.>
      set domain-name <string>
      set hostname <string>
      set type {ipv4|ipv6}
      set ip <class_ip>
      set ip6 <class_ip>
      set priority <integer>
   next
end
config ns-record
   edit <No.>
      set domain-name <string>
      set host-name <string>
      set type {ipv4|ipv6}
      set ip <class_ip>
      set ip6 <class_ip>
   next
end
config txt-record
   edit <No.>
      set name <string>
      set text <name>=<value>,<name>=<value>
   next
end
config srv-record
   edit 1
      set hostname 222
      set target-server 222
   next
end
config ptr-record
   edit <No.>
      set ptr-address <string>
      set fqdn <string>
   next
```

```
    end
  next
end
```

config global-dns-server zone	
type	• forward—The configuration allows you to apply DNS forwarding on a per-domain basis, overriding the forwarding settings in the “general” configuration. • fqdn-generate—The configuration has been generated by the global load balancing feature set. You cannot configure this type manually. • primary—The configuration contains the “primary” copy of data for the zone and is the authoritative server for it.
domain-name	The domain name must end with a period. For example: `example.com`.
negative-ttl	The last field in the SOA—the negative caching TTL. This informs other servers how long to cache no-such-domain (NXDOMAIN) responses from you. The default is 3600 seconds. The valid range is 0 to 2,147,483,647.
primary-server-ip	IP address of the primary server.
primary-server-ip6	IP address of the primary server.
primary-server-name	Sets the server name in the SOA record.
responsible-mail	Username of the person responsible for this zone, such as `root`.
ttl	The $TTL directive at the top of the zone file (before the SOA) gives a default TTL for every RR without a specific TTL set. The default is 86,400. The valid range is 1 to 2,147,483,647.
forward	• first—The DNS server queries the forwarder before doing its own DNS lookup. • only—Only query the forwarder. Do not perform a DNS lookup.
forwarders	Specify a remote server configuration object.
dnssec-status	Enable/disable DNSSEC. The Domain Name System Security Extensions (DNSSEC) is a feature of the Domain Name System (DNS) that authenticates responses to domain name lookups.
dnssec-algorithm	The **dnssec-algorithm** option is available if **dnssec-status** is enabled. Select the cryptographic algorithm to use for authenticating DNSSEC. • ECDSAP256SHA256 • ECDSAP384SHA384 • NSEC3RSASHA1 • RSASHA1 • RSASHA256 • RSASHA512
dnssec-keysize	The **dnssec-keysize** option is available if **dnssec-status** is enabled. Select the key size (number of bits) for the encryption algorithm. • 1024 bits

	• 2048 bits • 4096 bits **Note**: Prior to FortiADC 7.4.0, the DNSSEC key size only supported 512 bits, so configurations carried over from previous versions can continue using the 512-bit key. However, we recommend updating to the new 1024/2048/4096 bit keys as the 512-bit key is less secure and is no longer supported in the latest BIND 9 version.
`dsset-info`	It is generated by the system if DNSSEC is enabled for the zone.
`dssetinfo-filename`	The file is generated by the system if DNSSEC is enabled for the zone. The file generated by the zone configuration editor is the one you give to any parent zone or the registrar of your domain. The convention is dsset-<domain>, for example `dsset-example.com`.
`dsset-info-list`	Specify a DSSET info list configuration object.
`KSK`	Type characters for a string key. The file is generated by the system if DNSSEC is enabled for the zone.
`KSK-Filename`	The file is generated by the system if DNSSEC is enabled for the zone. To regenerate the KSK, disable DNSSEC and then re-enable DNSSEC.
`ZSK`	Type characters for a string key. The file is generated by the system if DNSSEC is enabled for the zone.
`ZSK-Filename`	The file is generated by the system if DNSSEC is enabled for the zone. To regenerate the ZSK, disable DNSSEC and then re-enable DNSSEC.
config a-aaaa-record	
`hostname`	The hostname part of the FQDN, such as `www`. **Note**: • You can specify the @ symbol to denote the zone root. The value substituted for @ is the preceding $ORIGIN directive. • A hostname can contain alphanumeric characters such as a–z, A–Z, and 0–9, but must NOT end with - (hyphen) or . (period). • You can also use * (wild card) in a domain name.
`source-type`	IPv4 or IPv6
`ip`	IP address of the virtual server.
`ip6`	IP address of the virtual server.
`method`	Weighted Round Robin is the only method supported.
`weight`	Assigns relative preference among members—higher values are more preferred and are assigned connections more frequently. The default is 1. The valid range is 1-255.
config cname-record	
`alias`	An alias name to another true or canonical domain name (the target). For instance, `www.example.com` is an alias for example.com.

target	The true or canonical domain name. For instance, example.com.
config mx-record	
hostname	The hostname part of the FQDN for a mail exchange server, such as mail.
type	IPv4 or IPv6
ip	IP address of the mail server.
ip6	IP address of the mail server.
priority	Preference given to this RR among others at the same owner. Lower values have greater priority.
config ns-record	
domain-name	The domain for which the name server has authoritative answers, such as example.com.
host-name	The hostname part of the FQDN, such as ns.
type	IPv4 or IPv6
ip	IP address of the name server.
ip6	IP address of the name server.
config txt-record	
name	Hostname. TXT records are name-value pairs that contain human readable information about a host. The most common use for TXT records is to store SPF records.
text	Comma-separated list of name=value pairs. An example SPF record has the following form: "v=spf1 +mx a:colo.example.com/28 -all" If you complete the entry from the CLI, put the string in quotes. (If you complete the entry from the the Web UI, you do not put the string in quotes.)
config srv-record	
hostname	The SRV Hostname.
target-server	The target server name (record).
config ptr-record	
PTR Address	A PTR address, such as 10.168.192.in-addr.arpa. or 1. If you use the number, the domain name is in the format "x.x.x.in-addr.arpa.".
FQDN	A fully qualified domain name, such as "www.example.com".

Example

```
FortiADC-VM # config global-dns-server zone
FortiADC-VM (zone) # edit wan-zone
```

```
Add new entry 'wan-zone' for node 2248

FortiADC-VM (wan-zone) # get
type : primary
domain-name :
dnssec-status : disable
ttl : 86400
responsible-mail :
negative-ttl : 3600
primary-server-name :
primary-server-ip : 0.0.0.0
primary-server-ip6 : ::

FortiADC-VM (wan-zone) # set domain-name www.fortiadc.com.
FortiADC-VM (wan-zone) # set responsible-mail root
FortiADC-VM (wan-zone) # set primary-server-name ns
FortiADC-VM (wan-zone) # set primary-server-ip 202.33.11.107

FortiADC-VM (wan-zone) # config a-aaaa-record
FortiADC-VM (a-aaaa-record) # edit 1
Add new entry '1' for node 2257
FortiADC-VM (1) # set hostname www
FortiADC-VM (1) # get
hostname : www
source-type : ipv4
weight : 1
ip : 0.0.0.0
method : wrr
FortiADC-VM (1) # set hostname www
FortiADC-VM (1) # set ip 202.33.11.1
FortiADC-VM (1) # end
FortiADC-VM (wan-zone) # end
```

config global-load-balance

The `config global-load-balance` commands configure the global load balancing feature settings. You configure global load balancing settings on the FortiADC instance that hosts the DNS server that is used for global load balancing. You do not configure these settings on the local FortiADC instances that are load balanced.

This chapter is a reference for the following commands:

config global load balance analytic

Creates a dynamic chart visible in Fortiview that shows the status of the data-center.

Syntax

```
config global-load-balance analytic
   edit <name>
      set type <data-center/link/server/virtual-server-pool>
      set data-center/link/server/virtual-server-pool <name>
      set range <1DAY/1HOUR/1MONTH/1WEEK/1YEAR/6HOURS>
   next
end
```

`<name>`	Configuration name. No spaces or special characters. You reference this name in the global load balance servers configuration.
`type`	Select type of analytic widget.
`range`	Set range to show activity of the data center.

Example

```
FortiADC-VM # config global-load-balance analytic
FortiADC-VM (analytic) edit 1
Add new entry '1'
```

```
FortiADC-VM (1) # get
type :
data-center :
range :
FortiADC-VM (1) # set type data-center
FortiADC-VM (1) # set data-center name1
FortiADC-VM (1) # set range 1DAY
next
end
```

config global-load-balance data-center

Use this command to create data center configurations that you associate with the server configurations for local FortiADCs. The data center configuration sets key properties: Location and/or ISP and ISP province. These properties are keys in the global load balancing algorithm that selects the FortiADC in closest proximity to the client.

The system includes the FortiGuard geolocation database and predefined ISP address books that you use in the configuration.

Before you begin:

- If you want to specify a user-defined ISP address book, you must create it before using this command.
- You must have read-write permission for global load balancing settings.

After you have created a data center configuration object, you can specify it in the global load balance servers configuration.

Syntax

```
config global-load-balance data-center
   edit <name>
      set location <datasource>
      set description <string>
   next
end
```

`<name>`	Configuration name. No spaces or special characters. You reference this name in the global load balance servers configuration.
`description`	Optional description to help administrators know the purpose or usage of the configuration.
`location`	Specify a location from the geolocation list. Note: Starting from FortiADC 5.0.0, you are required to the country or city code instead of the full name of the country or city.

Example

```
FortiADC-VM # config global-load-balance data-center
FortiADC-VM (data-center) # edit dc-china
Add new entry 'dc1' for node 2836
FortiADC-VM (dc1) # get
location :
description :
FortiADC-VM (dc1) # set location CN
FortiADC-VM (dc1) # end
```

config global-load-balance link

A link can be an access point of an ISP, and you can specify the data-center and the ISP in the link configuration. For the gateway in `config gateway`, you can specify the LLB gateway of each o f the SLB devices which are related to this link. A global load-balancing device can find out the status of the LLB link to this link according to the gateway configuration. At the same time, the RTT detection result of the same link could be shared.

Syntax

```
config global-load-balance link
   edit "link"
      set data-center <data-center name>
      set isp <isp name>
      set isp-province <isp province name>
      config gateway
         edit 1
            set server <server name>
            set gateway-name <gateway>
         next
         edit 2
            set server <server>
            set gateway-name <gateway name>
         next
      end
   next
end
```

`data-center`	Specify a data center for the link.
`isp`	Specify an ISP for the link.
`isp-province`	Specify a province in the selected ISP for the link.
`server`	Specify a global load-balancing server for the link.
`gateway-name`	Specify a gateway for the link.

Example

```
config global-load-balance link
   edit "link1"
      set data-center dc1
      set isp china-mobile
      set isp-province Henan
      config gateway
         edit 1
            set server slb48
            set gateway-name gw_81
         next
         edit 2
            set server slb48_dris
            set gateway-name gw81_dris
         next
      end
   next
end
```

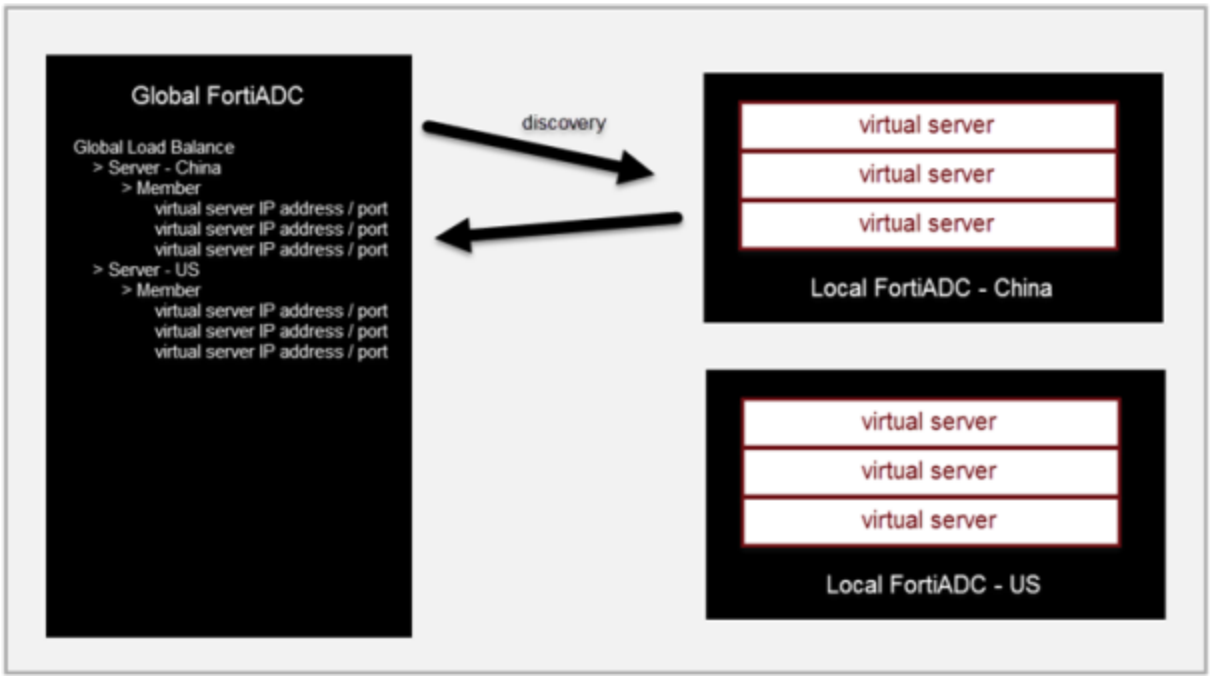
config global-load-balance servers

Use this command to configure global load balance servers.

In the context of the global server load balance configuration, servers are the local SLB (FortiADC instances or third-party servers) that are to be load balanced. For FortiADC instances, the GLB checks status and synchronizes configuration from the local SLB so that it can learn the set of virtual servers that are possible to include in the GLB virtual server pool.

Figure 1 illustrates configuration discovery. You use the execute discovery-glb-virtual-server command to populate the virtual-server-list configuration. Placement in this list does not include them in the pool. You also must name them explicitly in the virtual server pool configuration.

Figure 1: Virtual server discovery



Before you begin:

- You must have created the data center configuration objects that are associated with the local SLB.
- You must have created virtual server configurations on the local FortiADC SLB so that you can use execute discovery-glb-virtual-server command to discover them.
- You must have created an SDN connector configuration.
  Note: Currently, the SDN Connector option only supports AWS Connectors
- You must have read-write permission for global load balancing settings.

After you have created a server configuration object, you can specify it the global load balancing virtual server pool configuration.

Syntax

```
config global-load-balance servers
   edit <name>
      set server-type {FortiADC-SLB|Generic-Host|SDN-Connector}
      set auth-type {none|TCP_MD5SIG|auth_verify}
      set auth-key <string>
      set user-defined-certificate {enable|disable}
      set cert <datasource>
      set address-type {ipv4|ipv6}
      set ip <ip&netmask>
      set ip6 <ipv6&netmask>
      set port <integer>
      set sdn-connector <datasource>
      set use-sdn-private-ip {enable|disable}
      set data-center <datasource>
      set auto-sync {enable|disable}
      set health-check-ctrl {enable|disable}
```

```
      set health-check-relation {AND|OR}
      set health-check-list <datasource> <datasource> ...
      config virtual-server-list
         edit <name>
            set address-type {ipv4|ipv6}
            set ip <ip&netmask>
            set ip6 <ipv6&netmask>
            set gateway <string>
            set instance <datasource>
            set health-check-inherit {enable|disable}
            set health-check-ctrl {enable|disable}
            set health-check-list <datasource> <datasource> ...
            set health-check-relation {AND|OR}
         next
      end
   next
end
```

server-type	Select the remote server to use for global server load balancing: • FortiADC-SLB — use a FortiADC instance. • Generic-Host — use a third party ADC or server. • SDN-Connector — use an existing external connector that is connected to the FortiADC Security Fabric. **Note**: Currently, the SDN Connector option only supports AWS Connectors.
auth-type	The **auth-type** option is available if **server-type** is **FortiADC-SLB**. Select the authentication type: • none — No password. • TCP_MD5SIG — With password, but cannot be used if NAT is in between the client and server. This is because, when using the TCP MD5SIG authentication in a network with NAT in between, the IP layer is encrypted. So is every packet. Because the IP address will be changed, the encryption check will always fail. • auth_verify — The authentication key is sent to the server after a three-way handshake. The key is encrypted and NAT in between will not affect the authentication.
auth-key	The **auth-key** option is available if **server-type** is **FortiADC-SLB** and **auth-type** is **TCP_MD5SIG**. Enter the password to authenticate the key. The password you enter here must match the password configured on the FortiADC appliance in a global sever load-balancing configuration.
user-defined-certificate	The **user-defined-certificate** option is available if **Type** is **FortiADC SLB**. Enable to use a self-defined certificate for authentication.
cert	The **cert** option is available if **server-type** is **FortiADC-SLB** and **user-defined-certificate** is **enabled**. Select the local certificate object to use for the GSLB server.
address-type	The **address-type** option is available if **server-type** is **FortiADC-SLB**. IPv4 or IPv6.

ip/ip6	The **ip** or **ip6** option is available if **server-type** is **FortiADC-SLB**. Specify the IPv4 or IPv6 address for the FortiADC management interface. This IP address is used for synchronization and also status checks. If the management interface is unreachable, the virtual servers for that FortiADC are excluded from DNS responses.
port	The **port** option is available if **server-type** is **FortiADC-SLB**. Specify the port. Default: 5858 Range: 1-65535.
sdn-connector	The **sdn-connector** option is available if **server-type** is **SDN-Connector**. Select the SDN Connector to synchronize to the GSLB server. For public SDN type servers, GSLB can update the public IP dynamically. **Note**: Currently, only AWS connectors are supported.
use-sdn-private-ip	The **use-sdn-private-ip** option is available if **server-type** is **SDN-Connector**. Enable to use the SDN Private IP address.
data-center	Select a data center configuration object. The data center configuration object properties are used to establish the proximity of the servers and the client requests.
auto-sync	Enable/disable automatic synchronization with the remote server. When enabled, Global load balancing will synchronize automatically with the server member. If auto-sync is enabled for SDN Connector type servers, all instances from the SDN connector will be added as server members. **Note**: When disabling auto-sync, the server member will be cleared and re-synced.
health-check-ctrl	The **health-check-ctrl** option is available if **server-type** is **Generic-Host** or **SDN-Connector**. Enable/disable health checks for the virtual server list. The health check settings at this configuration level are the parent configuration. When you configure the list, you can specify whether to inherit or override the parent configuration. **Note**: Health checking is built-in, and you can optionally configure a gateway health check.
health-check-relation	The **health-check-relation** option is available if **server-type** is **Generic-Host** or **SDN-Connector**, and **health-check-ctrl** is **enabled**. • AND—All of the specified health checks must pass for the server to be considered available. • OR—One of the specified health checks must pass for the server to be considered available.
health-check-list	The **health-check-list** option is available if **server-type** is **Generic-Host** or **SDN-Connector**, and **health-check-ctrl** is **enabled**. Select one or more health check configuration objects.
config virtual-server-list	
When servers are FortiADC servers, use execute discovery-glb-virtual-server to populate the basic virtual-server-list configuration. After it has been populated, you can add a gateway health check. (optional).	

<name>	Must match the virtual server configuration name on the local FortiADC.
address-type	The **address-type** option is available if **server-type** is **FortiADC-SLB**. IPv4 or IPv6.
ip/ip6	The **ip** or **ip6** option is available if **server-type** is **FortiADC-SLB**. Virtual server IPv4 or IPv6 address.
gateway	The **gateway** option is available if **server-type** is **FortiADC-SLB**. Specify a gateway to enable an additional health check: is the gateway beyond the FortiADC reachable? Specify a string that matches the configuration name of a link load balancing gateway.
instance	The **instance** option is available if **server-type** is **SDN-Connector**. Select an instance from the SDN's instance list.
health-check-inherit	The **health-check-inherit** is available if **server-type** is **Generic-Host** or **SDN-Connector**. Enable to inherit the health check settings from the parent configuration. The Health Check Inherit option is enabled by default. Disable to specify health check settings in this member configuration.
health-check-ctrl	The **health-check-ctrl** is available if **server-type** is **Generic-Host** or **SDN-Connector**, and **health-check-inherit** is **disabled**. Enable health checking for the virtual server.
health-check-list	The **health-check-list** is available if **server-type** is **Generic-Host** or **SDN-Connector**, and **health-check-inherit** is **disabled**. Specify one or more health check configuration objects.
health-check-relation	The **health-check-relation** is available if **server-type** is **Generic-Host** or **SDN-Connector**, and **health-check-inherit** is **disabled**. • AND—All of the selected health checks must pass for the server to the considered available. • OR—One of the selected health checks must pass for the server to be considered available.

Example

```
FortiADC-VM # config global-load-balance servers

FortiADC-VM (servers) # edit FortiADC-2

FortiADC-VM (FortiADC-2) # set sync-status enable
FortiADC-VM (FortiADC-2) # auth-type TCP_MDFSIG
FortiADC-VM (FortiADC-2) # set auth-key
     ENC QVhOH9Wvq6q4BP2sqQMNJ6FDWWYcZA6THCj/sHFGHtAb6qO5nqy1SJ9PpEpc+yk/j8XWfXeORT8DsF8KDB
     hDL9K5Ms9sXs1y8gUQbtFnCIHKwIpf
FortiADC-VM (FortiADC-2) # set data-center United_States
FortiADC-VM (FortiADC-2) # set auto-sync enable
FortiADC-VM (FortiADC-2) # set ip 172.30.144.100
FortiADC-VM (FortiADC-2) # set server-type FortiADC-SLB
```

```
FortiADC-VM (FortiADC-2) # show
config global-load-balance servers
edit "FortiADC-2"
set ip 172.30.144.100
set data-center United_States
config virtual-server-list
end
next
end

FortiADC-VM (FortiADC-2) # end

FortiADC-VM # execute discovery-glb-virtual-server server FortiADC-2

FortiADC-VM # show global-load-balance servers FortiADC-2
config global-load-balance servers
edit "FortiADC-2"
set ip 172.30.144.100
set data-center United_States
config virtual-server-list
edit "mail_example_com"
set ip 192.0.2.2
set port 80
next
edit "www_example_com"
set ip 192.0.2.1
set port 811
next
end
next
end

FortiADC-VM # config global-load-balance servers

FortiADC-VM (servers) # edit FortiADC-2

FortiADC-VM (FortiADC-2) # config virtual-server-list

FortiADC-VM (virtual-server~l) # show
config virtual-server-list
edit "mail_example_com"
set ip 192.0.2.2
set port 80
next
edit "www_example_com"
set ip 192.0.2.1
set port 811
next
end

FortiADC-VM (virtual-server~l) # edit www_example_com
FortiADC-VM (www_example_com) # set gateway US-ISP1
FortiADC-VM (www_example_com) # end

FortiADC-VM (FortiADC-2) # end
```

config global-load-balance setting

Use this command to configure the following:

- Listen on interface/port on page 101
- Persistence on page 102
- Proximity on page 103
- Authentication on page 104

Listen on interface/port

The listen port means the port used for communication for GLB and SLB server. The GLB listen port is 5858 by default. User can change to other port from 1 to 65535.

Before you begin:

- You must have read-write permission for global load balancing settings.

Syntax

```
config global-load-balance setting
   set listen-on-all-interfaces {enable|disable}
   set ipv4-accessed-status {enable|disable}
   set ipv6-accessed-status {enable|disable}
   set port <integer>
end
```

`listen-on-all-interfaces`	Enable/disable IPv4/IPv6 network accessed status on all interfaces.
`port`	listen-on-port
`ipv4-accessed-status`	Enable/disable listening for DNS requests on the interface IPv4 address.
`ipv6-accessed-status`	Enable/disable listening for DNS requests on the interface IPv6 address.

Example

```
FortiADC-VM (setting) # get
ipv6-accessed-status: enable
ipv4-accessed-status: enable
listen-on-all-interface : enable
port : 5858
FortiADC-VM (setting) # set port 5858
FortiADC-VM (setting) # end
```

Persistence

Use this command to configure source address affinity and a timeout for GSLB persistence. You enable persistence per host in the GSLB host configuration.

If the DNS query is for a host that has persistence enabled, the DNS server replies with a response that has the virtual server IP addresses listed in the order determined by the GSLB proximity algorithms, and the client source IP address (for example 192.168.1.100) is recorded in the persistence table. If source address affinity is set to 24 bits, subsequent queries for the host from the 192.168.1.0/24 network are sent an answer with the virtual servers listed in the same order (unless a server becomes unavailable and is therefore omitted from the answer).

Persistence is required for applications that include transactions across multiple hosts, so the persistence table is also used for queries for other hosts with the same domain. For example, a transaction on a banking application might include connections to login.bank.com and transfer.bank.com. To support persistence in these cases, the GSLB persistence lookup accounts for domain as well. The first query for login.bank.com creates a mapping for the source address network 192.168.1.0/24 and the domain bank.com. When the DNS server receives subsequent requests, it consults the persistence table for a source network match, then a domain match and a hostname match. In this example, as long as you have created host configurations for both login.bank.com and transfer.bank.com, and persistence is enabled for each, the persistence table can be used to ensure the DNS answers to queries from the same network list the resource records in the same order.

Before you begin:

- You must have read-write permission for global load balancing settings.

Syntax

```
config global-load-balance setting
   set persistence-mask-length <integer>
   set persistence-mask-length6 <integer>
   set persistence-timeout <integer>
end
```

`persistence-mask-length`	Number of IPv4 netmask bits that define network affinity for the persistence table. The default is 24.
`persistence-mask-length6`	Number of IPv6 netmask bits that define network affinity for the persistence table. The default is 64.
`persistence-timeout`	This setting specifies the length of time in seconds for which the entry is maintained in the persistence table. The default is 86400. The valid range is 60-2,592,000 seconds.

Example

```
FortiADC-docs # config global-load-balance setting
FortiADC-docs (setting) # get
password : *
proximity-detect-protocol : icmp
proximity-detect-retry-count : 3
proximity-cache-mask-length : 24
proximity-cache-mask-length6 : 64
proximity-detect-interval : 3
proximity-cache-aging-period : 86400
persistence-mask-length : 24
persistence-mask-length6 : 64
persistence-timeout : 60
FortiADC-docs (setting) # set persistence-mask-length 24
FortiADC-docs (setting) # set persistence-mask-length6 64
FortiADC-docs (setting) # set persistence-timeout 60
FortiADC-docs (setting) # end
```

Proximity

Use this command to configure dynamic proximity. Dynamic proximity is used to order DNS lookup results based on the shortest application response time (RTT) for ICMP or TCP probes sent by the local SLB to the DNS resolver that sent the DNS request.

The system caches the RTT results for the period specified by the timeout. When there are subsequent requests from clients that have a source IP address within the same network (as specified by the netmask affinity), the RTT is taken from the results table instead of a new, real-time probe. This reduces response time.

Before you begin:

- You must have read-write permission for global load balancing settings.

The settings you configure are applied if the dynamic-proximity RTT option is enabled in the virtual server pool configuration.

Syntax

```
config global-load-balance setting
   set proximity-cache-aging-period <integer>
   set proximity-cache-mask-length <integer>
   set proximity-cache-mask-length6 <integer>
   set proximity-detect-interval <integer>
   set proximity-detect-protocol {icmp|icmp-and-tcp}
   set proximity-detect-retry-count <integer>
end
```

`proximity-cache-aging-period`	RTT results are cached. This setting specifies the length of time in seconds for which the RTT cache entry is valid. The default is 86400. The valid range is 60-2,592,000 seconds.

proximity-cache-mask-length	Number of IPv4 netmask bits that define network affinity for the RTT table. The default is 24. For example, if the GLB records an RTT for a client with source IP address 192.168.1.100, the record is stored and applies to all requests from the 192.168.1.0/24 network.
proximity-cache-mask-length6	Number of IPv6 netmask bits that define network affinity for the RTT table. The default is 64.
proximity-detect-interval	Interval between retries if the probe fails. The default is 3. The valid range is 1-3600 seconds.
proximity-detect-protocol	• icmp • icmp-and-tcp
proximity-detect-retry-count	Retry count if the probe fails. The default is 3. The valid range is 1-10 times.

Example

```
FortiADC-docs # config global-load-balance setting
FortiADC-docs (setting) # get
password : *
proximity-detect-protocol : icmp
proximity-detect-retry-count : 3
proximity-cache-mask-length : 24
proximity-cache-mask-length6 : 64
proximity-detect-interval : 3
proximity-cache-aging-period : 86400
persistence-mask-length : 24
persistence-mask-length6 : 64
persistence-timeout : 60
FortiADC-docs (setting) # set proximity-detect-protocol icmp
FortiADC-docs (setting) # set proximity-detect-retry-count 2
FortiADC-docs (setting) # set proximity-cache-mask-length 24
FortiADC-docs (setting) # set proximity-cache-mask-length6 64
FortiADC-docs (setting) # set proximity-detect-interval 2
FortiADC-docs (setting) # set proximity-cache-aging-period 200
FortiADC-docs (setting) # end
```

Authentication

Use this command to configure the authentication options for the GLB settings of the FQDN.

Before you begin:

- You must have read-write permission for global load balancing settings.

Syntax

```
config global-load-balance setting
```

```
    set auth-type {none|TCP_MD5SIG|auth_verify}
    set password <string>
    set ca-verify {enable|disable}
    set ca-group <datasource>
    set intermediate-ca-group <datasource>
end
```

auth-type	Select the authentication type: • none — No password. • TCP_MD5SIG — With password, but cannot be used if NAT is in between the client and server. This is because, when using the TCP MD5SIG authentication in a network with NAT in between, the IP layer is encrypted. So is every packet. Because the IP address will be changed, the encryption check will always fail. • auth_verify — The authentication key is sent to the server after a three-way handshake. The key is encrypted and NAT in between will not affect the authentication.
password	The **password** option is available if **auth-type** is **TCP_MD5SIG** or **auth_verify**. Enter the password to authenticate the key. This password is used for authentication between the GLB and the server. The same password must be set on both, otherwise the two will not be able to synchronize.
ca-verify	Enable/disable the root CA verification when synchronizing the SLB information to the GSLB server.
ca-group	The **ca-group** option is available if **ca-verify** is **enabled**. Select a trusted CA group to verify the peer certificate.
intermediate-ca-group	The **intermediate-ca-group** option is available if **ca-verify** is **enabled**. Select a trusted intermediate CA group to verify the peer certificate.

Example

```
FortiADC-docs # config global-load-balance setting
FortiADC-docs (setting) # get
password : *
proximity-detect-protocol : icmp
proximity-detect-retry-count : 3
proximity-cache-mask-length : 24
proximity-cache-mask-length6 : 64
proximity-detect-interval : 3
proximity-cache-aging-period : 86400
persistence-mask-length : 24
persistence-mask-length6 : 64
persistence-timeout : 60
set auth-type TCP_MD5SIG
FortiADC-docs (setting) #set password *
FortiADC-docs (setting) # end
```

config global-load-balance topology

Use this command to edit the member location list. This location is used when a vitual server pool is added to the host.

Starting from FortiADC 5.x.x, you can specify one location list as "any" to indicate any country other than the one that is already specified if you use DNS query origin method in a host.

Syntax

```
config global-load-balance topology
   edit <No.>
      set member <country or city>
   next
end
```

member	The country or city code.

Example

```
FortiADC-VM # config global-load-balance topology
FortiADC-VM (topology) # edit "1"
FortiADC-VM (1) # set member CN11
FortiADC-VM (1) # next
FortiADC-VM (topology) # end
```

Note: Starting from FortiADC 5.x.x, you must use country or city code to indicate a country or city. In the example above, "CN11" stands for "China, Beijing".

config global-load-balance virtual-server-pool

The virtual server pool configuration defines the set of virtual servers that can be matched in DNS resource records, so it should include, for example, all the virtual servers that can be answers for DNS requests to resolve www.example.com.

You also specify the key parameters of the global load balancing algorithm, including proximity options, status checking options, load balancing method, and weight.

The DNS response is an ordered list of answers. Virtual servers that are unavailable are excluded. Available virtual servers are ordered based on the following priorities:

1. Geographic proximity
2. Dynamic proximity
3. Weighted round robin

A client that receives DNS response with a list of answers tries the first and only proceeds to the next answers if the first answer is unreachable.

Before you begin:

- You must have created the global load balance server configuration and you must know the names of the virtual servers that have been populated in that configuration.
- You must have read-write permission for global load balancing settings.

After you have created a virtual server pool configuration object, you can specify it global load balancing host configuration.

Syntax

```
config global-load-balance virtual-server-pool
   edit <name>
      set check-server-status {enable|disable}
      set check-virtual-server-existence {enable|disable}
      set preferred {None | GEO | GEO-ISP | RTT | Least-Connections | Connection-Limit |
           Bytes-Per-Second | Server-Performance}
      set alternate {None | GEO | GEO-ISP | RTT | Least-Connections | Connection-Limit |
           Bytes-Per-Second | Server-Performance}
      set load-balance-method wrr
      config member
         edit <No.>
            set backup {enable|disable}
            set server <datasource>
            set server-member-name <string>
            set weight <integer>
         next
      end
   next
end

config global-load-balance virtual-server-pool
   edit "vsp1"
      set preferred Connection-Limit
      set preferred Bytes-Per-Second
   next
end
```

`check-server-status`	Enable/disable polling of the local FortiADC SLB. If the server is unresponsive, its virtual servers are not selected for DNS answers.
`check-virtual-server-existence`	Enable/disable checks on whether the status of the virtual servers in the virtual server list is known. Virtual servers with unknown status are not selected for DNS answers.
`preferred`	The preferred schedule method for this virtual server pool.
`alternate`	The alternate schedule method for this virtual server pool.
`load-balance-method`	Only weighted round-robin is supported.

	Note: The preferred method will be used first when scheduling a DNS query in a virtual server pool. If the preferred method is selected GEO/GEO-ISP and the configured GEO IP/ISP can't match source IP of DNS query, then alternate method will be used. Also, when the preferred method is set to None, the alternate method will be used. When the preferred and the alternate methods are both set to None or do not match any of the result, the load-balance-method WRR will be used to schedule by default.
`preferred Connection-Limit`	Scheduling is done according to the current connection number of each virtual server in the virtual-server-pool members and each virtual server's configured connection-limit value.
`preferred Bytes-Per-Second`	If the BPS method is used, the virtual server with the least throughput should be the answer responded by GLB.
config member	
`backup`	Enable to designate the pool as a backup member of the group. All backup members are inactive until all main members are down.
`server`	Specify a global server load balancing servers configuration object.
`server-member-name`	Specify the name of the virtual server that is in the servers virtual server list configuration.
`weight`	Assigns relative preference among members—higher values are more preferred and are assigned connections more frequently. The default is 1. The valid range is 1-255.

Example

```
FortiADC-VM # config global-load-balance virtual-server-pool
FortiADC-VM (virtual-server~p) # edit pool_test
Add new entry 'pool_test' for node 2858
FortiADC-VM (pool_test) # get
preferred : NONE
check-server-status : disable
check-virtual-server-existence: disable
load-balance-method : wrr
FortiADC-VM (pool_test) # set preferred GEO-ISP
FortiADC-VM (pool_test) # config member
FortiADC-VM (member) # edit 1
Add new entry '1' for node 2864
FortiADC-VM (1) # get
server :
server-member-name :
weight : 1
backup : disable
FortiADC-VM (1) # set server neighbor1
FortiADC-VM (1) # set server-member-name vs-1
FortiADC-VM (1) # set weight 2
FortiADC-VM (1) # next
FortiADC-VM (member) # edit 2
Add new entry '2' for node 2864
```

```
FortiADC-VM (2) # set server neighbor1
FortiADC-VM (2) # set server-member-name vs-2
FortiADC-VM (2) # next
FortiADC-VM (member) # edit 3
Add new entry '3' for node 2864
FortiADC-VM (3) # set server neighbor2
FortiADC-VM (3) # set server-member-name vs-3
FortiADC-VM (3) # set weight 2
FortiADC-VM (3) # next
FortiADC-VM (member) # edit 4
Add new entry '4' for node 2864
FortiADC-VM (4) # set server neighbor2
FortiADC-VM (4) # set server-member-name vs-4
FortiADC-VM (4) # end
FortiADC-VM (pool_test) # end

config global-load-balance virtual-server-pool
   edit "vsp1"
      set preferred GEO
      set alternate RTT
   next
end
```

config global-load-balance host

Use this command to create host configurations. Host settings are used to form the zone configuration and resource records in the generated DNS zone used for global load balancing.

Before you begin:

- You must have created the global virtual server pools you want to use.
- You must have read-write permission for global load balancing settings.

After you have created a host configuration object, it can be used to form the zone and resource records in the generated DNS zone configuration.

Syntax

```
config global-load-balance host
   edit <name>
      set domain-name <string>
      set host-name <string>
      set response-single-record {enable|disable}
      set persistence {enable|disable}
      set default-feedback-ip <ip>
      set default-feedback-ip6 <ip>
      set load-balance-method {global availability|none|topology}
      config virtual-server-pool-list
         edit <No.>
            set virtual-server-pool <string>
```

```
        next
      end
    next
end
```

domain-name	The domain name must end with a period. For example: example.com.
host-name	The hostname part of the FQDN, such as www. **Note**: You can specify the @ symbol to denote the zone root. The value substituted for @ is the preceding $ORIGIN directive.
respond-single-record	Enable/disable an option to send only the top record in response to a query. Disabled by default. By default, the response is an ordered list of records.
persistence	Enable/disable the persistence table. Disabled by default. If you enable persistence, the client source address is recorded in the persistence table, and subsequent requests from the same network or the same host or domain are sent an answer with the virtual servers listed in the same order (unless a server becomes unavailable and is therefore omitted from the answer).
default-feedback-ip	Specify an IP address to return in the DNS answer if no virtual servers are available.
default-feedback-ip6	Specify an IPv6 address to return in the DNS answer if no virtual servers are available.
load-balance-method	Specify a virtual server pool selection method. Set to "weight" by default.
virtual-server-pool	Specify a virtual server pool configuration object to host, i.e., a virtual server pool name, location list (optional), and/or ISP (optional).

Example

```
FortiADC-VM # config global-load-balance host
FortiADC-VM (host) # edit www_fadc_com
Add new entry 'www_fadc_com' for node 2869
FortiADC-VM (www_fadc_com) # get
host-name :
domain-name :
respond-single-record : disable
persistence : disable
load-balance-method : weight
default-feedback-ip : 0.0.0.0
default-feedback-ip6 : ::
FortiADC-VM (www_fadc_com) # set host-name www
FortiADC-VM (www_fadc_com) # set domain-name fadc.com
FortiADC-VM (www_fadc_com) # config virtual-server-pool-list
FortiADC-VM (virtual-server~p) # edit "1"
FortiADC-VM (1) # set virtual-server-pool test
FortiADC-VM (1) # next
FortiADC-VM (virtual-server~p) # end
FortiADC-VM (www_fadc_com) # end
```

config link-load-balance

The `config link-load-balance` commands configure the link load balancing feature settings.

This chapter is a reference for the following commands:

config link-load-balance flow-policy

Use this command to configure link load balancing policy rules.

A link policy matches traffic to rules that select a link group or virtual tunnel.

The policy uses a matching tuple: source, destination, service, and schedule. The policy match is a Boolean AND—All must match for the rule to be applied.

The elements of the tuple support specification by group objects. This is a Boolean OR—If source IP address belongs to member 1 OR member 2, then source matches.

The logical combinations enable you to subscribe multiple address spaces or services to a group of links, and create load balancing rules on that group basis.

The policy table is consulted from top to bottom. The first rule to match is applied.

The FortiADC system evaluates traffic to determine the routing rules to apply. With regard to link load balancing, the system evaluates rules in the following order and applies the first match:

1. LLB link policy
2. Policy route
3. Static/Dynamic route
4. LLB default link group

Before you begin:

- You must have configured any address, service, and schedule objects that you want to use as match criteria for your policy.
- You must have configured a link group or virtual tunnel group.
- You must have read-write permission for link load balancing settings.

Syntax

```
config link-load-balance flow-policy
   set default-link-group <datasource>
   config rule
      edit <name>
         set group-type {link-group | virtual-tunnel}
         set link-group <datasource>
         set virtual-tunnel <datasource>
         set destination-type {address|addrgrp|isp}
         set destination-address <datasource>
         set destination-addrgrp <datasource>
         set destination-isp <datasource>
         set in-interface <datasource>
         set schedule <datasource>
         set service-type {service|servicegrp}
         set service <datasource>
         set servicegrp <datasource>
         set source-type {address|addrgrp|isp}
         set source-address <datasource>
         set source-addrgrp <datasource>
         set source-isp <datasource>
   next
end
```

`default-link-group`	Specify a link group configuration object that is used as the default when traffic does not match policy rules.
config rule	
`group-type`	• link-group: Policy uses a link group. • virtual-tunnel: Policy uses a virtual tunnel.
`link-group`	If you specify the link group type, specify a link group configuration object.
`virtual-tunnel`	If you specify the virtual tunnel group type, specify a virtual tunnel configuration object.
`destination-type`	Specify whether to use address, address group, or ISP address objects for this rule.
`destination-address`	Specify an address object to match destination addresses. If you do not specify a destination address, the rule matches any destination.
`destination-addrgrp`	Specify an address object to match destination addresses. If you do not specify a destination address, the rule matches any destination.
`destination-isp`	Specify an address object to match destination addresses. If you do not specify a destination address, the rule matches any destination.
`in-interface`	Network interface to which the policy applies.
`schedule`	Specify the schedule object that determines the times the system uses the logic of this configuration. The link policy is active when the current time falls in a time period specified by one or more schedules in the schedule group. If you do not specify a schedule, the rule applies at all times.
`service-type`	Specify whether to use service or service group objects for this rule.

service	Specify a service object to match destination services. If you do not specify a service, the rule matches any service.
servicegrp	Specify a service group object to match destination services. If you do not specify a service, the rule matches any service.
source-type	Specify whether to use address, address group, or ISP address objects for this rule.
source-address	Specify an address object to match source addresses. If you do not specify a source address, the rule matches any source address.
source-addrgrp	Specify an address object to match source addresses. If you do not specify a source address, the rule matches any source address.
source-isp	Specify an address object to match source addresses. If you do not specify a source address, the rule matches any source address.

Example

```
FortiADC-docs # config link-load-balance flow-policy

FortiADC-docs (flow-policy) # config rule
FortiADC-docs (rule) # edit ISP-1
Add new entry 'ISP-1' for node 634

FortiADC-docs (ISP-1) # get
in-interface :
source-type : address
source-address :
destination-type : address
destination-address :
service-type : service
service :
schedule :
group-type : link-group
link-group :

FortiADC-docs (ISP-1) # set in-interface port2
FortiADC-docs (ISP-1) # set source-type addrgrp
FortiADC-docs (ISP-1) # set source-addrgrp LAN
FortiADC-docs (ISP-1) # set destination-type addrgrp
FortiADC-docs (ISP-1) # set destination-addrgrp WAN
FortiADC-docs (ISP-1) # set service-type servicegrp
FortiADC-docs (ISP-1) # set servicegrp Web
FortiADC-docs (ISP-1) # set link-group ISP1
FortiADC-docs (ISP-1) # end
FortiADC-docs (flow-policy) # end
```

config link-load-balance gateway

Use this command to configure gateway links.

The gateway link configuration enables you to specify health checks, bandwidth rate thresholds, and spillover threshold behavior for the gateway links you add to link groups.

Before you begin:

- You must know the IP addresses of the ISP gateway link used in the network segment where the FortiADC appliance is deployed.
- You must have added health check configuration objects that you want to use to probe the gateway links.
- You must have read-write permission for link load balancing settings.

After you have configured a gateway link configuration object, you can select it in the link group configuration.

Syntax

```
config link-load-balance gateway
   edit <name>
      set health-check-ctrl {enable|disable}
      set health-check-list {<datasource> ...}
      set health-check-relation {AND|OR}
      set inbound-bandwidth <integer>
      set ip <class_ip>
      set outbound-bandwidth <integer>
      set spillover-threshold-in <integer>
      set spillover-threshold-out <integer>
      set spillover-threshold-total <integer>
   next
end
```

`health-check-ctrl`	Enable/disable health checks.
`health-check-list`	Specify one or more health check configuration objects.
`health-check-relation`	• AND—All of the specified health checks must pass for the server to the considered available. • OR—One of the specified health checks must pass for the server to be considered available.
`inbound-bandwidth`	Maximum bandwidth rate for inbound traffic through this gateway link.
`ip`	IP address of the gateway link.
`outbound-bandwidth`	Maximum bandwidth rate for outbound traffic to this gateway link. If traffic exceeds this threshold, the FortiADC system considers the gateway to be full and does not dispatch new connections to it. The default is 2,000,000 Kbps. The valid range is 1 to 2,147,483,647. We recommend you tune bandwidth thresholds strategically, using the bandwidth rate and price structure agreement you have with your ISP to your advantage.
`spillover-threshold-in`	Maximum inbound bandwidth rate for a link in a spillover load balancing pool.
`spillover-threshold-out`	Maximum outbound bandwidth rate for a link in a spillover load balancing pool.

	If you enable spillover load balancing in the link group configuration, the system maintains a spillover list. It dispatches new connections to the link with the greatest priority until its spillover threshold is exceeded; then dispatches new connections to the link with the next greatest priority until its threshold is exceeded, and so on. The default is 2,000,000 Kbps. The valid range is 1 to 2,147,483,647.
`spillover-threshold-total`	Maximum total bandwidth rate (inbound plus outbound) for a link in a spillover load balancing pool.

Example

```
FortiADC-VM (gateway) # edit llb-gateway
Add new entry 'llb-gateway' for node 2501
FortiADC-VM (llb-gateway) # get
ip : 0.0.0.0
inbound-bandwidth : 2000000
outbound-bandwidth : 2000000
health-check-ctrl : disable
spillover-threshold-in: 2000000
spillover-threshold-out: 2000000
spillover-threshold-total: 2000000

FortiADC-VM (llb-gateway) # set ip 192.168.1.1
FortiADC-VM (llb-gateway) # end

FortiADC-VM # get link-load-balance gateway llb-gateway
ip : 192.168.1.1
inbound-bandwidth : 2000000
outbound-bandwidth : 2000000
health-check-ctrl : disable
spillover-threshold-in: 2000000
spillover-threshold-out: 2000000
spillover-threshold-total: 2000000
```

config link-load-balance link-group

Use this command to configure link groups.

Link groups include ISP gateways your company uses for outbound traffic. Grouping links reduces the risk of outages and provisions additional bandwidth to relieve potential traffic congestion.

The link group configuration specifies the load balancing algorithm and the gateway routers in the load balancing pool. You can enable LLB options, such as persistence rules and proximity routes.

Before you begin:

- You must have configured gateway links and persistence rules and before you can select them in the link group configuration.
- You must have read-write permission for link load balancing settings.

After you have configured a link group configuration object, you can select it in the link policy configuration.

Syntax

```
config link-load-balance link-group
   edit <name>
      set addr-type ipv4
      set persistence <datasource>
      set proximity-route {enable|disable}
      set route-method {consistent-hash-ip | least-connection | least-new-cps | least-
           throughput-all | least-throughput-in | least-throughput-out | spillover-
           throughput-all | spillover-throughput-in | spillover-throughput-out | wrr>
      config link-member
         edit <name>
            set backup {enable|disable}
            set gateway <datasource>
            set spillover-priority <integer>
            set status {enable|disable}
            set weight <integer>
         next
      end
   next
end
```

addr-type	Only IPv4 is supported.
persistence	Specify a persistence configuration. Optional.
proximity-route	Enable/disable use of proximity routes.
route-method	• consistent-hash-ip: Selects the gateway link based on a hash of the source IP address. • least-connection: Dispatches new connections to the link member with the lowest number of connections. • least-new-cps: Dispatches new connections to the link member that has the lowest rate of new connections per second. • least-throughput-all: Dispatches new connections to the link member with the least total traffic (that is, inbound plus outbound). • least-throughput-in: Dispatches new connections to the link member with the least inbound traffic. • least-throughput-out: Dispatches new connections to the link member with the least outbound traffic. • spillover-throughput-all: Spillover list based on total traffic (that is, inbound plus outbound). • spillover-throughput-in: Spillover list based on inbound traffic. • spillover-throughput-out: Dispatches new connections according to the spillover list based on outbound traffic. • wrr: Dispatches new connections to link members using a weighted round-robin method. This is the default.
config link member	
backup	• enable—Designate the link as a backup member of the group. All backup members are inactive until all main members are down.

	• disable—Designate the link as a main member of the group.
gateway	Specify a gateway configuration object.
spillover-priority	Assigns a priority to the link when using a spillover load balancing method. Higher values have greater priority. When a spillover method is enabled, the system dispatches new connections to the link that has the greatest spillover priority until its threshold is exceeded; then it dispatches new connections to the link with the next greatest priority until its threshold is exceeded, and so on. If multiple links in a link group have the same spillover priority, the system dispatches new connections among those links according to round robin. The default is 0. The valid range is 0-9.
status	• enable—The member is considered available for new traffic. • disable—The member is considered unavailable for new traffic.
weight	Assigns relative preference among members—higher values are more preferred and are assigned connections more frequently. The default is 1. The valid range is 1 to 255. All load balancing methods consider weight, except spillover, which uses its own priority configuration. Servers are dispatched requests proportional to their weight, relative to the sum of all weights. The following example shows the effect of weight on WRR: • Sever A, Weight 2; Server B, Weight 1: Requests are sent AABAAB. • Sever A, Weight 3; Server B, Weight 2: Requests are sent AABAB. For other methods, weight functions as a tie-breaker. For example, with the Least Connection algorithm, requests are sent to the server with the least connections. If the number of connections is equal, the request is sent to the server with the greater weight. For example: • Server A, Weight 1, 1 connection • Server B, Weight 2, 1 connection The next request is sent to Server B.

Example

```
FortiADC-VM (link-group) # edit llb-link-group
Add new entry 'llb-link-group' for node 618

FortiADC-VM (llb-link-group) # get
addr-type : ipv4
route-method : wrr
persistence :
proximity-route : disable

FortiADC-VM (llb-link-group) # config link-member

FortiADC-VM (link-member) # edit 1
Add new entry '1' for node 624

FortiADC-VM (1) # get
gateway :
weight : 1
spillover-priority : 0
```

```
status : enable

FortiADC-VM (1) # set gateway llb-gateway
FortiADC-VM (1) # end
```

config link-load-balance persistence

Use this command to configure persistence rules.

Persistence rules identify traffic that should be ignored by load balancing rules and instead be forwarded to the same gateway each time the traffic traverses the FortiADC appliance.

You should use persistence rules with applications that use a secure connection. Such applications drop connections when the server detects a change in a client's source IP address.

Before you begin:

- You must have an awareness of the types of outbound traffic from your network. Persistence rules are useful for traffic that requires an established session, such as secure connections (HTTPS and SSH, for example).
- You must have knowledge of the source and/or destination subnets to which the persistence rules should apply.
- You must have read-write permission for link load balancing settings.

You can use persistence rules in link groups but not virtual tunnels.

Syntax

```
config link-load-balance persistence
   edit <name>
      set timeout <integer>
      set type {destination-address | source-address | source-destination-address | source-
           destination-pair}
      set dst-ipv4-maskbits <integer>
      set src-ipv4-maskbits <integer>
   next
end
```

timeout	The default is 300 seconds.
type	• destination-address: Packets with a destination IP address that belongs to the same subnet take same outgoing gateway. • source-address: Packets with a source IP address that belongs to the same subnet take the same outgoing gateway. • source-destination-address: Packets with a source IP address and destination IP address that belong to the same subnet take the same outgoing gateway. • source-destination-pair: Packets with the same source IP address and destination IP address take same outgoing gateway.

	Note: Source address based persistence consumes a significant amount of memory, as calculated using the following formula per VS: *(max persistence entry size) x (size per entry of the table) x (content routing or pool count)* For example: If the max persistence entry size is 262144 (default), the per entry of the table is 44 bytes, and there is no content routing. 262144 x 44 = 11534336 bytes (which is ≈11MB)
`dst-ipv4-maskbits`	Number of bits in a subnet mask to specify a network segment that should following the persistence rule. For example, if you set this to 24, and the system chooses a particular gateway router for destination IP 192.168.1.100, the system will select that same gateway for traffic to all destination IPs in subnet 192.168.1.0/24.
`src-ipv4-maskbits`	Number of bits in a subnet mask to specify a network segment that should following the persistence rule. For example, if you set this to 24, and the system chooses a particular gateway router for client IP 192.168.1.100, the system will select that same gateway for subsequent client requests when the subsequent client belongs to subnet 192.168.1.0/24.

Example

```
FortiADC-VM # config link-load-balance persistence
FortiADC-VM (persistence) # edit llb-persistence
Add new entry 'llb-persistence' for node 674

FortiADC-VM (llb-persistence) # get
type : source-destination-pair
timeout : 300

FortiADC-VM (llb-persistence) # end
```

config link-load-balance proximity-route

Use this command to configure proximity routes.

The proximity route feature enables you to associate link groups with efficient routes. Proximity routes can improve user experience over the WAN because traffic is routed over fast routes.

You can use either or both of these methods:

- Dynamic Detection—The system polls the network for efficient routes. The algorithm selects a gateway based on latency. When the bandwidth usage of a gateway reaches 100%, the gateway is considered too busy and is not selected.
- Static Table—You specify the gateways to use for traffic on destination networks.

If you configure both, the system checks the static table first for a matching route and, if any, uses it. If there is no matching static route, the system uses dynamic detection.

Note: Adding a new static route does not affect existing sessions. Deleting or editing a static route causes the related sessions to be re-created.

Before you begin:

- You must have knowledge of IP addresses used in outbound network routes to configure a static route.
- You must have read-write permission for link load balancing settings.

Syntax

```
config link-load-balance proximity-route
     set mode {disable | dynamic-detect-only | static-table-first | static-table-only}
     set dynamic-cache-aging-period <integer>
     set dynamic-detect-protocol {icmp|icmp-and-tcp}
     set dynamic-detect-retry-count <class_ip>
     set dynamic-detect-retry-interval <integer>
     config static-table
        edit <No.>
           set type {isp|subnet}
           set ip-netmask <ip&netmask>
           set isp-name <datasource>
           set gateway <datasource>
        next
     end
   next
end
```

`mode`	• disable • dynamic-detect-only • static-table-first • static-table-only
`dynamic-cache-aging-period`	The default is 86,400 seconds (24 hours).
`dynamic-detect-protocol`	• icmp—Use ICMP to detect routes. Calculate proximity by the smaller RTT. • icmp-and-tcp—Some hosts do not response ICMP requests. Specify this option to use both ICMP and TCP to detect routes and RTT. For TCP detection, port 7 (TCP echo) is used. A connection refused or connection reset by the destination is treated as successful detection.
`dynamic-detect-retry-count`	The default is 3.
`dynamic-detect-retry-interval`	The default is 3.
config static-table	
`type`	Specify the IP and netmask manually or use an ISP address object. Routes that are specified manually have priority over ISP address object entries.
`ip-netmask`	Destination IP address and netmask.

isp-name	Specify an ISP address book configuration object. If an address exists in multiple ISP address books, the route entries have priority as follows: **1.** User-defined entries. **2.** Entries from an address book that has been imported. **3.** Entries from the predefined address book (default for the firmware image).
gateway	Specify a gateway configuration object. The gateway must be able to route packets to the destination IP address that you have specified.

Example

```
FortiADC-VM # config link-load-balance proximity-route
FortiADC-VM (proximity-route) # set mode static-table-first

FortiADC-VM (proximity-route) # get
mode : static-table-first
dynamic-detect-protocol: icmp
dynamic-detect-retry-count: 3
dynamic-detect-retry-interval: 3
dynamic-cache-aging-period: 86400

FortiADC-VM (proximity-route) # config static-table
FortiADC-VM (static-table) # edit 1
Add new entry '1' for node 687
FortiADC-VM (1) # set gateway 198.51.100.0
FortiADC-VM (1) # set destination 198.51.100.10
FortiADC-VM (1) # end

FortiADC-VM (proximity-route) # get
mode : static-table-first
dynamic-detect-protocol: icmp
dynamic-detect-retry-count: 3
dynamic-detect-retry-interval: 3
dynamic-cache-aging-period: 86400
== [ 1 ]

FortiADC-VM (proximity-route) # show
config link-load-balance proximity-route
set mode static-table-first
config static-table
edit 1
set destination 198.51.100.10/32
set gateway 198.51.100.0
next
end
end
```

config link-load-balance virtual-tunnel

Use this command to configure virtual tunnels.

Virtual tunnels enable reliable, site-to-site connectivity using Generic Routing Encapsulation (GRE) to tunnel traffic between pairs of FortiADC appliances.

The virtual tunnel group configuration sets the list of tunnel members, as well as load balancing options like algorithm and weight.

When you add members to a virtual tunnel configuration, you specify a local and remote IP address. These addresses are IP addresses assigned to a network interface on the local and remote FortiADC appliance.

Before you begin:

- You must have read-write permission for link load balance settings.

After you have configured a virtual tunnel configuration object, you can select it in the link policy configuration.

Syntax

```
config link-load-balance virtual-tunnel
   edit <name>
      set dispatch-method {vt-wrr|vt-chash}
      config vt-member
         edit <name>
            set health-check-ctrl {enable|disable}
            set status {enable|disable}
            set tunnel-local-addr <class_ip>
            set tunnel-remote-addr <class_ip>
            set weight <integer>
         next
      end
   next
end
```

`dispatch-method`	• vt-wrr: Dispatches packets to VT members using a weighted round-robin method. • vt-chash: Dispatches packets by source-destination IP address tuple.
`backup`	• enable—Designate the tunnel as a backup member of the group. All backup members are inactive until all main members are down. • disable—Designate the tunnel as a main member of the group.
`health-check-ctrl`	• enable—Send probes to test whether the link is available. • disable—Do not send probes to test the health of the link.
`status`	• enable—The member is considered available for new traffic. • disable—The member is considered unavailable for new traffic.
`tunnel-local-addr`	IP address for the network interface this system uses to form a VPN tunnel with the remote system.
`tunnel-remote-addr`	IP address that the remote FortiADC system uses to form a VPN tunnel with this system.

weight	Assigns relative preference among members—higher values are more preferred and are assigned connections more frequently.

Example

```
FortiADC-VM # config link-load-balance virtual-tunnel
FortiADC-VM (virtual-tunnel) # edit llb-vt
Add new entry 'llb-vt' for node 222

FortiADC-VM (llb-vt) # get
dispatch-method : vt-wrr

FortiADC-VM (llb-vt) # config vt-member
FortiADC-VM (vt-member) # edit vt-member-1
Add new entry 'vt-member-1' for node 225

FortiADC-VM (vt-member-1) # get
tunnel-local-addr : 0.0.0.0
tunnel-remote-addr : 0.0.0.0
weight : 1
status : enable
health-check-ctrl : disable

FortiADC-VM (vt-member-1) # set health-check-ctrl enable
FortiADC-VM (vt-member-1) # set tunnel-local-addr 192.0.2.10
FortiADC-VM (vt-member-1) # set tunnel-remote-addr 198.51.100.10
FortiADC-VM (vt-member-1) # end

FortiADC-VM (llb-vt) # get
dispatch-method : vt-wrr
== [ vt-member-1 ]

FortiADC-VM (llb-vt) # show
config link-load-balance virtual-tunnel
edit "llb-vt"
config vt-member
edit "vt-member-1"
set tunnel-local-addr 192.0.2.10
set tunnel-remote-addr 198.51.100.10
set health-check-ctrl enable
next
end
next
end
```

config load-balance

The `config load-balance` commands configure the load-balancing feature settings.

This chapter is a reference for the following commands:

config load-balance auth-policy

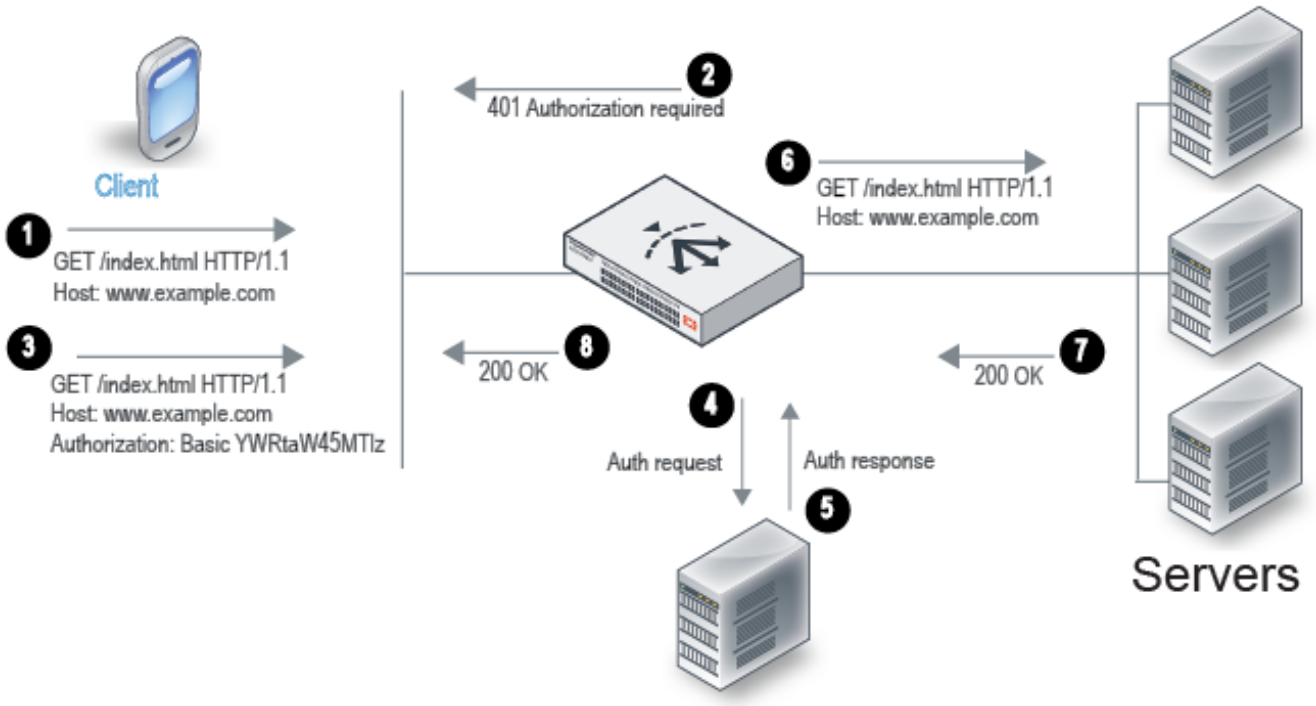
Use this command to configure an auth policy. The parameters of the policy set the matching terms that mandate authentication for the data source that has authorization. The authentication server type determines the type of data

source that would be referenced for the auth policy. There are three authentication server types: standard, SAML and OAuth.

For example, you can define an auth policy for a standard authentication server type that has the following logic: if the Host header matches example.com and the URI matches /index.html, then the group example-group is authorized. FortiADC supports the Basic Authentication Scheme described in RFC 2617.

Figure 3 illustrates the client-server communication when authorization is required.

Authorization and authentication



1. The client sends an HTTP request for a URL belonging to a FortiADC virtual server that has an authorization policy.
2. FortiADC replies with an HTTP 401 to require authorization. On the client computer, the user might be prompted with a dialog box to provide credentials.
3. The client reply includes an Authorization header that gives the credentials.
4. FortiADC sends a request to the server (local, LDAP, or RADIUS) to authenticate the user.
5. The authentication server sends its response, which can be cached according to your user group configuration.
6. If authentication is successful, FortiADC continues processing the traffic and forwards the request to the real server.
7. The real server responds with an HTTP 200 OK.
8. FortiADC processes the traffic and forwards the server response to the client.

Before you begin:

- You must have created the user groups to be authorized with the policy. You also configure users and authentication servers separately.
- You must have read-write permission for load balancing settings.

After you have configured an auth policy, you can select it in the virtual server configuration. Note the following requirements:

- Virtual server type must be Layer 2 or Layer 7.
- Profile type must be HTTP or HTTPS.

- The profile option once-only must be disabled.

Define an auth policy for a standard authentication server type

Define the auth policy to use standard authentication with `set type standard` and reference the user group data source.

Syntax

```
config load-balance auth-policy
   edit <name>
        config member
        edit 1
           set host-status {enable|disable}
           set host <hostname>
           set type standard
           set user-realm <string>
           set auth-path <path>
           set user-group <datasource>
        next
     end
   next
end
```

`host-status`	If enabled, require authorization only for the specified host. If disabled, ignore hostname in the HTTP request header and require authorization for requests with any Host header. Disabled by default.
`host`	Specify the HTTP Host header. If host-status is enabled, the policy matches only if the hostname header matches this value. Complete, exact matching is required. For example, `www.example.com` matches `www.example.com` but not `www.example.com.hk`.
`user-realm`	Realm to which the auth-path URI belongs. If a request is authenticated and a realm specified, the same credentials should be valid for all other requests within this realm.
`auth-path`	Require authorization only if the URI of the HTTP request matches this pathname. If none is specified, requests to any URI require authorization. The value is parsed as a match string prefix. For example, `/abc` matches `http://www.example.com/abcd` and `http://www.example.com/abc/11.html` but not `http://www.example.com/1abcd`.
`user-group`	User group that is authorized to access the protected resource.

Define an auth policy for a SAML authentication server type

Define the auth policy to use SAML authentication with `set type SAML` and reference the SAML SSO ID data source.

Syntax

```
config load-balance auth-policy
   edit <name>
        config member
        edit 1
           set host-status {enable|disable}
           set host <hostname>
           set type SAML
           set auth-path <path>
           set saml-sp <saml-sso-id>
        next
     end
   next
end
```

`host-status`	If enabled, require authorization only for the specified host. If disabled, ignore hostname in the HTTP request header and require authorization for requests with any Host header. Disabled by default.
`host`	Specify the HTTP Host header. If host-status is enabled, the policy matches only if the hostname header matches this value. Complete, exact matching is required. For example, `www.example.com` matches `www.example.com` but not `www.example.com.hk`.
`auth-path`	Require authorization only if the URI of the HTTP request matches this pathname. If none is specified, requests to any URI require authorization. The value is parsed as a match string prefix. For example, `/abc` matches `http://www.example.com/abcd` and `http://www.example.com/abc/11.html` but not `http://www.example.com/1abcd`.
`saml-sp`	SAML SSO ID that is authorized to access the protected resource.

Define an auth policy for an OAuth authentication server type

Define the auth policy to use OAuth authentication with `set type OAuth` and reference the OAuth data source.

Syntax

```
config load-balance auth-policy
   edit <name>
        config member
        edit 1
           set host-status {enable|disable}
           set host <hostname>
           set type OAuth
           set auth-path <path>
           set oauth <oauth-policy>
        next
     end
   next
```

```
end
```

`host-status`	If enabled, require authorization only for the specified host. If disabled, ignore hostname in the HTTP request header and require authorization for requests with any Host header. Disabled by default.
`host`	Specify the HTTP Host header. If host-status is enabled, the policy matches only if the hostname header matches this value. Complete, exact matching is required. For example, `www.example.com` matches `www.example.com` but not `www.example.com.hk`.
`auth-path`	Require authorization only if the URI of the HTTP request matches this pathname. If none is specified, requests to any URI require authorization. The value is parsed as a match string prefix. For example, `/abc` matches `http://www.example.com/abcd` and `http://www.example.com/abc/11.html` but not `http://www.example.com/1abcd`.
`oauth`	OAuth policy that is authorized to access the protected resource.

config load-balance caching

Use this command to configure the system cache.

The system RAM cache can store HTTP content and serve subsequent HTTP requests for that content without forwarding the requests to the backend servers, thereby reducing the load on the backend servers.

You can configure basic static caching or dynamic caching rules. For an overview of static and dynamic caching, see the FortiADC Handbook.

Before you begin:

- You must have a good understanding of caching and knowledge about the size of content objects clients access on the backend servers.
- You must have read-write permission for load balancing settings.

Caching is not enabled by default. After you have configured caching, you can select it in the profile configuration. To enable caching, select the profile when you configure the virtual server.

Syntax

```
config load-balance caching
   edit <name>
      set max-age <integer>
      set max-cache-size <integer>
      set max-entries <integer>
      set max-object-size <integer>
      config uri_exclude_list
         edit <No.>
            set uri <string>
         next
```

```
      end
      config dyn-cache-list
         edit <No.>
            set uri <string>
            set age <integer>
            set invalid-uri <string>
         next
      end
   next
end
```

max-age	The default is 43,200 seconds. The valid range is 60 to 86,400. The backend real server response header also includes a maximum age value. The FortiADC system enforces whichever value is smaller.
max-cache-size	The default is 100 MB. The valid range is 1 byte to 500 MB.
max-entries	The default is 10,000. The valid range is 1 to 262,144.
max-object-size	The default is 1 MB. The valid range is 1 byte to 10 MB.
config uri_exclude_list	
uri	Specify URIs to build a list or sites to exclude from caching. You can use regular expressions. This list has precedence over the Dynamic Cache Rule List. In other words, if a URI matches this list, it is ineligible for caching, even if it also matches the Dynamic Cache Rule list.
config dyn-uri-list	
uri	Pattern to match the URIs that have content you want cached and served by FortiADC. Be careful with matching patterns and the order rules in the list. Rules are consulted from lowest rule ID to highest. The first rule that matches is applied.
age	Timeout for the dynamic cache entry. The default is 60 seconds. The valid range is 1-86,400. This age applies instead of any age value in the backend server response header.
invalid-uri	Pattern to match URIs that trigger cache invalidation. Be careful with matching patterns and the order rules in the list. Rules are consulted from lowest rule ID to highest. The first rule that matches is applied. This list has precence over the Dynamic Cache URI list. In other words, if a URI matches this list, it is ineligible for caching, even if it also matches the Dynamic Cache URI list.

Example

```
FortiADC-VM # config load-balance caching

FortiADC-VM (caching) # edit lb-caching
Add new entry 'lb-caching' for node 2054
FortiADC-VM (lb-caching) # get
max-object-size : 1M
max-cache-size : 100M
max-entries : 10000
```

```
max-age : 43200

FortiADC-VM (lb-caching) # set max-cache-size 50M
FortiADC-VM (lb-caching) # end
```

config load-balance captcha-profile

FortiADC allows administrators to validate incoming users with CAPTCHAs to determine whether a client is a regular user or an attacker. FortiADC can configure the WAF/DoS Policy to issue CAPTCHAs only to clients who meet the attack rules.

Select a FortiADC default captcha profile from within the virtual server configuration or upload a customized captcha page if you want to use your own captcha verification page for when an WAF/DoS attack detected.

Syntax

```
config load-balance captcha-profile
   edit <captcha-profile-name>
      set vpath <string>
      set max-attempts <integer>
      set max-picture-changes <integer>
      set max-block-period <integer>
      set max-verify-period <integer>
      set max-valid-period <integer>
      set custom-captcha-page <enable/disable>
      set captcha-page-package <file-name>
      set picture-difficulty <hard/easy>
end
```

Paramter	Description
vpath	Virtual path of captcha function. This path is running on VS, so it will conflict with other configurations like errorpage's vpath and custom auth page. String type, not empty, maximum length 63, the default value is "/fortiadc_captcha/"
max-attempts	Maximum attempts for Captcha verification. Integer type, range 1-100, default 5. The client will be blocked upon exceeding max attempts.
max-picture-changes	The maximum number of times you can change another picture. Integer type, range 1-100, default 5. Attempts to change pictures upon exceeding the maximum number of attempts will not be successful.
picture-difficulty	There are two difficulty levels that can be selected: easy and hard. Hard level picture may fight AI picture recognition, but may cause difficulty in human identification. Default value is hard.
max-block-period	The length of time to block client. Integer type, range 10-2592000, default 86400. Client will be reset to untracked state once time has elapsed.
max-verify-period	The longest verification time from captcha verify action start. Unit second, range

Paramter	Description
	20-86400, default 1200. Exceed this time the client will be blocked.
custom-captcha-page	Enable/disable captcha page. The default is disabled. If disabled, the custom captcha package file option won't be valid.
captcha-page-package	File package for the customized captcha page. The file package must include index.html file, and in the index page, it must include a tag called "%%FORTIADC_CAPTCHA_IFRAME%%", that we will insert the verify page box on it. Note: This option is only available when custom-captcha-package is enabled.

config load-balance certificate-caching

Use the command to set certificate-caching related configuration. Certificate caching is used to restore re-signed certificates.

Note: This command is related to "config load-balance client-ssl-profile" on page 1.

Syntax

```
config load-balance certificate-caching
   edit "1"
      set max-certificate-cache-size <size>
      set max-entries <entries>
   next
```

`max-certificate-cache-size`	The maximum cache size used to store certificates. Valid values range from 10 Mb to 500 Mb.
`max-entries`	The maximum number of certificates that can be stored on the appliance (FortiADC), which can range from 1 to 2,621,444.

Example

```
config load-balance certificate-caching
   edit "1"
      set max-certificate-cache-size 100M
      set max-entries 10000
   next
```

config load-balance client-ssl-profile

Use this command to configure SSL-type real servers using the client-ssl-profile.

Profile	Description
LB_CLIENT_SSL_PROF_DEFAULT	This is the default client SSL load-balancing profile. It's a basic profile that can be used for all client SSL load-balancing scenarios. Recommended SSL versions: • SSLv3 • TLSv1.0 • TLSv1.1 • TLSv1.2 • TLSv1.3
LB_CLIENT_SSL_PROF_FORWARD_PROXY	This profile is used when the SSL Forward Proxy feature is enabled. It works in tandem with Forward Proxy Certificate Caching, i.e., LB_CERT_RAM_CACHING_DEFAULT), and Forward Proxy Local Signing CA, i.e., SSLPROXY_LOCAL_CA. Recommended SSL versions: • SSLv3 • TLSv1.0 • TLSv1.1 • TLSv1.2 • TLSv1.3
LB_CLIENT_SSL_PROF_HTTP2	This profile applies to HTTP2 protocol only. Recommended SSL version: • TLSv1.2 • TLSv1.3

Syntax

```
config load-balance client-ssl-profile
   edit <name>
      set client-certificate-verify <verify_profile_name>
      set client-sni-required {enable|disable}
      set forward-proxy {enable|disable}
      set local-certificate-group <local_certificate_group_name>
      set ssl-allowed-versions {sslv3 tlsv1.0 tlsv1.1 tlsv1.2 tlsv1.3}
      set ssl-ciphers <one or more ciphers>
      set ssl-customize-ciphers-flag {enable|disable}
      set forward-client-certificate {enable|disable}
      set forward-client-certificate-header <customized_header_name>
      set forward-proxy-certificate-caching <cache_name>
      set forward-proxy-local-signing-CA <local_ca>
      set forward-proxy-intermediate-ca-group <intermediate_ca>
      set forward-proxy-resign-cert-by-sni {enable|disable}
```

```
    set backend-ssl-sni-forward {enable|disable}
    set backend-ssl-customize-ciphers-flag {enable|disable}
    set backend-ssl-customized-ciphers <one or more ciphers>
    set backend-allow-ssl-versions {sslv3 tlsv1.0 tlsv1.1 tlsv1.2 tlsv1.3}
    set backend-ssl-OCSP-stapling-support {enable|disable}
    set reject-ocsp-stapling-with-missing-nextupdate {enable|disable}
    set reject-revoked-unknown-ocsp-stapling {enable|disable}
    set ocsp-stapling-skew-time <integer>
    set ssl-auto-chain-flag {enable|disable}
    set client-certificate-verify-option {required|optional}
    set ssl-session-cache-flag {enable|disable}
    set use-tls-tickets {enable|disable}
    set renegotiation {enable|disable}
    set rfc7919-comply {enable|disable}
    set supported-groups {secp256r1 secp384r1 secp521r1 x25519 x448 ffdhe2048 ffdhe3072
         ffdhe4096 ffdhe6144 ffdhe8192}
    set ssl-dynamic-record-sizing {enable|disable}
    set ssl-dh-param-size {1024bit|2048bit|4096bit}
    set ssl-auto-chain-flag {enable|disable}
  next
end
```

`client-certificate-verify`	Specify a certificate validation policy. **Note**: For VS configurations that reference a ZTNA Profile, ensure the corresponding EMS CA certificate is selected for the corresponding Client SSL profile.
`client-sni-required`	If enabled, clients are required to use the TLS server name indication (SNI) extension to include the server hostname in the TLS client hello message. This will allow FortiADC to select the appropriate local server certificate to present to the client.
`forward-proxy`	Enable/disable SSL forward proxy.
`local-certificate-group`	Configure the local certificate group that includes the certificates the virtual server presents to SSL/TLS clients. Note: This MUST be the backend server's certificate, NOT the appliance's GUI web server certificate.
`ssl-allowed-versions`	Specify the allowed SSL versions in a space-separated list. • sslv3 • tlsv1.0 • tlsv1.1 • tlsv1.2 • tlsv1.3 **Note**: • FortiADC does not support session reuse for SSLv2 at the client side. Instead, a new SSL session is started. Please make sure that the SSL versions are continuous. IF not, an error message should be returned. • RFC 7919 Comply cannot support SSLv3 and TLS 1.3. If **rfc7919-comply** is enabled and **sslv3** or **tlsv1.3** is selected in **ssl-allowed-versions**, an error message will display.
`ssl-ciphers`	Specify the supported SSL ciphers in a space-separated list. Ciphers are listed from strongest to weakest:

	• ECDHE-ECDSA-AES256-GCM-SHA384 • ECDHE-ECDSA-AES256-SHA384 • ECDHE-ECDSA-CAMELLIA256-SHA384 • *ECDHE-ECDSA-AES256-SHA • ECDHE-ECDSA-AES128-GCM-SHA256 • *ECDHE-ECDSA-AES128-SHA256 • ECDHE-ECDSA-CAMELLIA128-SHA256 • *ECDHE-ECDSA-AES128-SHA • ECDHE-ECDSA-DES-CBC3-SHA • ECDHE-ECDSA-RC4-SHA • ECDHE-RSA-AES256-GCM-SHA384 • ECDHE-RSA-AES256-SHA384 • ECDHE-RSA-CAMELLIA256-SHA384 • *ECDHE-RSA-AES256-SHA • DHE-RSA-AES256-GCM-SHA384 • *DHE-RSA-AES256-SHA256 • DHE-RSA-CAMELLIA256-SHA256 • *DHE-RSA-AES256-SHA • DHE-RSA-CAMELLIA256-SHA • AES256-GCM-SHA384 • *AES256-SHA256 • *AES256-SHA • ECDHE-RSA-AES128-GCM-SHA256 • *ECDHE-RSA-AES128-SHA256 • ECDHE-RSA-CAMELLIA128-SHA256 • DHE-RSA-AES128-GCM-SHA256 • *DHE-RSA-AES128-SHA256 • DHE-RSA-CAMELLIA128-SHA256 • *DHE-RSA-AES128-SHA • AES128-GCM-SHA256 • *AES128-SHA256 • *AES128-SHA • ECDHE-RSA-RC4-SHA • RC4-SHA • RC4-MD5 • EDH-RSA-DES-CBC3-SHA • DES-CBC3-SHA • eNULL *These ciphers are fully supported by hardware SSL (in 400F, 1200F, 2200F, 4200F and 5000F).
`ssl-customize-ciphers-flag`	Enable/disable the use of user-specified cipher suites.
`forward-client-certificate`	Enable/disable. If enabled, FortiADC will send the whole client certificate encoded in the BASE64 code in the specified HTTP header, which is either the X-Client-Cert or a

	user-defined header.
`forward-client-certificate-header`	The **forward-client-certificate-header** option is available if **forward-client-certificate** is **enabled**. The default is X-Client-Cert, but you can customize it using this command.
`forward-proxy-certificate-caching`	The **forward-proxy-certificate-caching** is available if **forward-proxy** is **enabled**. Select cache RAM to store re_signed certificates.
`forward-proxy-local-signing-CA`	The **forward-proxy-local-signing-CA** is available if **forward-proxy** is **enabled**. Set the CA used to sign the server certificate.
`forward-proxy-intermediate-ca-group`	The **forward-proxy-intermediate-ca-group** is available if **forward-proxy** is **enabled**. Set the intermediate CA group used to sign the server certificate.
`forward-proxy-resign-cert-by-sni`	The **forward-proxy-resign-cert-by-sni** is available if **forward-proxy** is **enabled**. Enabled by default, the **forward-proxy-resign-cert-by-sni** option allows SSL Forward Proxy to return re-signed local certificates with modified CN (or SAN) matching the ClientHello Server Server Name Indication (SNI). This allows the proxy to act automatically when the SNI is detected in the ClientHello message to return a re-signed local certificate to the client. The Common Name (CN) and/or Subject Alternative Name (SAN) of this certificate will be adeptly modified to align with the SNI, ensuring a seamless and trustworthy SSL handshake process. **Note**: This function is supported HTTP/S virtual servers and real server pools with HTTP/S topology.
`backend-ssl-sni-forward`	The **backend-ssl-sni-forward** is available if **forward-proxy** is **enabled**. Enable/disable forwarding the server's SNI.
`backend-ssl-customize-ciphers-flag`	The **backend-ssl-sni-forward** is available if **forward-proxy** is **enabled**. Enable/disable customized ciphers used to connect to the real server.
`backend-ssl-customized-ciphers ECDSA`	The **backend-ssl-customized-ciphers ECDSA** is available if **forward-proxy** is **enabled**. Set the cipher used to connect to the real server.
`backend-allow-ssl-versions`	The **backend-allow-ssl-versions** is available if **forward-proxy** is **enabled**. Set the SSL version used to connect to the real server.
`backend-ssl-OCSP-stapling-support`	Enable or disable. Disabled by default. Note: This parameter is available only when `backend-certificate-verify` is configured and `forward-proxy` is enabled.
`reject-ocsp-stapling-with-missing-nextupdate`	Enable or disable reject-ocsp-response-with-missing-nextupdate. Disabled by default. Note: When disabled, FortiADC will accept OCSP responses without the next-update time. If enabled, FortiADC will reject OCSP responses without the next-update time.
`reject-revoked-unknown-ocsp-stapling`	Enable or disable reject-revoked-unknown-ocsp-stapling. Enabled by default.

	Note: When enabled, FortiADC will reject OCSP responses whose status is revoked or unknown.
`ocsp-stapling-skew-time`	The default is 0 (in seconds). It means the skew time of this updated time and next updated time.
`ssl-auto-chain-flag`	Enabled by default. It means that when the configured certificate is used in the same client-ssl-profile as the local certificate, and the local certificate is issued by the CA set in the Client Certificate Verify section, ADC will automatically form a certificate chain to the client.
`client-certificate-verify-option`	Choose either of the following: • required—If this option is set as required, then a client certificate is required for verification. • optional—If this option is set as optional, then the system needs to work with a script such as OPTIONAL_CLIENT_AUTHENTICATION. In that case, FortiADC will accept SSL handshake for the initial transaction, and then lets the script to control the subsequent actions.
`ssl-session-cache-flag`	Enable to store SSL session in cache. This option is automatically disabled when the client-certificate-verify-option is set to optional.
`use-tls-tickets`	Enable to allow reusing SSL tickets. This option is automatically disabled when the client-certificate-verify-option is set to optional.
`renegotiation`	Enable or disable SSL renegotiation from the client side. **Note**: The feature is disabled by default.
`rfc7919-comply`	Enable/disable parameters to comply with RFC 7919. **Note**: • RFC 7919 Comply is not supported for Forward Proxy. If **rfc7919-comply** is enabled and **forward-proxy** is enabled, the RFC 7919 Comply feature will not apply to Forward Proxy functionality. • RFC 7919 Comply cannot support SSLv3 and TLS 1.3. If **rfc7919-comply** is enabled and **sslv3** or **tlsv1.3** is selected in **ssl-allowed-versions**, an error message will display. • When **rfc7919-comply** is enabled the **ssl-dh-param-size** option becomes unavailable.
`supported-groups`	The **supported-groups** option is available if **rfc7919-comply** is enabled. Specify the supported group objects from the following: • secp256r1 • secp384r1 • secp521r1 • x25519 • x448 • ffdhe2048 • ffdhe3072 • ffdhe4096 • ffdhe6144

	• ffdhe8192 At least one item from the FFDHE group must be selected. **Note**: The RFC 7919 Comply feature requires certain cipher selections to correspond with the Supported Group selection. • If a FFDHE group is selected (for example, ffdhe2048), then at least one cipher must be DHE-RSA (for example, DHE-RSA-AES256-SHA256). • If the Supported Group includes groups other than FFDHE (such as a SECP group, secp256r1), then at least one cipher must be ECDHE (for example, ECDHE-ECDSA-AES256-GCM-SHA384). • If a ECDHE cipher is selected (for example, ECDHE-ECDSA-AES256-GCM-SHA384), then the Supported Group must include at least one group that is not FFDHE (such as a SECP group, secp256r1).
`ssl-dynamic-record-sizing`	Allows ADC to dynamically adjust the size of TLS records based on the state of the connection, in order to prevent bottlenecks caused by the buffering of TLS record fragments. **Note**: The feature is disabled by default.
`ssl-dh-param-size`	Specify the pubkey length in Diffie Hellman. Default is 1024. **Note**: The **ssl-dh-param-size** option is not available when **rfc7919-comply** is enabled.
`ssl-auto-chain-flag`	Set it to disable to make ADC present only local certificates. **Note:** If the CA, when configured in "Client Certificate Verify," happens to accidentally issue the configured local certificates, the ADC will present chain certificates to the client. In this event, set `ssl-auto-chain-flag` to disable. Default is enable.

Example 1: Create a new client-SSL profile and quote it in virtual server configuration

Step 1: Configure a client SSL profile

```
config load-balance client-sssl-profile
   edit "csp1"
      set ssl-customize-ciphers-flag disable
      set ssl-ciphers DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES256-SHA256 DHE-RSA-AES256-SHA
           AES256-GCM-SHA384 AES256-SHA256 AES256-SHA DHE-RSA-AES128-GCM-SHA256 DHE-RSA-
           AES128-SHA256 DHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA RC4-
           SHA RC4-MD5 EDH-RSA-DES-CBC3-SHA DES-CBC3-SHA EDH-RSA-DES-CBC-SHA DES-CBC-SHA
      set ssl-allowed-versions sslv3 tlsv1.0 tlsv1.1 tlsv1.2
      set forward-proxy enable
      unset client-certificate-verify
      set forward-proxy-certificate-caching LB_CERT_RAM_CACHING_DEFAULT
      set forward-proxy-local-signing-CA SSLPROXY_LOCAL_CA
      unset forward-proxy-intermediate-ca-group
      unset backend-certificate-verify
      set backend-ssl-sni-forward enable
      set backend-ssl-customize-ciphers-flag enable
      set backend-ssl-customized-ciphers test
```

```
      set backend-ssl-allowed-versions sslv3 tlsv1.0 tlsv1.1 tlsv1.2
      set ssl-auto-chain-flag-enable
next
```

Step 2: Quote the client SSL profile in virtual server configuration:

```
config load-balance virtual-server
   edit "https_vS1"
      set client-ssl-profile csp1
   next
end
```

Example 2: Create a certificate-caching object and quote it in the client SSL profile

```
config load-balance certificate-caching
   edit "1"
      set max-certificate-cache-size 100M
      set max-entries 10000
      next
      config load-balance client-ssl-profile
      edit "test"
      set forward-proxy-certificate-caching 1
      set forward-proxy-local-signing-CA ca1
      set forward-proxy-intermediate-ca-group inter_group
      set backend-ssl-sni-forward enable
      set backend-ssl-customize-ciphers-flag disable
      set backend-ssl-customized-ciphers ECDHE-ECDSA-AES256-GCM-SHA384 (when backend-ssl-
           customize-ciphers-flag dis enable)
      set backend-ssl-customize-ciphers-flag enable/disable
      set backend-ssl-ciphers DHE-RSA-AES256-SHA DES-CBC3-SHA
      set backend-allow-ssl-versions tlsv1.1 tlsv1.2
End
```

Example 3: Create a client-certificate-verify object and quote it in the client SSL profile

```
config load-balance client-sssl-profile
   edit "csp1"
      set ssl-customize-ciphers-flag disable
      set ssl-ciphers DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES256-SHA256 DHE-RSA-AES256-SHA
           AES256-GCM-SHA384 AES256-SHA256 AES256-SHA DHE-RSA-AES128-GCM-SHA256 DHE-RSA-
           AES128-SHA256 DHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA RC4-
           SHA RC4-MD5 EDH-RSA-DES-CBC3-SHA DES-CBC3-SHA EDH-RSA-DES-CBC-SHA DES-CBC-SHA
      set ssl-allowed-versions sslv3 tlsv1.0 tlsv1.1 tlsv1.2
      set forward-proxy enable
      unset client-certificate-verify
      set forward-proxy-certificate-caching LB_CERT_RAM_CACHING_DEFAULT
      set forward-proxy-local-signing-CA SSLPROXY_LOCAL_CA
      unset forward-proxy-intermediate-ca-group
      set client-certificate-verify verify
      set client-certificate-verify-option required
      set ssl-session-cache-flag enable
      set use-tls-tickets enable
      set backend-ssl-resign-cert-by-sni enable
```

```
      set backend-ssl-sni-forward enable
      set backend-ssl-customize-ciphers-flag enable
      set backend-ssl-customized-ciphers test
      set backend-ssl-allowed-versions sslv3 tlsv1.0 tlsv1.1 tlsv1.2
   set ssl-auto-chain-flag-enable
next
```

config load-balance clone-pool

Use this command to create a new clone-pool, and to configure clone pool members inside it.

Syntax

```
config load-balance clone-pool
   edit <name>
      config pool_member
         edit <name>
            set mode <mirror-dst-mac-update/mirror-interface/mirror-ip-update/mirror=src-dst-
                 mac-update/mirror-src-mac-update>
            set destination-interface <port>
            set destination-mac <xx:xx:xx:xx:xx:xx>
   next
end
```

Clone Pool	
name	Specify a unique clone pool name
Pool Member	
name	Specify a unique pool member name. Note: A pool member is a clone server. So this name is essentially the name you give to the clone server.
interface	Select the interface (port) FortiADC uses to send out packets to the clone server.
mode	The headers of duplicated packets need to be updated when sent to monitor servers. There are several modes in which this occurs. Select one of the following: • Mirror Interface—This mode does not change the packet header at all. It is most commonly used; with it, the monitor does not look at the content of the packet, neither does it receive the payload, it merely looks at how much data is being passed, and counts the bytes of the data. The original Layer 2 Destination Address (DA) or Source Address (SA) and Layer 3 IP Addresses are left intact. In this mode the FortiADC simply sends the packets "as is" out from the specified interface. • Mirror Destination MAC Address Update—This mode uses Layer 2 forwarding. With the incoming packet, the ADC replaces the

destination MAC address with the specified destination MAC address. It is preferred when connecting the ADC to end devices like the IDS.
- Mirror Source MAC Update—This mode replaces the source MAC address in the incoming packet with the specified MAC address on the FortiADC device. This option is recommended where not changing the source MAC address could cause a loop.
- Mirror Source Destination MAC Update—This mode replaces both the source and destination MAC addresses at Layer 2, but does not change the Layer-3 IP addressing information.
- Mirror IP Update—This mode replaces the incoming packet's IP address with the specified IP address and then forwards the duplicated packet to those servers. This mode may also change the Layer 4 source and destination ports. If the virtual server port isn't set to wildcard port 0 while the port IS specified, the Layer 4 destination port on the duplicated packets will be changed to the specified value. This option is recommended for scenarios in which monitor servers are not directly connected to the ACOS device.

Example

```
FortiADC-VM (root) # config load-balance clone-pool
FortiADC-VM (clone-pool) # edit 1
FortiADC-VM (1) # config pool_member
FortiADC-VM (pool_member) # edit name
FortiADC-VM (name) # set
FortiADC-VM (name) # set mode mirror-dst-mac-update

FortiADC-VM (name) # get
mode : mirror-dst-mac-update
destination-interface :
destination-mac : 00:00:00:00:00:00

next
end
```

config load-balance compression

Use this command to configure compression options.

The following content types can be compressed:

- application/javascript
- application/soap+xml
- application/x-javascript
- application/xml

- text/css
- text/html
- text/javascript
- text/plain
- text/xml

Not all HTTP responses should be compressed. Compression offers the greatest performance improvements when applied to URLs whose media types compress well, such as repetitive text such as tagged HTML, and scripts such as JavaScript. Files that already contain efficient compression such as GIF images usually should not be compressed, as the CPU usage and time spent compressing them will result in an increased delay rather than network throughput improvement. Plain text files where no words are repeated, such as configurations with unique URLs or IPs, also may not be appropriate for compression.

Before you begin:

- You must have a good understanding of HTTP compression and knowledge of the content types served from the backend real servers.
- You must have read-write permission for load balancing settings.

Compression is not enabled by default. After you have configured a compression rule, you can select it in the profile configuration. To enable compression, select the profile when you configure the virtual server.

Syntax

```
config load-balance compression
   edit <name>
      set cpu-limit <integer>
      set max-cpu-usage <integer>
      set min-content-length <integer>
      set uri-list-type {include | exclude>
      config uri_list
         edit <No.>
            set uri <string>
         next
      end
      config content_types
         edit <No.>
            set content-type {application/javascript | application/soap+xml | application/x-
                 javascript | application/xml | text/css | text/html | text/javascript |
                 text/plain | text/xml}
         next
      end
   next
end
```

cpu-limit	Enable/disable application of a CPU limit.
max-cpu-usage	Maximum CPU usage for compression operations. The default is 80.
min-content-length	Do not compress files smaller than this size. The default is 1024 bytes.
uri-list-type	Specify whether to include or exclude items in the list from compression.
config uri_list	

uri	Specify URIs to build a list or sites to include/exclude from compression. You can use regular expressions.
config content_type	
content-type	• application/javascript • application/soap+xml • application/x-javascript • application/xml • text/css • text/html • text/javascript • text/plain • text/xml

Example

```
FortiADC-VM (compression) # config load-balance compression
FortiADC-VM (compression) # edit lb-compression
Add new entry 'lb-compression' for node 1627

FortiADC-VM (lb-compression) # get
min-content-length : 1024
cpu-limit : enable
max-cpu-usage : 80
uri-list-type : exclude

FortiADC-VM (lb-compression) # set max-cpu-usage 50
FortiADC-VM (lb-compression) # end
```

config load-balance connection-pool

Use this command to configure connection pool settings.

A connection pool enables Layer 7 load balancing virtual servers to “reuse” existing TCP connections. Using a connection pool can reduce the impact of TCP overhead on web server and application performance.

Before you begin:

- You must have read-write permission for load balancing feature settings.

After you have created a connection pool configuration, you can specify it in a virtual server configuration.

Note: The feature is not supported for profiles with the Source Address option enabled.

Syntax

```
config load-balance connection-pool
   edit <name>
```

```
      set age <integer>
      set reuse <integer>
      set size <integer>
      set timeout <integer>
   next
end
```

age	Maximum duration of a connection in seconds. The recommended value is 3000.
reuse	Maximum number of times that the virtual server can reuse the connection. The recommended value is 2000.
size	Maximum number of connections in the connection pool. The recommended value is 0, which specifies that there is no limit on the connection size.
timeout	Maximum number of seconds a connection can be idle before the system deletes it. The recommended value is 30.

Predefined connection-pool

```
config load-balance connection-pool
   edit "LB_CONNECTION_POOL_DEFAULT"
      set size 10000
      set age 86400
      set reuse 10000
      set timeout 50
   next
end
```

Example

```
FortiADC-VM # config load-balance connection-pool

FortiADC-VM (connection-pool) # edit lb-connection-pool
Add new entry 'lb-connection-pool' for node 1698

FortiADC-VM (lb-connec~i) # get
size : 10000
age : 86400
reuse : 10000
timeout : 50

FortiADC-VM (lb-connec~i) # set age 3000
FortiADC-VM (lb-connec~i) # set reuse 2000
FortiADC-VM (lb-connec~i) # set size 0
FortiADC-VM (lb-connec~i) # set timeout 30

FortiADC-VM (lb-connec~i) # get
size : 0
age : 3000
reuse : 2000
timeout : 30

FortiADC-VM (example-connec~i) # end
```

config load-balance content-rewriting

Use this command to configure content rewriting rules.

You might rewrite the HTTP headers for various reasons, including the following:

- Redirect HTTP to HTTPS—You can use the content rewriting feature to send redirects when the requested resource requires a secure connection. For example, create a rule that matches requests to `http://example.com/resource` with an action to send a redirect that has the secure URL in the Location header: `https://example.com/resource`.
- External-to-internal URL translation—It is standard for web servers to have external and internal domain names. You can use content-based routing to forward HTTP requests to example.com to a server pool that includes server1.example.com, server2.example.com, and server3.example.com. When you use content routing like this, you should also rewrite the Location header in the HTTP response so that the client receives HTTP with example.com in the header and not the internal domain server1.example.com. Create a rule that matches the regular expression `server.*\.example\.com` in the Location header of the HTTP response with an action to rewrite the Location header with the public URL `http://example.com`.
- Other security reasons—Another use case for external-to-internal URL translation involves masking pathnames that give attackers information about your web applications. For example, the unmasked URL for a blog might be http://www.example.com/wordpress/?feed=rss2, which exposes that the blog is a wordpress application. In this case, you want to publish an external URL that does not have clues of the underlying technology. For example, in your web pages, you create links to http://www.example.com/blog. On FortiADC, you create a rule that matches requests to http://www.example.com/resource2 with an action to rewrite the URL to the internal URL `http://www.example.com/wordpress/?feed=rss2`. For the return traffic, you create another rule that matches `http://www.example.com/wordpress/?feed=rss2` in the Location header of the HTTP response with an action to rewrite it with the public URL `http://www.example.com/blog`.

Table 7 summarizes the HTTP header fields that can be rewritten.

HTTP header rewriting

Direction	HTTP Header
HTTP Request	Host Referer
HTTP Redirect	Location
HTTP Response	Location

The first line of an HTTP request includes the HTTP method, relative URL, and HTTP version. The next lines are headers that communicate additional information. The following example shows the HTTP request for the URL http://www.example.com/index.html:

```
GET /index.html HTTP/1.1
Host: www.example.com
Referer: http://www.google.com
```

The following is an example of an HTTP redirect including the HTTP Location header:

```
HTTP/1.1 302 Found
Location: http://www.iana.org/domains/example/
```

You can use literal strings or regular expressions to match traffic to rules. To match a request URL such as http://www.example.com/index, you create two match conditions: one for the Host header `www.example.com` and another for the relative URL that is in the GET line: `/index.html`.

For HTTP redirect rules, you can specify the rewritten location as a literal string or as a regular expression. For all other types or rules, you must specify the complete URL as a literal string.

Before you begin:

- You must have a good understanding of HTTP header fields.
- You must have a good understanding of Perl-compatible regular expressions (PCRE) if you want to use them in rule matching or rewriting.
- You must have read-write permission for load balancing settings.

After you have configured a content rewriting rule, you can select it in the virtual server configuration.

Note: You can select multiple content rewriting rules in the virtual server configuration. Rules you add to that configuration are consulted from top to bottom. The first to match is applied. If the traffic does not match any of the content rewriting rule conditions, the header is not rewritten.

Syntax

```
config load-balance content-rewriting
   edit <name>
      set action-type {request|response>
      set action {add_http_header | delete_http_header | redirect | rewrite_http_header |
           rewrite_http_location | send-403-forbidden}
      set header-name [string/regular express]
      set header-value [string/regular express]
      set redirect <string>
      set host-status {enable|disable}
      set host <string>
      set referer-status {enable|disable}
      set referer <string>
      set url-status {enable|disable}
      set url <string>
      set location <string>
      set comments <string>
      config match-condition
         edit <No.>
            set content <string>
            set object {http-host-header | http-location-header | http-referer-header | http-
                 request-url | ip-source-address}
            set reverse {enable|disable}
            set type {string | regular-expression}
            set ignore-case {enable | disable}
         next
      end
   next
end
```

`action-type`	Specify whether to rewrite the HTTP request or HTTP response.
`action`	If you configure a rule based on the HTTP request, you can specify the following actions: • add_http_header

	• delete_http_header • rewrite_http_header • redirect • send-403-forbidden If you configure a rule based on the HTTP response, you can specify the following action: • rewrite_http_location
`header-name`	Creates a new header or deletes an existing header with the header name. Use this command only if action is set to add_http_header or delete_http_header.
`header-value`	Creates a new header or deletes an existing header with the header value. Use this command only if action is set to add_http_header or delete_http_header.
`redirect`	Sends a redirect with the URL you specify in the HTTP Location header field. For Redirect rules, specify an absolute URL. For example: `https://example.com/content/index.html` Use this command only if action is set to redirect. **Note**: The rewrite string can be a literal string or a regular expression.
`host-status`	Enable/disable rewriting the Host header by replacing the hostname with the string you specify. Use this command only if action is set to rewrite_http_header.
`host`	Rewrites the Host header by replacing the hostname with the string you specify. For Host rules, specify a replacement domain and/or port. Use this command only if action is set to rewrite_http_header. **Note**: The rewrite string is a literal string. Regular expression syntax is not supported.
`referer-status`	Enable/disable rewriting the Referer header with the URL you specify. Use this command only if action is set to rewrite_http_header.
`referer`	Rewrites the Referer header with the URL you specify. For Referer rules, you must specify an absolute URL. **Note**: The rewrite string is a literal string. Regular expression syntax is not supported.
`url-status`	Enable/disable rewriting the Host header by replacing the whole URL with the string you specify.
`url`	Rewrites the request URL and Host header using the string you specify. For URL rules, specify a URL in one of the following formats: • Absolute URL — `https://example.com/content/index.html` • Relative URL — `content/index.html` If you specify a relative URL, the host header is not rewritten. **Note**: The rewrite string is a literal string. Regular expression syntax is not supported.
`location`	For Location rules, specify an absolute URL. For example: `https://example.com/content/index.html` **Note**: The rewrite string is a literal string. Regular expression syntax is not supported.
`comments`	Optional administrator note.

config match-condition	
content	Specify the string or regular expression syntax.
object	Specify content matching conditions based on the following parameters: • http-host-header • http-location-header • http-referer-header • http-request-url • ip-source-address **Note**: When you add multiple conditions, FortiADC joins them with an AND operator. For example, if you specify both a HTTP Host Header and HTTP Request URL to match, the rule is a match only for traffic that meets both conditions.
reverse	Rule matches if traffic does not match the expression.
type	• string • regular-expression
ignore case	If the match rule is case sensitive, it will ignore case.

Example

The following example creates a configuration to rewrite a literal string:

```
FortiADC-VM # config load-balance content-rewriting
FortiADC-VM (content-rewrit~n) # edit c-rewrite-0
Add new entry 'c-rewrite-0' for node 1737

FortiADC-VM (c-rewrite-0) # set action redirect
FortiADC-VM (c-rewrite-0) # set redirect https://example.com/resource
FortiADC-VM (c-rewrite-0) # set comments http-to-https

FortiADC-VM (c-rewrite-0) # config match-condition
FortiADC-VM (match-condition) # edit 1
FortiADC-VM (1) # set type string
FortiADC-VM (1) # set object http-host-header
FortiADC-VM (1) # set content www.example.com
FortiADC-VM (1) # next

FortiADC-VM (match-condition) # edit 2
FortiADC-VM (2) # set type string
FortiADC-VM (2) # set object http-request-url
FortiADC-VM (2) # set content /resource
FortiADC-VM (2) # end
```

The following example creates a configuration to rewrite using a regular expression:

```
FortiADC-VM (content-rewrit~n) # edit c-rewrite-1

FortiADC-VM (c-rewrite-1) # set action redirect
FortiADC-VM (c-rewrite-1) # set redirect https://$0/$1
FortiADC-VM (c-rewrite-1) # set comments http-to-https

FortiADC-VM (c-rewrite-1) # config match-condition
```

```
FortiADC-VM (match-condition) # edit 1
FortiADC-VM (1) # set type regular-expression
FortiADC-VM (1) # set object http-host-header
FortiADC-VM (1) # set content (.*)
FortiADC-VM (1) # next

FortiADC-VM (match-condition) # edit 2
FortiADC-VM (2) # set type regular-expression
FortiADC-VM (2) # set object http-request-url
FortiADC-VM (2) # set content ^/(.*)$
FortiADC-VM (2) # end
```

config load-balance content-routing

Use this command to configure content routing.

Content routes select the backend server pool based on matches to TCP/IP or HTTP header values.

Layer 7 content route rules are based on matches to the following header values:

- HTTP Host
- HTTP Referer
- HTTP Request URL
- SNI
- Source IP address

You might want to use Layer 7 content routes to simplify front-end coding of your web pages or to obfuscate the precise server names from clients. For example, you can publish links to a simple URL named example.com and use content route rules to direct traffic for requests to example.com to a server pool that includes server1.example.com, server2.example.com, and server3.example.com.

Layer 4 and Layer 2 content route rules are based on matches to the following header values:

- Source IP address

Note: Layer 4 content rules can be used by both Layer 4 virtual servers and Layer 2 virtual servers.

Before you begin:

- You must have a good understanding of HTTP header fields.
- You must have a good understanding of Perl-compatible regular expressions (PCRE) if you want to use them in rule matching.
- You must have read-write permission for load balancing settings.

After you have configured a content routing rule, you can select it in the virtual server configuration.

Note: You can select multiple content routing rules in the virtual server configuration. Rules you add to that configuration are consulted from top to bottom. The first rule to match is applied. If the traffic does not match any of the content routing rule conditions specified in the virtual server configuration, the system behaves unexpectedly. Therefore, it is important that you create a “catch all” rule that has no match conditions. In the virtual server configuration, this rule should be ordered last so it can be used to forward traffic to a default pool.

Syntax

```
config load-balance content-routing
   edit <name>
      set type {l4-content-routing|l7-content-routing}
      set ip <ip&netmask>
      set ip6 <ip&netmask>
      set connection-pool inherit {enable|disable}
      set connection-pool <datasource>
      set load-balance-pool <datasource>
      set method-inherit {enable|disable}
      set load-balance-method <datasource>
      set persistence-inherit {enable|disable}
      set load-balance-persistence <datasource>
      set comments <string>
      set schedule-list {enable|disable}
      set schedule-pool-list <datasource>
      set packet-forwarding-method {inherit|FullNAT}
      set ippool-list <datasource>
      config match-condition
         edit <No.>
            set content <string>
            set object {http-host-header | http-referer-header | http-request-url | ip-
                 source-address | sni}
            set reverse {enable|disable}
            set type {string | regular-expression}
            set ignore-case {enable|disable}
         next
      end
   next
end
```

type	l4-content-routing l7-content-routing
ip	The **ip** option is available if **type** is **l4-content-routing**. Address/mask notation to match the source IPv4 address in the packet header.
ip6	The **ip6** option is available if **type** is **l4-content-routing**. Address/mask notation to match the source IPv6 address in the packet header.
connection-pool-inherit	Enable to use the connection pool configuration object specified in the virtual server configuration.
connection-pool	If not using inheritance, specify the connection pool.
load-balance-pool	Specify a real server pool.
method-inherit	Enable (default) to use the method specified in the virtual server configuration.
load-balance-method	If not using inheritance, select a load balancing method type.
persistence-inherit	Enable (default) to use the persistence object specified in the virtual server configuration.
load-balance-persistence	If not using inheritance, select a session persistence type.

comments	Optional administrator note.
schedule-list	Enable/disable schedule pool list.
schedule-pool-list	Specify the schedule-pool.
packet-forwarding-method	The **packet-forwarding-method** option is available if **type** is **l4-content-routing**. Select the packet forwarding method: • inherit • FullNAT
ippool-list	The **ippool-list** option is available if **type** is **l4-content-routing** and **packet-forwarding-method** is **FullNAT**. Specify the source pool objects for packet forwarding.
config match-condition The **config match-condition** is only applicable if **type** is **l7-content-routing**.	
content	Specify the string or regular expression syntax.
object	Specify content matching conditions based on the following parameters: • http-host-header • http-referrer-header • http-request-url • sni • ip-source-address **Note**: When you add multiple conditions, FortiADC joins them with an AND operator. For example, if you specify both a HTTP Host Header and HTTP Request URL to match, the rule is a match only for traffic that meets both conditions.
reverse	Rule matches if traffic does not match the expression.
type	The **type** option is **not available** if **object** is **ip-source-address**. • string • regular-expression
ignore-case	The **ignore-case** option is **not available** if **object** is **ip-source-address**. If the match case is case sensitive, it will ignore case.

Example

```
FortiADC-VM # config load-balance content-routing
FortiADC-VM (content-routing) # edit example.com
Add new entry 'example.com' for node 1756

FortiADC-VM (example.com) # get
type : l7-content-routing
persistence-inherit : enable
load-balance-persistence:
method-inherit : enable
load-balance-method :
connection-pool :
connection-pool-inherit: disable
load-balance-pool :
```

```
comments : comments

FortiADC-VM (example.com) # set persistence-inherit enable
FortiADC-VM (example.com) # set method-inherit enable
FortiADC-VM (example.com) # set load-balance-pool example-pool
FortiADC-VM (example.com) # set comments external-to-internal-name-map
FortiADC-VM (example.com) # config match-condition
FortiADC-VM (match-condition) # edit 1
Add new entry '1' for node 1768

FortiADC-VM (1) # get
object : http-host-header
type : regular-expression
content : match
reverse : disable

FortiADC-VM (1) # set type string
FortiADC-VM (1) # set content http://example.com
FortiADC-VM (1) # set object http-request-url
FortiADC-VM (1) # end


FortiADC-VM (example.com) # get
type : l7-content-routing
persistence-inherit : enable
method-inherit : enable
connection-pool :
connection-pool-inherit: disable
load-balance-pool : example-pool
== [ 1 ]
comments : external-to-internal-name-map

FortiADC-VM (example.com) # show
config load-balance content-routing
edit "example.com"
set persistence-inherit enable
set method-inherit enable
set load-balance-pool example-pool
config match-condition
edit 1
set object http-request-url
set type string
set content http://example.com
next
end
set comments external-to-internal-name-map
next
end
FortiADC-VM (example.com) # end
```

config load-balance error-page

Deprecated. You must use the web UI to upload an error page and create an error page configuration object.

config load-balance geoip-list

Use this command to configure the Geo IP address block list.

The FortiGuard Geo IP service provides a database that maps IP addresses to countries, satellite providers, and anonymous proxies. The database is updated periodically.

The Geo IP block list is a policy that takes the action you specify when the virtual server receives requests from IP addresses in the blocked country's IP address space.

For Layer 4 virtual servers, FortiADC blocks access when the first TCP SYN packet arrives. For Layer 7 virtual servers, FortiADC blocks access after the handshake, allowing it to redirect the traffic if you have configured it to do so.

Basic Steps

1. Configure the connection to FortiGuard so the system can receive periodic Geo IP Database updates.
2. Create rules to block traffic from locations.
3. Maintain a allowlist to allow traffic from specified subnets even if they belong to the address space blocked by the Geo IP block list.
4. Select the Geo IP block list and allowlist in the profiles you associate with virtual servers.

Before you begin:

- You must have read-write permission for load balancing settings.

Syntax

```
config load-balance geoip-list
   edit <name>
      set action {deny | pass | redirect | send-403-forbidden}
      set log {enable|disable}
      set severity {high | low | medium}
      set status {enable|disable}
      config geoip-member
      edit <No.>
         set region-list <country-code>
      next
   next
end
```

action	• Pass • Deny • Redirect (you can specify a redirect URL in the virtual server configuration)

	• Send 403 Forbidden **Note**: Layer 4 and TCPS virtual servers do not support Redirect or Send 403 Forbidden. If you apply a configuration that uses these options to a Layer 4 or TCPS virtual server, FortiADC logs the action as Redirect or Send 403 Forbidden, but in fact denies the traffic.
`log`	Enable/disable logging.
`severity`	The severity to apply to the event. Severity is useful when you filter and sort logs: • low • medium • high
`status`	Enable/disable the list.
config geoip-member	
`region-list`	Specify a geolocation object. Type `?` to see a list. The list includes countries as well as selections for anonymous proxies and satellite providers.

Example

```
FortiADC-VM # config load-balance geoip-list

FortiADC-VM (geoip-list) # edit demo
Add new entry 'demo' for node 2883

FortiADC-VM (demo) # get
log : disable
action : deny
severity : low
status : enable

FortiADC-VM (demo) # set log enable
FortiADC-VM (demo) # set severity high

FortiADC-VM (demo) # config geoip-member
FortiADC-VM (geoip-member) # edit 1
Add new entry '1' for node 2888

FortiADC-VM (1) # set region-list ?
ZZ Reserved
A1 Anonymous Proxy
A2 Satellite Provider
O1 Other Country
AD Andorra
AE United Arab Emirates
AF Afghanistan
AG Antigua and Barbuda
AI Anguilla
AL Albania
AM Armenia
AN Netherlands Antilles
AO Angola
AP Asia/Pacific Region
AQ Antarctica
```

```
AR Argentina
AS American Samoa
AT Austria
AU Australia
AW Aruba
AX Aland Islands
AZ Azerbaijan
BA Bosnia and Herzegovina
BB Barbados
BD Bangladesh
BE Belgium
BF Burkina Faso
BG Bulgaria
BH Bahrain
BI Burundi
BJ Benin
BL Saint Bartelemey
BM Bermuda
BN Brunei Darussalam
BO Bolivia
BQ Bonaire, Saint Eustatius and Saba
BR Brazil
BS Bahamas
BT Bhutan
BV Bouvet Island
BW Botswana
BY Belarus
BZ Belize
CA Canada
CC Cocos (Keeling) Islands
CD Congo, The Democratic Republic of the
CF Central African Republic
CG Congo
CH Switzerland
CI Cote d'Ivoire
CK Cook Islands
CL Chile
CM Cameroon
CN China
CO Colombia
CR Costa Rica
CU Cuba
CV Cape Verde
CW Curacao
CX Christmas Island
CY Cyprus
CZ Czech Republic
DE Germany
DJ Djibouti
DK Denmark
DM Dominica
DO Dominican Republic
DZ Algeria
EC Ecuador
EE Estonia
EG Egypt
EH Western Sahara
```

```
ER Eritrea
ES Spain
ET Ethiopia
EU Europe
FI Finland
FJ Fiji
FK Falkland Islands (Malvinas)
FM Micronesia, Federated States of
FO Faroe Islands
FR France
GA Gabon
GB United Kingdom
GD Grenada
GE Georgia
GF French Guiana
GG Guernsey
GH Ghana
GI Gibraltar
GL Greenland
GM Gambia
GN Guinea
GP Guadeloupe
GQ Equatorial Guinea
GR Greece
GS South Georgia and the South Sandwich Islands
GT Guatemala
GU Guam
GW Guinea-Bissau
GY Guyana
HK Hong Kong
HM Heard Island and McDonald Islands
HN Honduras
HR Croatia
HT Haiti
HU Hungary
ID Indonesia
IE Ireland
IL Israel
IM Isle of Man
IN India
IO British Indian Ocean Territory
IQ Iraq
IR Iran, Islamic Republic of
IS Iceland
IT Italy
JE Jersey
JM Jamaica
JO Jordan
JP Japan
KE Kenya
KG Kyrgyzstan
KH Cambodia
KI Kiribati
KM Comoros
KN Saint Kitts and Nevis
KP Korea, Democratic People's Republic of
KR Korea, Republic of
```

```
KW Kuwait
KY Cayman Islands
KZ Kazakhstan
LA Lao People's Democratic Republic
LB Lebanon
LC Saint Lucia
LI Liechtenstein
LK Sri Lanka
LR Liberia
LS Lesotho
LT Lithuania
LU Luxembourg
LV Latvia
LY Libyan Arab Jamahiriya
MA Morocco
MC Monaco
MD Moldova, Republic of
ME Montenegro
MF Saint Martin
MG Madagascar
MH Marshall Islands
MK Macedonia
ML Mali
MM Myanmar
MN Mongolia
MO Macao
MP Northern Mariana Islands
MQ Martinique
MR Mauritania
MS Montserrat
MT Malta
MU Mauritius
MV Maldives
MW Malawi
MX Mexico
MY Malaysia
MZ Mozambique
NA Namibia
NC New Caledonia
NE Niger
NF Norfolk Island
NG Nigeria
NI Nicaragua
NL Netherlands
NO Norway
NP Nepal
NR Nauru
NU Niue
NZ New Zealand
OM Oman
PA Panama
PE Peru
PF French Polynesia
PG Papua New Guinea
PH Philippines
PK Pakistan
PL Poland
```

```
PM Saint Pierre and Miquelon
PN Pitcairn
PR Puerto Rico
PS Palestinian Territory
PT Portugal
PW Palau
PY Paraguay
QA Qatar
RE Reunion
RO Romania
RS Serbia
RU Russian Federation
RW Rwanda
SA Saudi Arabia
SB Solomon Islands
SC Seychelles
SD Sudan
SE Sweden
SG Singapore
SH Saint Helena
SI Slovenia
SJ Svalbard and Jan Mayen
SK Slovakia
SL Sierra Leone
SM San Marino
SN Senegal
SO Somalia
SR Suriname
SS South Sudan
ST Sao Tome and Principe
SV El Salvador
SX Sint Maarten
SY Syrian Arab Republic
SZ Swaziland
TC Turks and Caicos Islands
TD Chad
TF French Southern Territories
TG Togo
TH Thailand
TJ Tajikistan
TK Tokelau
TL Timor-Leste
TM Turkmenistan
TN Tunisia
TO Tonga
TR Turkey
TT Trinidad and Tobago
TV Tuvalu
TW Taiwan
TZ Tanzania, United Republic of
UA Ukraine
UG Uganda
UM United States Minor Outlying Islands
US United States
UY Uruguay
UZ Uzbekistan
VA Holy See (Vatican City State)
```

```
VC Saint Vincent and the Grenadines
VE Venezuela
VG Virgin Islands, British
VI Virgin Islands, U.S.
VN Vietnam
VU Vanuatu
WF Wallis and Futuna
WS Samoa
XK Kosovo
YE Yemen
YT Mayotte
ZA South Africa
ZM Zambia
ZW Zimbabwe
CN11 China,Beijing
CN12 China,Tianjin
CN13 China,Hebei
CN14 China,Shanxi(Taiyuan)
CN15 China,Neimenggu
CN21 China,Liaoning
CN22 China,Jilin
CN23 China,Heilongjiang
CN31 China,Shanghai
CN32 China,Jiangsu
CN33 China,Zhejiang
CN34 China,Anhui
CN35 China,Fujian
CN36 China,Jiangxi
CN37 China,Shandong
CN41 China,Henan
CN42 China,Hubei
CN43 China,Hunan
CN44 China,Guangdong
CN45 China,Guangxi
CN46 China,Hainan
CN50 China,Chongqing
CN51 China,Sichuan
CN52 China,Guizhou
CN53 China,Yunnan
CN54 China,Xizang
CN61 China,Shaanxi(Xian)
CN62 China,Gansu
CN63 China,Qinghai
CN64 China,Ningxia
CN65 China,Xinjiang

FortiADC-VM (1) # set region-list FM

FortiADC-VM (1) # get
region-list : FM
FortiADC-VM (1) # end

FortiADC-VM (demo) # get
log : enable
action : deny
severity : high
status : enable
```

```
== [ 1 ]

FortiADC-VM (demo) # end
```

config load-balance http2-profile

This command is used by HTTP or HTTPS profiles. You must enable the HTTP/2 gateway function to use this profile.

Profile	Description
LB_HTTP2_PROFILE_DEFAULT	Set max-concurrent-stream 5 max-receive-window 65535 max-frame-size 16384 header-table-size 4096 max-header-list-size 65536 ssl-constraint enabled

Syntax

```
config load-balance http2-profile
   edit <profile name>
      set priority-mode best-effort
      set upgrade-mode upgradeable
      set max-concurrent-stream <integer>
      set max-receive-window <integer>
      set max-frame-size <integer>
      set header-table-size <integer>
      set max-header-list-size <integer>
      set ssl-constraint disable/enable
```

`header-table-size`	The size of header table for HPACK.
`max-concurrent-stream`	The maximum number of concurrent streams.
`max-frame-size`	The maximum size of a frame.
`max-header-list-size`	The maximum size of the header list.
`max-receive-window`	The maximum size of the receive window.
`priority-mode`	The priority of stream mode.
`ssl-constraint`	The SSL constraint check.
`upgrade-mode`	The protocol upgrade to HTTP/2 mode.

Example

```
config load-balance http2-profile
```

```
   edit "http2"
      set priority-mode best-effort
      set upgrade-mode upgradeable
      set max-concurrent-stream 5
      set max-receive-window 32767
      set max-frame-size 16384
      set header-table-size 4096
      set max-header-list-size 65536
      set ssl-constraint disable
   next
end
```

config load-balance http3-profile

HTTP/3 is the latest version of the HTTP protocol and unlike previous versions which relied on TCP to handle streams in the HTTP layer, HTTP/3 uses QUIC (Quick UDP Internet Connections), a multiplexed transport protocol built on UDP. The HTTP/3 protocol has a lower latency as a result of using QUIC, allowing it to have a quicker handshake for establishing a secure session compared to HTTP/2 which achieves this using TCP and TLS.

Use the `config load-balance http3-profile` command to configure an HTTP3 Profile configuration that can then be referenced by HTTPS application profiles. Once referenced, the HTTPS profile becomes a HTTP/3 load balance profile and the virtual server that references the profile becomes a HTTP/3 VS. This HTTP/3 VS can only operate under L7-HTTPS VS.

HTTP/3 VS listens to the TCP port and the corresponding UDP port at the same time.

FortiADC does not support server side HTTP/3, instead support is provided for client HTTP/3 to the ADC and then converted to HTTP/1 (conversion to HTTP/2 is not supported).

In version 7.4.0, FortiADC is introducing HTTP/3 support as an experimental feature with limited HTTP/3 functionality, so it is not recommended to be used in production environments. For details, see HTTP/3 supported functionality and limitations on page 162.

A predefined profile is available to be referenced in HTTPS application profiles. All values in the predefined profile is view-only and cannot be modified.

Profile	Description
LB_HTTP3_PROFILE_DEFAULT	max-streams — 5 max-idle-timeout — 50 connection-tx-buffers — 30 quic-cc-algo — cubic

Syntax

```
config load-balance http3-profile
   edit <name>
      set max-streams <integer>
      set max-idle-timeout <integer>
```

```
    set connection-tx-buffers <integer>
    set quic-cc-algo {cubic|newreno}
  next
end
```

max-streams	Specify the maximum allowable number of HTTP/3 QUIC streams. The default value is 5, and the range is 1-200.
max-idle-timeout	Specify the HTTP/3 QUIC connection idle timeout in seconds. When no data is transmitted over the HTTP/3 connection after the specified time has elapsed, the HTTP/3 connection will timeout. The HTTP/3 connection is tracked using a unique connection-ID instead of a UDP session. The default value is 50 seconds, and the range is 1-86400 seconds.
connection-tx-buffers	Specify the number of buffers to send on the HTTP/3 QUIC connection. This parameter significantly affects the performance of the HTTP/3 response direction. The higher the number of buffers are sent, the higher the performance will be. However, the memory usage increases. The default value is 30, and the range is 5-100.
quic-cc-algo	FortiADC supports Cubic and New Reno loss-based congestion control for QUIC, where the congestion control responds to packet loss events. Select the QUIC congestion algorithm to use: • cubic • newreno Cubic is the default congestion control algorithm.

Example

```
config load-balance http3-profile
  edit 1
    set max-streams 5
    set max-idle-timeout 50
    set connection-tx-buffers 30
    set quic-cc-algo cubic
  next
end
```

HTTP/3 supported functionality and limitations

HTTP/3 support is currently an experimental feature with limited HTTP/3 functionality, so it is not recommended to be used in production environments.

Key limitations:

- HTTP/3 only operates under L7-HTTPS VS.
- HTTP/3 VS does not support dynamic configuration.
- HTTP/3 VS does not support session and persistence table display.

- HTTP/3 VS does not support HTTP detailed information statistics.
- HTTP/3 is only supported on VS, and the backend (RS) only supports HTTP/1.1.

The current iteration of the HTTP/3 feature is supported in limited or conditional capacity. The following lists the configurations that currently support HTTP/3 functionality and in what capacity.

Configuration	Supported HTTP/3 functionality
load-balance profile	Profile **type** must be **https** to reference HTTP3 profiles.
load-balance virtual-server	• VS **type** must be Layer 7 to reference HTTP3 profiles. • Number of ports must be set to one port only, multiple ports is not supported. • Alone mode must be enabled.
load-balance method	Supported load balancing methods: • round-robin • least-connection • host-hash • host-domain-hash • uri-hash • full-uri-hash • dynamic load balance
load-balance persistence	Supported persistence types: • consistent-hash-ip • embedded-cookie • hash-cookie • hash-http-header • hash-http-request • hash-source-address-port • insert-cookie • persistent-cookie • rewrite-cookie • ssl-session-id
Client SSL Profile	Allowed SSL Versions — SSLv3, TLSv1.0, TLSv1.1, TLSv1.2 and TLSv1.3
Real Server SSL Profile	Allowed SSL Versions — SSLv3, TLSv1.0, TLSv1.1, TLSv1.2 and TLSv1.3

config load-balance ippool

Use this command to configure a NAT IP address range pool to be used in a Layer 4 virtual server deployment

In a Layer 4 virtual server configuration, you select a “packet forwarding method” that includes the following network address translation (NAT) options:

- Direct Routing—Does not rewrite source or destination IP addresses.
- DNAT—Rewrites the destination IP address for packets before it forwards them.
- Full NAT—Rewrites both the source and destination IP addresses. Use for standard NAT, when client and server IP addresses are all IPv4 or all IPv6.
- NAT46—Rewrites both the source and destination IP addresses. Use for NAT 46, when client IP addresses are IPv4 and server IP addresses are IPv6.
- NAT64—Rewrites both the source and destination IP addresses. Use for NAT 64, when client IP addresses are IPv6 and server IP addresses are IPv4.

In a Layer 7 virtual server configuration, you do not select a packet forwarding option. Layer 7 virtual servers use NAT46 and NAT64 to support those traffic flows, but they do not use the Source Pool configuration.

See the FortiADC Handbook for example usage.

Before you begin:

- You must have a good understanding of NAT. You must know the address ranges your network has provisioned for NAT.
- Be sure to configure the backend servers to use the FortiADC address as the default gateway so that server responses are also rewritten by the NAT module.
- You must have read-write permission for load balancing settings.

After you have configured a source pool IP address range configuration object, you can select it in the virtual server configuration. You can assign a virtual server multiple source pools (with the same or different source pool interface associated with it).

There are no validation checks for duplicate addresses in the NAT source pool for HA synchronization, SNAT, 1-to-1 NAT, and VIP, as the ha-mgmt-ip is not synchronized between the HA nodes. For these configurations, ensure the starting and ending IPs in the address range of the NAT source pool are not duplicates.

Syntax

```
config load-balance ippool
   edit <No.>
      set interface <datasource>
      set addr-type {ipv4|ipv6}
      set ip-min <class_ip>
      set ip-max <class_ip>
      config node-member
         edit <name>
            set ha-node <integer>
            set interface <datasource>
            set addr-type {ipv4|ipv6}
            set ip-min <class_ip>
            set ip-max <class_ip>
         next
      end
   next
end
```

interface	Interface to receive responses from the backend server. The interface used for the initial client traffic is determined by the virtual server configuration.
addr-type	IPv4 or IPv6
ip-min	The first address in the address pool.
ip-max	The last address in the address pool.
config node-member	
<name>	Create a node member list to be used in an HA active-active deployment when the node interfaces have multiple IP addresses. Name is a configuration name. Valid characters are A-Z, a-z, 0-9, _, and -. No spaces. You reference this name in the virtual server configuration. **Note**: After you initially save the configuration, you cannot edit the name.
ha-node	Specify the HA cluster node ID.
interface	Interface to receive responses from the backend server. The interface used for the initial client traffic is determined by the virtual server configuration.
addr-type	IPv4 or IPv6
ip-min	The first address in the address pool.
ip-max	The last address in the address pool.

config load-balance l2-exception-list

Use this command to configure an exception list for SSL forward proxy decryption. You can leverage FortiGuard web filter categories, and you can configure a list of additional destinations.

Before you begin:

- You must have created a web-filter-profile configuration if you want to specify it in the exception list.
- You must have hostname or IP address details on additional destinations you want to exclude from SSL decryption.
- You must have read-write permission for load balancing settings.

After you have configured an exception list, you can specify it in the virtual server configuration.

Syntax

```
config load-balance l2-exception-list
   edit <name>
      set description <string>
      set web-filter-profile <datasource>
      config member
         edit <No.>
            set type {host|ip}
```

```
            set host-pattern <string>
            set ip-netmask <ip&netmask>
            set ip-netmask-role {destination|source}
          next
        end
    next
end
```

`description`	A string to describe the purpose of the configuration, to help you and other administrators more easily identify its use. Put phrases in quotes. For example: `"Customer ABC"`.
`web-filter-profile`	Specify a web filter profile.
config member	
`type`	How you want to define the exception: • host • ip
`host-pattern`	The **host-pattern** option is available if **type** is **host**. Specify a wildcard pattern, such as `*.example.com`.
`ip-network`	The **ip-network** option is available if **type** is **ip**. Specify the IP address and CIDR-formatted subnet mask, separated by a forward slash, such as 192.0.2.0/24. Dotted quad formatted subnet masks are not accepted. IPv6 addresses are not supported.
`set ip-netmask-role`	The **ip-network** option is available if **type** is **ip**. Specify the role of the IP/Netmask: • destination — The IP/Netmask is set as the Destination IP, and the L2 SSL Forward Proxy VS will be bypassed based on this Destination IP. • source — The IP/Netmask is set as the Source IP, and the L2 SSL Forward Proxy VS will be bypassed based on this Source IP.

Example

```
FortiADC-docs # config load-balance l2-exception-list
FortiADC-docs (l2-exception-l~s) # edit financial
Add new entry 'financial' for node 3880
FortiADC-docs (financial) # set description "financial websites"
FortiADC-docs (financial) # config member
FortiADC-docs (member) # edit 1
Add new entry '1' for node 3883
FortiADC-docs (1) # set type host
FortiADC-docs (1) # set host-pattern *.bankofamerica.com
FortiADC-docs (1) # next
FortiADC-docs (member) # edit 2
Add new entry '2' for node 3883
FortiADC-docs (2) # set type host
FortiADC-docs (2) # set host-pattern *.schwab.com
FortiADC-docs (2) # end
FortiADC-docs (financial) # end
```

config load-balance method

Use this command to add method configuration objects.

The system includes predefined configuration objects for all supported load balancing methods, and there is no need to create additional configuration objects. You may choose to do so, however, for various reasons, for example, to use a naming convention that makes the purpose of the configuration clear to other administrators.

Table 8 describes the predefined methods.

Predefined methods

Predefined	Description
LB_METHOD_ROUND_ROBIN	Selects the next server in the series: server 1, then server 2, then server 3, and so on.
LB_METHOD_LEAST_CONNECTION	Selects the server with the least connections.
LB_METHOD_FASTEST_RESPONSE	Selects the server with the fastest response to health check tests.
LB_METHOD_URI	Selects the server based on a hash of the URI found in the HTTP header, excluding hostname.
LB_METHOD_FULL_URI	Selects the server based on a hash of the full URI string found in the HTTP header. The full URI string includes the hostname and path.
LB_METHOD_HOST	Selects the server based on a hash of the hostname in the HTTP Request header Host field.
LB_METHOD_HOST_DOMAIN	Selects the server based on a hash of the domain name in the HTTP Request header Host field.
LB_METHOD_DEST_IP_HASH	Selects the next hop based on a hash of the destination IP address. This method can be used with the Layer 2 virtual server.

Before you begin:

- You must have read-write permission for load balancing settings.

Syntax

```
config load-balance method
   edit <name>
      set type {dest-ip-hash | fastest-response | full-uri-hash | host-domain-hash | host-
           hash | least-connection | round-robin | uri-hash}
   next
end
```

`type`	Specify the method.

config load-balance pagespeed

Use this command to set which kind of HTTP requests will be handled by PageSpeed and how to accelerate.

Syntax

```
config load-balance pagespeed
   edit <name>
      set file-cache-inode-limit <1-100000>
      set file-cache-size-limit <1-512>
      set profile <datasource>

   config page-control
      edit <id>
         set type include/exclude
         set uri-pattern <uri regex>
      next
   end

   config resource-control
      edit <id>
         set fetch-domain <string>
         set origin-domain-pattern <regex string>
         set rewrite-domain <string>
      next
   end
end
```

Parameter	Description
`file-cache-inode-limit`	The maximum file cache inode.
`file-cache-size-limit`	The maximum file cache size (1M ~512M).
`profile`	The name of the PageSpeed profile.
config page-control	
`Type`	The include/exclude type.
`full-uri-pattern`	The full URI regex, such as http(s)://*example.com/*/htmls/*.html. **Note:** Wildcards include * which matches any 0 or more characters, and ?, which matches exactly one character. Unlike Unix shells, the / directory separator is not special, and can be matched by either * or ?. The resources are always expanded into their absolute form before expanding.
Config resource-control	

origin-domain-pattern	The origin domain regex, such as (http(s)://)*.example.com.
fetch-domain	The fetch domain string, such as http://www.example.com.
rewrite-domain	The rewrite domain string, such as http://www.example.com. **Note:** In the HTTP response body, the HTML sometimes links with certain resource URL. If the resource contain a domain name, FortiADC will perform the activity according to the fetch-domain setting or the rewrite-domain setting.

Example:

```
config load-balance pagespeed
   edit "all"
      set profile all
      set file-cache-inode-limit 10000
      set file-cache-size-limit 128
      config page-control
         edit 1
            set type include
            set uri-pattern *
         next
      end
      config resource-control
      end
   next
end
```

config load-balance pagespeed-profile

Use this command to specify the resources that will be handled by PageSpeed.

Syntax

```
config load-balance pagespeed-profile
   edit <name>
      set html enable/disable
      set css enable/disable
      set image enable/disable
      set combine-css enable/disable
      set max-combine-css-byte <1-10240>
      set jpeg-sampling enable/disable
      set resize-image enable/disable
      set move-css-to-head enable/disable
   next
end
```

Parameter	Description
html	Enable/disable HTML optimizer.
move-css-to-head	Moves CSS elements above the script tags.
css	Enable/disable CSS optimizer.
combine-css	Combines multiple CSS elements into one.
max-combine-css-byte Image	The maximum number of combined CSS bytes.
jpeg-sampling	Reduces the color sampling of .jpeg images to 4:2:0.
resize-image	Resizes images when the corresponding img tag specifies a smaller width and height.

Example:

```
config load-balance pagespeed-profile
   edit "all"
      set html enable
      set css enable
      set image enable
      set combine-css enable
      set max-combine-css-byte 4096
      set jpeg-sampling disable
      set resize-image enable
      set move-css-to-head enable
   next
end
```

config load-balance persistence

Use this command to configure persistence rules.

Persistence rules identify traffic that should not be load-balanced, but instead forwarded to the same backend server that has seen requests from that source before. Typically, you configure persistence rules to support server transactions that depend on an established client-server session, like e-commerce transactions or SIP voice calls.

The system maintains persistence session tables to map client traffic to backend servers based on the session attribute specified by the persistence rule.

The persistence table is evaluated before load balancing rules. If the packets received by the ADC match an entry in the persistence session table, the packets are forwarded to the server that established the connection, and load balancing rules are not applicable.

Most persistence rule types have a timeout. When the time that has elapsed since the system last received a request from the client IP address is greater than the timeout, the system does not use the mapping table to forward the request. Instead, it again selects the server using the method specified in the virtual server configuration. Hash-based rule types have a timeout built into the hash algorithm. For other types, you can specify the timeout.

Table 9 describes the predefined persistence rules. You can get started with these commonly used persistence methods or create custom objects.

Predefined persistence rules

Predefined	Description
LB_PERSIS_SRC_ADDR	Persistence based on source IP address or subnet.
LB_PERSIS_HASH_SRC_ADDR	Persistence based on a hash of source IP address.
LB_PERSIS_HASH_SRC_ADDR_PORT	Persistence based on a hash that includes source IP address and port.
LB_PERSIS_HASH_COOKIE	Persistence is based on a hash of a cookie provided by client request.
LB_PERSIS_RDP_COOKIE	Persistence based on RDP cookie sent by RDP clients in the initial connection request.
LB_PERSIS_SSL_SESS_ID	Persistence based on the SSL session ID.
LB_PERSIS_SIP_CALL_ID	Persistence based on the SIP call ID.
LB_PERSIS_PASSIVE_COOKIE	Persistence based on a passive cookie generated by the server. FortiADC does not generate or manage the cookie, but only observes it in the HTTP stream, thus the name "passive cookie". Also known as "server cookie".

Before you begin:

- You must have a good understanding and knowledge of the applications that require persistent sessions and the methods that can be used to identify application sessions.
- You must have read-write permission for load balancing settings.

FortiADC supports the following persistence rule types.

- Source Address on page 172
- Source Address Hash on page 173
- Source Address-Port Hash on page 173
- HTTP Header Hash on page 173
- HTTP Request Hash on page 174
- Cookie Hash on page 174
- Persistent Cookie on page 175
- Passive Cookie on page 177
- Insert Cookie on page 178
- Rewrite Cookie on page 180
- Embedded Cookie on page 180
- RADIUS Attribute on page 181
- SSL Session ID on page 183
- SIP Call ID on page 183
- RDP Cookie on page 184
- ISO8583 Bitmap on page 184

Each persistence rule type requires specific parameters to be applied. Use the links above to navigate to the CLI commands for each persistence rule type.

After you have configured a persistence rule, you can select it in the virtual server configuration.

Source Address

Use this command to create a Source Address persistence rule. Persistence is based on the source IP address.

Syntax

```
config load-balance persistence
   edit <name>
      set type source-address
      set timeout <integer>
      set ipv4-maskbits <integer>
      set ipv6-maskbits <integer>
      set match-across-servers {enable|disable}
      set persistent-ippool {enable|disable}
   next
end
```

`timeout`	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 1-86400, default: 300). When the time that has elapsed since the system last received a request from the client IP is greater than the timeout, the system would not use the mapping table to forward the request. Instead, it again selects the server using the method specified in the virtual server configuration.
`ipv4-maskbits`	Specify the number of bits in a IPv4 subnet mask to specify a network segment that should follow the persistence rule. (Range: 1-32, default: 32). For example, if IPv4 maskbits is set to 24, and the backend server A responds to a client with the source IP 192.168.1.100, server A also responds to all clients from subnet 192.168.1.0/24.
`ipv6-maskbits`	Specify the number of bits in a IPv6 subnet mask to specify a network segment that should follow the persistence rule. (Range: 1-128, default: 128).
`match-across-servers`	If enabled, clients will continue to access the same backend server through different virtual servers for the duration of a session. **Note:** The persistence rule with **match-across-servers** enabled works only with L4 virtual servers or the L7 virtual server whose Profile is LB_PROF_RADIUS.
`persistent-ippool`	If enabled, both the source IP and the IP selected from the IP pool will persist together for the duration of a session. **Note**: The **persistent-ippool** option is supported only in L4 virtual servers.

Example

```
config load-balance persistence
   edit "source-address"
      set type source-address
      set timeout 300
      set ipv4-maskbits 32
```

```
      set ipv6-maskbits 128
      set match-across-servers enable
      set persistent-ippool enable
   next
end
```

Source Address Hash

Use this command to create a Source Address Hash persistence rule. Persistence is based on a hash of the IP address of the client making an initial request.

Syntax

```
config load-balance persistence
   edit <name>
      set type consistent-hash-ip
   next
end
```

Source Address-Port Hash

Use this command to create a Source Address-Port Hash persistence rule. Persistence is based on a hash of the IP address and port of an initial client request.

Syntax

```
config load-balance persistence
   edit <name>
      set type hash-source-address-port
   next
end
```

HTTP Header Hash

Use this command to create a HTTP Header Hash persistence rule. Persistence is based on a hash of the specified header value found in an initial client request.

Syntax

```
config load-balance persistence
   edit <name>
      set type hash-http-header
      set keyword <string>
   next
end
```

`keyword`	Specify the HTTP header value.

Example

```
config load-balance persistence
   edit "hash-http-header"
      set type hash-http-header
      set keyword <string>
   next
end
```

HTTP Request Hash

Use this command to create a HTTP Request Hash persistence rule. Persistence is based on a hash of the specified URL parameter in an initial client request.

Syntax

```
config load-balance persistence
   edit <name>
      set type hash-http-request
      set keyword <string>
   next
end
```

`keyword`	Specify the URL parameter.

Example

```
config load-balance persistence
   edit "hash-http-request"
      set type hash-http-request
      set keyword <string>
   next
end
```

Cookie Hash

Use this command to create a Cookie Hash persistence rule. Persistence is based on a hash of a cookie provided by client request.

Syntax

```
config load-balance persistence
   edit <name>
      set type hash-cookie
      set keyword <string>
   next
end
```

keyword	Specifies the cookie name. If the specified cookie keyword is a valid cookie name from which the cookie value can be extracted, it will be used to calculate the hash. Without the keyword, the hash will be calculated using the whole cookie. Otherwise, the default round robin method will be used.

Example

```
config load-balance persistence
   edit "hash-cookie"
      set type hash-cookie
      set keyword myCookieName
   next
end
```

Persistent Cookie

Use this command to create a Persistent Cookie persistence rule. Persistence is based on the cookie provided in the backend server response. It forwards subsequent requests with this cookie to the original backend server.

Syntax

```
config load-balance persistence
   edit <name>
      set type persistent-cookie
      set keyword <string>
      set timeout <integer>
      set cookie-domain <string>
      set cookie-httponly {enable|disable}
      set cookie-secure {enable|disable}
      set cookie-samesite {nothing|none|lax|strict}
      set cookie-custom-attr {enable|disable}
      set cookie-custom-attr-value <string>
   next
end
```

keyword	Specify the backend server cookie name.
timeout	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 0-86400, default: 300). If the timeout is set to 0 seconds, the persistence cookie will function as a session cookie, expiring at the end of the session. Typically, when the timeout value is set to 1 or more seconds, a timestamp is applied to the cookie to track the timeout. When the timeout is set to 0, the cookie will not have a timestamp and instead will expire along with the session.
cookie-domain	Specifies the domain attribute of the cookie. If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.

`cookie-httponly`	Enable/disable to add the "HTTPOnly" flag to cookies. This option is disabled by default. The HttpOnly attribute limits the scope of the cookie to HTTP requests. In particular, the attribute instructs the user agent to omit the cookie when providing access to cookies via "non-HTTP" APIs (such as a web browser API that exposes cookies to scripts). If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-secure`	Enable/disable to add the Secure flag to cookies. This option is disabled by default. The Secure attribute limits the scope of the cookie to "secure" channels (where "secure" is defined by the user agent). When a cookie has the Secure attribute, the user agent will include the cookie in an HTTP request only if the request is transmitted over a secure channel (typically HTTP over Transport Layer Security (TLS)). If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-samesite`	Add a SameSite attribute to prevent the browser from sending cookies along with cross-site requests, to mitigate the risk of cross-origin information leakage. It provides Strict, Lax, and None values for this attribute: • nothing — Do not add Samesite attribute to cookies. The default value is nothing. • none — set the value as none if a cookie is required to be sent by cross origin. **Note**: If **cookie-secure** is enabled, then **cookie-samesite** should be set to **none**. • lax — any request from the third parties will not carry such cookies except for GET requests that navigate to the destination URL. • strict — any request from the third parties will not carry such cookies If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-custom-attr`	Enable to specify custom attributes. This option is disabled by default. When **cookie-custom-attr** is enabled, the following options become unavailable: **cookie-domain**, **cookie-httponly**, **cookie-secure**, and **cookie-samesite**.
`cookie-custom-attr-value`	The **cookie-custom-attr-value** option appears if **cookie-custom-attr** is enabled. Specify the full cookie attributes, including any of the standard attributes and any of the custom attributes. If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.

Example

```
config load-balance persistence
   edit "persistent-cookie"
      set type persistent-cookie
      set keyword myCookieName
```

```
      set timeout 300
      set cookie-domain example.com
      set cookie-httponly enable
      set cookie-secure enable
      set cookie-samesite nothing
      set cookie-custom-attr disable
   next
end
```

Passive Cookie

Use this command to create a Passive Cookie persistence rule. Persistence is based on the cookie which is generated from the server.

Syntax

```
config load-balance persistence
   edit <name>
      set type passive-cookie
      set sess-keyword {auto|PHPSESSID|JSESSIONID|CFID+CFTOKEN|ASP.NET_SessionId|custom}
      set keyword <string>
      set timeout <integer>
   next
end
```

`sess-keyword`	Specify the cookie which is generated from the server. Select either of the following: • auto — includes all options except custom. • PHPSESSID • JSESSIONID • CFID+CFTOKEN • ASP.NET_SessionId • custom — use a custom defined cookie name.
`keyword`	The **keyword** option is available if **sess-keyword** is **custom**. Specify the backend server cookie name.
`timeout`	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 1-86400, default: 300).

Example

```
config load-balance persistence
   edit "passive-cookie"
      set type passive-cookie
      set sess-keyword auto
      set timeout 300
   next
end
```

Insert Cookie

Use this command to create an Insert Cookie persistence rule. Persistence is based on a cookie inserted by the FortiADC system.

The system inserts a cookie whose name is the value specified by Keyword and whose value is the real server pool member Cookie value and expiration date (if the client does not already have a cookie).
For example, if the value of Keyword is `sessid` and the real server pool member Cookie value is `rs1`, FortiADC sends the cookie `sessid=rs1|U6iFN` to the client, where `U6iFN` is the expiration date as a base64 encoded string.

Syntax

```
config load-balance persistence
   edit <name>
      set type insert-cookie
      set keyword <string>
      set timeout <integer>
      set cookie-domain <string>
      set cookie-httponly {enable|disable}
      set cookie-secure {enable|disable}
      set cookie-samesite {nothing|none|lax|strict}
      set cookie-custom-attr {enable|disable}
      set cookie-custom-attr-value <string>
   next
end
```

`keyword`	Specify the backend server cookie name.
`timeout`	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 0-86400, default: 300). If the timeout is set to 0 seconds, the insert cookie will function as a session cookie, expiring at the end of the session. Typically, when the timeout value is set to 1 or more seconds, a timestamp is applied to the cookie to track the timeout. When the timeout is set to 0, the cookie will not have a timestamp and instead will expire along with the session.
`cookie-domain`	Specifies the domain attribute of the cookie. If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-httponly`	Enable/disable to add the "HTTPOnly" flag to cookies. This option is disabled by default. The HttpOnly attribute limits the scope of the cookie to HTTP requests. In particular, the attribute instructs the user agent to omit the cookie when providing access to cookies via "non-HTTP" APIs (such as a web browser API that exposes cookies to scripts). If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-secure`	Enable/disable to add the Secure flag to cookies. This option is disabled by default.

	The Secure attribute limits the scope of the cookie to "secure" channels (where "secure" is defined by the user agent). When a cookie has the Secure attribute, the user agent will include the cookie in an HTTP request only if the request is transmitted over a secure channel (typically HTTP over Transport Layer Security (TLS)). If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-samesite`	Add a SameSite attribute to prevent the browser from sending cookies along with cross-site requests, to mitigate the risk of cross-origin information leakage. It provides Strict, Lax, and None values for this attribute: • nothing — Do not add Samesite attribute to cookies. The default value is nothing. • none — set the value as none if a cookie is required to be sent by cross origin. **Note**: If **cookie-secure** is enabled, then **cookie-samesite** should be set to **none**. • lax — any request from the third parties will not carry such cookies except for GET requests that navigate to the destination URL. • strict — any request from the third parties will not carry such cookies If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.
`cookie-custom-attr`	Enable to specify custom attributes. This option is disabled by default. When **cookie-custom-attr** is enabled, the following options become unavailable: **cookie-domain**, **cookie-httponly**, **cookie-secure**, and **cookie-samesite**.
`cookie-custom-attr-value`	The **cookie-custom-attr-value** option appears if **cookie-custom-attr** is enabled. Specify the full cookie attributes, including any of the standard attributes and any of the custom attributes. If the cookie attribute does not match or is not applicable, the user agent will ignore the cookie and will not enforce persistence.

Example

```
config load-balance persistence
   edit "insert-cookie"
      set type insert-cookie
      set keyword MyInsCookieName
      set timeout 300
      set cookie-domain example.com
      set cookie-httponly enable
      set cookie-secure enable
      set cookie-samesite nothing
      set cookie-custom-attr disable
   next
end
```

Rewrite Cookie

Use this command to create a Rewrite Cookie persistence rule. Persistence is based on the cookie provided in the backend server response, but the system rewrites the cookie.

Syntax

```
config load-balance persistence
   edit <name>
      set type rewrite-cookie
      set keyword <string>
   next
end
```

keyword	Specify the backend server cookie name.

Example

```
config load-balance persistence
   edit "rewrite-cookie"
      set type rewrite-cookie
      set keyword myCookieName
   next
end
```

Embedded Cookie

Use this command to create a Embedded Cookie persistence rule. Persistence is based on the cookie provided in the backend server response.

The system checks the HTTP response for a `Set-Cookie:` value that matches the value specified by Keyword in the persistence configuration. However, it preserves the original value and adds the real server pool member Cookie value and a ~ (tilde) as a prefix.

For example, the value of Keyword is `sessid`. The real server pool member Cookie value is `rs1`. After an initial client request, the response from the server contains `Set-Cookie: sessid=666`, which the system changes to `Set-Cookie: sessid=rs1~666`. It uses this rewritten value to forward subsequent requests to the same backend server as the original request.

Syntax

```
config load-balance persistence
   edit <name>
      set type embedded-cookie
      set keyword <string>
   next
end
```

keyword	Specify the backend server cookie name.

Example

```
config load-balance persistence
   edit "embedded-cookie"
      set type embedded-cookie
      set keyword myCookieName
   next
end
```

RADIUS Attribute

Use this command to create a RADIUS Attribute persistence rule. Persistence is based on a specified RADIUS attribute.

Syntax

```
config load-balance persistence
   edit <name>
      set type radius-attribute
      set timeout <integer>
      set match-across-servers {enable|disable}
      set override-connection-limit {enable|disable}
      set radius-attribute-relation {AND|OR}
      config radius-attribute
         edit <No.>
            set type {1-user-name|4-nas-ip-address|5-nas-port|6-service-type|7-framed-
                 protocol|8-framed-ip-address|9-framed-ip-netmask|12-framed-mtu|13-framed-
                 compression|14-login-ip-host|19-callback-number|24-state|26-vendor-
                 specific|30-called-station-id|31-calling-station-id|32-nas-identifier|33-
                 proxy-state|34-login-lat-service|35-login-lat-node|36-login-lat-group|60-
                 chap-challenge|61-nas-port-type|62-port-limit|63-login-lat-port}
            set vendor-id <integer>
            set vendor-type <integer>
         next
      end
   next
end
```

`timeout`	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 1-86400, default: 300). When the time that has elapsed since the system last received a request from the client IP is greater than the timeout, the system would not use the mapping table to forward the request. Instead, it again selects the server using the method specified in the virtual server configuration.
`match-across-servers`	An option for radius-attribute and source-address persistence methods. Enable so clients continue to access the same backend server through different virtual servers for the duration of a session.
`override-connection-limit`	An option for radius-attribute only. Disabled by default. If the real server connection limit is reached and this option is enabled, the new connection will neither persist to the new server nor go to another node.

`radius-attribute-relation`	Specify the relation when multiple radius attributes are configured. • AND — All of the specified radius attributes must be the same in the hash table to be persistent. • OR — Search the first radius attribute in the hash table for persistence if the first radius attribute exists. If not, search the following radius attributes in sequence.
config radius-attribute	
`type`	Specify the Radius attribute type: • 1-user-name • 4-nas-ip-address • 5-nas-port • 6-service-type • 7-framed-protocol • 8-framed-ip-address • 9-framed-ip-netmask • 12-framed-mtu • 13-framed-compression • 14-login-ip-host • 19-callback-number • 24-state • 26-vendor-specific • 30-called-station-id • 31-calling-station-id • 32-nas-identifier • 33-proxy-state • 34-login-lat-service • 35-login-lat-node • 36-login-lat-group • 60-chap-challenge • 61-nas-port-type • 62-port-limit • 63-login-lat-port
`vendor-id`	The **vendor-id** option is available if the radius attribute **type** is **26-vendor-specific**. The number specifies the vendor id. 0 means the entire attribute will be used as a persistence input.
`vendor-type`	The **vendor-id** option is available if the radius attribute **type** is **26-vendor-specific**. The number specifies the vendor type. 0 means the entire attribute will be used as a persistence input.

Example

```
config load-balance persistence
   edit "radius-attribute"
      set type radius-attribute
```

```
      set timeout 300
      set match-across-servers enable
      set override-connection-limit enable
      set radius-attribute-relation AND
      config radius-attribute
         edit 1
            set type 4-nas-ip-address
         next
      end
   next
end
```

SSL Session ID

Use this command to create a SSL Session ID persistence rule. Persistence is based on SSL session ID.

Syntax

```
config load-balance persistence
   edit <name>
      set type ssl-session-id
      set timeout <integer>
   next
end
```

`timeout`	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 1-86400, default: 300).

Example

```
config load-balance persistence
   edit "ssl-session-id"
      set type ssl-session-id
      set timeout 300
   next
end
```

SIP Call ID

Use this command to create a SIP Call ID persistence rule. Persistence is based on SIP Call ID. For SIP services, you can establish persistence using Source Address, Source Address Hash, or SIP caller ID.

Syntax

```
config load-balance persistence
   edit <name>
      set type sip-call-id
      set timeout <integer>
   next
end
```

timeout	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 1-86400, default: 300).

Example

```
config load-balance persistence
   edit "sip-call-id"
      set type sip-call-id
      set timeout 300
   next
end
```

RDP Cookie

Use this command to create a RDP Cookie persistence rule. Persistence based on RDP cookie sent by RDP clients in the initial connection request.

Syntax

```
config load-balance persistence
   edit <name>
      set type rdp-cookie
   next
end
```

ISO8583 Bitmap

Use this command to create a ISO8583 Bitmap persistence rule. Persistence is based on the bitmap.

Syntax

```
config load-balance persistence
   edit <name>
      set type iso8583-bitmap
      set timeout <integer>
      set iso8583-bitmap-relation {AND|OR}
      set keyvalue-relation {AND|OR}
      config iso8583-bitmap
         edit <No.>
            set type {2-primary-acct-num|3-process-code|7-date-time-trans|11-sys-trace-audit-
                  num|12-date-tm-loc-trans|18-merchant-type|19-acq-inst-cy-code|20-PAN-ext-cy-
                  code|21-fwd-inst-cy-code|22-POS-date|23-card-seq-num|24-function-code|32-
                  acq-inst-id-code|33-fwd-inst-id-code|37-retrieval-ref-num|40-serv-restrict-
                  code|41-card-accept-term-id|42-card-accept-id-code|52-PIN-data|67-ext-
                  payment-code|68-recv-inst-cy-code|69-settle-inst-cy-code|73-date-action|74-
                  credits-num|75-credits-reversal-num|76-debits-num|77-debits-reversal-num|78-
                  tranfer-num|79-transfer-reversal-num|80-inquiries-num}
         next
      end
   next
```

```
end
```

`timeout`	Specify the timeout (in seconds) for an inactive persistence session table entry. (Range: 1-86400, default: 300).
`iso8583-bitmap-relation`	Specify the relation among the bitmap type be AND/OR. The default is OR.
`keyvalue-relation`	The **keyvalue-relation** option is available if **iso8583-bitmap-relation** is **OR**. Specify the relation of keyvalue be AND/OR. The default is AND.
config iso8583-bitmap	
`type`	Specify the bitmap type from the following 30 supported types: • 2-primary-acct-num • 3-process-code • 7-date-time-trans • 11-sys-trace-audit-num • 12-date-tm-loc-trans • 18-merchant-type • 19-acq-inst-cy-code • 20-PAN-ext-cy-code • 21-fwd-inst-cy-code • 22-POS-date • 23-card-seq-num • 24-function-code • 32-acq-inst-id-code • 33-fwd-inst-id-code • 37-retrieval-ref-num • 40-serv-restrict-code • 41-card-accept-term-id • 42-card-accept-id-code • 52-PIN-data • 67-ext-payment-code • 68-recv-inst-cy-code • 69-settle-inst-cy-code • 73-date-action • 74-credits-num • 75-credits-reversal-num • 76-debits-num • 77-debits-reversal-num • 78-tranfer-num • 79-transfer-reversal-num • 80-inquiries-num

Example

```
config load-balance persistence
   edit "iso8583-bitmap"
      set type iso8583-bitmap
      set timeout 300
      set iso8583-bitmap-relation OR
      set keyvalue-relation AND
      config iso8583-bitmap
         edit 1
            set type 2-primary-acct-num
         next
      end
   next
end
```

config load-balance pool

Use this command to configure real server pool settings.

A server pool is a group of the real servers that host the applications that you load balance.

To configure a server pool:

1. Create a server pool object.
2. Add members.

Before you begin:

- You must have a good understanding and knowledge of the backend server boot behavior, for example, how many seconds it takes to “warm up” after a restart before it can process traffic.
- You must know the IP address and port of the applications.
- You must have already created real server SSL profiles if you want to specify them in the real server configuration.
- You must have read-write permission for load balancing settings.

After you have configured a real server pool, you can select it in the virtual server configuration.

Syntax

```
config load-balance pool
   edit <name>
      set addr-type {ipv4|ipv6}
      set health-check-ctrl {enable|disable}
      set health-check-list {<datasource> ...}
      set health-check-relation {AND|OR}
      set type {static|dynamic}
      set sdn-connector <string>
      set service <string>
      set use-private-addr {enable|disable}
      set real-server-ssl-profile <datasource>
      config pool_member
         edit <No.>
            set backup {enable|disable}
```

```
            set connection-limit <integer>
            set connection-rate-limit <integer>
            set health-check-inherit {enable|disable}
            set health-check-ctrl {enable|disable}
            set health-check-list {<datasource> ...}
            set health-check-relation {AND|OR}
            set ip <class_ip>
            set ip6 <class_ip>
            set pool_member_cookie <string>
            set pool_member_server_name <string>
            set pool_member_service_port <integer>
            set pool_member_weight <integer>
            set recover <integer>
            set rs-profile-inherit {enable|disable}
            set real-server-profile <datasource>
            set ssl {enable|disable}
            set status {enable|disable|maintain}
            set warm-rate <integer>
            set warm-up <integer>
            set modify-host {enable|disable}
            set host <string>
        next
    end
  next
end
```

`addr-type`	• IPv4 • IPv6
`set type {static|dynamic}`	Select whether the real servers use static or dynamic IP addresses.
`sdn-connector`	Select the SDN Connector which is created in **Security Fabric**. Available only when type is dynamic.
`Service`	Select the service protocol that FortiADC uses to communicate with the instances. Available only when type is dynamic.
`use-private-addr {enable|disable}`	When FortiADC is deployed on public cloud platforms: • Enable this option to get the private address of the instances. • Disable this option to get the public address of the instances. Available only when type is dynamic.
`health-check-ctrl`	Enable health checking for the pool. The health check settings at this configuration level are the parent configuration. When you configure the pool members, you can specify whether to inherit or override the parent configuration.
`health-check-list`	Specify one or more health check configuration objects.
`health-check-relation`	• AND—All of the specified health checks must pass for the server to be considered available. • OR—One of the specified health checks must pass for the server to be considered available.
`real-server-profile`	Specify a real server profile. Real server profiles determine settings for communication between FortiADC and the backend real servers.

config pool_member	
backup	Server that the ADC directs traffic to only when other servers in the pool are down. The backup server receives connections when all the other pool members fail the health check or you have manually disabled them, for example. **Note**: Not applicable for SIP servers.
connection-limit	Maximum number of concurrent connections to the backend server. The default is 0 (disabled). The valid range is 1 to 1,048,576 concurrent connections. **Note**: Connection Limit is not supported for FTP or SIP servers.
connection-rate-limit	Limit the number of new connections per second to this server. The default is 0 (disabled). The valid range is 1 to 86,400 connections per second. In Layer 4 deployments, you can apply a connection rate limit per real server and per virtual server. Both limits are enforced. **Note**: The connection rate limit applies only when the real servers belong to a Layer 4 virtual server. If you add a real server pool with this setting configured to a Layer 7 virtual server, for example, the setting is ignored. **Note**: Connection Rate Limit is not supported for FTP or SIP servers.
health-check-inherit	Enable to inherit the health check settings from the parent configuration. Disable to specify health check settings in this member configuration.
health-check-ctrl	Enable health checking for the pool.
health-check-list	Specify one or more health check configuration objects.
health-check-relation	• AND—All of the selected health checks must pass for the server to the considered available. • OR—One of the selected health checks must pass for the server to be considered available.
ip	Backend server IP address. In a Layer 2 virtual server deployment, specify the IP address of the next hop to the destination server. Configure a pseudo default gateway in the static route since Layer 2 virtual servers need to use this default route internally to match all the destinations that the client wants to access. However, this default gateway is not used because the next hop is the pool member and not the pseudo gateway. In a Layer 2 virtual server deployment, ensure the backend servers have been configured to route responses through the FortiADC IP address.
ip6	Backend server IP address. In a Layer 2 virtual server deployment, specify the IP address of the next hop to the destination server. Configure a pseudo default gateway in the static route since Layer 2 virtual servers need to use this default route internally to match all the destinations that the client wants to access. However, this default gateway is not used because the next hop is the pool member and not the pseudo gateway. In a Layer 2 virtual server deployment, ensure the backend servers have been configured to route responses through the FortiADC IP address.

`pool_member_cookie`	Cookie name to be used when cookie-based Layer 7 session persistence is enabled. The cookie is used to create a FortiADC session ID, which enables the system to forward subsequent related requests to the same backend server. If you do not specify a cookie name, it is set to the pool member server name string. **Note**: Not applicable for SIP servers.
`pool_member_server_name`	Real server member configuration name to appear in logs and reports. Alphabetic, numeric, underscore (_), and hyphen (-) characters are allowed. The setting is required.
`pool_member_service_port`	Backend server listening port number. Usually HTTP is 80, HTTPS is 443, FTP is 21, SMTP is 25, DNS is 53, POP3 is 110, IMAP4 is 143, RADIUS is 1812, and SNMP is 161. **Tip**: The system handles port 0 as a “wildcard” port. When configured to use port 0, the system uses the destination port from the client request. For example, if you specify 0, and the destination port in the client request is 50000, the traffic is forwarded to port 50000.
`pool_member_weight`	Assigns relative preference among members—higher values are more preferred and are assigned connections more frequently. The default is 1. The valid range is 1 to 256. All load balancing methods consider weight. Servers are dispatched requests proportional to their weight, relative to the sum of all weights. The following example shows the effect of weight on Round Robin: • Sever A, Weight 2; Server B, Weight 1: Requests are sent AABAAB. • Sever A, Weight 3; Server B, Weight 2: Requests are sent AABAB. For other methods, weight functions as a tie-breaker. For example, with the Least Connection algorithm, requests are sent to the server with the least connections. If the number of connections is equal, the request is sent to the server with the greater weight. For example: • Server A, Weight 1, 1 connection • Server B, Weight 2, 1 connection The next request is sent to Server B.
`recover`	Seconds to postpone forwarding traffic after downtime, when a health check indicates that this server has become available again. The default is 0 (disabled). The valid range is 1 to 86,400 seconds. After the recovery period elapses, the FortiADC assigns connections at the warm rate. Examples of when the server experiences a recovery and warm-up period: • A server is coming back online after the health check monitor detected it was down. • A network service is brought up before other daemons have finished initializing and therefore the server is using more CPU and memory resources than when startup is complete. To avoid connection problems, specify the separate warm-up rate, recovery rate, or both.

	Tip: During scheduled maintenance, you can also manually apply these limits by setting Status to Maintenance instead of Enable. **Note**: Not applicable for SIP servers.
`rs-profile-inherit`	Enable to inherit the real server profile from the pool configuration. Disable to specify the real server profile in this member configuration.
`real-server-profile`	If not configured to inherit the pool setting, specify a real server profile. Real server profiles determine settings for communication between FortiADC and the backend real servers.
`status`	• enable—The server can receive new sessions. • disable—The server does not receive new sessions and closes any current sessions as soon as possible. • maintain—The server does not receive new sessions but maintains any current connections.
`warm-rate`	Maximum connection rate while the server is starting up. The default is 100 connections per second. The valid range is 1 to 86,400 connections per second. The warm up calibration is useful with servers that have the network service brought up before other daemons have finished initializing. As the servers are brought online, CPU and memory are more utilized than they are during normal operation. For these servers, you define separate rates based on warm-up and recovery behavior. For example, if Warm Up is 5 and Warm Rate is 2, the number of allowed new connections increases at the following rate: • 1st second—Total of 2 new connections allowed (0+2). • 2nd second—2 new connections added for a total of 4 new connections allowed (2+2). • 3rd second—2 new connections added for a total of 6 new connections allowed (4+2). • 4th second—2 new connections added for a total of 8 new connections allowed (6+2). • 5th second—2 new connections added for a total of 10 new connections allowed (8+2). **Note**: Not applicable for SIP servers.
`warm-up`	If the server cannot initially handle full connection load when it begins to respond to health checks (for example, if it begins to respond when startup is not fully complete), indicate how long to forward traffic at a lesser rate. The default is 0 (disabled). The valid range is 1 to 86,400 seconds. **Note**: Not applicable for SIP servers.
`modify-host`	Enable to allow FortiADC to modify the HTTP header according to the "host" field of the real server. This is disabled by default.
`host`	The **host** option is available if **modify-host** is enabled. Specify the host as a string. This field cannot be left empty. The input validation regex is `([0-9A-Za-z+/?%$#&\=.:_-]|\[|])+$`, maximum of 255 characters. The default value is `host`.

Example

```
FortiADC-VM # config load-balance pool
FortiADC-VM (pool) # edit lb-pool
Add new entry 'lb-pool' for node 1705

FortiADC-VM (lb-pool) # get
addr-type : ipv4
health-check-ctrl : disable

FortiADC-VM (lb-pool) # set health-check-ctrl enable
FortiADC-VM (lb-pool) # set ?
addr-type address type
health-check-ctrl health check control
*health-check-list health check list
health-check-relation health check relationship

FortiADC-VM (lb-pool) # set health-check-list lb-health-check

FortiADC-VM (lb-pool) # config pool_member
FortiADC-VM (pool_member) # edit 1
Add new entry '1' for node 1710

FortiADC-VM (1) # get
health-check-inherit: enable
status : enable
ssl : disable
backup : disable
ip : 0.0.0.0
ip6 :
pool_member_service_port: 80
pool_member_weight : 1
connection-limit : 0
recover : 0
warm-up : 0
warm-rate : 10
connection-rate-limit: 0
pool_member_cookie : cookie

FortiADC-VM (1) # set ip 192.168.100.1
FortiADC-VM (1) # end
FortiADC-VM (lb-pool) # end
```

config load-balance profile

Use this command to configure virtual server profiles. A profile is a configuration object that defines how you want the FortiADC virtual server to handle traffic for specific protocols. Virtual server profiles determine settings used in network communication on the client-FortiADC segment, in contrast to real server profiles, which determine the settings used in network communication on the FortiADC-real server segment.

The **Application Profile Usage** table describes the usage by application profile type, including the compatible virtual server types, load-balancing methods, persistence methods, and content routing types.

Application Profile Usage

Profile	Usage	VS Type	LB Methods	Persistence
FTP	Use with FTP servers.	Layer 7, Layer 4	Layer 7: Round Robin, Least Connections Layer 4: Same as Layer 7, plus Fastest Response, Dynamic Load	Layer 7: Source Address, Source Address Hash Layer 4: Same as Layer 7, plus Source Address-Port Hash
HTTP	Use for standard, unsecured web server traffic.	Layer 7, Layer 2	Layer 7: Round Robin, Least Connections, URI Hash, Full URI Hash, Host Hash, Host Domain Hash, Dynamic Load Layer 2: Same as Layer 7, plus Destination IP Hash	Source Address, Source Address Hash, Source Address-Port Hash, HTTP Header Hash, HTTP Request Hash, Cookie Hash, Persistent Cookie, Insert Cookie, Embedded Cookie, Rewrite Cookie, Passive Cookie
HTTPS	Use for secured web server traffic when offloading TLS/SSL from the backend servers. You must import the backend server certificates into FortiADC and select them in the HTTPS profile.	Layer 7, Layer 2	Same as HTTP	Same as HTTP, plus SSL Session ID
TURBO HTTP	Use for unsecured HTTP traffic that does not require advanced features like caching, compression, content rewriting, rate limiting, Geo IP blocking, or source NAT. The profile can be used with content routes and destination NAT, but the HTTP request must be in the first data packet. This profile enables packet-based forwarding that reduces network latency and system CPU usage. However, packet-based forwarding for HTTP is advisable only when you do not anticipate dropped packets or out-of-order packets.	Layer 7	Round Robin, Least Connections, Fastest Response	Source Address

Profile	Usage	VS Type	LB Methods	Persistence
RADIUS	Use with RADIUS servers.	Layer 7	Round Robin	RADIUS attribute
RDP	Use with Windows Terminal Service(remote desktop protocol).	Layer 7	Round Robin, Least Connections	Source Address, Source Address Hash, Source Address-Port Hash, RDP Cookie
SIP	Use with applications that use session initiation protocol (SIP), such as VoIP, instant messaging, and video.	Layer 7	Round Robin, URI Hash, Full URI Hash	Source Address, Source Address Hash, Source Address-Port Hash, SIP Call ID
TCP	Use for other TCP protocols.	Layer 4, Layer 2	Layer 4: Round Robin, Least Connections, Fastest Response, Dynamic Load Layer 2: Round Robin, Least Connections, Fastest Response, Destination IP Hash, Dynamic Load	Source Address, Source Address Hash, Source Address-Port Hash
TCPS	Use for secured TCP when offloading TLS/SSL from the backend servers. Like the HTTPS profile, you must import the backend server certificates into FortiADC and select them in the TCPS profile.	Layer 7, Layer 2	Layer 7: Round Robin, Least Connections, Dynamic Load Layer 2: Round Robin, Least Connections, Destination IP Hash, Dynamic Load	Source Address, Source Address Hash, Source Address-Port Hash, SSL Session ID
UDP	Use with UDP servers.	Layer 4, Layer 2	Layer 4: Round Robin, Least Connections, Fastest Response, Dynamic Load Layer 2: Same as Layer 4, plus Destination IP Hash	Source Address, Source Address Hash, Source Address-Port Hash
IP	Combines with Layer 2 TCP/UDP/HTTP virtual server to balance the rest of the IP packets passed through FortiADC. When running the IP protocol 0 VS, the traffic always tries to match none protocol 0 VS first.	Layer 2	Round Robin, Dynamic Load	Source Address, Source Address Hash, Source Address-Port Hash
DNS	Use with DNS servers.	Layer 7	Round Robin, Least Connections	Not supported yet.

Profile	Usage	VS Type	LB Methods	Persistence
SMTP	Use with SMTP servers.	Layer 7	Round Robin, Least Connections	Source Address, Source Address Hash
RTMP	A TCP-based protocol used for streaming audio, video, and data over the Internet	Layer 7	Round Robin, Least Connection	Source Address, Source Address Hash
ISO8583	Use with ISO8583 servers	Layer 7	Round Robin	N/A
RTSP	A network control protocol used for establishing and controlling media sessions between end points	Layer 7	Round Robin, Least Connection	Source Address, Source Address Hash
MySQL	MySQL network protocol stack (i.e., MySQL-Proxy) which parses and builds MySQL protocol packets	Layer 7	Round Robin, Least Connection	N/A
DIAMETER	A successor to RADIUS, DIAMETER is the next-generation Authentication, Authorization and Accounting (AAA) protocol widely used in IMS and LTE.	Layer 7	Round Robin	Source Address. DIAMETER Session ID (default)
MSSQL	MSSQL network protocol stack, which parses and builds MSSQL protocol packets	Layer 7	Least connection	N/A
EXPLICIT_HTTP	A simple explicit/forward HTTP proxy mode. In this mode, you don't need to add backend real server pool. The destination IP address of the downstream is specified by the URL or Host field of the client request.	Layer 7	N/A	N/A
L7 TCP	Use for other TCP protocols.	Layer 7	Layer 7: Round Robin, Least Connections	Source Address, Source Address Hash
L7 UDP	Use with UDP servers.	Layer 7	Layer 7: Round Robin, Least Connections	Source Address, Source Address Hash

The **Predefined Profiles** table lists the default values of each predefined profile. All values in the predefined profiles are view-only, and cannot be modified. You can select predefined profiles in the virtual server configuration, or you can create user-defined profiles to include configuration objects such as certificates, caching settings, compression options, and IP reputation.

Predefined Profiles

Profile	Defaults
LB_PROF_DIAMETER	Origin Host—Blank Origin Realm—Blank Vendor ID—0 Product Name—Blank Idle Timeout—300 (seconds) (Note: This refers to the built-in session ID persistence timeout.) Server Close Propagation—OFF (Note: This means that the connection on the client side stays open when the server closes any connection on its side.) Client SSL—Off
LB_PROF_TCP	Timeout TCP Session—100 Timeout TCP Session after FIN—100 IP Reputation—Disabled Geo IP block list—None Geo IP Allowlist—None
LB_PROF_UDP	Timeout UDP Session—100 IP Reputation—Disabled Stateless—Disabled Geo IP block list—None Geo IP Allowlist—None
LB_PROF_HTTP	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Client Address—Disabled X-Forwarded-For—Disabled X-Forwarded-For Header—Blank IP Reputation—Disabled HTTP Mode—Keep Alive Compression—None. Decompression—None Caching—None Geo IP Block List—None Geo IP Allowlist—None Geo IP Redirect URL—http:// HTTP Send Timeout—5 HTTP2—None

Profile	Defaults
LB_PROF_HTTP_SERVERCLOSE	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Buffer Pool—Enabled Client Address—Disabled X-Forwarded-For—Disabled X-Forwarded-For Header—None IP Reputation—Disabled HTTP Mode—Server Close Customized SSL Ciphers Flag—Disabled Compression—None Decompression—None Caching—None Geo IP Block List—None Geo IP Allowlist—None Geo IP Redirect URL—http:// HTTP Send Timeout—0 HTTP2—None
LB_PROF_TURBOHTTP	Timeout TCP Session—100 Timeout TCP Session after FIN—100 IP Reputation—Disabled Geo IP Block List—None Geo IP Allowlist—None
LB_PROF_FTP	Timeout TCP Session—100 Timeout TCP Session after FIN—100 IP Reputation—Disabled Geo IP Block List—None Geo IP Allowlist—None Client Address—Off Security Mode—None
LB_PROF_RADIUS	Client Address—Off Source Port—Off Dynamic Auth—Disable RADIUS Session—300 Geo IP Block List—None Geo IP Allowlist—None

Profile	Defaults
LB_PROF_SIP	SIP Max Size—65535 Server Keepalive Timeout—30 Server Keepalive—Enabled Client Keepalive—Disabled Client Protocol—UDP Server Protocol—None Failed Client Type—Drop Failed Server Type—Drop Insert Client IP—Disabled Geo IP Block List—None Geo IP Allowlist—None Client Address—Off Media Address—0.0.0.0
LB_PROF_RDP	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 Source Address—Disabled IP Reputation—Disabled Geo IP Block List—None Geo IP Allowlist—None
LB_PROF_IP	IP Reputation—Disabled Geo IP Block List—None Geo IP Allowlist—None Timeout IP Session—100
LB_PROF_DNS	Client Address—Off DNS Cache Flag—Enabled DNS Cache Ageout Time—3600 DNS Cache Size—10 DNS Cache Entry Size—512 DNS Cache Response Type—All Records DNS Malform Query Action—Drop DNA Max Query Length—512 DNS Authentication Flag—Disabled
LB_PROF_TCPS	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 Client Address—Disabled

Profile	Defaults
	IP Reputation—Disabled Geo IP block list—None
LB_PROF_HTTPS	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Client Address—Disabled X-Forwarded-For—Disabled X-Forwarded-For Header—None IP Reputation—Disabled HTTP Mode—Keep Alive SSL Proxy Mode—Disabled Compression—None Decompression—None Caching—None Geo IP Block List—None Geo IP Allowlist—None Geo IP Redirect URL—http:// HTTP Send Timeout—0 HTTP2—None
LB_PROF_HTTPS_SERVERCLOSE	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Client Address—Disabled X-Forwarded-For—Disabled X-Forwarded-For Header—None IP Reputation—Disabled HTTP Mode—Server Close Compression—None Decompression—None Caching—None Geo IP Block List—None Geo IP Allowlist—None Geo IP Redirect URL—http://

Profile	Defaults
	HTTP Send Timeout—0 HTTP2—None
LB_PROF_SMTP	Starttls Active Mode—require Forbidden Command—expn, turn, vrfy Local Certificate Group—LOCAL_CERT_GROUP Client Address—Disable Forbidden Command Status—Enable Domain Name—default.com Geo IP Block List—None Geo IP Allowlist—None
LB_PROF_RTSP	Max Header Size—Default is 4096. Valid values range from 2048 to 65536. Client Address—Disabled by default. When enabled, FortiADC will use the client address to connect to the server pool.
LB_PROF_RTMP	Client Address—Disabled by default. When enabled, FortiADC will use the client address to connect to the server pool.
LB_PROF_HTTP2_H2	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 HTTP Send Timeout—0 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Client Address—Disabled X-Forwarded-For—Disabled IP Reputation—Disabled HTTP Mode—Keep Alive Compression—None Decompression—None HTTP2—LB_HTTP2_PROFILE_DEFAULT Caching—None Geo IP Block List—None Geo IP Allow list—None Geo IP Redirect URL—http:// Tune Buffer Size—17418 Max HTTP Headers—200 Response Half Closed Connection—Disabled
LB_PROF_HTTP2_H2C	Client Timeout—50 Server Timeout—50

Profile	Defaults
	Connect Timeout—5 Queue Timeout—5 HTTP Send Timeout—0 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Client Address—Disabled X-Forwarded-For—Disabled IP Reputation—Disabled HTTP Mode—Keep Alive Compression—None Decompression—None HTTP2—LB_HTTP2_PROFILE_DEFAULT Caching—None Geo IP Block List—None Geo IP Allowlist—None Geo IP Redirect URL—http:// Tune Buffer Size—17418 Max HTTP Headers—200 Response Half Closed Connection—Disabled
LB_PROF_HTTP3	Client Timeout—50 Server Timeout—50 Connect Timeout—5 Queue Timeout—5 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 X-Forwarded-For—Disabled HTTP Mode—Keep Alive HTTP3—LB_HTTP3_PROFILE_DEFAULT Tune Buffer Size—32768 Max HTTP Headers—200
LB_PROF_ISO8583	Timeout TCP Session—100 Message Encode Type—ASCII Length Indicator Type—binary Length Indicator Shift—0 Length Indicator Size—2 Optional Header Length—2 Optional Trailer Hex—None Geo IP Block List—None Geo IP Allowlist—None

Profile	Defaults
LB_PROF_EXPLICIT_HTTP	Client Timeout—50 Server Timeout—50 Connect Timeout—50 Queue Timeout—50 HTTP Send Timeout—0 HTTP Request Timeout—50 HTTP Keepalive Timeout—50 Client Address—Disabled X-Forwarded-For—Disabled X-Forwarded-For Header—None IP Reputation—Disabled HTTP Mode—Keep Alive Decompression—None Geo IP Block List—None Geo IP Allowlist—None Geo IP Redirect URL—http:// Tune Buffer Size—8030 Max HTTP Headers—100 Response Half Closed Connection—Disabled
LB_PROF_L7_TCP	Timeout TCP Session—100 IP Reputation—Disabled Geo IP Block List—None Geo IP Allowlist—None

Before you begin:

- You must have already created configuration objects for certificates, caching, and compression if you want the profile to use them.
- You must have read-write permission for load balance settings.

Syntax

```
config load-balance profile
   edit <name>
      set type {diameter | dns | explicit_http | ftp | http | turbohttp | https | ip |
           iso8583 | l7-tcp | l7-udp | mssql | mysql | radius | rdp | rtmp | rtsp | sip |
           smtp | tcp | tcps | udp}
      set timeout_tcp_session <integer>
      set timeout_tcp_session_after_FIN <integer>
      set timeout_send_rst {enable|disable}
      set timeout-radius-session <integer>
      set timeout_udp_session <integer>
      set buffer-pool {enable|disable}
      set caching <datasource>
      set cache-response-type {single-answer | round-robin}
```

```
        set client-address {enable|disable}
        set client-timeout <integer>
        set compression <datasource>
        set connect-timeout <integer>
        set http-keepalive-timeout <integer>
        set http-mode {KeepAlive|OnceOnly|ServerClose}
        set http-request-timeout <integer>
        set http-x-forwarded-for {enable|disable}
        set http-x-forwarded-for-header <string>
        set queue-timeout <integer>
        set server-timeout <integer>
        set tune-bufsize <integer>
        set tune-maxrewrite <integer>
        set ip-reputation {enable|disable}
        set geoip-list <datasource>
        set allowlist <datasource>
        set security-mode {none|explicit|implicit}
        set geoip-redirect <string>
        set client-keepalive {enable|disable}
        set client-protocol {tcp|udp}
        set failed-client {drop|send}
        set failed-client-str <string>
        set failed-server {drop|send}
        set failed-server-str <string>
        set max-size <integer>
        set server-keepalive {enable|disable}
        set server-keepalive-timeout <integer>
        set server-protocol {tcp|udp}
        set sip-insert-client-ip {enable|disable}
        set media-addr <ip address>
        set dynamic-auth {enable|disable}
        set dynamic-auth-port <integer>
        config client-request-header-erase
           edit <No.>
              set type {all|first}
              set string <string>
           next
        end
        config client-request-header-insert
           edit <No.>
              set type {append-always | append-if-not-exist | insert-always insert-if-not-
                   exist}
              set string <string>
           next
        end
        config client-response-header-erase
           edit <No.>
              set type {all|first}
              set string <string>
           next
        end
        config client-response-header-insert
           edit <No.>
              set type {append-always | append-if-not-exist | insert-always insert-if-not-
                   exist}
              set string <string>
           next
```

```
      end
      config server-request-header-erase
         edit <No.>
            set type {all|first}
            set string <string>
         next
      end
      config server-request-header-insert
         edit <No.>
            set type {append-always | append-if-not-exist | insert-always insert-if-not-
                 exist}
            set string <string>
         next
      end
      config server-response-header-erase
         edit <No.>
            set type {all|first}
            set string <string>
         next
      end
      config server-response-header-insert
         edit <No.>
            set type {append-always | append-if-not-exist | insert-always insert-if-not-
                 exist}
            set string <string>
         next
      end
   next
end
```

The following commands are used to invoke the "LB_PROF_DNS" profile in Layer-7 virtual servers.

```
config load-balance profile
   edit "dns"
      set type dns
      set cache-response-type {all-records | round-robin}
      set caching {enable|disable}
      set client-address {enable|disable}
      set malform-query-action {drop|forward}
      set max-cache-age <integer>
      set max-cache-entry-size <integer>
      set max-cache-size <integer>
      set max-query-length <integer>
      set redirect-to-tcp-port {enable|disable}
   next
end

config load-balance virtual-server
   edit "vs1"
      set load-balance-profile LB_PROF_DNS
   next
end
```

The following commands are used to invoke the "LB_PROF_IP" profile in Layer-2 virtual servers. When the profile of a Layer-2 virtual server is set to "LB_PROF_IP", you must specify the protocol numbers the virtual server can accept.

```
config load-balance profile
   edit "ip"
      set type ip
      set timeout-ip-session <integer>
      set ip-reputation {enable|disable}
      set geoip-list <string>
      set allowlist <string>
   next
end

config load-balance virtual-server
   edit "LB_PROF_IP"
      set type l2-load-balance
      set load-balance-profile LB_PROF_IP
      set protocol-numbers <value> protocol range "A-B" or single protocol number "A"
   next
end
```

The following commands are used to configure MySQL load-balancing:

```
config load-balance profile
   edit "mysql"
      set type mysql
      set mysql-mode {single-primary|sharding}
   next
end
```

The following commands are used to create a new MySQL profile (basic configuration):

```
config load-balance profile
   edit <name>
      config mysql-user-password
         edit <id>
               set username <username>
               set password <password>
         next
      end
   next
end
```

The following commands are used to configure a MySQL profile in basic single-primary mode:

```
config load-balance profile
   edit <name>
      config mysql-rule
         edit <rule id>
               set type [primary| secondary]
               set database <database name> <database name> ...
               set user <user name> <user name> ...
               set table <table name> <table name> ...
               set client-ip <client ip> <client ip> ...
               set sql <sql statement> <sql statement> ...
         next
      end
   next
end
```

The following commands are used to configure a MySQL profile in data-sharding mode:

```
config load-balance profile
```

```
   edit <name>
      set mysql-mode sharding
      config mysql-sharding
         edit <id>
               set type range
               set table <table name>
               set key <column name>
               set group <group id>:<range> <group id>:<range> ... # such as set groups 0:0-
                    999 1:1000-9999
         next
         edit <id>
               set type hash
               set database <database name>
               set table <table name>
               set key <column name>
               set group <group id> <group id>
         next
      end
   next
end
```

The following commands are used to configure MySQL profile-specific pool members:

```
config load-balance pool
   edit <pool name>
      config pool_member
         edit 1
               set mysql-group-id <group id> #for Data Sharding
               set mysql-read-only enable #for secondary
         next
      end
   next
end
```

The following commands are used to create an RTSP profile:

```
config load-balance profile
edit "RTSP"
      set type rtsp
      set max-header-size <size>
      set client-address <enable/disable>
   next
```

The following commands are used to configure an RTMP profile:

```
config load-balance profile
edit "RTMP"
      set type rtmp
      set client-address <enable/disable>
   next
```

The following commands are used to configure a diameter proxy_mode profile:

```
config load-balance profile
   edit "diameter_proxy"
      set type diameter
      set origin-host <string>
      set origin-realm <string>
      set client-ssl {enable|disable}
      set vendor-id <integer>
      set product-name <string>
```

```
      set idle-timeout <integer>
      set server-close-propagation <enable/disable>
   next
end
```

The following commands are used to configure a diameter relay_mode profile:

```
config load-balance profile
   edit "diameter_proxy"
      set type diameter
      set idle-timeout <integer>
      set server-close-propagation <enable/disable>
   next
end
```

The following commands are used to configure an explicit HTTP profile:

```
config load-balance profile
   edit <name>
      set type explicit_http
      set caching <string>
      set client-address {enable|disable}
      set client-timeout <integer>
      set connect-timeout <integer>
      set decompression <string>
      set geoip-list <string>
      set geoip-redirect <string>
      set http-keepalive-timeout <integer>
      set http-request-timeout <integer>
      set http-send-timeout <integer>
      set http-x-forwarded-for {enable|disable}
      set http-x-forwarded-for-header <string>
      set ip-reputation {enable|disable}
      set max-http-headers {enable|disable}
      set queue-timeout <integer>
      set response-half-closed-request {enable|disable}
      set server-timeout <integer>
      set tune-bufsize <integer>
      set tune-maxrewrite {enable|disable}
      set allowlist <string>
   next
end
```

`type`	Specify the profile type. After you have specified the type, the CLI commands are constrained to the ones that are applicable to the specified type, not all of the settings described in this table.
set type tcp	
`timeout_tcp_session`	Client-side timeout for connections where the client has not sent a FIN signal, but the connection has been idle. The default is 100 seconds. The valid range is 1 to 86,400.
`timeout_tcp_session_ after_FIN`	Client-side connection timeout. The default is 100 seconds. The valid range is 1 to 86,400.
`timeout_send_rst`	Enable to send TCP RST to the client and real server when the TCP session expires. This is disabled by default.

	Note: This function is supported for both IPv4 and IPv6 in L4 and L2 virtual servers. For L4 virtual servers, `timeout_send_rst` is supported for DNAT/FullNAT/NAT46/NAT64 packet forwarding methods.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.
`geoip-list`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
set type ip	
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.
`geoip-list`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
`timeout-ip-session`	Client-side session timeout. The default is 100 seconds. The valid range is 1 to 86,400.
set type dns	
`client-address`	Enable/disable to use the original client IP address as the source address when connecting to the real server.
`caching`	Enable/disable the cache for the DNS virtual server.
`max-cache-age`	Specify the cache age-out time (in seconds). The default is 3,600. The valid range is 0 to 65,535.
`max-cache-size`	Specify the maximum cache size (in Megabytes). The default is 10. The valid range is 1 to 100.
`max-cache-entry-size`	Specify the maximum cache entry size. The default is 512. The valid range is 256 to 4,096.
`cache-response-type`	Select either of the following cache response types: • single • round-robin
`malform-query-action`	Select either of the following reactions for the malformed requests: • drop • forward
`max-query-length`	Specify the maximum query length. The default is 512. The valid range is 256 to 4,096.
`redirect-to-tcp-port`	Enable/disable to authenticate client by redirecting UDP query to TCP.
set type udp	
`stateless`	Enable to apply the UDP stateless function.
`timeout_udp_session`	Client-side session timeout. The default is 100 seconds. The valid range is 1 to 86,400.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.

`geoip-list`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
set type rtsp	
`max-header-size`	Specify the maximum size of RTSP packets, which can range from 16 to 65, 536.
`client-address`	Enable/disable to use the original client IP address as the source address when connecting to the real server.
set type rtmp	
`client-address`	Enable/disable to use the original client IP address as the source address when connecting to the real server.
set type ftp	
`timeout_tcp_session`	Client-side timeout for connections where the client has not sent a FIN signal, but the connection has been idle. The default is 100 seconds. The valid range is 1 to 86,400.
`timeout_tcp_session_ after_FIN`	Client-side connection timeout. The default is 100 seconds. The valid range is 1 to 86,400.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.
`geoip-list`	Specify a Geo IP block list configuration object.
`allowlist`	Specify a Geo IP allowlist configuration object.
`security-mode`	Select either of the following: • none • explicit • implicit
set type http and **set type https**	
`tune-bufsize`	Specify the buffer size for a session when buffer-pool is enabled. Specify lower values to allow more sessions to coexist in the same amount of RAM, and higher values for traffic with larger HTTP body content. The default is 8,030 bytes. The valid range is 128 to 2,147,483,647. **Note**: The tune-bufsize factors into the total allowable size for an HTTP request. The default maximum length of an HTTP request is calculated as *tune-bufsize <integer> - tune-maxrewrite <integer>*. For example, tune-bufsize 8030 - tune-maxrewrite 1024 = 7006 bytes is the maximum for the HTTP request.
`caching`	Specify the name of the caching configuration object.
`client-address`	Use the original client IP address as the source address in the connection to the real server.

`client-timeout`	Client-side TCP connection timeout. The default is 50 seconds. The valid range is from 1 to 3,600.
`compression`	Specify a compression configuration object.
`connect-timeout`	Multiplexed server-side TCP connection timeout. Usually less than the client-side timeout. The default is 5 seconds. The valid range is 1 to 3,600.
`http-keepalive-timeout`	The default is 50 seconds. The valid range is 1 to 3,600.
`http-mode`	• KeepAlive. Do not close the connection to the real server after each HTTP transaction. Instead, keep the connection between FortiADC and the real server open until the client-side connection is closed. This option is required for applications like Microsoft SharePoint. • OnceOnly. An HTTP transaction can consist of multiple HTTP requests (separate requests for an HTML page and the images contained therein, for example). To improve performance, the "once only" flag instructs the FortiADC to evaluate only the first set of headers in a connection. Subsequent requests belonging to the connection are not load balanced, but sent to the same server as the first request. • ServerClose. Close the connection to the real server after each HTTP transaction.
`http-request-timeout`	Client-side HTTP request timeout. The default is 50 seconds. The valid range is 1 to 3,600.
`http-x-forwarded-for`	Enable this option to append the client IP address found in IP layer packets to the HTTP header, for example, `X-forwarded-for: 192.168.161.100`. The default header name is `X-forwarded-for`. If you prefer a different name, use `http-x-forwarded-for-header` to define a custom name.
`http-x-forwarded-for-header`	Specify a custom name for the HTTP header which carries the client IP address. Do not include the 'X-' prefix. Examples: Forwarded-For, Real-IP, or True-IP.
`queue-timeout`	Specifies how long connection requests to a backend server remain in a queue if the server has reached its maximum number of connections. If the timeout period expires before the client can connect, FortiADC drops the connection and sends a 503 error to the client. The default is 5 seconds. The valid range is 1 to 3,600.
`server-timeout`	Server-side IP session timeout. The default is 50 seconds. The valid range is 1 to 3,600.
`tune-maxrewrite`	Specify the buffer space reserved for content rewriting. The default is 1,024 bytes. The valid range is 128 to 2,147,483,647. **Note**: The tune-maxrewrite factors into the total allowable size for an HTTP request. The default maximum length of an HTTP request is calculated as *tune-bufsize <integer> - tune-maxrewrite <integer>*. For example, tune-bufsize 8030 - tune-maxrewrite 1024 = 7006 bytes is the maximum for the HTTP request.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.

`geoip-list`	Specify a Geo IP block list configuration object.
`geoip-redirect`	For HTTP/HTTPS, if you have configured a Geo IP redirect action, specify a redirect URL.
`allowlist`	Specify a Geo IP allowlist configuration object.
`http2-profile`	Specify an HTTP2 profile configuration object.
`http3-profile`	The **http3-profile** option is only available if **type** is **https**. Specify an HTTP3 Profile configuration object. See config load-balance http3-profile on page 161.
set type radius	
`timeout-radius-session`	The default is 300 seconds. The valid range is 1 to 3,600.
`dynamic-auth`	Enable/disable RADIUS dynamic authorization (CoA, Disconnect messages).
`dynamic-auth-port`	Dynamic auth port.
`client-address`	Enable/disable the use of a client IP as the source IP to connect to the real server.
`geoip-list`	Specify a Geo IP block list configuration object.
`allowlist`	Specify a Geo IP allowlist configuration object.
set type rdp	
`client-timeout`	Client-side TCP connection timeout. The default is 50 seconds. The valid range is 1 to 3,600.
`server-timeout`	Server-side IP session timeout. The default is 50 seconds. The valid range is 1 to 3,600.
`connect-timeout`	Multiplexed server-side TCP connection timeout. Usually less than the client-side timeout. The default is 5 seconds. The valid range is 1 to 3,600.
`queue-timeout`	Specifies how long connection requests to a backend server remain in a queue if the server has reached its maximum number of connections. If the timeout period expires before the client can connect, FortiADC drops the connection and sends a 503 error to the client. The default is 5 seconds. The valid range is 1 to 3,600.
`client-address`	Use the original client IP address as the source address in the connection to the real server.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.
`geoip-list`	Specify a Geo IP block list configuration object.
`allowlist`	Specify a Geo IP allowlist configuration object.
set type tcps	
`client-timeout`	Client-side TCP connection timeout. The default is 50 seconds. The valid range is 1 to 3,600.

server-timeout	Server-side IP session timeout. The default is 50 seconds. The valid range is 1 to 3,600.
connect-timeout	Multiplexed server-side TCP connection timeout. Usually less than the client-side timeout. The default is 5 seconds. The valid range is 1 to 3,600.
queue-timeout	Specifies how long connection requests to a backend server remain in a queue if the server has reached its maximum number of connections. If the timeout period expires before the client can connect, FortiADC drops the connection and sends a 503 error to the client. The default is 5 seconds. The valid range is 1 to 3,600.
client-address	Use the original client IP address as the source address in the connection to the real server.
ip-reputation	Enable to apply the FortiGuard IP reputation service.
geoip-list	Specify a Geo IP block list configuration object.
allowlist	Specify a Geo IP allowlist configuration object.
set type turbohttp	
timeout_tcp_session	Client-side timeout for connections where the client has not sent a FIN signal, but the connection has been idle. The default is 100 seconds. The valid range is 1 to 86,400.
timeout_tcp_session_after_FIN	Client-side connection timeout. The default is 100 seconds. The valid range is 1 to 86,400.
ip-reputation	Enable to apply the FortiGuard IP reputation service.
geoip-list	Specify a Geo IP block list configuration object.
allowlist	Specify a Geo IP allowlist configuration object.
set type sip	
client-keepalive	Enable/disable a keepalive period for new client-side requests. Supports CRLF ping-pong for TCP connections. Disabled by default.
client-address	Use the original client IP address as the source address in the connection to the real server.
media-addr	Change the media address of SIP payload to specified address. 0.0.0.0 is default.
client-protocol	Client-side transport protocol: • tcp • udp (default)
failed-client	Action when the SIP client cannot be reached: • drop—Drop the connection. • send—Drop the connection and send a message, for example, a status code and error message.
fail-client-str	Message string. Use double-quotation marks for strings with spaces.
failed-server	Action when the SIP server cannot be reached:

	• drop—Drop the connection. • send—Drop the connection and send a message, for example, a status code and error message.
`fail-server-str`	Message string. Use double-quotation marks for strings with spaces. For example: `"404 Not Found"`
`max-size`	Maximum message size. The default is 65535 bytes. The valid range is 1-65535.
`server-keepalive`	Enable/disable a keepalive period for new server-side requests. Supports CRLF ping-pong for TCP connections. Enabled by default.
`server-keepalive-timeout`	Maximum wait for a new server-side request to appear. The default is 30 seconds. The valid range is 5-300.
`server-protocol`	Server-side transport protocol. • tcp • udp Default is "unset", so the client-side protocol determines the server-side protocol.
`sip-insert-client-ip`	Enable/disable option to insert the client source IP address into the X-Forwarded-For header of the SIP request.
set type explicit_http	
`caching`	Caching name.
`client-address`	Use client address to connect to pool.
`client-timeout`	The maximum inactivity time on the client side.
`connect-timeout`	The maximum time to wait for a connection attempt to a server to succeed.
`decompression`	The decompression name.
`geoip-list`	The geography IP block list.
`geoip-redirect`	Redirect URL for IP geography.
`http-keepalive-timeout`	The maximum allowed time to wait for a new HTTP request to appear.
`http-request-timeout`	The maximum allowed time to wait for a complete HTTP request.
`http-send-timeout`	The timeout (in seconds) of HTTP send out all the buffered data.
`http-x-forwarded-for`	Insert X-Forwarded-For header to request.
`http-x-forwarded-for-header`	Change X-Forwarded-For header name.
`ip-reputation`	Use IP Reputation
`max-http-headers`	Max HTTP headers limit. **Note:** If enlarge this limit, you may meet parse failure because the buffer size limit.
`queue-timeout`	The maximum time to wait in the queue for a connection slot to be free.

`response-half-closed-request`	If enabled, FortiADC will continue serving the request in half closed connection until the response completes.
`server-timeout`	The maximum inactivity time on the server side.
`tune-bufsize`	Specify the buffer size for a session when buffer-pool is enabled. Specify lower values to allow more sessions to coexist in the same amount of RAM, and higher values for traffic with larger HTTP body content. The default is 8,030 bytes. The valid range is 128 to 2,147,483,647. **Note**: The tune-bufsize factors into the total allowable size for an HTTP request. The default maximum length of an HTTP request is calculated as *tune-bufsize <integer> - tune-maxrewrite <integer>*. For example, tune-bufsize 8030 - tune-maxrewrite 1024 = 7006 bytes is the maximum for the HTTP request.
`tune-maxrewrite`	Specify the buffer space reserved for content rewriting. The default is 1,024 bytes. The valid range is 128 to 2,147,483,647. **Note**: The tune-maxrewrite factors into the total allowable size for an HTTP request. The default maximum length of an HTTP request is calculated as *tune-bufsize <integer> - tune-maxrewrite <integer>*. For example, tune-bufsize 8030 - tune-maxrewrite 1024 = 7006 bytes is the maximum for the HTTP request.
`allowlist`	The geography IP allowlist.
config client-request-header-erase	Configuration to erase headers from client requests. Table setting. Maximum 4 members.
`type`	• all—Parse all headers for a match. • first—Parse the first header for a match.
`string`	Header to be erased.
config client-request-header-insert	Configuration to insert headers into client requests. Table setting. Maximum 4 members.
`type`	• append-always—Append after the last header. • append-if-not-exist—Append only if the header is not present. • insert-always—Insert before the first header even if the header is already present. • insert-if-not-exist—Insert before the first header only if the header is not already present.
`string`	The header:value pair to be inserted.
config client-response-header-erase	Configuration to erase headers from client responses. Table setting. Maximum 4 members.
`type`	• all • first

`string`	Header to be erased.
config client-response-header-insert	Configuration to insert headers into client responses. Table setting. Maximum 4 members.
`type`	• append-always • append-if-not-exist • insert-always • insert-if-not-exist
`string`	The header:value pair to be inserted.
config server-request-header-erase	Configuration to erase headers from server requests. Table setting. Maximum 4 members.
`type`	• all • first
`string`	Header to be erased.
config server-request-header-insert	Configuration to insert headers into server requests. Table setting. Maximum 4 members.
`type`	• append-always • append-if-not-exist • insert-always • insert-if-not-exist
`string`	The header:value pair to be inserted.
server-response-header-erase	Configuration to erase headers from server responses. Table setting. Maximum 4 members.
`type`	• all • first
`string`	Header to be erased.
server-response-header-insert	Configuration to insert headers into server responses. Table setting. Maximum 4 members.
`type`	• append-always • append-if-not-exist • insert-always • insert-if-not-exist
`string`	The header:value pair to be inserted.
set type diameter	
`origin-host`	Sets the value of Diameter AVP 264. This AVP can be a character string and specifies the identity of the originating host for Diameter messages. ADC will modify the origin-host avp on client request with the setting value, then transfer it to RS.

	ADC will modify the origin-host avp on server response with the setting value, then transfer it to the client. Specify the identity in the following format: vs.realm The host is a string unique to the client. The realm is the Diameter realm, specified by the Realm option (described below). If origin-host is set with an empty value (nothing), ADC will not change the value of the origin-host in the client or server when it transfers them. The default is empty value.
`origin-realm`	Sets the value of Diameter AVP 296. This AVP can be a character string and specifies the Diameter realm from which Diameter messages, including requests, are originated. ADC will modify the origin-realm avp on client request with the setting value, then transfer it to RS. ADC will modify the origin-realm avp on server response with the setting value, then transfer it to the client. If origin-realm is set with an empty value(nothing), ADC will not change the value of the origin-realm in the client or server when it transfers them. The default is empty value.
`product-name`	Sets the value of Diameter AVP 269. This AVP can be a character string and specifies the product; for example, “fortiadc”. ADC will modify the Product-Name avp on client request with the setting value, then transfer it to RS. ADC will modify the Product-Name avp on server response with the setting value, then transfer it to the client. If product-name is set with an empty value(nothing), ADC will not change the value of the origin-realm in the client or server when it transfers them. The default is empty value.
`Vendor-id`	Sets the value of Diameter AVP 266. This AVP can be a character string and specifies the vendor; for example, “156”. ADC will modify the vendor-id avp on client request with the setting value, then transfer it to RS. ADC will modify the vendor-id avp on server response with the setting value, then transfer it to the client. If vendor-ide is set to 0, ADC will not change the value of vendor-id in the client or server when it transfers them. The default is 0. The valid range is 0-4294967295.
`Idle-timeout`	Default, for different requests with the same session_id avp, if their interval is less than idle-timeout, ADC will dispatch them to the same RS. The default is 300 seconds. The valid range is 1-86400. When this parameter is set, ADC will act in proxy mode.
`server-close-propagation`	When transferring diameter traffic with server-close-propagation enabled, if one of the servers resets or sends DPR to ADC, ADC will close connection with the client and other servers at the same time.

	When transferring diameter traffic with server-close-propagation disabled, if one of the servers resets or sends DPR to ADC, ADC will transfer the requests from the client to the other servers. Disabled by default.
set type iso8583	
`timeout_tcp_session`	Client-side timeout for connections where the client has not sent a FIN signal, but the connection has been idle. The default is 100 seconds. The valid range is 1 to 86,400 seconds.
`msg-encode-type`	Specify the encode type for protocol message, default ASCII.
`length-indicator-type`	Specify the encode type of length indicator, default binary.
`length-indicator-shift`	Specify bytes to shift from the beginning of payload to read length value, range 0-32.
`length-indicator-size`	Specify total bytes reading to calculate length, range 0-8.
`opt-header-length`	Specify length of optional header before MTI, including the length-indicator, range 0-32.
`opt-trailer-hex`	Specify hex string of optional trailer, maximum length 16, i.e. 8 bytes in binary.
`geo-ip`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
set type mssql	
`client-timeout`	This timeout is counted as the amount of time when the client did not send a complete request HTTP header to the FortiADC after the client connected to the FortiADC. If this timeout expires, FortiADC will send a 408 message to client and close the connection to the client. The default is 50 seconds. The valid range is 1 to 86,400 seconds.
`server-age`	Specify the maximum inactivity time for MS SQL server on the server side. The default is 600 seconds. The valid range is 1 to 86,400 seconds.
`server-max-size`	Specify the maximum connections that can connect to the MS SQL server on the server side. The default is 10,000. The valid range is 1 to 30,000.
`geo-ip`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
set type smtp	
`client-address`	Enable/disable to use the original client IP address as the source address when connecting to the real server. **Note**: When using the NAT Source Pool for SMTP VS, ensure the SMTP application profile is disabled for Client Address. When the SMTP is enabled for Client Address, it will use the original client IP address as the source address when connecting to the real server, which cannot be done when the NAT source pool is used at the same time.

`starttls-active-mode`	Select one of the following: • allow—The client can either use or not use the STARTTLS command. • require—The STARTTLS command must be used to encrypt the connection first. • none—The STARTTLS command is NOT supported.
`disable-command-status`	Enable/disable to forbid the command(s) selected in forbidden-command.
`disable-command`	Select any, all, or none of the commands (i.e., expn, turn, vrfy). If selected, the command or commands will be rejected by FortiADC; otherwise, the command or commands will be accepted and forwarded to the back end.
`geo-ip`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
`domain-name`	Specify the domain name.
set type mysql	**Note**: The system does not provide default MyQSL profiles as it does with the other protocols.
`mysql-mode`	Select either of the following MySQL modes: • single-primary — The profile will use the single-primary mode. You will then need to specify and configure the primary server and secondary servers. • sharding— The profile will use the sharding mode to load-balance MySQL traffic.
set type l7-tcp	
`timeout_tcp_session`	Client-side timeout for connections where the client has not sent a FIN signal, but the connection has been idle. The default is 100 seconds. The valid range is 1 to 86,400.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.
`geoip-list`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.
set type l7-udp	
`timeout_udp_session`	Client-side session timeout. The default is 100 seconds. The valid range is 1 to 86,400.
`ip-reputation`	Enable to apply the FortiGuard IP reputation service.
`geoip-list`	Select a Geo IP block list configuration object.
`allowlist`	Select an allowlist configuration object.

Example

The following example shows the list of predefined profiles:

```
FortiADC-VM # get load-balance profile
== [ LB_PROF_TCP ]
== [ LB_PROF_UDP ]
```

```
== [ LB_PROF_HTTP ]
== [ LB_PROF_TURBOHTTP ]
== [ LB_PROF_FTP ]
== [ LB_PROF_RADIUS ]
== [ LB_PROF_SIP ]
== [ LB_PROF_TCPS ]
== [ LB_PROF_HTTPS ]
== [ LB_PROF_HTTP2_H2C]
== [ LB_PROF_HTTP2_H2 ]
== [ LB_PROF_SMTP ]
== [ LB_PROF_RTSP ]
== [ LB_PROF_RTMP ]
== [ LB_PROF_DIAMETER ]
== [ LB_PROF_IP ]
== [ LB_PROF_RDP ]
== [ LB_PROF_HTTP_SERVERCLOSE ]
== [ LB_PROF_HTTPS-SERVERCLOSE ]
== [ LB_PROF_DNS ]
```

The following example shows the details of the predefined HTTPS profile:

```
FortiADC-VM (profile) # get load-balance profile LB_PROF_HTTPS
type : https
tune-bufsize : 8030
tune-maxrewrite : 1024
client-timeout : 50
server-timeout : 50
connect-timeout : 5
queue-timeout : 5
http-request-timeout : 50
http-keepalive-timeout : 50
buffer-pool : enable
client-address : disable
http-x-forwarded-for : disable
http-x-forwarded-for-header :
http-mode : ServerClose
compression :
caching :
ip-reputation : disable
geoip-list :
allowlist :
geoip-redirect : http://
```

The following example creates a user-defined SIP profile:

```
FortiADC-VM # config load-balance profile
FortiADC-VM (profile) # edit sip-profile
Add new entry 'sip-profile' for node 1643
FortiADC-VM (sip-profile) # set type sip
FortiADC-VM (sip-profile) # get
type : sip
max-size : 65535
server-keepalive-timeout : 30
server-keepalive : enable
client-keepalive : disable
client-protocol : udp
server-protocol :
```

```
sip-insert-client-ip : disable
failed-client : drop
failed-server : drop
FortiADC-VM (sip-profile) # set timeout 120
FortiADC-VM (sip-profile) # set max-size 2048
FortiADC-VM (sip-profile) # set server-keepalive-timeout 180
FortiADC-VM (sip-profile) # set failed-server send
FortiADC-VM (sip-profile) # set fail-server-str "404 Not Found"
FortiADC-VM (sip-profile) # config ?
client-request-header-erase erase header from client request
client-request-header-insert insert header into client request
client-response-header-erase erase header from client response
client-response-header-insert insert header into client response
server-request-header-erase erase header from server request
server-request-header-insert insert header into server request
server-response-header-erase erase header from server response
server-response-header-insert insert header into server response
FortiADC-VM (sip-profile) # config client-request-header-insert
FortiADC-VM (client-request~h) # edit 1
Add new entry '1' for node 4554
FortiADC-VM (1) # set type insert-if-not-exist
FortiADC-VM (1) # set string "Via: SIP/2.0/UDP 1.1.1.100:5060"
FortiADC-VM (1) # end
FortiADC-VM (sip-profile) # end
FortiADC-VM #
```

The following example creates a DNS profile:

```
config load-balance profile
   edit "dns"
      set type dns
      set malform-query-action drop
      set redirect-to-tcp-port disable
      set caching enable
      set max-query-length 512
      set max-cache-age 3600
      set max-cache-entry-size 512
      set max-cache-size 10
   next
end

config load-balance virtual-server
   edit "vs1"
      set load-balance-profile dns
   next
end
```

The following example creates an IP profile:

```
config load-balance profile
   edit "ip"
      set type ip
      set timeout-ip-session 100
   next
end

config load-balance virtual-server
```

```
   edit "vs2"
      set type l2-load-balance
      set protocol-numbers 0 1
      set load-balance-profile ip
   next
end
```

The following example creates a MySQL profile:

```
config system health-check
   edit mysql
      set type mysql
      set user root
      set password fortinet
      set port 3306
   next
end

config load-balance real-server
   edit "rs1"
      set ip 192.168.1.1
   next
end


config load-balance pool
   edit "pool_mysql"
      set health-check-ctrl enable
      set health-check-list icmp
      set real-server-ssl-profile NONE
      config pool_member
         edit 1
              set pool_member_cookie rs1
              set real-server rs1
         next
      end
   next
end


config load-balance virtual-server
   edit "mysql"
      set type l7-load-balance
      set interface port2
      set ip 10.1.1.1
      set port 3306
      set load-balance-profile mysql
      set load-balance-method LB_METHOD_ROUND_ROBIN
      set load-balance-pool pool_mysql
   next
end
```

The following example creates an RTSP profile:

```
config load-balance profile
edit "RTSP"
```

```
    set type rtsp
    set max-header-size 2048
    set client-address enable
  next
```

The following example creates an RTMP profile:

```
config load-balance profile
edit "RTMP"
    set type rtmp
    set client-address enable
  next
```

config load-balance real-server-ssl-profile

Use this command to configure real server profiles. A real server profile determines settings used in network communication on the FortiADC-server segment, in contrast to a virtual server profile, which determines the settings used in network communication on the client-FortiADC segment.

Table 12 provides a summary of the predefined profiles. You can select predefined profiles in the real server configuration, or you can create user-defined profiles.

Predefined real server profiles

Profile	Defaults
LB_RS_SSL_PROF_DEFAULT	• Allow version: SSLv3, TLSv1.0, TLSv1.1, and TLSv1.2 • Cipher suite list: custom
LB_RS_SSL_PROF_ECDSA	• Allow version: SSLv3, TLSv1.0, TLSv1.1, and TLSv1.2 • Cipher suite list: ECDHE-ECDSA-AES256-GCM-SHA384, ECDHE-ECDSA-AES256-SHA384, ECDHE-ECDSA-AES256-SHA, ECDHE-ECDSA-AES128-GCM-SHA256,ECDHE-ECDSA-AES128-SHA256,ECDHE-ECDSA-AES128-SHA,ECDHE-ECDSA-RC4-SHA,ECDHE-ECDSA-DES-CBC3-SHA
LB_RS_SSL_PROF_ECDSA_SSLV3	• Allow version: SSLv3 • Cipher suite list: ECDHE-ECDSA-AES256-SHA, ECDHE-ECDSA-AES128-SHA,ECDHE-ECDSA-RC4-SHA,ECDHE-ECDSA-DES-CBC3-SHA
LB_RS_SSL_PROF_ECDSA_TLS12	• Allow version: TLSv1.2 • Cipher suite list: ECDHE-ECDSA-AES256-GCM-SHA384, ECDHE-ECDSA-AES256-SHA384, ECDHE-ECDSA-AES128-GCM-SHA256,ECDHE-ECDSA-AES128-SHA256
LB_RS_SSL_PROF_ENULL	• Allow version: SSLv3, TLSv1.0, TLSv1.1, and TLSv1.2 • Cipher suite list: eNull

Profile	Defaults
	Recommended for Microsoft Direct Access servers where the application data is already encrypted and no more encryption is needed.
LB_RS_SSL_PROF_HIGH	• Allow version TLSv1.2 • Cipher suite list: ECDHE-RSA-AES256-GCM-SHA384 ECDHE-RSA-AES256-SHA384 ECDHE-RSA-AES256-SHA DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES256-SHA256 AES256-GCM-SHA384 AES256-SHA256
LB_RS_SSL_PROF_LOW_SSLV3	• Allow version SSLv3 • Cipher suite list: DHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES128-SHA256 DHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA ECDHE-RSA-RC4-SHA RC4-MD5 ECDHE-RSA-DES-CBC3-SHA EDH-RSA-DES-CBC3-SHA DES-CBC3-SHA EDH-RSA-DES-CBC-SHA DES-CBC-SHA
LB_RS_SSL_PROF_MEDIUM	• Allow version: TLSv1.0, TLSv1.1, and TLSv1.2 • Cipher suite list: ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA DHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES128-SHA256 DHE-RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA RC4-SHA EDH-RSA-DES-CBC3-SHA DES-CBC3-SHA
LB_RS_SSL_PROF_NONE	SSL is disabled.

Before you begin:

- You must have read-write permission for load balance settings.

Syntax

```
config load-balance real-sever-ssl-profile
   edit <name>
      set ssl {enable|disable}
      set allow-ssl-versions {sslv3 tlsv1.0 tlsv1.1 tlsv1.2 tlsv1.3}
      set local-cert <datasource>
      set server-cert-verify <datasource>
      set ssl-ciphers {ECDHE-ECDSA-AES256-GCM-SHA384 ECDHE-ECDSA-AES256-SHA384 ECDHE-ECDSA-
           AES256-SHA ECDHE-ECDSA-AES128-GCM-SHA256 ECDHE-ECDSA-AES128-SHA256 ECDHE-ECDSA-
           AES128-SHA ECDHE-ECDSA-DES-CBC3-SHA ECDHE-ECDSA-RC4-SHA ECDHE-RSA-AES256-GCM-
           SHA384 ECDHE-RSA-AES256-SHA384 ECDHE-RSA-AES256-SHA DHE-RSA-AES256-GCM-SHA384
           DHE-RSA-AES256-SHA256 DHE-RSA-AES256-SHA AES256-GCM-SHA384 AES256-SHA256 AES256-
           SHA ECDHE-RSA-AES128-GCM-SHA256 ECDHE-RSA-AES128-SHA256 ECDHE-RSA-AES128-SHA DHE-
           RSA-AES128-GCM-SHA256 DHE-RSA-AES128-SHA256 DHE-RSA-AES128-SHA AES128-GCM-SHA256
           AES128-SHA256 AES128-SHA ECDHE-RSA-RC4-SHA RC4-SHA RC4-MD5 ECDHE-RSA-DES-CBC3-SHA
           EDH-RSA-DES-CBC3-SHA DES-CBC3-SHA EDH-RSA-DES-CBC-SHA DES-CBC-SHA eNULL }
      set ssl-customize-ciphers-flag {enable|disable}
      set ssl-customized-ciphers <string>
      set ssl-session-reuse {enable|disable}
```

```
      set ssl-session-reuse-limit <integer>
      set ssl-sni-forward {enable|disable}
      set ssl-tls-ticket-reuse {enable|disable}
      server-OCSP-stapling-support {enable|disable}
      set rfc7919-comply {enable|disable}
      set supported-groups {secp256r1 secp384r1 secp521r1 x25519 x448 ffdhe2048 ffdhe3072
           ffdhe4096 ffdhe6144 ffdhe8192}
   next
end
```

`ssl`	Enable/disable SSL for the connection between the FortiADC and the real server.
`allow-ssl-versions`	Specify a space-separated list of allowed SSL versions. • sslv3 • tlsv1.0 • tlsv1.1 • tlsv1.2 • tlsv1.3 **Note**: • Please make sure that the SSL version is continuous. If not, an error message should be returned. • RFC 7919 Comply cannot support SSLv3 and TLS 1.3. If **rfc7919-comply** is enabled and **sslv3** or **tlsv1.3** is selected in **ssl-allowed-versions**, an error message will display.
`local-cert`	Specify a local certificate object.
`server-cert-verify`	Specify a Certificate Verify configuration object to validate server certificates. This Certificate Verify object must include a CA group and can include OCSP and CRL checks.
`ssl-ciphers`	Specify the supported SSL ciphers in a space-separated list. Ciphers are listed from strongest to weakest: • ECDHE-ECDSA-AES256-GCM-SHA384 • ECDHE-ECDSA-AES256-SHA384 • ECDHE-ECDSA-CAMELLIA256-SHA384 • *ECDHE-ECDSA-AES256-SHA • ECDHE-ECDSA-AES128-GCM-SHA256 • *ECDHE-ECDSA-AES128-SHA256 • ECDHE-ECDSA-CAMELLIA128-SHA256 • *ECDHE-ECDSA-AES128-SHA • ECDHE-ECDSA-DES-CBC3-SHA • ECDHE-ECDSA-RC4-SHA • ECDHE-RSA-AES256-GCM-SHA384 • ECDHE-RSA-AES256-SHA384 • ECDHE-RSA-CAMELLIA256-SHA384 • *ECDHE-RSA-AES256-SHA • DHE-RSA-AES256-GCM-SHA384 • *DHE-RSA-AES256-SHA256

	• DHE-RSA-CAMELLIA256-SHA256 • *DHE-RSA-AES256-SHA • DHE-RSA-CAMELLIA256-SHA • AES256-GCM-SHA384 • *AES256-SHA256 • *AES256-SHA • ECDHE-RSA-AES128-GCM-SHA256 • *ECDHE-RSA-AES128-SHA256 • ECDHE-RSA-CAMELLIA128-SHA256 • *ECDHE-RSA-AES128-SHA • DHE-RSA-AES128-GCM-SHA256 • *DHE-RSA-AES128-SHA256 • DHE-RSA-CAMELLIA128-SHA256 • *DHE-RSA-AES128-SHA • AES128-GCM-SHA256 • *AES128-SHA256 • *AES128-SHA • ECDHE-RSA-RC4-SHA • RC4-SHA • RC4-MD5 • ECDHE-RSA-DES-CBC3-SHA • EDH-RSA-DES-CBC3-SHA • DES-CBC3-SHA • EDH-RSA-DES-CBC-SHA • DES-CBC-SHA • eNULL *These ciphers are fully supported by hardware SSL (in 400F, 1200F, 2200F, 4200F and 5000F).
`ssl-customize-ciphers-flag`	Enable/disable use of user-specified cipher suites.
`ssl-customized-ciphers`	If the customize cipher flag is enabled, specify a colon-separated, ordered list of cipher suites. An empty string is allowed. If empty, the default cipher suite list is used.
`ssl-session-reuse`	Enable/disable SSL session reuse.
`ssl-session-reuse-limit`	The default is 0 (disabled). The valid range is 0-1048576.
`ssl-sni-forward`	Enable/disable forwarding the client SNI value to the server. The SNI value will be forwarded to the real server only when the client-side ClientHello message contains a valid SNI value; otherwise, nothing is forwarded.
`ssl-tls-ticket-reuse`	Enable/disable TLS ticket-based session reuse.
`server-OCSP-stapling-support`	Enable/disable server `ocsp_stapling`. The default is disable. Note: Only when verify is enabled does this command take effect.
`rfc7919-comply`	Enable/disable parameters to comply with RFC 7919. **Note**:

	• RFC 7919 Comply cannot support SSLv3 and TLS 1.3. If **rfc7919-comply** is enabled and **sslv3** or **tlsv1.3** is selected in **ssl-allowed-versions**, an error message will display.
`supported-groups`	The **supported-groups** option is available if **rfc7919-comply** is enabled. Specify the supported group objects from the following: • secp256r1 • secp384r1 • secp521r1 • x25519 • x448 • ffdhe2048 • ffdhe3072 • ffdhe4096 • ffdhe6144 • ffdhe8192 At least one item from the FFDHE group must be selected. **Note**: The RFC 7919 Comply feature requires certain cipher selections to correspond with the Supported Group selection. • If a FFDHE group is selected (for example, ffdhe2048), then at least one cipher must be DHE-RSA (for example, DHE-RSA-AES256-SHA256). • If the Supported Group includes groups other than FFDHE (such as a SECP group, secp256r1), then at least one cipher must be ECDHE (for example, ECDHE-ECDSA-AES256-GCM-SHA384). • If a ECDHE cipher is selected (for example, ECDHE-ECDSA-AES256-GCM-SHA384), then the Supported Group must include at least one group that is not FFDHE (such as a SECP group, secp256r1).

Example

```
FortiADC-VM # config load-balance real-server-ssl-profile
FortiADC-VM (real-server-ss~-) # get
== [ LB_RS_SSL_PROF_NONE ]
== [ LB_RS_SSL_PROF_LOW_SSLV2 ]
== [ LB_RS_SSL_PROF_LOW_SSLV3 ]
== [ LB_RS_SSL_PROF_MEDIUM ]
== [ LB_RS_SSL_PROF_HIGH ]
== [ LB_RS_SSL_PROF_ECDSA ]
== [ LB_RS_SSL_PROF_ECDSA_SSLV3 ]
== [ LB_RS_SSL_PROF_ECDSA_TLS12 ]
== [ LB_RS_SSL_PROF_ENULL ]
== [ LB_RS_SSL_PROF_DEFAULT ]

FortiADC-VM (real-server-ss~-) # edit RS-SSL-PROFILE-USER-DEFINED
Add new entry 'RS-SSL-PROFILE-USER-DEFINED' for node 3862
FortiADC-VM (RS-SSL-PROFILE~U) # set ssl enable
FortiADC-VM (RS-SSL-PROFILE~U) # get
```

```
ssl : enable
local-cert:
server-cert-verify :
ssl-sni-forward : disable
ssl-session-reuse : disable
ssl-customize-ciphers-flag : disable
ssl-ciphers : DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES256-SHA256 DHE-RSA-AES256-SHA AES256-GCM-
     SHA384 AES256-SHA256 AES256-SHA DHE-RSA-AES128-GCM-SHA256 DHE-RSA-AES128-SHA256 DHE-
     RSA-AES128-SHA AES128-GCM-SHA256 AES128-SHA256 AES128-SHA RC4-SHA RC4-MD5 EDH-RSA-DES-
     CBC3-SHA DES-CBC3-SHA EDH-RSA-DES-CBC-SHA DES-CBC-SHA
allow-ssl-versions : sslv3 tlsv1.0 tlsv1.1 tlsv1.2
FortiADC-VM (RS-SSL-PROFILE~U) # set ssl-session-reuse enable
FortiADC-VM (RS-SSL-PROFILE~U) # set allow-ssl-versions tlsv1.2
FortiADC-VM (RS-SSL-PROFILE~U) # end
FortiADC-VM #
```

config load-balance reputation

Use this command to configure IP reputation policies.

The FortiGuard IP Reputation service provides a regularly updated data set that identifies compromised and malicious clients.

The IP reputation configuration allows you to specify the action the system takes when it receives traffic from a client with an IP address on the list. Table 13 lists limitations for IP reputation actions.

IP reputation actions

Action		Profile Limitations
Pass	IPv4 only	Not supported for RADIUS.
Deny	IPv4 only	Not supported for RADIUS.
Redirect	IPv4 only	Not supported for RADIUS, FTP, TCP, UDP.
Send 403 Forbidden	IPv4 only	Not supported for RADIUS, FTP, TCP, UDP.

Note: IP reputation is also not supported for Layer 4 virtual servers when the Packet Forwarding Mode is Direct Routing.

Basic Steps

1. Configure the connection to the FortiGuard IP Reputation Service.
2. Optionally, customize the actions you want to take when the system encounters a request from an IP source that matches the list; and add exceptions. If a source IP appears on the exceptions list, the system does not look it up on the IP reputation list. See below.
3. Enable IP reputation in the profiles you associate with virtual servers.

Before you begin:

- You must have read-write permission for load balancing settings.

Syntax

```
config load-balance reputation
   edit <No.>
      set action {deny | pass | redirect | send-403-forbidden}
      set category <string>
      set log {enable|disable}
      set severity {high | low | medium}
      set status {enable|disable}
   next
end
```

action	• Pass • Deny • Redirect • Send 403 Forbidden *Note*: Layer 4 and TCPS virtual servers do not support Redirect or Send 403 Forbidden. If you apply an IP reputation configuration that uses these options to a Layer 4 or TCPS virtual server, FortiADC logs the action as Redirect or Send 403 Forbidden, but in fact denies the traffic.
category	Specify a FortiGuard IP Reputation category: • Botnet • Anonymous Proxy • Phishing • Spam • Block List • Others
log	Enable/disable logging.
severity	The severity to apply to the event. Severity is useful when you filter and sort logs: • Low • Medium • High
status	Enable/disable the category.

Example

```
FortiADC-VM # get load-balance reputation
== [ 1 ]
== [ 2 ]
== [ 3 ]
== [ 4 ]
== [ 5 ]
== [ 6 ]

FortiADC-VM # get load-balance reputation 1
category : Botnet
status : enable
action : pass
severity : low
log : disable
```

```
FortiADC-VM # get load-balance reputation 2
category : "Anonymous Proxy"
status : enable
action : pass
severity : low
log : disable

FortiADC-VM # get load-balance reputation 3
category : Phishing
status : enable
action : pass
severity : low
log : disable

FortiADC-VM # get load-balance reputation 4
category : Spam
status : enable
action : pass
severity : low
log : disable

FortiADC-VM # get load-balance reputation 5
category : Others
status : enable
action : pass
severity : low
log : disable

FortiADC-VM # get load-balance reputation 6
category : "Block List"
status : enable
action : deny
severity : low
log : disable
```

config load-balance reputation-exception

Use this command to add exceptions to IP reputation rules. If enabled, the specified IP address or range of IP addresses will be allowed to pass through.

Before you begin:

- You must have read-write permission for load balancing feature settings.

Syntax

```
config load-balance reputation-exception
   edit <No.>

      set status {enable|disable}
      set type {ip-netmask | ip-range}
```

```
      set ip-network <ip&netmask>
      set start-ip <classip>
      set end-ip <classip>
   next
end
```

status	Enable or disable the exception. You might have occasion to toggle to exception off and on.
type	• ip-netmask: address block • ip-range: address range
ip-network	Specify a subnet using the address/mask notation.
start-ip	Specify the start of an address range.
end-ip	Specify the end of an address range.

config load-balance reputation-block-list

Use this command to create a new ip-reputation block list. If enabled, the specified IP address or range of IP addresses will not be allowed to pass through.

Before you begin:

- You must have read-write permission for load balancing feature settings.

Syntax

```
config load-balance reputation-block-list
   config entries
   edit <No.>
      set status {enable|disable}
      set type { ip-netmask|ip-range}
      set ip-network <ip&netmask>
      set end-ip <ip>
      set start-ip <ip>
   next
end
```

status	Enable or disable the exception. You might have occasion to toggle to exception off and on.
type	• ip-netmask: address block • ip-range: address range
IP/Netmask	If IP/netmask is selected in the Type field above, specify a subnet using the address/mask notation.
start-ip	Specify the start of an address range.
end-ip	Specify the end of an address range.

config load-balance schedule-pool

Use this command to create a new schedule-pool which can control the working real server by schedule-group (config system schedule-group).

Syntax

```
config load-balance schedule-pool
   edit <name>
      set load-balance-pool <datasource>
      set schedule <datasource>
   next
end
```

load-balance-pool	Specify a real server pool.
schedule	Schedule-group (configured by "config system schedule-group").

Example

```
config load-balance schedule-pool
   edit "new"
      set load-balance-pool example-pool
      set schedule example-schedule
   next
end
```

config load-balance virtual-server

Use this command to configure virtual servers.

The virtual server configuration supports three classes of application delivery control:

- Layer 7—Persistence, load-balancing, and routing are based on Layer-7 objects, such as HTTP headers, cookies, and so on.
- Layer 4—Persistence, load-balancing, and network address translation are based on Layer-4 objects, such as source and destination IP address.
- Layer 2—This feature is useful when the request's destination IP is unknown and you need to load-balance connections between multiple next-hop gateways.

Before you begin:

- You must have a deep understanding of the backend servers and your load balancing objectives.
- You must have configured a real server pool (required) and other configuration objects that you can incorporate into the virtual server configuration, such as persistence rules, user-defined profiles, source IP address pools if you are deploying full NAT, content routes and rewriting rules, and error messages.
- You must have read-write permission for load balancing settings.

Unlike virtual IPs on FortiGate or virtual servers on FortiWeb, virtual servers on FortiADC are activated as soon as you configure them and set status to `enable`. You do not apply them by selecting them in a policy.

Syntax

```
config load-balance virtual-server
   edit <vs-name>
      set type {l2-load-balance | l4-load-balance | l7-load-balance}
      set addr-type {ipv4|ipv6}
      set alone {enable|disable}
      set auth-policy <datasource>
      set clone-pool <datasource>
      set clone-traffic-type {both-sides|client-side|server-side}
      set comments <string>
      set connection-limit <integer>
      set connection-pool <datasource>
      set connection-rate-limit <integer>
      set content-rewriting {enable|disable}
      set content-rewriting-list <string>
      set content-routing {enable|disable}
      set content-routing-list <string>
      set error-msg <string>
      set geoip-block-list <datasource>
      set allowlist <datasource>
      set interface <datasource>
      set ip <class_ip>
      set l2-exception-list <datasource>
      set port <value> port range "portA-portB" or single port number "portA"
      set port <number>
      set load-balance-method <datasource>
      set load-balance-persistence <datasource>
      set load-balance-pool <datasource>
      set load-balance-profile <datasource>
      set client-ssl-profile <datasource>
      set multi-process <integer>
      set packet-forwarding-method {FullNAT|NAT|NAT46|NAT64|direct_routing| tunneling}
      set ippool-list <datasource> <datasource> ...
      set scripting-flag enable
      set scripting-list <datasource> <datasource> ...
      set status {enable|disable|maintain}
      set traffic-log {enable|disable}
      set event-log {enable|disable}
      set trans-rate-limit <integer>
      set waf-profile <datasource>
      set warm-rate <integer>
      set warm-up <integer>
      set traffic-group <string>
      set ssl-mirror {enable|disable}
      set ssl-mirror-intf <port>
      set pagespeed <datasource>
      set http2https enable
      set http2https-port <portA-portB portC portD>
      set max-persistence-entries <integer>
```

```
      set schedule-list {enable|disable}
      set schedule-pool-list <datasource>
      set dos-profile <datasource>
      set ztna-profile <datasource>
      set one-click-gslb-server-option {enable|disable}
    next
end
```

type	Specify the virtual server type: • l7-load-balance: Persistence, load balancing, and routing are based on Layer 7 objects, such as HTTP headers, cookies, and so on. • l4-load-balance: Persistence, load balancing, and network address translation are based on Layer 4 objects, such as source and destination IP address. • l2-load-balance: This feature is useful when the request's destination IP is unknown and you need to load balance connections between multiple next-hop gateways. After you have specified the type, the CLI commands are constrained to the ones that are applicable to the specified type, not all of the settings described in this table.
addr-type	IPv4 or IPv6 **Note**: IPv6 is not supported for layer 4 FTP, layer 2 FTP, HTTP Turbo or RDP.
alone	Enable/disable alone mode. Enabled by default. When enabled, the virtual server is handled by a separate httproxy daemon. When disabled, the virtual server belongs to a group that is handled by one httproxy daemon. Alone mode boosts performance but impacts memory utilization. If memory utilization becomes an issue, consider enabling alone mode only for key virtual servers and disabling for less important ones. **Note**: HTTP, HTTPS, and TCPS only.
auth-policy	Specify an auth policy configuration object. HTTP/HTTPS only.
comments	A string to describe the purpose of the configuration, to help you and other administrators more easily identify its use. Put phrases in quotes. For example: "Customer ABC".
connection-limit	Limit the number of concurrent connections. The default is 0 (disabled). The valid range is 1 to 1,048,576 concurrent connections. You can apply a connection limit per real server and per virtual server. Both limits are enforced. Attempted connections that are dropped by security rules are not counted. **Note**: Not supported for FTP or SIP profiles.
connection-pool	Specify a connection pool configuration object. **Note**: Not supported for SIP profiles.
connection-rate-limit	With all Layer 4 profiles, and with the Layer 2 TCP profile, you can limit the number of new connections per second. The default is 0 (disabled). The valid range is 1 to 86,400 connections per second.

	You can apply a connection rate limit per real server and per virtual server. Both limits are enforced. Attempted connections that are dropped by security rules are not counted. **Note**: Not supported for FTP profiles.
`content-rewriting`	Enable to rewrite HTTP headers. **Note**: Not supported for SIP profiles.
`content-rewriting-list`	Specify content rewriting rules. **Note**: You can select multiple content rewriting rules in the virtual server configuration. Rules that you add are consulted from top to bottom. The first rule to match is applied. If the traffic does not match any of the content rewriting rule conditions, the header is not rewritten.
`content-routing`	Enable to route packets to backend servers based on IP address (Layer 4) or HTTP headers (Layer 7 content). **Note**: Not supported for SIP profiles. Supports L2 TCP/UDP/IP profiles.
`content-routing-list`	Specify content route configuration objects. **Note**: You can specify multiple content routing rules in the virtual server configuration. Rules that you add are consulted from top to bottom. The first rule to match is applied. If the traffic does not match any of the content routing rule conditions specified in the virtual server configuration, the system behaves unexpectedly. Therefore, it is important that you create a “catch all” rule that has no match conditions. In the virtual server configuration, this rule should be ordered last so it can be used to forward traffic to a default pool.
`error-msg`	Specify an error page configuration object. **Note**: Not supported for SIP profiles.
`error-page`	If you do not use an error page, you can enter an error message to be returned to clients in the event no server is available. **Note**: Not supported for SIP profiles.
`geoip-blocklist`	Specify a geography IP address block list configuration object.
`allowlist`	Specify a geography IP address allowlist configuration object.
`interface`	Network interface that receives client traffic for this virtual server.
`ip`	IP address provisioned for the virtual server. **Note**: You do not specify an IP address for a Layer 2 virtual server. A Layer 2 virtual server is not aware of IP addresses. Instead of routing data for a specific destination, this type of server simply forwards data from the specified network interface and port.
`port`	Port number to listen for client requests. **Note**: If a Layer 2 virtual server is assigned a network interface that uses port 80 or 443, ensure that the HTTPS and HTTP administrative access options are not enabled for the interface. Note: A L7 virtual server can have up to 256 ports, but there is no such a limit for L4 virtual servers.

	Note: Port number can be set to 0 if load-balance type is L4 or L2 and the profile is TCP or UDP.
`port range`	Specify the number of ports in a port range. For example, if port is 80, and port-range is 254, then the virtual port range starts at 80 and goes to 334. The default is 0 (no range). The valid range is 0-255. For SIP, the valid range is 0-5. The port-range option is useful in deployments where it is desirable to have a virtual IP address with a large number of virtual ports, such as data centers or web hosting companies that use port number to identify their specific customers. Statistics and configurations are applied to the virtual port range as a whole and not to the individual ports within the specified range. **Note**: Not supported for FTP, HTTP Turbo, RADIUS, or Layer 2 TCP profiles Note: You can define up to eight port ranges.
`load-balance-method`	Specify a predefined or user-defined method configuration object.
`load-balance-persistence`	Specify a predefined or user-defined persistence configuration object.
`load-balance-pool`	Specify a server pool configuration object.
`load-balance-profile`	Specify a predefined or user-defined profile configuration object. After you have specified the profile, the CLI commands are constrained to the ones that are applicable to the specified profile type, not all of the settings described in this table.
`client-ssl-profile`	Specify a predefined or user-defined client SSL profile configuration object. **Note**: • This setting applies to **HTTPS**, **TCPS**, HTTP2 H2, **SMTP**, and **FTPS** applications only. In the case of HTTPS, it becomes available only when SSL is enabled. • If a ZTNA Profile is referenced in the VS, ensure the client SSL profile has enabled client certificate verification for the corresponding EMS CA certificate object. See config load-balance client-ssl-profile on page 133.
`l2-exception-list`	Specify a user-defined SSL forward proxy exception configuration object.
`multi-process`	If your system has a multicore CPU, you can assign the number of CPU cores to handle traffic for a virtual server. The valid range is 1 to 15. **Note**: HTTP, HTTPS, and TCPS only.
`packet-forwarding-method`	In Layer 4 virtual server deployments, select one of the following packet forwarding methods: • direct_routing — Forwards the source and destination IP addresses with no changes. **Note**: For FTP profiles, when Direct Routing is selected, you must also configure a persistence method. • NAT— Replaces the destination IP address with the IP address of the backend server selected by the load balancer. The destination IP address of the initial request is the IP address of the virtual server. Be sure to configure FortiADC as the default gateway on the backend server so that the reply goes through FortiADC and can also be translated.

	• FullNAT—Replaces both the destination and source IP addresses. IPv4 to IPv4 or IPv6 to IPv6 translation. • NAT46—Replaces both the destination and source IP addresses, translating IPv4 addresses to IPv6 addresses. • NAT64—Replaces both the destination and source IP addresses, translating IPv6 addresses to IPv4 addresses. • Tunneling- In tunnel mode, the load balancer sends requests to real servers through an IP tunnel. When a user accesses the virtual server, a packet destined for the virtual IP address arrives, a real server is chosen from the cluster according to the connection scheduling algorithm. Then the load balancer encapsulates the packet within an IP datagram and forwards it to the chosen server. **Note**: For Full NAT, NAT46, and NAT64, the source IP address is replaced by an IP address from the pool you specify with `ippool`. The destination IP address is replaced with the IP address of the backend server selected by the load balancer.

`ippool-list`	If you are configuring a Layer 4 virtual server and enable Full NAT, NAT46, or NAT64, specify a space-separated list of IP address pool configuration objects to be used for SNAT.
	Note:
	By default, the same IP pool cannot be set up in different virtual servers. However, you can enable IP address sharing through the CLI to allow the source pool to be set up in different virtual servers.
	To enable IP address sharing:
	`config system global`
	`  set share-ip-address enable`
	`end`

`scripting-flag`	Enable by default.
`scripting-list`	Specify a scripting policy configuration object. HTTP/HTTPS only. Note: The maximum number of scripts in "`set scripting-list <>`" is 256.
`status`	• enable—The server can receive new sessions. • disable—The server does not receive new sessions and closes any current sessions as soon as possible. • maintain—The server does not receive new sessions but maintains any current connections.
`traffic-log`	Enable to record traffic logs for this virtual server. **Note**: Local logging is constrained by available disk space. We recommend that if you enable traffic logs, you monitor your disk space closely. We also recommend that you use local logging during evaluation and verification of your initial deployment, and then configure remote logging to send logs to a log management repository.
`event-log`	Enable to record event logs for this virtual server.
`trans-rate-limit`	Limit the number of HTTP or SIP requests per second. The default is 0 (disabled). The valid range is 1 to 1,048,567 transactions per second.

	The system counts each client request against the limit. When the request rate exceeds the limit, the virtual server sends an HTTP 503 error response to the client. **Note**: Not supported for HTTP Turbo profiles.
`waf-profile`	Specify a web application firewall (WAF) profile configuration object. HTTP/HTTPS only.
`warm-rate`	Maximum connection rate while the virtual server is starting up. The default is 100 connections per second. The valid range is 1 to 86,400 connections per second. If Warm Up is 5 and Warm Rate is 2, the number of allowed new connections increases at the following rate: • 1st second—Total of 2 new connections allowed (0+2). • 2nd second—2 new connections added for a total of 4 new connections allowed (2+2). • 3rd second—2 new connections added for a total of 6 new connections allowed (4+2). • 4th second—2 new connections added for a total of 8 new connections allowed (6+2). • 5th second—2 new connections added for a total of 10 new connections allowed (8+2). **Note**: Not supported for SIP profiles.
`warm-up`	If the server cannot initially handle full connection load when it begins to respond to health checks (for example, if it begins to respond when startup is not fully complete), indicate how long to forward traffic at a lesser rate. The default is 0 (disabled). The valid range is 1 to 86,400 seconds. **Note**: Not supported for SIP profiles.
`ssl-mirror`	Enable/disable SSL mirroring. When ssl-mirror is enabled, FortiADC will mirror the client HTTPS/TCPS packets traffic by the SSL-mirror-interface port after decrypting the SSL. Note: Use this command send mirror packets of HTTPS or TCPS virtual servers to third-party solutions via the designated network interfaces. See below.
`ssl-miror-intf`	Specify the outgoing interfaces be ssl-mirror interfaces. You can set up to four outgoing interfaces.
`pagespeed`	Set PageSpeed to let FortiADC speed up HTTP responses using its Web Performance Optimization solutions.
`http2https`	Enable/disable redirect HTTP request to HTTPS
`http2https-port`	HTTP service port list for redirecting HTTP to HTTPS. Format: portA-portB portC portD.
`max-persistence-entries`	Maximum persistence entries size. This command only works if load-balance-persistence is enabled with type source-address.
`schedule-list`	Enable/disable schedule pool list.
`schedule-pool-list`	Specify the schedule-pool.

clone-pool	Specify the clone-pool.
clone-traffic-type	Specify the clone-traffic-type.
dos-profile	LB process will get all the configurations of this profile and write the parameters to the configuration file of HTTProxy.
ztna-profile	Specify a ZTNA profile configuration object. **Note**: This setting applies to **Layer 7 HTTPS** and **TCPS** applications only.
one-click-gslb-server-option	Enable/disable the FortiGSLB function.

Example

```
FortiADC-VM # config load-balance virtual-server
FortiADC-VM (virtual-server) # edit lb-vs1
Add new entry 'lb-vs1' for node 1775

config load-balance virtual-server
   edit “l7vs”
      set type l7-load-balance
      set interface port1
      set ip 172.1.1.2
      set traffic-group traffic-group-1
   next
end

config load-balance virtual-server
   edit "VS"
      set type l7-load-balance
      set interface port3
      set ip 192.168.1.1
      set load-balance-profile LB_PROF_HTTPS
      set load-balance-method LB_METHOD_ROUND_ROBIN
      set load-balance-pool pool
      set scripting-flag enable
      set scripting-list HTTP_2_HTTPS_REDIRECTION REWRITE_HOST_n_PATH REDIRECTION_by_STATUS_
           CODE
      set traffic-group default
      set clone-pool 1
      set clone-traffic-type both-sides
      set dos-profile dos-profile
      set ztna-profile ztna-profile
   next
end


FortiADC-VM (lb-vs1) # get
status : enable
type : l4-load-balance
multi-process : 1
packet-forwarding-method: NAT
interface :
addr-type : ipv4
```

```
ip : 0.0.0.0
port : 80
connection-limit : 10000
load-balance-profile:
content-routing : disable
load-balance-persistence:
load-balance-method :
load-balance-pool :
traffic-log : disable
warm-up : 0
warm-rate : 10
connection-rate-limit: 0
id : 0
clone-pool : 1
clone-traffic-type : both-sides
FortiADC-VM (lb-vs1) # set ip 192.168.200.1
FortiADC-VM (lb-vs1) # set interface port4
FortiADC-VM (lb-vs1) # set load-balance-profile LB_PROF_TCP
FortiADC-VM (lb-vs1) # set load-balance-method LB_METHOD_ROUND_ROBIN
FortiADC-VM (lb-vs1) # set load-balance-pool lb-pool
FortiADC-VM (lb-vs1) # end

FortiADC-VM # get load-balance virtual-server lb-vs1
status : enable
type : l4-load-balance
multi-process : 1
packet-forwarding-method: NAT
interface : port4
addr-type : ipv4
ip : 192.168.200.1
port : 80
connection-limit : 10000
load-balance-profile: LB_PROF_TCP
content-routing : disable
load-balance-persistence:
load-balance-method : LB_METHOD_ROUND_ROBIN
load-balance-pool : lb-pool
traffic-log : disable
warm-up : 0
warm-rate : 10
connection-rate-limit: 0
id : 1
```

config load-balance web-category

Read-only. Displays the web filter categories imported from FortiGuard. You specify web categories when you create web filter groups with the config load-balance web-filter-profile command.

For information on FortiGuard web categories, go to the FortiGuard website:

http://fortiguard.com/webfilter

Before you begin:

- You must have read permission for load balancing settings.

Example

```
docs-1 # get load-balance web-category
                                        == [ Potentially Liable ]
                                        == [ Adult/Mature Content ]
                                        == [ Bandwidth Consuming ]
                                        == [ Security Risk ]
                                        == [ General Interest - Personal ]
                                  == [ General Interest - Business ]
```

See Also

- config system web-filter

config load-balance web-filter-profile

Use this command to configure web filter profile. The web filter profile should include categories that should not be processed by the outbound L2 SSL forward proxy feature. To address privacy concerns, you can include categories such as "Personal Privacy", "Finance and Banking", "Health and Wellness", and Medicine.

Before you begin:

- You must have read-write permission for load balancing settings.

After you have configured web filter profile, you can specify it in an L2 exception list.

Syntax

```
config load-balance web-filter-profile
   edit <name>
      set description <string>
      config category-members
         edit <No.>
            set category <datasource>
         next
      end
   next
end
```

description	A string to describe the purpose of the configuration, to help you and other administrators more easily identify its use. Put phrases in quotes. For example: "Customer ABC".
config member	
category	Specify a FortiGuard category or subcategory. Put phrases in quotes. For example: "Personal Privacy".

Example

```
FortiADC-docs # config load-balance web-filter-profile
FortiADC-VM (web-filter-pro~i) # edit fortiguard-categories2passthrough
Add new entry 'fortiguard-categories2passthrough' for node 4622
FortiADC-VM (fortiguard-cat~g) # set description "Finance and Banking Personal Privacy
     Health and Wellness"
FortiADC-VM (fortiguard-cat~g) # config category-members
FortiADC-VM (category-members) # edit 1
Add new entry '1' for node 4625
FortiADC-VM (1) # set category "Finance and Banking"

FortiADC-VM (1) # next
FortiADC-VM (category-members) # edit 2
Add new entry '2' for node 4625
FortiADC-VM (2) # set category "Personal Privacy"
FortiADC-VM (2) # next
FortiADC-VM (category-members) # edit 3
Add new entry '3' for node 4625
FortiADC-VM (3) # set category "Health and Wellness"
FortiADC-VM (3) # next
FortiADC-VM (category-members) # edit 4
Add new entry '4' for node 4625
FortiADC-VM (4) # set category Medicine
FortiADC-VM (4) # end
FortiADC-VM (fortiguard-cat~g) # end
```

config load-balance web-sub-category

Read-only. Displays the web filter subcategories imported from FortiGuard. You specify web subcategories when you create web filter groups with the config load-balance web-filter-profile command.

For information on FortiGuard web categories, go to the FortiGuard website:

http://fortiguard.com/webfilter

Before you begin:

- You must have read permission for load balancing settings.

Example

```
docs-1 # get load-balance web-sub-category
                                == [ Unrated ]
                                == [ Drug Abuse ]
                                == [ Alternative Beliefs ]
                                == [ Hacking ]
                                == [ Illegal or Unethical ]
                                == [ Discrimination ]
                                == [ Explicit Violence ]
                                == [ Abortion ]
                                == [ Other Adult Materials ]
                                == [ Advocacy Organizations ]
                                == [ Gambling ]
                                == [ Extremist Groups ]
                                == [ Nudity and Risque ]
                                == [ Pornography ]
                                == [ Dating ]
                                == [ Weapons (Sales) ]
                                == [ Advertising ]
                                == [ Brokerage and Trading ]
                                == [ Freeware and Software Downloads ]
                                == [ Games ]
                                == [ Web-based Email ]
                                == [ File Sharing and Storage ]
                                == [ Streaming Media and Download ]
                                == [ Malicious Websites ]
                                == [ Entertainment ]
                                == [ Arts and Culture ]
                                == [ Education ]
                                == [ Finance and Banking ]
                                == [ Health and Wellness ]
                                == [ Job Search ]
                                == [ Medicine ]
                                == [ News and Media ]
                                == [ Social Networking ]
                                == [ Political Organizations ]
                                == [ Reference ]
                                == [ Global Religion ]
                                == [ Search Engines and Portals ]
                                == [ Shopping ]
                                == [ General Organizations ]
                                == [ Society and Lifestyles ]
                                == [ Sports ]
                                == [ Travel ]
                                == [ Personal Vehicles ]
                                == [ Business ]
                                == [ Information and Computer Security ]
                                == [ Government and Legal Organizations ]
                                == [ Information Technology ]
                                == [ Armed Forces ]
                                == [ Dynamic Content ]
                                == [ Meaningless Content ]
                                == [ Web Hosting ]
                                == [ Marijuana ]
                                == [ Folklore ]
```

```
== [ Proxy Avoidance ]
== [ Phishing ]
== [ Plagiarism ]
== [ Sex Education ]
== [ Alcohol ]
== [ Tobacco ]
== [ Lingerie and Swimsuit ]
== [ Sports Hunting and War Games ]
== [ Web Chat ]
== [ Instant Messaging ]
== [ Newsgroups and Message Boards ]
== [ Digital Postcards ]
== [ Peer-to-peer File Sharing ]
== [ Internet Radio and TV ]
== [ Internet Telephony ]
== [ Child Education ]
== [ Real Estate ]
== [ Restaurant and Dining ]
== [ Personal Websites and Blogs ]
== [ Secure Websites ]
== [ Content Servers ]
== [ Child Abuse ]
== [ Web-based Applications ]
== [ Domain Parking ]
== [ Spam URLs ]
== [ Personal Privacy ]
== [ Dynamic DNS ]
== [ Auction ]
```

See Also

- config system web-filter

config load-balance allowlist

Use this command to configure the Geography IP address allowlist.You use the allowlist to permit requests from clients that otherwise might be denied by the Geography IP address block list. For example, you might have a good reason to block requests from the whole address range for a country, except for the addresses for your known customers.

Before you begin:

- You must have read-write permission for load balancing settings.

After you have configured a Geography IP address allowlist, you can specify it in the virtual server configuration.

Syntax

```
config load-balance allowlist
   edit <name>
      set description <string>
      set status {enable|disable}
      config allowlist-member
         edit <No.>
            set description <string>
            set type {ip-netmask|ip-range}
            set ip-network <ip&netmask>
            set start-ip <ip>
            set end-ip <ip>
      next
   next
end
```

`description`	A string to describe the purpose of the configuration, to help you and other administrators more easily identify its use. Put phrases in quotes. For example: `"Customer ABC"`.
`status`	Enable/disable the list.
config allowlist-member	
`description`	Enter a brief description of the IP subnet or IP range, depending on which Type you choose. The description can be up to 1023 characters long. Valid characters are `A-Z, a-z, 0-9, _, -, .,` and `:`. No space is allowed.
`type`	Select and configure either of the following: • ip-netmask — Specify the IP address and CIDR-formatted subnet mask, separated by a forward slash (/), such as 192.0.2.0/24. Dotted quad formatted subnet masks are not accepted. IPv6 addresses are not supported. • ip-range — Specify the Start IP and the End IP addresses of the IP range.
`ip-network`	The **ip-network** option is available if **type** is **ip-netmask**. Specify the IP address and CIDR-formatted subnet mask, separated by a forward slash, such as 192.0.2.0/24. Dotted quad formatted subnet masks are not accepted. IPv6 addresses are not supported.
`start-ip`	The **start-ip** option is available if **type** is **ip-range**. Specify the Start IP address of the IP range. IPv6 addresses are not supported.
`end-ip`	The **end-ip** option is available if **type** is **ip-range**. Specify the End IP address of the IP range. IPv6 addresses are not supported.

Example

```
FortiADC-VM # config load-balance allowlist

FortiADC-VM (allowlist) # edit demo
Add new entry 'demo' for node 2893

FortiADC-VM (demo) # get
```

```
description : IP-geo-allow-list
status : enable

FortiADC-VM (demo) # set description "Customer ABC."

FortiADC-VM (demo) # config allowlist-member

FortiADC-VM (allowlist-member) # edit 1
Add new entry '1' for node 2897

FortiADC-VM (1) # get
ip-network : 0.0.0.0/0

FortiADC-VM (1) # set ip-network 192.0.2.0/24

FortiADC-VM (1) # end

FortiADC-VM (demo) # get
description : "Customer ABC."
status : enable
== [ 1 ]

FortiADC-VM (demo) # end
```

config log

The `config log` commands configure logging.

This chapter is a reference for the following commands:

config log fast_report

Use this command to configure fast reports.

Before you begin:

- You must have read-write permission for log settings.

Syntax

```
config log fast_report
   edit <Name>
      set module {slb|attack}
      set history_runchart {enable|disable}
      set range {1DAY | 1HOUR | 1MONTH| 1WEEK | 10MINS}
      set traffic_data_type {bytes|sessions}
      set slb_subtype {top_browser | top_dest | top_dev | top_domain | top_os | top_referrer
           | top_session | top_source_country | top_src | top_url }
      set filter_object {srccountry|dstcountry}
      set filter_value <string>
      set topx <integer>
      set topy <integer>
   next
end
```

`module`	Either of the following modules: • slb • attack

history_runchart	Enable/disable the history runchart.
range	Past day, hour, month, week, or 10 minutes.
traffic_data_type	Query by session count or bytes.
slb_subtype	Query subtype.
filter_object	Filter by source country or destination country. *Optional.*
filter_value	The country to be filtered.
topx	The number of the top x results.
topy	The number of the top y results.

Example

```
FortiADC-VM # config log fast_report
FortiADC-VM (fast_report) # edit fast-report
Add new entry 'fast-report' for node 4590
FortiADC-VM (fast-report) # get
module : slb
history_runchart : disable
range : 10MINS
traffic_data_type : bytes
slb_subtype : top_src
filter_object :
filter_value :
FortiADC-VM (fast-report) # set filter_object srccountry
FortiADC-VM (fast-report) # set filter_value "United States"
FortiADC-VM (fast-report) # end
```

Example

```
FortiADC-VM # config log fast_report
FortiADC-VM (fast_report) # edit "all_attack"
Add new entry 'all_attack' for node 4590
FortiADC-VM (all_attack) # set module attack
FortiADC-VM (all_attack) # set history_runchart enable
FortiADC-VM (all_attack) # set attack_sort_type count
FortiADC-VM (all_attack) # set attack_subtype top_attack_type_for_all
FortiADC-VM (all_attack) # unset filter_object
FortiADC-VM (all_attack) # unset filter_value
FortiADC-VM (all_attack) # set topx 5
FortiADC-VM (all_attack) # set topy 5
FortiADC-VM (all_attack) # get
module : attack
history_runchart : enable
attack_sort_type : count
attack_subtype : top_attack_type_for_all
filter_object :
filter_value :
topx : 5
```

```
topy : 5
FortiADC-VM (all_attack) #set filter_object srccountry
FortiADC-VM (all_attack) # set filter_value "United States"
FortiADC-VM (all_attack) # end
```

config log report

Use this command to configure on-demand or scheduled reports.

Before you begin:

- You must have read-write permission for log settings.

Syntax

```
config log report
   edit <name>
      set email-format pdf
      set email-attachname <string>
      set email-body <string>
      set email-compress {enable|disable}
      set email-subject <string>
      set on-schedule {enable|disable}
      set period-relative {absolute|last-2-weeks|last-7-days|last-14-days|last-30-days|last-
           N-days|last-N-hours|last-N-weeks| last-month|last-quarter|last-week|this-
           month|this-quarter|this-week|this-year|today|yesterday}
      set period-absolute-from <YYYY-MM-DD-HH:MM:SS>
      set period-absolute-to <YYYY-MM-DD-HH:MM:SS>
      set queryset <datasource>
      set schedule-hour <integer>
      set schedule-type {daily|weekdays}
      set schedule-weekdays {friday monday saturday sunday thursday tuesday wednesday}
   next
end
```

email-format	Attachment format. Only PDF is supported. If you schedule reports and set this option, the report is sent on schedule to all addresses in the config log report email list.
email-attachname	Filename for attachment.
email-body	Message body.
email-compress	Enable/disable compression of the attachment.
email-subject	Message subject.
on-schedule	Enable/disable reporting on schedule.
period-relative	Report period relative to the time it is generated.

period-absolute-from	If period-relative is set to absolute, specify from and to timestamps for one-time reports for a specified time range.
period-absolute-to	
queryset	Specify a space-separated list of queries to include in the report attachment. There are many predefined queries, and you can configure user-defined queries with the config log report_queryset command.
schedule-hour	0-23.
schedule-type	Daily or on specified days.
schedule-weekdays	If you do not schedule the report daily, specify the days on which to run it.

Example

```
FortiADC-docs # config log report
FortiADC-docs (report) # edit my_report
Add new entry 'my_report' for node 1962

FortiADC-docs (my_report) # get
on-schedule : enable
queryset :
email-format :
period-relative : yesterday
schedule-type : schedule-hour : 12

FortiADC-docs (my_report) # set queryset ?
<datasource> query list
SLB-Top-Policy-By-Bytes log.report_queryset
SLB-Top-Source-By-Bytes log.report_queryset
SLB-Top-Source-Country-By-Bytes log.report_queryset
SLB-History-Flow-By-Bytes log.report_queryset
LLB-Top-Link-by-Bytes log.report_queryset
LLB-History-Flow-By-Bytes log.report_queryset
DNS-Top-Policy-by-Count log.report_queryset
DNS-Top-Source-by-Count log.report_queryset
Attack-Top-Destination-For-IPReputation-By-Count log.report_queryset
Attack-Top-Source-For-IPReputation-By-Count log.report_queryset
Attack-Top-Source-Country-For-IPReputation-By-Count log.report_queryset
Attack-Top-Destination-For-GEO-By-Count log.report_queryset
Attack-Top-Source-For-GEO-By-Count log.report_queryset
Attack-Top-Source-Country-For-GEO-By-Count log.report_queryset
Attack-Top-Destination-For-WAF-By-Count log.report_queryset
Attack-Top-Source-For-WAF-By-Count log.report_queryset
Attack-Top-Source-Country-For-WAF-By-Count log.report_queryset
Attack-Top-Destination-For-Synflood-By-Count log.report_queryset
Event-Top-Admin-Login-By-Count log.report_queryset
Event-Top-Failed-Admin-Login-By-Count log.report_queryset
Event-Top-Admin-Config-By-Count log.report_queryset

FortiADC-docs (my_report) # set queryset SLB-Top-Source-Country-By-Bytes Attack-Top-Source-
     Country-For-WAF-By-Count

FortiADC-docs (my_report) # set email-format pdf
```

```
FortiADC-docs (my_report) # set schedule-type daily

FortiADC-docs (my_report) # set email-attachname "Daily_Country_Report"

FortiADC-docs (my_report) # set email-body "This report was sent by your website admin.
     Please contact admin@example.com to request changes to daily report metrics."

FortiADC-docs (my_report) # get
on-schedule : enable
queryset : SLB-Top-Source-Country-By-Bytes Attack-Top-Source-Country-For-WAF-By-Count
email-format : pdf
email-subject : "Daily Country Report" email-body : "This report was sent by your website
     admin. Please contact admin@example.com to request changes to daily report metrics."
email-attachname : Daily_Country_Report
email-compress : enable period-relative : yesterday
schedule-type : daily schedule-hour : 12

FortiADC-docs (my_report) # end
```

config log report email

Use this command to add email addresses for alert recipients.

Before you begin, make sure you have read-write permission for log settings.

Syntax

```
config log report_email
   edit <name>
      set from <string>
      set to <string>
   next
end
```

`from`	The sender's email address to be used in report email
`to`	The recipient's email address to be used in report email.

config log report_queryset

Use this command if you need to configure report queries that are different from the predefined queries.

Before you begin:

- You must have read-write permission for log settings.

After you have configured a query, you can select it in the report configuration.

Syntax

```
config log report_queryset
   edit <name>
      set module {attack|dns|event|llb|slb}
      set attack_sort_type count
      set attack_subtype {top_destip_for_geo|top_destip_for_ipreputation|top_destip_for_
           sysflood|top_destip_for_waf|top_source_country_for_geo|top_source_country_for_
           ipreputation|top_source_country_for_waf|top_source_for_geo|top_source_for_
           ipreputation|top_source_for_waf}
      set dns_sort_type count
      set dns_subtype {top_policy|top_source}
      set event_sort_type count
      set event_subtype {top_admin_config|top_admin_login|top_failed_admin_login}
      set llb_subtype {top_link|slb_history_flow}
      set slb_subtype {slb_history_flow|top_policy|top_source|top_source_country}
      set traffic_data_type {sessions|bytes}
   next
end
```

module	Set the reporting module. This setting also filters the commands so that only relevant options are available.
attack_sort_type	Results are ordered by count.
attack_subtype	Key query term.
dns_sort_type	Results are ordered by count.
dns_subtype	Key query term.
event_sort_type	Results are ordered by count.
event_subtype	Key query term.
llb_subtype	Key query term.
slb_subtype	Key query term.
traffic_data_type	Query by session count or bytes.

Example

```
FortiADC-docs # config log report_queryset
FortiADC-docs (report_queryset) # edit my_slb_query
Add new entry 'my_slb_query' for node 2514

FortiADC-docs (my_slb_query) # get
module : slb
traffic_data_type : bytes
slb_subtype : top_policy
```

```
FortiADC-docs (my_slb_query) # set slb_subtype ?
slb_history_flow slb_history_flow
top_policy top_policy
top_source top_source
top_source_country top_source_country

FortiADC-docs (my_slb_query) # set slb_subtype top_source_country

FortiADC-docs (my_slb_query) # next

FortiADC-docs (report_queryset) # edit my_attack_query
Add new entry 'my_attack_query' for node 2514

FortiADC-docs (my_attack_query) # set module attack
FortiADC-docs (my_attack_query) # set attack_subtype ?
top_destip_for_geo top_destip_for_geo
top_destip_for_ipreputation top_destip_for_ipreputation
top_destip_for_sysflood top_destip_for_sysflood
top_destip_for_waf top_destip_for_waf
top_source_country_for_geo top_source_country_for_geo
top_source_country_for_ipreputation top_source_country_for_ipreputation
top_source_country_for_waf top_source_country_for_waf
top_source_for_geo top_source_for_geo
top_source_for_ipreputation top_source_for_ipreputation
top_source_for_waf top_source_for_waf

FortiADC-docs (my_attack_query) # set attack_subtype top_source_country_for_waf

FortiADC-docs (my_attack_query) # get
module : attack
attack_sort_type : count
attack_subtype : top_source_country_for_waf

FortiADC-docs (my_attack_query) # end
FortiADC-docs #
```

config log setting fast_stats

Use this command to enable or disable real-time statistics collection for fast reports. Enabled by default. Can be disabled if you encounter issues.

Before you begin:

- You must have read-write permission for log settings.

Syntax

```
config log setting fast_stats
   set status {enable|disable}
   set traffic-log-status {enable|disable}
```

```
   set traffic-log-category slb
   set attack-log-status {enable|disable}
   set attack-log-category synflood
   ipreputation waf geo av
end
```

status	Enable/disable fast statistics. Disabled by default.
traffic-log-status	Enable/disable fast statistics for traffic logs. Disabled by default.
traffic-log-category	Enable/disable fast statistics for traffic categories. SLB is enabled by default.
attack-log-status	Enable/disable fast statistics for attack logs. Disabled by default.
attack-log-category	Enable/disable fast statistics for attack categories. Syn flood, IP reputation, WAF, GEO, and AV are enabled by default.

Example

```
docs-2 # config log setting fast_stats
docs-2 (fast_statis) # get
status : enable
traffic-log-status : enable
traffic-log-category : slb
attack-log-status : enable
attack-log-category : synflood ipreputation waf geo
```

config log setting local

Use this command to configure basic log settings.

The local log is a datastore hosted on the FortiADC system.

Typically, you use the local log to capture information about system health and system administration activities. We recommend that you use local logging during evaluation and verification of your initial deployment, and then configure remote logging to send logs to a log management repository where they can be stored long term and analyzed using preferred analytic tools.

Local log disk settings are configurable. You can select a subset of system events, traffic, and security logs.

Before you begin:

- You must have read-write permission for log settings.

Syntax

```
config log setting local
   set attack-log-cached-lines {0|100|500|800|1000|2000|5000|10000}
   set attack-log-status {enable|disable}
   set attack-log-category {av|ddos|geo|ipreputation|ips|waf|fw|ztna}
   set disk-full {overwrite | nolog}
```

```
    set event-log-cached-lines {0|100|500|800|1000|2000|5000|10000}
    set event-log-status {enable|disable}
    set event-log-category {admin|configuration|fw|glb|health-check|llb|slb|system|user}
    set loglevel {alert|critical|debug|emerge|error|information|notification|warning}
    set rate_limit <integer>
    set rotation-size <integer>
    set status {enable|disable}
    set traffic-log-cached-lines {0|100|500|800|1000|2000|5000|10000}
    set traffic-log-status {enable|disable}
    set traffic-log-category {slb|dns|llb}
    set script-log-status {enable|disable}
    set script-log-category {slb}
end
```

`attack-log-cached-lines`	Limit the number of logs that are cached. The default is 0 (disabled). Valid multiples are 100, 500, 800, 1000, 2000, 5000, 10000. If 0, every generated log is written to disk immediately. If 1000, logs are written to disk in batches of 1000.
`attack-log-status`	Enable/disable logging for security events.
`attack-log-category`	If **attack-log-status** is enabled, the **attack-log-category** becomes configurable. Select one or more of the following security categories to include in the security logs export: • ddos — DoS protection logs. • ipreputation — IP Reputation logs. • waf — WAF logs. • geo — Geo IP blocking logs. • av — AV logs. • ips — IPS logs. • fw — Firewall logs. • ztna — ZTNA logs.
`disk-full`	Specify log behavior when the maximum disk space for local logs (30% of total disk space) is reached: • overwrite—Continue logging. Overwrite the earliest logs. • nolog—Stop logging.
`event-log-cached-lines`	Limit the number of logs that are cached. The default is 0 (disabled). Valid multiples are 100, 500, 800, 1000, 2000, 5000, 10000. If 0, every generated log is written to disk immediately. If 1000, logs are written to disk in batches of 1000.
`event-log-status`	Enable/disable logging for the category.
`event-log-category`	If **event-log-status** is enabled, the **event-log-category** becomes configurable. Select one or more of the following event categories to include in the event logs export: • configuration — Configuration changes. • admin — Administrator actions. • system — System operations, warnings, and errors. • user — Authentication results logs. • health-check — Health check results and client certificate validation check results. • slb — Notifications, such as connection limit reached.

	• llb — Notifications, such as bandwidth thresholds reached. • glb — Notifications, such as the status of associated local SLB and virtual servers. • fw — Notifications for the Firewall module, such as SNAT source IP pool is using all of its addresses.
`loglevel`	Specify the lowest severity for which alerts are sent: • Emergency—The system has become unstable. • Alert—Immediate action is required. • Critical—Functionality is affected. • Error—An error condition exists and functionality could be affected. • Warning—Functionality might be affected. • Notification—Information about normal events. • Information—General information about system operations. • Debug—Detailed information about the system that can be used to troubleshoot unexpected behavior. For example, if you select `error`, the system sends alerts with level Error, Critical, Alert, and Emergency. If you select `alert`, the system sends alerts with level Alert and Emergency.
`rate_limit`	Rate limit logging (logs/second). The default is 0 (disabled).
`rotation-size`	Maximum size for a local log file. The default is 200 MB. When the current log file reaches this size, a new file is created.
`status`	Enable/disable local logging.
`traffic-log-cached-lines`	Limit the number of logs that are cached. The default is 0 (disabled). Valid multiples are 100, 500, 800, 1000, 2000, 5000, 10000. If 0, every generated log is written to disk immediately. If 1000, logs are written to disk in batches of 1000.
`traffic-log-status`	Enable/disable logging for the category.
`traffic-log-category`	If **traffic-log-status** is enabled, the **traffic-log-category** becomes configurable. Select one or more of the following traffic categories to include in the traffic logs export: • slb — Server Load Balancing traffic logs related to sessions and throughput. • dns — Global Load Balancing traffic logs related to DNS requests. • llb — Link Load Balancing traffic logs related to session and throughput.
`script-log-status`	Enable/disable script log.
`script-log-category`	Set script log category.

Example

```
FortiADC-VM (root) # get log setting local
status : enable
rotation-size : 199
disk-full : overwrite
loglevel : information
event-log-status : enable
event-log-category : configuration admin health_check system user slb llb glb fw
```

```
traffic-log-status : enable
traffic-log-category : slb dns
attack-log-status : enable
attack-log-category : synflood ipreputation waf geo
script-log-status : enable
script-log-category : slb
event-log-cached-lines : 0
traffic-log-cached-lines : 0
attack-log-cached-lines : 0
rate_limit : 0
```

config log setting remote

Use this command to configure logging to a remote syslog server.

To configure from global, see config log setting global_remote on page 67. Global has preset configurations that users may use for easy configuration, which apply to all VDOMs. However, in config log setting remote, the user can customize the configuration for the individual VDOM, overriding the global remote config.

You can enable override_global_remote here:

```
FortiADC-VM (root) # config log setting general
FortiADC-VM (general) # show full-configuration
config log setting general
set override_global_remote enable
end
```

A remote syslog server is a system provisioned specifically to collect logs for long term storage and analysis with preferred analytic tools.

Before you begin:

- You must have read-write permission for log settings.

Syntax

```
config log setting remote
   edit <name>
      set address_type {ip|fqdn}
      set attack-log-status {enable|disable}
      set attack-log-category {av|ddos|geo|ipreputation|ips|waf|fw|ztna}
      set comma-separated-value {enable|disable}
      set event-log-status {enable|disable}
      set event-log-category {admin|configuration|fw|glb|health-check|llb|slb|system|user}
      set facility {alert | audit | auth | authpriv | clock | cron | daemon | ftp | kern |
           local0, local1, local2, local3, local4, local5, local6, local7, lpr, mail, news,
           ntp}
      set fqdn <string>
      set loglevel {alert|critical|debug|emergency|error|information|notification|warning}
      set proto [udp|tcp|tcpssl]
```

```
      set tcp_framing {traditional|octet_counted}
      set port <integer>
      set server <string>
      set status {enable|disable}
      set traffic-log-status {enable|disable}
      set traffic-log-category {slb|dns|llb}
    next
end
```

attack-log-status	Enable/disable logging for security events.
attack-log-category	If **attack-log-status** is enabled, the **attack-log-category** becomes configurable. Select one or more of the following security categories to include in the security logs export: • ddos — DoS protection logs. • ipreputation — IP Reputation logs. • waf — WAF logs. • geo — Geo IP blocking logs. • av — AV logs. • ips — IPS logs. • fw — Firewall logs. • ztna — ZTNA logs.
comma-separated-value	Send logs in CSV format. Do not use with FortiAnalyzer.
event-log-status	Enable/disable logging for system events.
event-log-category	If **event-log-status** is enabled, the **event-log-category** becomes configurable. Select one or more of the following event categories to include in the event logs export: • configuration — Configuration changes. • admin — Administrator actions. • system — System operations, warnings, and errors. • user — Authentication results logs. • health-check — Health check results and client certificate validation check results. • slb — Notifications, such as connection limit reached. • llb — Notifications, such as bandwidth thresholds reached. • glb — Notifications, such as the status of associated local SLB and virtual servers. • fw — Notifications for the Firewall module, such as SNAT source IP pool is using all of its addresses.
facility	Identifier that is not used by any other device on your network when sending logs to FortiAnalyzer/syslog.
fqdn	The **fqdn** option is available if **address_type** is **fqdn**. Specify the FQDN of the syslog server.
loglevel	Select the lowest severity to log from the following options: • emergency — The system has become unstable. • alert — Immediate action is required. • critical — Functionality is affected. • error — An error condition exists and functionality could be affected. • warning — Functionality might be affected.

	• notification — Information about normal events. • information — General information about system operations. • debug — Detailed information about the system that can be used to troubleshoot unexpected behavior. The exported logs will include the selected severity level and above. For example, if you select **error**, the system collects logs with severity level error, critical, alert, and emergency. If you select **alert**, the system collects logs with severity level alert and emergency.
`port`	Listening port number of the syslog server. Usually this is UDP/TCP/TCPSSL port 514.
`server`	The **server** option is available if **address_type** is **ip**. IP address of the syslog server.
`status`	Enable/disable the configuration.
`proto`	Use protocol to transfer log messages.
`tcp_framing`	The frame in which the log message is stored in tcp/tcpssl packets.
`traffic-log-status`	Enable/disable logging for traffic processed by the load balancing modules.
`traffic-log-category`	If **traffic-log-status** is enabled, the **traffic-log-category** becomes configurable. Select one or more of the following traffic categories to include in the traffic logs export: • slb — Server Load Balancing traffic logs related to sessions and throughput. • dns — Global Load Balancing traffic logs related to DNS requests. • llb — Link Load Balancing traffic logs related to session and throughput.

Example

```
FortiADC-VM # config log setting remote
FortiADC-VM (remote) # edit 1
Add new entry '1' for node 547

FortiADC-VM (1) # get
status : disable
server : 0.0.0.0
port : 514
loglevel : information
comma-separated-value : disable
facility : kern
event-log-status : disable
traffic-log-status : disable
attack-log-status : disable

FortiADC-VM (1) # set status enable
FortiADC-VM (1) # set address_type ip
FortiADC-VM (1) # set server 203.0.113.10
FortiADC-VM (1) # set loglevel notification

FortiADC-VM (1) # set event-log-status enable
FortiADC-VM (1) # set event-log-category admin configuration system

FortiADC-VM (1) # set traffic-log-status enable
```

```
FortiADC-VM (1) # set traffic-log-category slb dns llb
FortiADC-VM (1) # end

FortiADC-VM # get log setting remote
== [ 1 ]
status: enable
server: 203.0.113.10
port: 514
loglevel: notification
facility: kern

FortiADC-VM # show log setting remote
config log setting remote
edit 1
set status enable
set server 203.0.113.10
set loglevel notification
set event-log-status enable
set event-log-category configuration admin system
set traffic-log-status enable
set traffic-log-category slb dns llb
next
end
```

config log setting fortianalyzer

Use this command to configure logging to a FortiAnalyzer server using OFTP.

The Optimized Fabric Transfer Protocol (OFTP) is used when information is synchronized between FortiAnalyzer and FortiADC, as well as for other Fortinet products. Remote logging and archiving can be configured on the FortiADC to send logs to a FortiAnalyzer unit.

OFTP listens on port TCP/514.

To configure from global, see config log setting global_faz on page 68. Global has preset configurations that users may use for easy configuration, which apply to all VDOMs. However, in config log setting fortianalyzer, the user can customize the configuration for the individual VDOM, overriding the global FortiAnalyzer config.

You can enable override_global_faz here:

```
FortiADC-VM (root) # config log setting general
FortiADC-VM (general) # show full-configuration
config log setting general
set override_global_faz enable
end
```

Requirements:

- Read-write permission for log settings.
- The FortiAnalyzer service is required to be exposed on External IP.

FortiADC supports integration with FortiAnalyzer versions 7.0.2 or later. As earlier versions of FortiAnalyzer is not optimally compatible with FortiADC, unexpected behavior may occur.

Syntax

```
config log setting fortianalyzer
   edit <name>
      set status {enable|disable}
      set server <string>
      set loglevel {alert|critical|debug|emergency|error|information|notification|warning}
      set event-log-status {enable|disable}
      set event-log-category {admin|configuration|fw|glb|health-check|llb|slb|system|user}
      set traffic-log-status {enable|disable}
      set traffic-log-category {slb|dns|llb}
      set attack-log-status {enable|disable}
      set attack-log-category {av|ddos|geo|ipreputation|ips|waf|fw}
   next
end
```

`status`	Enable/disable the configuration.
`server`	Specify the IP address of the FortiAnalyzer Log server.
`loglevel`	Select the lowest severity to log from the following options: • emergency — The system has become unstable. • alert — Immediate action is required. • critical — Functionality is affected. • error — An error condition exists and functionality could be affected. • warning — Functionality might be affected. • notification — Information about normal events. • information — General information about system operations. • debug — Detailed information about the system that can be used to troubleshoot unexpected behavior. The exported logs will include the selected severity level and above. For example, if you select **error**, the system collects logs with severity level error, critical, alert, and emergency. If you select **alert**, the system collects logs with severity level alert and emergency.
`event-log-status`	Enable/disable logging for events.
`event-log-category`	If **event-log-status** is enabled, the **event-log-category** becomes configurable. Select one or more of the following event categories to include in the event logs export: • configuration — Configuration changes. • admin — Administrator actions. • system — System operations, warnings, and errors. • user — Authentication results logs. • health-check — Health check results and client certificate validation check

	results. • slb — Notifications, such as connection limit reached. • llb — Notifications, such as bandwidth thresholds reached. • glb — Notifications, such as the status of associated local SLB and virtual servers. • fw — Notifications for the Firewall module, such as SNAT source IP pool is using all of its addresses.
`traffic-log-status`	Enable/disable logging for traffic processed by the load-balancing modules.
`traffic-log-category`	If **traffic-log-status** is enabled, the **traffic-log-category** becomes configurable. Select one or more of the following traffic categories to include in the traffic logs export: • slb — Server Load Balancing traffic logs related to sessions and throughput. • dns — Global Load Balancing traffic logs related to DNS requests. • llb — Link Load Balancing traffic logs related to session and throughput.
`attack-log-status`	Enable/disable logging for traffic processed by the security modules.
`attack-log-category`	If **attack-log-status** is enabled, the **attack-log-category** becomes configurable. Select one or more of the following security categories to include in the security logs export: • ddos — DoS protection logs. • ipreputation — IP Reputation logs. • waf — WAF logs. • geo — Geo IP blocking logs. • av — AV logs. • ips — IPS logs. • fw — Firewall logs.

Example

```
config log setting fortianalyzer
   edit 1
      set status enable
      set server 192.8.8.8
      set loglevel information
      set event-log-status enable
      set event-log-category system slb
      set traffic-log-status enable
      set traffic-log-category slb dns
      set attack-log-status enable
      set attack-log-category waf av
   next
end
```

config router

This chapter is a reference for the following commands:

config router isp

Network systems maintain route tables to determine where to forward TCP/IP packets. Use this command to configure ISP routes. ISP routes can be used for outbound traffic and link load balancing traffic.

Routes for outbound traffic are chosen according to the following priorities:

1. Link local routes—Self-traffic uses link local routes.
2. LLB policy route—Configured policy routes have priority over default routes.
3. System policy route—Configured policy routes have priority over default routes.
4. Static route / ISP route / OSPF route—Priority is based on the distance metric. By default, distance for static routes is 10, for ISP routes is 20, and for OSPF routes is 110. The distance metric is configurable for static routes and OSPF routes, but not ISP routes.
5. Default LLB route—Default routes have lower priority than configured routes.
6. Default static route / OSPF route—Default routes have lower priority than configured routes.

Before you begin:

- You must have read-write permission for system settings.

Note: Adding a new ISP route does not affect existing sessions. Deleting or editing an ISP route causes the related sessions to be re-created.

Syntax

```
config router isp
   edit <No.>
      set destination <datasource>
      set gateway <class_ip>
   next
end
```

destination	Specify an ISP address book configuration object. **Note**: Two ISP routes cannot reference the same ISP address book. The ISP routing feature does not support multipath routing.
gateway	IP address of the gateway router that can route packets to the destination IP address that you have specified.

Example

```
FortiADC-VM # config router isp
```

See also

- get router info routing-table

config router bgp

Use these commands to configure BGP-related options, such as AS ID, router ID, distance of routes, redistribute route , etc., including BGP network, neighbor, and ha-router-id-list configurations.

Before you begin:

- You must know how BGP has been implemented in your network, i.e., the configuration details of the implementation.
- You must have read-write permission for system settings.
- You must have configured all the required access (IPv6) lists and prefix (IPv6) lists.

Syntax

```
config router bgp
   set as <id>
   set router-id <ipv4 address>
   set distance-external <1-255>
   set distance-external6 <1-255>
   set distance-internal <1-255>
   set distance-internal6 <1-255>
   set distance-local <1-255>
   set distance-local6 <1-255>
   set redistribute-ospf {enable|disable}
   set redistribute-connected {enable|disable}
   set redistribute-static {enable|disable}
   set redistribute-connected6 {enable|disable}
   set redistribute-static6 {enable|disable}
```

```
    set always-compare-med {enable|disable}
    set deterministic-med {enable|disable}
    set bestpath-as-path-ignore {enable|disable}
    set bestpath-cmp-routerid {enable|disable}
    set bestpath-med-missing-as-worst {enable|disable}
    config network
       edit <id>
          set type {ipv4|ipv6}
          set prefix <ipv4-netmask>
          set prefix6 <ipv6-netmask>
       next
    end
    config neighbor
       edit <id>
          set remote-as <id>
          set addr-type {ipv4|ipv6}
          set ip <ipv4 address>
          set ip6 <ipv6 address>
          set interface <interface name>
          set port <0-65535>
          set keepalive-timer <0-65535>
          set holdtime-timer <0-65535>
          set default-originate {enable|disable}
          set distribute-list-in <access list name>
          set distribute-list-out <access list name>
          set distribute-list-in6 <ipv6 access list name>
          set distribute-list-out6 <ipv6 access list name>
          set prefix-list-in <prefix list name>
          set prefix-list-out <prefix list name>
          set prefix-list-in6 <ipv6 prefix list name>
          set prefix-list-out6 <ipv6 prefix list name>
          set ebgp-multihop <1-255 >
          set next-hop-self {enable|disable}
          set passive {enable|disable}
          set password <password>
          set shutdown {enable|disable}
          set ttl-security <1-254>
          set update-source-type {interface|address}
          set update-source-interface <interface name>
          set update-source-ip <ipv4 address>
          set update-source-ip6 <ipv6 address>
          set weight <0-65535>
          set bfd {enable|disable}
       next
    end
    config ha-router-id-list
       edit <id>
          set router-id <ipv4 address>
          set node <0-7>
       next
    end
 end
```

`as <id>`	Specify the AS (Autonomous System) number.
`router-id`	Specify a unique value to identify the router, using an IPv4 address.

distance-external	Specify the distance for routes external to the AS.
distance-external6	Specify the distance for IPv6 routes external to the AS.
distance-internal	Specify the distance for routes internal to the AS.
distance-internal6	Specify the distance for IPv6 routes internal to the AS.
distance-local	Specify the distance for routes local to the AS.
distance-local6	Specify the distance for IPv6 routes local to the AS.
redistribute-ospf	Enable/disable the redistribute OSPF route to the BGP server.
redistribute-connected	Enable/disable the redistribute connected route to the BGP server.
redistribute-static	Enable/disable the redistribute static route to the BGP server.
redistribute-connected6	Enable/disable the redistribute connected IPv6 route to the BGP server.
redistribute-static6	Enable/disable the redistribute static IPv6 route to the BGP server.
always-compare-med	Enable/disable always compare MED (Multi-Exit Discriminator) for BGP decision.
deterministic-med	Enable/disable enforce deterministic comparison of MED for BGP decision.
bestpath-as-path-ignore	Enable/disable ignore AS path for BGP decision.
bestpath-cmp-routerid	Enable/disable compare router ID for identical EBGP paths for BGP decision.
bestpath-med-missing-as-worst	Enable/disable treat missing MED as least preferred for BGP decision.
Network	
type	Specify the address type: IPv4 or IPv6.
prefix	Specify the network prefix when (address) type is IPv4, using the IP/mask format.
prefix6	Specify the network prefix when (address) type is IPv6, using the IPv6/mask format.
Neighbor	
remote-as	The AS number of the neighbor.
addr-type	Address type used to configure the neighbor
ip	IP address of the neighbor.
ip6	IPv6 address of the neighbor.
interface	Interface that connected to neighbor
port	Port number that communicate with the neighbor.
keepalive-timer	Frequency to send keep alive requests.
holdtime-timer	Number of seconds to mark peer as dead.
default-originate	Enable/disable originate default route to this neighbor.
distribute-list-in	Filter for IP updates from this neighbor.

distribute-list-out	Filter for IP updates to this neighbor.
distribute-list-in6	Filter for IPv6 updates from this neighbor.
distribute-list-out6	Filter for IPv6 updates to this neighbor.
prefix-list-in	IP Inbound filter for updates from this neighbor.
prefix-list-out	IP Outbound filter for updates to this neighbor.
prefix-list-in6	IPv6 Inbound filter for updates from this neighbor.
prefix-list-out6	IPv6 Outbound filter for updates to this neighbor.
ebgp-multihop	Specify the maximum multi-hops allowed for EBGP neighbors. Only need for EBGP neighbor, cannot set with ttl-security.
next-hop-self	Enable/disable IP next-hop calculation for this neighbor.
passive	Enable/disable sending of open messages to this neighbor.
password	Set Password.
shutdown	Enable/disable shutdown for this neighbor.
update-source-type	Type of source for routing updates.
update-source-interface	Interface Source for routing updates.
update-source-ip	IP address Source for routing updates.
update-source-ip6	IPv6 address Source for routing updates.
weight	Default weight for routes from this neighbor. Range is <0-65535>.
bfd	Enable to activate Bidirectional Forwarding Detection (BFD) on the BGP session. When BFD detects a path failure, a neighbor Down event is notified immediately to the BGP process, triggering a BGP neighbor status change.
HA router ID list	
router-id	Specify the router ID, using IPv4 address.
node <0-7>	Specify Node ID of HA Node.

Examples for IPv4 BGP configuration

Configure BGP router

```
FortiADC-VM (root) # config router bgp
FortiADC-VM (bgp) # set as 101
FortiADC-VM (bgp) # set router-id 10.0.6.217
FortiADC-VM (bgp) # set distance-internal 300
FortiADC-VM (bgp) # set redistribute-static enable
```

Configure BGP network

```
FortiADC-VM (bgp) # config network
```

```
FortiADC-VM (network) # edit 1
FortiADC-VM (1) # set type ipv4
FortiADC-VM (1) # set prefix 172.15.1.0/24
FortiADC-VM (1) # next
FortiADC-VM (network) # edit 2
FortiADC-VM (1) # set type ipv4
FortiADC-VM (1) # set prefix 192.168.11.0/24
FortiADC-VM (1) # next
FortiADC-VM (network) # end
```

Configure BGP neighbor

```
FortiADC-VM (bgp) # config neighbor
FortiADC-VM (neighbor) # edit 1
FortiADC-VM (1) # set remote-as 101
FortiADC-VM (1) # set ip 172.15.11.218
FortiADC-VM (1) # set interface port2
FortiADC-VM (1) # next
FortiADC-VM (neighbor) # end
FortiADC-VM (bgp) # get
as : 101
router-id : 10.0.6.217
distance-external : 20
distance-internal : 250
distance-local : 200
redistribute-ospf : disable
redistribute-connected : disable
redistribute-static : enable
redistribute-connected6 : disable
redistribute-static6 : disable
always-compare-med : disable
deterministic-med : disable
bestpath-as-path-ignore : disable
bestpath-cmp-routerid : disable
bestpath-med-missing-as-worst : disable
== [ 1 ]
== [ 2 ]
== [ 1 ]
FortiADC-VM (bgp) # end
```

Examples for IPv6 BGP configuration

Configure BGP router (IPv6)

```
FortiADC-VM (root) # config router bgp
FortiADC-VM (bgp) # set as 101
FortiADC-VM (bgp) # set router-id 10.0.6.217
FortiADC-VM (bgp) # config network #configure BGP network
FortiADC-VM (network) # edit 1
FortiADC-VM (1) # set type ipv6
FortiADC-VM (1) # set prefix6 2015::/64
FortiADC-VM (1) # next
FortiADC-VM (network) # edit 2
FortiADC-VM (1) # set type ipv4
```

```
FortiADC-VM (1) # set prefix6 2016::/64
FortiADC-VM (1) # next
FortiADC-VM (network) # end
```

Configure BGP network (IPv6)

```
FortiADC-VM (bgp) # config network
FortiADC-VM (network) # edit 1
FortiADC-VM (1) # set type ipv6
FortiADC-VM (1) # set prefix6 2015::/64
FortiADC-VM (1) # next
FortiADC-VM (network) # edit 2
FortiADC-VM (1) # set type ipv4
FortiADC-VM (1) # set prefix6 2016::/64
FortiADC-VM (1) # next
FortiADC-VM (network) # end
```

Configure BGP neighbor (IPv6)

```
FortiADC-VM (bgp) # config neighbor #configure BGP neighbor
FortiADC-VM (neighbor) # edit 1
FortiADC-VM (1) # set remote-as 101
FortiADC-VM (1) # set addr-type ipv6
FortiADC-VM (1) # set ip6 2016::2
FortiADC-VM (1) # set interface port2
FortiADC-VM (1) # next
FortiADC-VM (neighbor) # end
FortiADC-VM (bgp) # end
```

config router bfd

Use this command to configure Bidirectional Forwarding Detection (BFD) to quickly detect Border Gateway Protocol (BGP) session failures by enabling quicker rerouting of traffic in the event of a link or peer failure. After you configure a BFD object, you can enable BFD in the BGP Neighbor configuration.

Before you begin:

- You must know how BGP has been implemented in your network, i.e., the configuration details of the implementation.
- You must have read-write permission for system settings.

Syntax

```
config router bfd
   config interface
      edit <name>
         set interface <interface name>
         set desired-min-tx <integer>
         set detect-mult <integer>
```

```
        set required-min-rx <integer>
      next
   end
end
```

interface	Specify the Interface to assign for BFD. The BFD Interface can refer to the Link Load Balance Ingress Interface specified in the Link Policy. Any Layer 3 interfaces that receive and send external packets can be assigned for BFD except for Loopback interfaces.
desired-min-tx	Specify the desired minimum transmit interval for BFD liveness detection in milliseconds. Default: 750ms, Range: 200ms-30000ms. This refers to the interval that the FortiADC would like to use when transmitting BFD Control packets.
detect-mult	Specify the detection time multiplier. Default: 3 Range: 1-20. The negotiated transmit interval, multiplied by this value, provides the Detection Time for the receiving system.
required-min-rx	Specify the required minimum receive interval for BFD liveness detection in milliseconds. Default: 500ms, Range: 200ms-30000ms. This refers to the minimum interval after which the local routing device must receive a reply from a neighbor with which it has established a BFD session.

Example

```
config router bfd
   config interface
      edit "BFD_port3"
         set interface port3
         set desired-min-tx 500
         set detect-mult 3
         set required-min-rx 1000
      next
   end
end
```

config router md5-ospf

Use this command to configure a table of MD5 keys used in OSPF cryptographic authentication. The table can include up to 256 entries. All OSPF interfaces that want to learn routes from each other must be configured with the same authentication type and password or MD5 key (one match is enough).

OSPF cryptographic authentication involves the use of a shared secret key to authenticate all router traffic on a network. The key is never sent over the network in the clear—a packet is sent and a condensed and encrypted form of the packet is appended to the end of the packet. A non-repeating sequence number is included in the OSPF packet to protect against replay attacks that could try to use already sent packets to disrupt the network. When a packet is accepted as

authentic, the authentication sequence number is set to the packet sequence number. If a replay attack is attempted, the packet sent will be out of sequence and ignored.

Before you begin:

- You must have read-write permission for router settings.

After you have configured an MD5 key configuration object, you can specify it in the OSPF router configuration.

Syntax

```
config router md5-ospf
   edit <name>
      config md5-member
         edit <No.>
            set md5-key <string>
         next
      end
   next
end
```

<No.>	A number 1-255. Each member key ID must be unique to its member list.
md5-key	A string of up to 16 characters to be hashed with the cryptographic MD5 hash function.

Example

```
FortiADC-docs # config router md5-ospf
FortiADC-docs (md5-ospf) # edit md5-key-pool
Add new entry 'md5-key-pool' for node 3752
FortiADC-docs (md5-key-pool) # config md5-member
FortiADC-docs (md5-member) # edit 1
Add new entry '1' for node 3754
FortiADC-docs (1) # set key 0123456789abcdef
FortiADC-docs (1) # end
FortiADC-docs (md5-key-pool) # end
FortiADC-docs #
```

config router ospf

Use this command to configure OSPF. FortiADC supports OSPF version 2. OSPF (Open Shortest Path First) is described in RFC2328.

OSPF is a link-state interior routing protocol. Compared with RIP, OSPF can provide scalable network support and faster convergence times. OSPF is widely used in large networks such as ISP backbone and enterprise networks.

Before you begin:

- You must have read-write permission for router settings.

Syntax

```
config router ospf
   set router-id <integer>
   set default-metric <integer>
   set distance <integer>
   set default-information-originate {always|enable|disable}
   set default-information-metric-type {1|2}
   set default-information-metric <integer>
   set redistribute-connected {enable|disable}
   set redistribute-connected-metric-type {1|2}
   set redistribute-connected-metric <integer>
   set redistribute-static {enable|disable}
   set redistribute-static-metric-type {1|2}
   set redistribute-static-metric <integer>
   config area
      edit <class_ip>
         set authentication {md5|none|text}
         set type {regular|stub}
         set stub-type {summary|no-summary}
      next
   end
   config network
      edit <No.>
         set area <datasource>
         set prefix <ip&netmask>
      next
   end
   config ospf-interface
      edit <name>
         set authentication {md5|none|text}
         set authentication-md5 <datasource>
         set authentication <text>
         set cost <integer>
         set dead-interval <integer>
         set hello-interval <integer>
         set interface <datasource>
         set mtu-ignore {enable|disable}
         set network-type {broadcast | point-to-multipoint | point-to-point}
         set priority <integer>
         set retransmit-interval <integer>
         set transmit-delay <integer>
      next
   end
end
```

`router-id`	32-bit number that identifies the router. The router ID uses dotted decimal notation. sets the router-ID of the OSPF process. The router-ID must be an IP address of the router, and it must be unique within the entire OSPF domain to the OSPF speaker.
`default-metric`	The default is 10.
`distance`	The default is 110.

`default-information-originate`	• enable—Originate an AS-External (type-5) LSA describing a default route into all external routing capable areas of the specified metric and metric type. • always—The default is always advertised, even when there is no default present in the routing table. • disable
`default-information-metric-type`	• 1 • 2
`default-information-metric`	The default is -1.
`redistribute-connected`	Enable/disable to redistribute connected routes into OSPF, with the metric type and metric set if specified. Redistributed routes are distributed into OSPF as Type-5 External LSAs into links to areas.
`redistribute-connected-metric-type`	• 1 • 2
`redistribute-connected-metric`	Specify a metric.
`redistribute-static`	Enable/disable to redistribute static routes into OSPF, with the metric type and metric set if specified. Redistributed routes are distributed into OSPF as Type-5 External LSAs into links to areas.
`redistribute-static-metric-type`	• 1 • 2
`redistribute-static-metric`	Specify a metric.
config area	
<class_id>	32-bit number that identifies the OSPF area. An OSPF area is a smaller part of the larger OSPF AS. Areas are used to limit the link-state updates that are sent out. The flooding used for these updates would overwhelm a large network, so it is divided into these smaller areas for manageability.
`authentication`	Specify an authentication type: • none—Also called null authentication. No authentication is used. In this case the 16-byte Authentication field is not checked, and can be any value. However checksumming is still used to locate errors. • text—A simple password is used. The password is a plain text string of characters. The same password is used for all transactions on a network. The main use of this type of authentication is to prevent routers from accidently joining the network. Simple password authentication is vulnerable to many forms of attack, and is not recommended as a secure form of authentication. • md5—Use OSPF cryptographic authentication. A shared secret key is used to authenticate all router traffic on a network. The key is never sent over the network in the clear—a packet is sent and a condensed and encrypted form of the packet is appended to the end of the packet. A non-repeating sequence number is included in the OSPF packet to protect against replay

	attacks that could try to use already sent packets to disrupt the network. When a packet is accepted as authentic, the authentication sequence number is set to the packet sequence number. If a replay attack is attempted, the packet sent will be out of sequence and ignored.
`type`	Area type setting: • regular—A normal area • stub—An area where no router originates routes external to OSPF and hence all external routes are via the ABRs.
`stub-type`	If using stub area, select a stub summary setting: • summary—allow an ABR to send summary LSAs into a stub area • no-summary—Prevent an ABR sending summary LSAs into a stub area
config network	
`area`	Specify an area configuration name.
`prefix`	Address/mask notation to specify the subnet.
config ospf-interface	
`authentication`	Specify an authentication type. All OSPF interfaces that want to learn routes from each other must be configured with the same authentication type and password or MD5 key (one match is enough). Options are: • none—Use the authentication type referenced by the area included in the network configuration. • md5—Override the authentication type referenced by the area included in the network configuration with the MD5 configuration specified here. • text—Override the authentication type referenced by the area included in the network configuration with the text configuration specified here.
`authentication-md5`	Specify an MD5 configuration name.
`authentication-text`	Specify a password string. Passwords are limited to 8 characters.
`cost`	Set link cost for the specified interface. The cost value is set to router-LSA's metric field and used for SPF calculation. The default is 0.
`dead-interval`	Number of seconds for RouterDeadInterval timer value used for Wait Timer and Inactivity Timer. This value must be the same for all routers attached to a common network. The default is 40 seconds.
`hello-interval`	Number of seconds between hello packets sent on the configured interface. This value must be the same for all routers attached to a common network. The default is 10 seconds.
`interface`	Specify the interface to enable OSPF for it.
`mtu-ignore`	Enable/disable to ignore the interface MTU. Disabled by default.
`network-type`	• broadcast • point-to-point • point-to-multipoint

priority	The router with the highest priority will be more eligible to become Designated Router. Setting the value to 0 makes the router ineligible to become Designated Router. The default is 1.
retransmit-interval	Interval for retransmitting Database Description and Link State Request packets. The default is 5 seconds.
transmit-delay	Increment LSA age by this value when transmitting. The default is 1 second.

Example

FortiADC1

```
FortiADC-VM # config router ospf
FortiADC-VM (ospf) # set router-id 1.1.1.2
FortiADC-VM (ospf) # set default-metric 5
FortiADC-VM (ospf) # config network
FortiADC-VM (network) # edit 1
Add new entry '1' for node 2090
FortiADC-VM (1) # set prefix 1.1.1.1/32
FortiADC-VM (1) # set area 0.0.0.0
FortiADC-VM (1) # end

FortiADC-VM (ospf) # get
router-id : 1.1.1.2
default-information-originate: disable
default-information-metric: -1
default-information-metric-type: 2
default-metric : 5
distance : 110
redistribute-connected: disable
redistribute-connected-metric: -1
redistribute-connected-metric-type: 2
redistribute-static : disable
redistribute-static-metric: -1
redistribute-static-metric-type: 2
== [ 1 ]

FortiADC-VM (ospf) # show
config router ospf
set router-id 1.1.1.2
set default-metric 5
config network
edit 1
set prefix 1.1.1.1/32
next
end
config ospf-interface
end
end
```

FortiADC2

```
FortiADC-VM # config router ospf
FortiADC-VM (ospf) # set router-id 1.1.1.3
FortiADC-VM (ospf) # config network
FortiADC-VM (network) # edit 1
Add new entry '1' for node 2090
FortiADC-VM (1) # set prefix 1.1.1.1/32
FortiADC-VM (1) # set area 0.0.0.0
FortiADC-VM (1) # end

FortiADC-VM (ospf) # get
router-id : 1.1.1.2
default-information-originate: disable
default-information-metric: -1
default-information-metric-type: 2
default-metric : 10
distance : 110
redistribute-connected: disable
redistribute-connected-metric: -1
redistribute-connected-metric-type: 2
redistribute-static : disable
redistribute-static-metric: -1
redistribute-static-metric-type: 2
== [ 1 ]

FortiADC-VM (ospf) # show
config router ospf
set router-id 1.1.1.2
config network
edit 1
set prefix 1.1.1.1/32
next
end
config ospf-interface
end
end
```

See Also

- config router md5-ospf
- get router info ospf

config router policy

Network systems maintain route tables to determine where to forward TCP/IP packets. Use this command to configure system policy routes. Policy routes are based on IP layer values, specifically the source and/or destination fields.

Routes for outbound traffic are chosen according to the following priorities:

1. Link local routes—Self-traffic uses link local routes.
2. LLB policy route—Configured policy routes have priority over default routes.

3. System policy route—Configured policy routes have priority over default routes.
4. Static route / ISP route / OSPF route—Priority is based on the distance metric. By default, distance for static routes is 10, for ISP routes is 20, and for OSPF routes is 110. The distance metric is configurable for static routes and OSPF routes, but not ISP routes.
5. Default LLB route—Default routes have lower priority than configured routes.
6. Default static route / OSPF route—Default routes have lower priority than configured routes.

The system evaluates policy routes, then static routes. The packets are routed to the first route that matches. The policy route table, therefore, need not include a “default route” for packets that do not match your policy because those packets can be forwarded to the default route set in the static route table.

Most policy route settings are optional, so a matching route might not provide enough information to forward the packet. In that case, the FortiADC appliance may refer to the routing table in an attempt to match the information in the packet header with a route in the routing table. For example, if the destination address is the only match criteria in the policy route, the FortiADC appliance looks up the IP address of the next-hop router in its routing table. This situation could occur when interfaces are dynamic (such as DHCP or PPPoE) and you do not want or are unable to specify a static IP address of the next-hop router.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config router policy
   edit <No.>
      set destination <ip&netmask>
      set gateway <class_ip>
      set source <ip&netmask>
   next
end
```

`destination`	Address/mask notation to match the destination IP in the packet header. To match any value, leave it blank or enter 0.0.0.0/32.
`gateway`	IP address of the gateway router that can route packets to the destination IP address that you have specified.
`source`	Address/mask notation to match the source IP in the packet header. To match any value, either leave it blank or enter 0.0.0.0/32.

config router setting

Use this command to change basic routing settings. However, the default settings are recommended for most deployments.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config router setting
   set rt-cache-strict {enable | disable}
   set rt-cache-reverse {enable | disable}
   set rt-best-match {enable | disable}
   set ip-forward {enable | disable}
   set ip6-forward {enable | disable}
   set icmp-redirect-send {enable | disable}
   set ip-forward-use-pmtu {enable | disable}
   set layer4-vs-ignore-df {enable | disable}
   config rt-cache-reverse-exception
      edit <No.>
         set addr-type {ipv4 | ipv6}
         set ip-netmask <ip&netmask>
         set ipv6-netmask <ipv6&netmask>
      next
   end
end
```

`rt-cache-strict`	Enable it when you want to send reply packets only via the same interface that received the request packets. When enabled, source interface becomes part of the matching tuple FortiADC uses to identify sessions, so reply traffic is forwarded from the same interface that received the traffic. Normally each session is identified by a 5-tuple: source IP, destination IP, protocol, source port, and destination port. Disabled by default.
`rt-cache-reverse`	When enabled, forwards reply packets to the ISP link that forwarded the corresponding request packet. Enabled by default. When not enabled, forwards all packets based on the results of routing look-up. The rt-cache-reverse function is useful when your site gets traffic routed to it from multiple ISP links.
`rt-best-match`	Ignore the route type. The route with the longest prefix will be selected. Must restart for change to take effect. Disabled by default.
`ip-forward`	Enabled by default. Do not disable under normal circumstances. If disabled, functions related to routing, like link load balancing, static routing, policy routing, and OSPF routing cannot function.
`ip6-forward`	Enabled by default. Do not disable under normal circumstances. If disabled, functions related to routing, like link load balancing, static routing, policy routing, and OSPF routing cannot function.
`icmp-redirect-send`	When enabled, the icmp redirect package will be sent to the sender if the packet is being routed suboptimally so that the subsequent packets forward through a different gateway. Disabled by default.

ip-forward-use-pmtu	Forward package use path mtu. Path mtu is the mtu between two hosts. Can be used to avoid IP fragmentation. Disabled by default.
layer4-vs-ignore-df	Does not fragment layer4 vs packages. Disabled by default.
config rt-cache-reverse-exception	
addr-type	When rt-cache-reverse is enabled, you can specify the source IP address type that should be handled differently. Select one of the following: • ipv4 • ipv6 **Note**: If **addr-type** is not set, IPv4 will be set as the default source IP address type.
ip-netmask	The **ip-netmask** option is available if **addr-type** is **ipv4** or **addr-type** is not set. Specify a subnet IPv4 address and netmask for each reverse route cache exception. For example, if you configure an exception for 192.168.1.0/24, FortiADC will not maintain a pointer to the ISP for traffic from source 192.168.1.18. Reply packets will be forwarded based on the results of routing look-up.
ipv6-netmask	The **ipv6-netmask** option is available if **addr-type** is **ipv6**. Specify a subnet IPv6 address and netmask for each reverse route cache exception. For example, if you configure an exception for 200a::/64, FortiADC will not maintain a pointer to the ISP for traffic from source 200a::1:1aaa. Reply packets will be forwarded based on the results of routing look-up.

Example

```
FortiADC-VM # config router setting
FortiADC-VM (setting) # get
rt-cache-strict : disable
rt-cache-reverse : enable
ip-forward : enable
ip6-forward : enable

FortiADC-VM (setting) # config rt-cache-reverse-exception
FortiADC-VM (rt-cache-reverse) # edit 1
Add new entry '1' for node 3740
FortiADC-VM (1) # set addr-type ipv6
FortiADC-VM (1) # set ipv6-netmask 200a::/64
FortiADC-VM (1) # end
FortiADC-VM (setting) # end
```

config router static

Network systems maintain route tables to determine where to forward TCP/IP packets. Use this command to configure static routes. Static routes are based on destination IP addresses.

Routes for outbound traffic are chosen according to the following priorities:

1. Link local routes—Self-traffic uses link local routes.
2. LLB Link Policy route—Configured policy routes have priority over default routes.
3. Policy route—Configured policy routes have priority over default routes.
4. Static route / ISP route / OSPF route—Priority is based on the distance metric. By default, distance for static routes is 10, for ISP routes is 20, and for OSPF routes is 110. The distance metric is configurable for static routes and OSPF routes, but not ISP routes.
5. Default LLB Link Policy route—Default routes have lower priority than configured routes.
6. Default static route / OSPF route—Default routes have lower priority than configured routes.

The system evaluates policy routes, then static routes. The packets are routed to the first route that matches. The static route table, therefore, is the one that must include a “default route” to be used when no more specific route has been determined.

Static routes specify the IP address of a next-hop router that is reachable from that network interface. Routers are aware of which IP addresses are reachable through various network pathways, and can forward those packets along pathways capable of reaching the packets’ ultimate destinations. The FortiADC system itself does not need to know the full route, as long as the routers can pass along the packet.

You must configure at least one static route that points to a router, often a router that is the gateway to the Internet. You might need to configure multiple static routes if you have multiple gateway routers, redundant ISP links, or other special routing cases.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config router static
   edit <No.>
      set destination <ip&netmask>
      set distance <integer>
      set gateway <class_ip>
   next
end
```

`destination`	Address/mask notation to match the destination IP in the packet header. Specify 0.0.0.0/0 or ::/0 to set a default route for all packets. It is a best practice to include a default route. If there is no other, more specific static route defined for a packet’s destination IP address, a default route will match the packet, and pass it to a gateway router so that any packet can reach its destination. If you do not define a default route, and if there is a gap in your routes where no route matches a packet’s destination IP address, packets passing through the FortiADC towards those IP addresses will, in effect, be null routed. While this can help to ensure that unintentional traffic cannot leave your FortiADC and therefore can be a type of security measure, the result is that you must modify your routes every time that a new valid destination is added to your network. Otherwise, it will be unreachable. A default route ensures that this kind of locally-caused “destination unreachable” problem does not occur.

distance	The default administrative distance is 10, which makes it preferred to OSPF routes that have a default of 110. We recommend you do not change these settings unless your deployment has exceptional requirements.
gateway	Specify the IP address of the gateway router that can route packets to the destination IP address that you have specified.

Example

```
FortiADC-VM # config router static

FortiADC-VM (static) # edit 1
FortiADC-VM (1) # set gateway 192.168.1.1
FortiADC-VM (1) # end

FortiADC-VM # get router static 1
destination : 0.0.0.0/0
gateway : 192.168.1.1
distance : 10
```

config security

The `config security` commands configure web application firewall (WAF) settings.

This chapter is a reference for the following commands:

config security antivirus profile

Use this command to configure an anti-virus profile.

In many cases, you can use a predefined AV profile, and you are not required to create a new AV profile of your own.

Before you begin, make sure that you have read-write permission to configure the system's security settings.

After you have created an anti-virus profile, you can include it in HTTP or HTTPS virtual service profiles.

Syntax

```
config security antivirus profile
   edit <name>
      set comments <string>
      set uncomp-size-limit <integer>
      set uncomp-nest-limit <integer>
      set scan-bzip2 {enable | disable}
      set streaming-content-bypass {enable | disable}
      set oversize-limit <integer>
      set oversize {bypass | log | block}
      set options {avmonitor | quarantine}
      set emulator {enable | disable}
      set fsa-analytics {disable | suspicious | all}
      set analytics-max-upload <integer>
      set analytics-db {disable | enable}
      set av-virus-log {disable | enable}
   next
end
```

`uncomp-size-limit`	The maximum size in MB of the memory buffer used to temporarily decompress files. (Range: 1 - 2000 MB, default: 2 MB).

uncomp-nest-limit	The maximum number of levels of nesting (compression) allowed to decompress.
scan-bzip2	Enable or disable bzip2 scanning algorithm.
streaming-content-bypass	Enable or disable bypass streaming content (rather than buffering it).
oversize-limit	The maximum in-memory file size in KB to be scanned. (Range: 1 - 12000000 KB, default: 1024 KB). **Note**: For AV files larger than 1000 KB, the device memory must be larger than 32 GB to support the scan.
options	Select an option for the system to handle infected files.
emulator	Enable or disable Win32 Emulator.
fsa-analytics	Select an option to submit files to FortiSandbox.
analytics-max-upload	The maximum file size in KB allowed to upload to FortiSandbox.
analytics-db	Enable or disable FortiSandbox signature database.
av-virus-log	The maximum file size in KB allowed to upload to FortiSandbox.

Example

```
FortiADC-docs # config security antivirus profile
FortiADC-docs (profile) # edit av_profile_01
FortiADC-docs (av_profile_01) # set comments test_for_doc
FortiADC-docs (av_profile_01) # set uncomp-size-limit 10
FortiADC-docs (av_profile_01) # set uncomp-nest-limit 5
FortiADC-docs (av_profile_01) # set scan-bzip2 enable
FortiADC-docs (av_profile_01) # set streaming-content-bypass enable
FortiADC-docs (av_profile_01) # set oversize-limit 1024
FortiADC-docs (av_profile_01) # set oversize log
FortiADC-docs (av_profile_01) # set options quarantine
FortiADC-docs (av_profile_01) # set emulator enable
FortiADC-docs (av_profile_01) # set fsa-analytics suspicious
FortiADC-docs (av_profile_01) # set analytics-max-upload 1024
FortiADC-docs (av_profile_01) # set analytics-db enable
FortiADC-docs (av_profile_01) # set av-virus-log enable
FortiADC-docs (av_profile_01) # next
FortiADC-docs (profile) # end
```

Reference to an AV profile

Use the following commands to reference an AV profile to a HTTP/HTTPs or SMTP virtual service.

Syntax

```
config load-balance virtual-server
   edit <name>
      set av-profile <profile-name>
   end
```

```
end
```

Example

```
FortiADC-docs # config load-balance virtual-server
FortiADC-docs (virtual-server) # edit vs1
FortiADC-docs (vs1) # set load-balance-profile LB_PROF_HTTP
FortiADC-docs (vs1) # set av-profile av_profile_01
FortiADC-docs (vs1) # end
```

config security antivirus quarantine

Use this command to configure anti-virus quarantine.

Syntax

```
config security antivirus quarantine
   set destination {NULL | disk}
   set agelimit <integer>
   set maxfilesize <integer>
   set quarantine-quota <integer>
   set drop-infected {http | https | smtp}
   set lowspace {drop-new | ovrw-old|
end
```

destination	The destination for quarantined files, which could be either of the following: • NULL—Disable quarantine. • Disk—Send quarantined files to the hard disk.
age-limit	The number of hours that quarantined files are kept on the hard disk. The default is 1 hour. Valid values range form 0 to 336 hours. Note: If the age limit is set to 0 (zero), it means that there is no age limit and quarantined files will remain on the hard disk forever.
maxfilesize	The maximum size (in KB) of a single file that can be quarantined. The default is 1024 (KB). Valid values range from 1 to 2048 KB. Note: Files larger than the set Max File Size will not be quarantined. In reality, this value is subject the available quarantine quota that remains on the hard disk. For example, when there is less than 1024 KB of quarantine quota (disk space reserved for quarantined files) remaining, a file of 1024 KB in size still will not be quarantined even though you've set Max File Size to 1024.
quarantine-quota	The amount of disk space reserved for quarantining files. The default is 512 MB. Valid values range from 0 to 1024 MB. If the value is set to 0, no files are quarantined

drop-infected	Select either or both of the following: • HTTP • HTTPS • SMTP Note: By default neither option is selected, which means that both types of files are quarantined. If selected, files involving the specified protocol or protocols will be dropped (not quarantined).
lowspace	Specify the way in which new files are handled when the system disk space is running low, which could be either of the following: • Override Old—Override old quarantine files with new ones. • Drop New—Drop new quarantine files to retain old ones.

Example

```
FortiADC-docs # config security antivirus quarantine
FortiADC-docs (quarantine) # set destination disk
FortiADC-docs (quarantine) # set agelimit 1
FortiADC-docs (quarantine) # set maxfilesize 1
FortiADC-docs (quarantine) # set quarantine-quota 256
FortiADC-docs (quarantine) # set drop-infected http
FortiADC-docs (quarantine) # set lowspace drop-new
FortiADC-docs (quarantine) # end
```

config security antivirus settings

FortiADC's antivirus (AV) service relies on the system's AV engine and signature databases. The AV engine is upgraded whenever new functions are added. The Updated daemon is responsible for updating the AV engine and the signature databases.

The system offers three types of AV signature databases, namely, Normal, Extended, and Extreme. They represent different levels of AV services. In order for FortiADC to provide you with the level of AV service that you desire, you must choose the right signature database.

Syntax

```
config security antivirus settings
   set default-db {extended | extreme | normal}
   set ramdisk-size <integer>
end
```

default-db	Set the default signature database. • Normal: The regular virus database, which includes “In the Wild” viruses and most

	commonly seen viruses on the network. It provides regular protection. • Extended: The extended virus database, which includes both “In the Wild” viruses and a large collection of zoo viruses that are no longer seen in recent virus studies. It provides enhanced security protection. • Extreme: The extreme virus database, which includes both “In the Wild” viruses and all known zoo viruses that are no longer seen in recent virus studies. It provides the highest level of security protection.
`ramdisk-size`	Specify the ramdisk size in GB. (Range: 0 - 12 GB, default: 0 GB). **Note**: When the changes to the ramdisk configuration is committed, you will be prompted to restart the device for the configuration to take effect.

config security dos dos-protection-profile

A DoS Protection profile references the DoS policies that are to be enforced.

Syntax

```
configure security dos dos-protection-profile
   edit <name>
      set http-access-limit <datasource>
      set http-connection-flood-protection <datasource>
      set http-request-flood-protection <datasource>
      set tcp-access-flood-protection <datasource>
      set tcp-slowdata-attack-protection <datasource>
      set dns-query-flood-protection <datasource>
      set dns-reverse-flood-protection <datasource>
      set http-send-timeout <integer>
   next
end
```

`http-access-limit`	Specify an HTTP Access Limit policy. Limit the request number per second from an IP.
`http-connection-flood-protection`	Specify an HTTP Connection Flood policy. Limit the number of connections from a client, which is marked by a cookie.
`http-request-flood-protection`	Specify an HTTP Request Flood policy. Limit the request number per second from a client, which is marked by a cookie.
`tcp-access-flood-protection`	Specify TCP Connection Access Flood Protection policy. A TCP connection flood attempts to prevent legitimate requests from being established by flooding the server with requests for new connections. By setting a threshold limit for TCP requests, FortiADC can detect and take action to protect against a TCP connection flood.
`tcp-slowdata-attack-protection`	Specify a TCP Slow Data Flood Protection policy.

	After the TCP connection is established (the three-way handshake is completed), if FortiADC sends data to the client but the client returns a zero window (a zero window appears when, for example, the client does not take the data out of the TCP receive queue of the client OS when the data sent by the FortiADC fills up the queue), FortiADC will stop sending data. In this case, FortiADC can actively abort TCP connections and release related resources to avoid occupying its resources for a long time.
`dns-query-flood-protection`	Specify a DNS Query Flood Protection policy. The DNS Query Flood Protection policy can limit the number of DNS request per second to mitigate against DNS query flood attacks that aim to overwhelm DNS servers with high volumes of illegitimate DNS queries.
`dns-reverse-flood-protection`	Specify a DNS Reverse Flood Protection policy. The DNS Reverse Flood Protection policy can limit the number of ANY type DNS requests per second to mitigate against DNS reverse flood attacks that aim to overwhelm network resources with high volumes of DNS responses.
`http-send-timeout`	After receiving an HTTP request, FortiADC may forward a response which comes from the backend server. If FortiADC cannot send out all the response messages, it will save the rest of the data in a buffer, and will try to send out again when possible. When there occurs a timeout, if the buffer still has data to be sent, FortiADC will abort this TCP connection.

Example

```
configure security dos dos-protection-profile
   edit dos-profile
      set http-access-limit access-limit
      set http-connection-flood-protection conn-limit
      set http-request-flood-protection req-limit
      set http-send-timeout 3
   next
   end
   configure security dos dos-protection-profile
   edit dos-profile
      set http-access-limit access-limit
      set http-connection-flood-protection conn-limit
      set http-request-flood-protection req-limit
   next
end
```

config security dos http-access-limit

Syntax

```
configure security dos http-access-limit
edit <name>
set status [enable | disable]
set access-limit-per-ip <integer>
```

```
set action [ pass | deny | block-period]
set block-period <integer>
set log [enable | disable]
set severity [ high | medium | low | info]
next
end
```

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
access-limit-per-ip	The access limitation per IP	integer	0-65535	0	No
action	Action taken when the limit is reached.	choice	Pass deny block-period	deny	No
block-period	Number of seconds during which FortiADC blocks the connection action,	integer	1-3600	60	No
severity	Severity of the Log	choice	info low medium high	high	No
log	Records log message	choice	enable disable	disable	No

CLI Parameter	Visible condition	Special value	Effective condition
access-limit-per-ip	always visible	0, means no limit	Attach this config to a DoS protection profile, and attach the DoS profile to a virtual server
action	always visible	N/A	
block-period	action == block-period	N/A	
severity	log == enable	N/A	
log	always visible	N/A	

Function description

CLI Parameter	Description
access-limit-per-ip	If FortiADC receives some HTTP request which has the same source IP in a second, it will check the number to see if it reaches the limit or not. If it has, then FortiADC takes action. The “one second” times when the first request arrived; the count block will be released after one second.
action	DoS protect action.

CLI Parameter	Description
block-period	Block the TCP connection for a period (seconds). During this period if the TCP connection's source IP is blocked, this connection will be aborted. If FortiADC reboots, this block action will be invalid.
severity	Log severity level
log	Enable or disable log

Example

```
configure security dos http-access-limit
edit access-limit
set access-limit-per-ip 10
set action block-period
set block-period 30
set log enable
set severity info
next
end
configure security dos http-access-limit
edit access-limit
set access-limit-per-ip 10
set action Pass
set log enable
set severity info
next
end
configure security dos http-access-limit
edit access-limit
set access-limit-per-ip 10
next
end
```

config security dos dns-query-flood-protection

Use this command to configure a DNS Query Flood Protection policy to limit the number of DNS request per second which can mitigate against DNS query flood attacks.

Syntax

```
config security dos dns-query-flood-protection
   edit <name>
      set status {enable | disable}
      set dns-query-rate-limit <integer>
      set action {pass | deny}
      set log {enable | disable}
      set severity {high | medium | low}
   next
end
```

`status`	Enable/disable the status of this DNS Query Flood Protection policy.
`dns-query-rate-limit`	Specify the allowable number of DNS requests per second. The range is 0 to 1048567. The default is 0, which means that no limit is placed on the number of DNS queries that can be made per second. **Note**: Multiple "rate limit" type of operations may be executed through various configurations, however, they cannot be executed all at once. Priority is given to certain rate limit operations. The following lists the execution sequence. **1.** Transaction Rate Limit (from the virtual server configuration). **2.** DNS Query Rate Limit (from DNS Query Flood Protection policy). **3.** ANY Query Rate Limit (from DNS Reverse Flood Protection policy).
`action`	Select the corresponding action to take when the DNS Query Rate Limit is exceeded: • pass — Allow the traffic. • deny — Drop the traffic, send a 400 Bad request to the client. **deny** is the default option.
`log`	Enable/disable logging for the Action. This is disabled by default.
`severity`	Select the event severity to log when the DNS Query Flood Protection policy is triggered: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is **high**.

Example

```
config security dos dns-query-flood-protection
   edit "DNS_Query_Flood_1"
      set status enable
      set dns-query-rate-limit 2
      set action deny
      set log disable
      set severity high
   next
end
```

config security dos dns-reverse-flood-protection

Use this command to configure a DNS Reverse Flood Protection policy to limit the number of ANY type DNS requests per second which can mitigate against DNS reverse flood attacks.

Syntax

```
config security dos dns-reverse-flood-protection
```

```
   edit <name>
      set status {enable | disable}
      set dns-query-any-rate-limit <integer>
      set action {pass | deny}
      set log {enable | disable}
      set severity {high | medium | low}
   next
end
```

status	Enable/disable the status of this DNS Reverse Flood Protection policy.
dns-query-any-rate-limit	Specify the allowable number of DNS requests per second, with query type ANY. The range is 0 to 1048567. The default is 0, which means that no limit is placed on the number of DNS queries that can be made per second. **Note**: Multiple "rate limit" type of operations may be executed through various configurations, however, they cannot be executed all at once. Priority is given to certain rate limit operations. The following lists the execution sequence. **1.** Transaction Rate Limit (from the virtual server configuration). **2.** DNS Query Rate Limit (from DNS Query Flood Protection policy). **3.** ANY Query Rate Limit (from DNS Reverse Flood Protection policy).
action	Select the corresponding action to take when the ANY Query Rate Limit is exceeded: • pass — Allow the traffic. • deny — Drop the traffic, send a 400 Bad request to the client. **deny** is the default option.
log	Enable/disable logging for the Action. This is disabled by default.
severity	Select the event severity to log when the DNS Reverse Flood Protection policy is triggered: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is **high**.

Example

```
config security dos dns-query-flood-protection
   edit "DNS_Reverse_Flood_1"
      set status enable
      set dns-query-rate-limit 1
      set action pass
      set log enable
      set severity high
   next
end
```

config security dos http-connection-flood-protection

HTTP Connection Flood policy can limit connections from a client that are marked by a cookie.

Syntax

```
configure security dos http-request-flood-protection
edit <name>
set status [enable | disable]
set request-limit-per-session <integer>
set action [ Pass | deny | Pass&deny | block-period]
set block-period <integer>
set severity [ high | medium | low | info]
next
end
```

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
request-limit-per-session	the request limitation of per HTTP session	integer	0-65535	0	No
action	action when reach the limit	choice	Pass deny block-period	deny	No
block-period	number of seconds that block the connection action	integer	1-3600	60	No
severity	severity of the Log	choice	info low medium high	high	No
log	record log message	choice	enable disable	disable	No

Function description

CLI Parameter	Description
request-limit-per-session	If ADC receives a HTTP request, first match the URL and host. If matched, it will insert a cookie to the header when the response arrives. If a new request arrives ADC and carry a cookie which is inserted by ADC, ADC will find a block to record the number that all the TCP connection which use the same cookie, if reach the limit, then take action.
action	DoS protect action
block-period	Block the HTTP request for a period(second), timing when tack the action. During this period if the TCP connection whose request have the blocked cookie will be aborted. If ADC reboot, this block action is still valid.

CLI Parameter	Description
severity	Log severity level
log	Enable or disable log

Example

configure security dos http-request-flood-protection

edit req-limit

set request-limit-per-session 2

set action block-period

set block-period 20

set log enable

set severity medium

next

end

configure security dos http-request-flood-protection

edit req-limit

set request-limit-per-session 2

set action Pass

next

end

configure security dos http-request-flood-protection

edit req-limit

set request-limit-per-session 2

next

end

config security dos http-request-flood-protection

HTTP Request Flood policy can limit the speed of HTTP requests from a client that is marked by a cookie.

Syntax

configure security dos http-request-flood-protection

edit <name>

set status [enable | disable]

set request-limit-per-session <integer>

set action [Pass | deny | Pass&deny | block-period]

set block-period <integer>

set severity [high | medium | low | info]

next

end

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
request-limit-per-session	The request limitation of per HTTP session	integer	0-65535	0	No
action	Action when limit is reached	choice	Pass deny block-period	deny	No
block-period	Number of seconds during which to block the connection action	integer	1-3600	60	No
severity	Severity of the Log	choice	info low medium high	high	No
log	Record log message	choice	enable disable	disable	No

Function description

CLI Parameter	Description
request-limit-per-session	If FortiADC receives a HTTP request, it will first match the URL and host. If these match, it will insert a cookie to the header when the response arrives. If a new request arrives and carries a cookie which is inserted by FortiADC, FortiADC will find a block to record the number of all the TCP connections which use the same cookie; if it reaches the limit, FortiADC will take action.
action	DoS protect action
block-period	Block the HTTP request for a period (second). During this period if the TCP connection's request has the blocked cookie, it will be aborted. If FortiADC reboots, this block action is still valid.
severity	Log severity level
log	Enable or disable log

Example

```
configure security dos http-request-flood-protection
edit req-limit
set request-limit-per-session 2
set action block-period
set block-period 20
set log enable
set severity medium
next
end
configure security dos http-request-flood-protection
edit req-limit
set request-limit-per-session 2
set action Pass
next
end
configure security dos http-request-flood-protection
edit req-limit
set request-limit-per-session 2
next
end
```

config security dos ip-fragmentation-protection

IP Packet fragmentation assures that IP data grams can flow through any other type of network. It allows data grams created as a single packet to be split into many smaller packets for transmission and reassembled at a receiving host. A DDoS attack can deny services to the network by creating a fragmented data gram of a large enough size to overrun the buffers in your router.

The attack purpose is to consume the system memory and network bandwidth in the shortest possible time. We can limit the maximum usage of memory in each socket, the maximum distance counters between fragmentation packages from the same source IP, and the receiving timeout for an entire package.

Syntax

```
config security dos ip-fragmentation-protection
set max-memory-size <integer>
set min-memory-size <integer>
set time <integer>
end
```

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
max-memory-size	ip fragmentation maximum memory size limit(KB)	integer	0-4096	4096	No

CLI Parameter	Help message	Type	Scope	Default	Must
min-memory-size	ip fragmentation minimum memory size limit(KB)	integer	0-4096	3072	No
time	fragment package alive time	char	0-256	30	No

Function description

CLI Parameter	Description
max-memory-size	Maximum memory size of the IP fragmentation packet for the vdom. If it reaches this limit, FortiADC will stop doing IP fragmentation reassemble.
min-memory-size	When total IP fragmentation memory size drops to min-memory-size, it will start to do fragmentation reassemble again.
time	Max life time for each fragmentation queue. All the fragmentation packets in the queue will be dropped if the queue exceed this timeout.

Example

```
configure security dos ip-fragmentation-protection
set max-memory-size 4096
set max-memory-size 3072
set time 30
end
```

config security dos tcp-access-flood-protection

A Connection Flood refers to an overwhelming amount of connections attempting to flood a victimized FortiADC at the same time. This can be from a single IP address or from a botnet.

The purpose of the attack is to consume the amount of connections in a shorter time. To prevent this, we can limit the numbers of connections from the same IP address.

Example

```
config security dos tcp-access-flood-protection
edit <name>
set max-access-count <integer>
set action [ pass | deny | block-period]
set block-period <integer>
set severity [ high | medium | low ]
set log [enable | disable]
next
end
```

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
max-access-count	Limit the number of TCP connection per source IP address	integer	0-65535	0	No
action	action when reach the limit	choice	Pass deny block-period	deny	No
block-period	number of seconds that block the connection action	integer	1-3600	60	No
severity	severity of the Log	choice	info low medium high	high	No
log	record log message	choice	enable disable	disable	No

Function description

CLI Parameter	Description
max-access-count	Set the TCP connection limit for each source IP address
action	DoS protect action when TCP connection number exceed the limit Pass – allow the new connection from this IP address Deny – deny the new connection from this IP address Block-period -- deny the new connection from this IP address for a period of time
block-period	Block the connection creating for a period, timing when tack the action. During this period, the new connection will abort.
severity	Log severity level
log	Enable or disable log

Example

```
configure security dos tcp-access-flood-protection
edit tcp-conn
set max-access-count 256
set action block-period
set block-period 20
set log enable
set severity medium
next
end
```

config security dos tcp-slowdata-attack-protection

A Slow Data attack sends legitimate application layer requests but reads responses very slowly. With that, it may attempt to exhaust the target' s connection pool. Slow reading advertises a very small number for the TCP Receive Window size and at the same time by emptying the client's TCP receive buffers slowly. That ensures a very low data flow rate.

The purpose of the attack is to consume the system resources (memory, CPU time) slowly. We can disable the connection when it fails to send probe packages within the zero-window timer.

Syntax

```
config security dos tcp-slowdata-attack-protection
edit <name>
set probe-interval-time <integer>
set probe-count <integer>
set action [ pass | deny | block-period]
set block-period <integer>
set severity [ high | medium | low ]
set log [enable | disable]
next
end
```

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
probe-interval-time	Probe internal timer for zero-window probe	char	0-256	30	No
probe-count	Max count for zero-window probe	char	0-256	5	No
action	Action taken when probe count exceeds limit and still no >0 windows packet received	choice	Pass deny block-period	deny	No
block-period	Number of seconds to block the connection action if you choose block-period as action	integer	1-3600	60	No
severity	Severity of the Log	choice	info low medium high	high	No
log	Record log message	choice	enable disable	disable	No

Function description

CLI Parameter	Description
probe-interval-time	Set probe interval time for the TCP zero window timer. After receiving a zero window packet, FortiADC will probe the peer side periodically until it receives a >0 window, or probe count exceeds the max probe-count.
probe-count	Max consecutive zero window probe count
action	Action taken after exceeding max probe count Pass –if the probe count exceeds probe-count, FortiADC stops the probe and passes all the packets in both direction. Deny – deny the connection with RST Block-period – deny the connection, and block any new connection from the peer side for a period of time
block-period	Block the new connection from peer side for a period. During this period, the new connection will abort.
severity	Log severity level
log	Enable or disable log

Example

```
configure security dos tcp-slowdata-attack-protection
edit zero-window-limit
set probe-interval-time 30
set probe-count 5
set action block-period
set block-period 20
set log enable
set severity medium
next
end
```

config security dos tcp-synflood-protection

TCP SYN flood protection is a global setting to protect all virtual server traffic from SYN flood attack. After the SYN Cookie option is enabled, each virtual server will monitor SYN rate. If the average SYN rate in 10 seconds exceeds Maximum Half-Open Sockets, it will perform SYN Cookie on all subsequent new connections (SYN packets) of this virtual server until the rate drops to below Maximum Half-Open Sockets.

Syntax

```
config security dos tcp-synflood-protection
Set syncookie enable | disable
set max-half-open <integer>
set max-stale-timeout <integer>
end
```

CLI Parameter	Description
syncookie	Enable/disable syn flood protection
Max-half-open	If average halfopen connection rate in 10 seconds for each VS exceeds this setting, it will enable syncookie for all new following TCP connections for this VS. If the average rate drops to bellow it, it will disable syncookie then for this VS.

Example

```
config security dos tcp-synflood
set syncookie enable
Set max-half-open 1024
end
```

config security ips profile

The FortiADC Intrusion Prevention System (IPS) combines signature detection and prevention with low latency and excellent reliability. With intrusion protection, you can create multiple IPS profiles, each containing a complete configuration based on signatures. Then, you can apply any IPS profile to any L4 VS.

Intrusion Prevention System (IPS) technology protects your network from cybercriminal attacks by actively seeking and blocking external threats before they can reach potentially vulnerable network devices.

Use this command to configure an IPS profile.

Syntax

```
config security ips profile
   edit <profile>
      set comment {comment}
      config entries
            edit {id}
            set rule {id1 id2 ….}
            set status {disable | enable | default}
            set log {disable | enable}
            set action {pass | block | default}
            set location {loc1 loc2…}
            set severity {sev1 serv2…}
            set protocol {proto1 proto2…}
            set application {app1 app2…}
            set os {os1 os2…}
            set rate-count {count}
            set rate-duration {duration}
            set rate-mode {periodical | continuous}
            set rate-track {field}
         next
end

config load-balance virtual-server
```

```
    set type l4-load-balance
      set ips-profile {name}
    next
end
```

rule	Use rule ID to identify the predefined IPS signatures to add to profile.
status	Specify status of the signatures included in filter. Default is default. • Default enables the filter and only use filters with default status of enable. Filters with default status of disable will not be used. • Enable • Disable
log	Specify the logging status of the signatures included in the filter. • Default is the default. Default enables logging only for filters with default status that is set to enable. Filters with a default logging status of disable will not be logged. • Enable • Disable
action	Specify what action is taken with traffic in which signatures are detected. Default is the default. • `block` will drop the session with offending traffic. • `pass` allow the traffic. • `default` either pass or drop matching traffic, depending on the default action of each signature.
location	Specify the type of system to be protected. Default is all. • All • Client • Server
severity	Relative importance of signature, from info to critical. Default is all. • all • info • low • medium • high • critical
protocol	Specify protocols to be examined. • `?` lists available protocols. • `all` includes all protocols. • `other` includes all unlisted protocols
application	Specify applications to be protected. • `?` lists available applications. • `all` includes all applications. • `other` includes all unlisted applications.
os	Specify operating systems to be protected. Default is all. • `all` includes all operating systems.

	• `other` includes all unlisted operating systems
rate-count	Count of the rate. range[0-65535]
rate-duration	Duration (sec) of the rate. range[1-65535]
rate-mode	Rate limit mode. • `periodical` Allow configured number of packets every rate-duration. • `continuous` Block packets once the rate is reached.
rate-track	Track the packet protocol field. • `none` none • `src-ip` Source IP. • `dest-ip` Destination IP.

Example

```
ADC-6 (profile) # show full
config security ips profile
edit "default"
set comments "Prevent critical attacks."
config entries
edit 1
unset rule
set log enable
set status default
set action default
set location ALL
set severity medium high critical
set protocol ALL
set os ALL
set application ALL
next
end
next
```

config security wad profile

Use this command to configure a security wad profile.

Syntax

```
config security wad profile
edit <name>
     set description “anti-defacement profile” // default is blank
     set monitor [ enable | disable ] // default is disable
     *set host <ip-addr or hostname>
     *set connect-type [ ftp | ssh ]
```

```
      *set port <port-num>
      *set folder <folder-path>
      set user <user-name>
      set password <passwd> // should not show
      set interval-root <num> // unit is seconds
      set interval-other <num> // unit is seconds
      set monitor-depth <num>
      set skip-max-size <num> // unit is KB
      set skip-file-type <extension-name>
      set auto [ restore | acknowledge | disable ] // default disable
   end
end

config system alert-policy
   edit <policy-name>
   config alert-member
   edit <member-name>
   set SEC_Web_Page_Defacement_Detected // add new built-in alert-member
   end
end

config system alert
   edit <alert-name>
   set alert-source-type event
   set event SEC_Bot_Detected // add new event type
   set comments "Web page defacement is detected on virtual server"
end
```

description	Description of WAD profile, default is blank.
monitor	Enable or disable defacement monitoring, default is disable.
host	The website's IPv4 address or hostname for connecting and monitoring.
connect-type	Connect type to host
port	Host port number
folder	Root directory path to perform monitoring
user	Username to connect to the host.
password	Password to connect to the host; shouldn't show.
interval-root	Monitor interval for files in root directory, unit is seconds.
interval-other	Monitor interval for files in subdirectories under root directory; unit is seconds.
monitor-depth	Maximum directory hierarchy depth that can be monitored.
skip-max-size	Skip monitoring files that have a size larger than the maximum number; unit is KB.
skip-file-type	Skip monitoring files that have the specified extension name.
auto	Restore—Automatically restore to the original content once defacement is found. Acknowledge—Automatically confirm the defacement and consider it as new original content Disable—Do not perform any automatic action. Default.

Example

```
ADC-6 # config security wad profile
ADC-6 (profile) # edit 1

ADC-6 (1) #
ADC-6 (1) # set description "profile"

ADC-6 (1) # set monitor enable

ADC-6 (1) # set host 1.1.1.1

ADC-6 (1) # set connect-type ftp

ADC-6 (1) # set port 1

ADC-6 (1) # set folder "folder"

ADC-6 (1) # set user test1

ADC-6 (1) # set password password

ADC-6 (1) # set interval-root 30

ADC-6 (1) # set interval-other 30

ADC-6 (1) # set monitor-depth 1

ADC-6 (1) # set skip-max-size 2

ADC-6 (1) # set skip-file-type "extension"

ADC-6 (1) # set auto restore

ADC-6 (1) # end
ADC-6 (1) # get
description : profile
monitor : enable
host : 1.1.1.1
connect-type : ftp
port : 1
folder : folder
username : test1
password : *
interval-root : 30
interval-other : 30
monitor-depth : 1
skip-max-size : 2
skip-file-type : extension
auto : restore
```

config security waf api-gateway-policy

An API Gateway is an intermediate level or tier of indirection (Gateway) for handling multiple client calls to microservice endpoints. In a microservices architecture, the gateway keeps client apps from directly consuming functionality from multiple microservices, which can become very difficult when there are many microservices. Without API Gateways, the client apps must send requests directly to the microservices and that raises problems, such as coupling, too many round trips, security issues, all microsercvices handle same concerns such as authorization, SSL, etc. Having an API gateway addresses these issues and simplifies the process.

Syntax

```
config security waf api-gateway-policy
   edit <policy-name>
      config rule-list
         edit <policy-rule-id>
            set rule <datasource_api_rule>
            set status {enable|disable}
         next
      end
   next
end
```

CLI Parameter	Description
rule	Specify one or more rules created in API Gateway Rule to be used in policy. The rules will be checked one by one from top to bottom until URL in request is matched to the Full URL Pattern in a rule.
status	Enable or disable the policy rule

See also:

- config security waf api-gateway-rule on page 306
- config security waf api-gateway-user on page 308

config security waf api-gateway-rule

Use this command to create API Gateway rules.

Syntax

```
config security waf api-gateway-rule
   edit <api-rule-name>
      set url-pattern <string>
      set http-method {get|post|head|options|trace|connect|delete|put|patch|other}
      set api-key-verification {enable|disable}
      set api-key-location {http-parameter|http-header}
      set header-field-name <string>
```

```
        set parameter-name <string>
        set action <datasource_action>
        set severity {high|medium|low}
        set host <string>
        set host-status {enable|disable}
        set exception <datasource_exception>
        set rate-limit-status {enable|disable}
        set rate-limit-period <integer>
        set rate-limit-requests <integer>
        config user-list
           edit <user-list-id>
              set user <datasource_api_user>
              set status {enable|disable}
           next
        end
        config attach-http-header
           edit <attach-http-header-id>
              set http-header-name <http-header-name_str>
              set http-header-value <http-header-value_str>
           next
        end
     next
  end
```

CLI Parameter	Description
url-pattern	Matching string. Regular expressions are supported.
http-method	Select one or more HTTP methods that are allowed when accessing the API.
api-key-verification	When a user makes an API request, the API key will be included in the HTTP header or parameter. FortiWeb obtains the API key from the request. When this option is enabled, FortiWeb verifies the key tocheck whether the key belongs to an valid API user.
api-key-location	Indicate where to find the API key in HTTP request: • HTTP Parameter • HTTP Header Note: Available only when API Key Verification is enabled.
header-field-name	Enter the header field name of the API key.
parameter-name	Enter the parameter name of the API key.
action	Select the action profile that you want to apply. See config security waf action on page 346 The default is Alert.
severity	When FortiADC records violations of this rule in the attack log, each log message contains a Severity Level (severity_level) field. Select which severity level FortiADC uses when using Input Validation: • Low • Medium • High The default value is Low.

CLI Parameter	Description
host	Select the name of a protected host that the Host: field of an HTTP request must be in to match the API gateway rule. This option is available only if Host Status is enabled.
host-status	Enable/Disable for applying this rule only to HTTP requests for specific web hosts
exception	Select a user-defined exception configuration object. Exceptions identify specific hosts or URL patterns that are not subject to processing by this rule.
rate-limit-status	Enable/Disable to do rate limit for API calls
rate-limit-period	range 1-600 seconds, default 60
rate-limit-requests	range 1-100000, default 600
user-list (available when api-key-verification is enabled)	
user	Specify one or more users created in API Gateway User to define which users have the persmission to access the API.
status	enable or disable user list
attach-http-header	
http-header-name	Field name of specific header lines to be inserted into HTTP header.
http-header-value	Value of specific header lines to be inserted into HTTP header.

See also:

- config security waf api-gateway-policy on page 306
- config security waf api-gateway-user on page 308

config security waf api-gateway-user

Syntax

```
config security waf api-gateway-user
   edit <api-user_name>
      set comments <string>
      set uuid <string>
      set api-key <string>
      config ip-access-list
         edit <ip-access-id>
            set ip-netmask <ip&netmask>
         next
      end
      config http-referer-list
         edit <http-referer-id>
            set http-referer <string>
         next
      end
   next
```

```
end
```

CLI Parameter	Description
comments	(Optional) Enter a description or comments for the user
uuid	Note: cannot be changed once set
api-key	Note: cannot be changed once set
ip-access-list	
ip-netmask	Restricted Access IPs. Specify the IP addresses that can access the API key.
http-referer-list	
http-referer	Restrict HTTP Referers. Specify the URLs that can use the API key when present in the Referer HTTP header. This can be used to prevent an API key from being reused on other client-side web applications that don't match this URL. URLs must be complete and begin with 'http://' or 'https://'

See also:

- config security waf api-gateway-policy on page 306
- config security waf api-gateway-rule on page 306

config security waf api-discovery

Use this command to configure API Discovery policies that allow FortiADC to automatically discover external API endpoints from HTTP/HTTPS requests and responses that have passed through API validity checks, wherein the API is parsed for information including the Host, Paths, parameters and their schemas from query requests or entity bodies, as well as classify parameters that match PII (Personal Identifiable Information) signatures. API Discovery also supports manually imported OAS files compliant with OpenAPI 3.0 and Swagger 2.0 standard to parse and discover as internal API endpoints that can also be matched by incoming API requests or responses. The discovered external and internal API endpoints can then be directly applied in API security rules based on the Host, Path, and request rate. Once the API requests and responses pass the API validity check that matches the rule, the specified security action will be triggered to protect against the malicious APIs.

API Discovery policies depend entirely on internal or external API endpoints to function. For external API endpoints to be discovered, the API Discovery policy must be referenced in a WAF Profile used in an active virtual server. When the virtual server referencing the API Discovery policy receives API responses/requests, external API endpoint discovery is automatically activated. The API Discovery policy will automatically begin validating APIs and parsing endpoints to build your API endpoints database which can then be viewed in the API View page (in the GUI). From the API Discovery policy, you can add API Security rules to trigger alerts and actions against APIs identified as malicious through API Discovery.

Use the `config security waf api-discovery` command to configure automatic discovery for external API endpoints. To manually import internal API endpoints, see execute oas-file import on page 576.

API Discovery is based on VDOMs, where API endpoints are discovered and stored per VDOM.

API Discovery is based on VDOMs, where API endpoints are discovered and stored per VDOM. The total API endpoints database size for each VDOM is 1 GB.

The maximum number of API Discovery policies is 256.

Syntax

```
config security waf api-discovery
   edit <name>
      set api-discovery {enable|disable}
      config api-security-rule
         edit <No.>
            set host <string>
            set path <string>
            set rate-limit <integer>
            set severity {high|medium|low}
            set action <datasource>
         next
      end
   next
end
```

`api-discovery`	Enable or disable the API Discovery configuration. This is disabled by default.
config api-security-rule	
`host`	Specify the HTTP Host header. This is required. Maximum length is 255 characters. Example: 192.168.0.253, [2001:1234::a41:6e]:8443, or demo.fortinet.com. Once the API Discovery policy is activated, the policy matches only if the Host header matches this value. Complete, exact matching is required. For example, www.example.com matches www.example.com but not www.example.com.hk.
`path`	Specify the API resource path. Text string and simple regex is supported. Example: /login. Begin with '/'.
`rate-limit`	Specify the allowable requests per second. Default: 0. Range: 0 - 100000000. **Note**: 0 means there is no limit.
`severity`	Select the action profile to apply when a bot is detected. See config security waf action on page 346. The default action is alert.
`action`	Select the event severity to log when a bot is detected: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is low.

Example

```
config security waf api-discovery
   edit "api-discovery-test"
      set api-discovery enable
      config api-security-rule
         edit 1
            set host 192.168.0.253
            set path /api/*
            set rate-limit 0
            set severity low
            set action alert
         next
      end
   next
end
```

config security waf bot-detection

Use this command to configure Bot Detection policies. Bot Detection policies use heuristics to detect client traffic likely to be generated by robots instead of genuine clients. You can use predefined blocklists and allowlists to get started. You can use the user-specified allowlist table to fine-tune detection.

After you have created a bot detection policy, you can specify it in a WAF profile configuration.

Before you begin:

- You must have read-write permission for security settings.

Syntax

```
config security waf bot-detection
   edit <name>
      set status {enable|disable}
      set bad-robot {enable|disable}
      set search-engine-crawler {enable|disable}
      set search-engine list [Ask|Baidu|Bing|DuckDuckGo|Google|Sogou|Yahoo|Yandex ]
      set action {datasource}
      set http-request-rate <integer>
      set severity {high|low|medium}
      config allowlist
         edit <No.>
            set cookie-name-pattern <string>
            set ip <subnet>
            set url-pattern <string>
            set url-parameter-name-pattern <string>
            set user-agent-pattern <string>
         next
      end
   next
```

```
end
```

status	Enable/disable bot detection.
bad-robot	Enable/disable the predefined bad robot blocklist.
search-engine-crawler	Enable/disable the predefined search engine spider allowlist.
search-engine-list	Set list of search engines. Default value is all search engines.
action	Specify a WAF action object.
http-request-rate	The default is 0 (off). The valid range is 0-100,000,000 requests per second.
severity	• high • medium • low
config allowlist	
cookie-name-pattern	Matching string. Regular expressions are supported.
ip	Matching subnet (CIDR format).
url-pattern	Matching string. Regular expressions are supported.
url-parameter-name-pattern	Matching string. Regular expressions are supported.
user-agent-pattern	Matching string. Regular expressions are supported.

Example

```
ADC-3 (root) # config security waf bot-detection
ADC-3 (bot-detection) # edit waf-bot-detection-policy
ADC-3 (waf-bot-detect~y) # get
status : disable
ADC-3 (waf-bot-detect~y) # set status enable
ADC-3 (waf-bot-detect~y) # get
status : enable
search-engine-crawler : enable
search-engine-list : Bing Google Yahoo
bad-robot : enable
http-request-rate : 0
action :
severity : low
ADC-3 (waf-bot-detect~y) # config allowlist
ADC-3 (allowlist) # edit 1
ADC-3 (1) # get
ip : 0.0.0.0/0
url-pattern :
url-parameter-name-pattern :
user-agent-pattern :
cookie-name-pattern :
ADC-3 (1) # set ip 10.1.1.0/24
ADC-3 (1) # end
ADC-3 (waf-bot-detect~y) # end
```

config security waf threshold-based-detection

Use this command to configure Threshold Based Detection policies. FortiADC uses Threshold Based Detection policies to determine whether requests are generated by robots instead of a human by detecting suspicious behavior patterns that exceed the normal threshold defined in the policy. Threshold Based Detection rules are defined by the number of times a type of behavior is allowed to occur within a specified amount of time. Once the number of occurrence exceeds the defined threshold value, an action is triggered in response to detecting the suspicious behavior.

FortiADC supports the following three types of Threshold Based Detection:

- Crawler Detection — Detects web crawlers that are usually used to map out your application structure by monitoring the frequency of HTTP response codes. If the occurrence of a specified HTTP response code exceeds the allowable threshold in the specified time frame, FortiADC will execute the relevant action for the traffic.
- Content Detection — Detects malicious tools that try to download large amounts of content such as text/HTML and application/ XML from your website by monitoring the frequency of download activities. If the occurrence of the download activity exceeds the allowable threshold within the specified time frame, FortiADC will execute the relevant action for the traffic.
- Attack Detection — Detects suspicious attack behavior patterns indicative of a bot attack by monitoring the frequency of attacks detected in specific WAF Attack modules. If the occurrence of specific attacks exceeds the allowable threshold within the specified time frame, FortiADC will execute the relevant action for the traffic.

FortiADC offers Predefined Threshold Based Detection policy configurations on page 313 you can use to get started.

Predefined Threshold Based Detection policy configurations

Name	Comments	Predefined settings
Bot_Detect	Detect suspicious bot with CAPTCHA action	**Crawler Status — Enabled** Response Code — 403,404 Crawler Action — captcha Crawler Severity — Medium Crawler Occurrence Limit — 100 Crawler Occurrence Within — 60 (seconds) **Content Scraping Status — Enabled** Content Type — Text/HTML, Text/Plain, Text/XML, Application/XML, Application/Soap+XML, Application/JSON Content Action — captcha Content Severity — Medium Content Occurrence Limit — 100 Content Occurrence Within — 60 (seconds) **Attack Detection Status — Enabled** Attack Modules — Web Attack Signature, Input Validation, Brute Force Attack Detection, URL Protection, HTTP Protocol Constraint, Credential Stuffing Defense Attack Action — captcha Attack Severity — Medium

Name	Comments	Predefined settings
		Attack Occurrence Limit — 100 Attack Occurrence Within — 60 (seconds)
Content_Scraping_Detect	Monitor the frequency of illegal content scraping with ALERT action	**Crawler Status — Disabled** **Content Scraping Status — Enabled** Content Type — Text/HTML, Text/Plain, Text/XML, Application/XML, Application/Soap+XML, Application/JSON Content Action — alert Content Severity — Low Content Occurrence Limit — 100 Content Occurrence Within — 60 (seconds) **Attack Detection Status — Disabled**
Crawler_Detect	Monitor the frequency of 403 and 404 response codes with ALERT action	**Crawler Status — Enabled** Response Code — 403,404 Crawler Action — alert Crawler Severity — Low Crawler Occurrence Limit — 100 Crawler Occurrence Within — 60 (seconds) **Content Scraping Status — Disabled** **Attack Detection Status — Disabled**
High-Level-Security	Block all suspicious threshold violations	**Crawler Status — Enabled** Response Code — 403,404 Crawler Action — deny Crawler Severity — High Crawler Occurrence Limit — 100 Crawler Occurrence Within — 60 (seconds) **Content Scraping Status — Enabled** Content Type — Text/HTML, Text/Plain, Text/XML, Application/XML, Application/Soap+XML, Application/JSON Content Action — deny Content Severity — High Content Occurrence Limit — 100 Content Occurrence Within — 60 (seconds) **Attack Detection Status — Enabled** Attack Modules — Web Attack Signature, Input Validation, Brute Force Attack Detection, URL Protection, HTTP Protocol Constraint, Credential Stuffing Defense • Advanced — Data Leak Prevention, SQL/XSS Injection Detection, Cookie Security, CSRF

Name	Comments	Predefined settings
		Protection, CORS Protection, JSON Validation, OpenAPI Validation, XML Protection, API Gateway Attack Action — deny Attack Severity — High Attack Occurrence Limit — 100 Attack Occurrence Within — 60 (seconds)
Illegal_User_Detect	Detect illegal user with CAPTCHA action	**Crawler Status — Disabled** **Content Scraping Status — Disabled** **Attack Detection Status — Enabled** Attack Modules — Brute Force Attack Detection, Credential Stuffing Defense Attack Action — captcha Attack Severity — Medium Attack Occurrence Limit — 100 Attack Occurrence Within — 60 (seconds)
Vulnerability_Scan	Monitor the frequency of web attack signature violations with CAPTCHA action	**Crawler Status — Disabled** **Content Scraping Status — Disabled** **Attack Detection Status — Enabled** Attack Modules — Web Attack Signature Attack Action — captcha Attack Severity — Medium Attack Occurrence Limit — 100 Attack Occurrence Within — 60 (seconds)

After you have configured Threshold Based Detection policies, you can select them in WAF profiles.

Before you begin:

- You must have read-write permission for security settings.

Syntax

```
config security waf threshold-based-detection
   edit <name>
      set crawler-status {enable|disable}
      set response-code <integer>
      set crawler-action <datasource>
      set crawler-severity {high|medium|low}
      set crawler-occurrence-limit <integer>
      set crawler-occurrence-within <integer>
      set content-scraping-status {enable|disable}
      set content-type
           [text/html|text/plain|text/xml|application/xml|application/soap+xml|application/j
           son]
      set content-action <datasource>
      set content-severity {high|medium|low}
```

```
      set content-occurrence-limit <integer>
      set content-occurrence-within <integer>
      set attack-detection-status {enable|disable}
      set attack-modules [web-attack-signature|http-protocol-constraint|sql-xss-injection-
           detection|url-protection|xml-validation|json-validation|openapi-
           validation|cookie-security|csrf-protection|brute-force-login|data-leak-
           prevention|input-validation|credential-stuffing-defense|http-header-security|api-
           gateway|cors-protection]
      set attack-action <datasource>
      set attack-severity {high|medium|low}
      set attack-occurrence-limit <integer>
      set attack-occurrence-within <integer>
      set comment <string>
    next
end
```

`crawler-status`	Enable/Disable Crawler Detection. This is disabled by default.
`response-code`	The **response-code** option is available if **crawler-status** is enabled. Specify the 3 digit HTTP response code(s) to check. Enter as a single code (e.g. 403), multiple codes (e.g. 403,404), or as a range (e.g. 500-503). Range: 100-599.
`crawler-action`	The **crawler-action** option is available if **crawler-status** is enabled. Select the action profile to apply when a web crawler bot is detected. See config security waf action on page 346. The default action is alert.
`crawler-severity`	The **crawler-severity** option is available if **crawler-status** is enabled. Select the event severity to log when a web crawler bot is detected: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is low.
`crawler-occurrence-limit`	The **crawler-occurrence-limit** option is available if **crawler-status** is enabled. Specify the maximum number of responses that can be received from the specified **response-code** within the time frame (set in **crawler-occurrence-limit**). If the limit is exceeded, the specified **crawler-action** will be triggered. Default: 100, Range: 1-100000.
`crawler-occurrence-within`	The **crawler-occurrence-within** option is available if **crawler-status** is enabled. Specify the time span during which to count how many times a response is received from the specified **response-code**. Default: 60 seconds, Range: 1-600 seconds.
`content-scraping-status`	Enable/disable Content Detection. This is disabled by default.
`content-type`	The **content-type** option is available if **content-scraping-status** is enabled. Select one or more content type to monitor for content scraping: • text/html • text/plain • text/xml

	• application/xml • application/soap+xml • application/json
`content-action`	The **content-action** option is available if **content-scraping-status** is enabled. Select the action profile to apply when a content scraping bot is detected. See config security waf action on page 346.. The default action is alert.
`content-severity`	The **content-severity** option is available if **content-scraping-status** is enabled. Select the event severity to log when a content scraping bot is detected: • High — Log as high severity events. • Medium — Log as a medium severity events. • Low — Log as low severity events. The default is low.
`content-occurrence-limit`	The **content-occurrence-limit** option is available if **content-scraping-status** is enabled. Specify the maximum number of responses that can be received from the specified **content-type** within the time frame (set in **content-occurrence-within**). If the limit is exceeded, the specified **content-action** will be triggered. Default: 100, Range: 1-100000.
`content-occurrence-within`	The **content-occurrence-within** option is available if **content-scraping-status** is enabled. Specify the time span during which to count how many times a response is received from the specified **content-type**. Default: 60 seconds, Range: 1-600 seconds.
`attack-detection-status`	Enable/disable Attack Detection. This is disabled by default.
`attack-modules`	The **attack-modules** option is available if **attack-detection-status** is enabled. Select one or more attack modules to monitor for bot attacks: • web-attack-signature • http-protocol-constraint • sql-xss-injection-detection • url-protection • xml-validation • json-validation • openapi-validation • cookie-security • csrf-protection • brute-force-login • data-leak-prevention • input-validation • credential-stuffing-defense • http-header-security • api-gateway • cors-protection

attack-action	The **attack-action** option is available if **attack-detection-status** is enabled. Select the action profile to apply when a bot attack is detected. See config security waf action on page 346. The default action is alert.
attack-severity	The **attack-severity** option is available if **attack-detection-status** is enabled. Select the event severity to log when a bot attack is detected: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is low.
attack-occurrence-limit	The **attack-occurrence-limit** option is available if **attack-detection-status** is enabled. Specify the maximum number of responses that can be received from the specified **attack-modules** within the time frame (set in **attack-occurrence-within**). If the limit is exceeded, the specified **attack-action** will be triggered. Default: 100, Range: 1-100000.
attack-occurrence-within	The **attack-occurrence-within** option is available if **attack-detection-status** is enabled. Specify the time span during which to count how many times a response is received from the specified **attack-modules**. Default: 60 seconds, Range: 1-600 seconds.
comment	Optionally, enter comments about the Threshold Based Detection policy.

Example

```
config security waf threshold-based-detection
   edit "attack_1"
      set crawler-status enable
      set response-code 404
      set crawler-action alert
      set crawler-severity low
      set crawler-occurrence-limit 1
      set crawler-occurrence-within 60
      set content-scraping-status enable
      set content-type text/html
      set content-action alert
      set content-severity low
      set content-occurrence-limit 1
      set content-occurrence-within 60
      set attack-detection-status enable
      set attack-modules url-protection
      set attack-action alert
      set attack-severity low
      set attack-occurrence-limit 2
      set attack-occurrence-within 60
   next
end
```

config security waf biometrics-based-detection

Use this command to configure Biometrics Based Detection policies. FortiADC uses Biometrics Based Detection policies to determine whether requests are generated by robots instead of a human by checking client events within a specified period. With JavaScript enabled on the client browser, FortiADC can collect behavioral biometrics (such as mouse movement, keyboard, screen touch, and scroll) and monitor as client events for a specified period. FortiADC can then determine whether the behavioral biometrics from the request is indicative of a bot or a human.

After you have configured Biometrics Based Detection policies, you can select them in WAF profiles.

Before you begin:

- You must have read-write permission for security settings.

Syntax

```
config security waf biometrics-based-detection
   edit <name>
      set ignore-js-check {enable|disable}
      set monitor-client-event [mouse-movement|click|keyboard|screen-touch|scroll]
      set event-collection-time <integer>
      set bot-effective-time <integer>
      set js-request-url <string>
      set action <datasource>
      set severity {high|medium|low}
      set exception <datasource>
      config url-list
         edit <No.>
            set host-status {enable/disable}
            set host <string>
            set request-url <string>
         next
      end
   next
end
```

`ignore-js-check`	Enable/disable redirect to a warning page to enable JavaScript. This is disabled by default. • disable — FortiADC will check if JavaScript is enabled on the client browser. If JavaScript is not enabled, then FortiADC will redirect to a warning page to let the user enable JavaScript. If the client does not enable JavaScript within 10 seconds, the traffic may be recognized as a bad bot. • enable — FortiADC will not check if JavaScript is enabled on the client browser. If JavaScript is enabled on the client browser, events can be collected normally and FortiADC can determine if it is a bot or not. But if JavaScript is disabled on the client browser, the client will be recognized as a bot after the Event Collection Time, since events cannot be collected by FortiADC.
`monitor-client-event`	Select one or more client events to monitor: • mouse-movement

	• click • keyboard • screen-touch • scroll By default, mouse-movement, click, and keyboard are preselected. If the configuration is saved with no monitor-client-event selected, it will default to the preselected client events.
`event-collection-time`	Specify for how long the events will be collected from the client. Default: 60 Range: 10-3600 seconds.
`bot-effective-time`	Specify the time interval before FortiADC tests and verifies a bot again, once a bot has been detected. Default: 5 Range: 1-60 minute(s).
`js-request-url`	Specify the URL to use to insert JavaScript code to the client machine. Default: `/fadc_client/default_index.js`.
`action`	Select the action profile to apply when a bot is detected. See config security waf action on page 346. The default action is alert.
`severity`	Select the event severity to log when a bot is detected: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is low.
`exception`	Select an exception configuration object. Exceptions identify specific hosts or URL patterns that are not subject to processing by this rule.
config url-list	
`host-status`	If enabled, require authorization only for the specified host. If disabled, ignore hostname in the HTTP request header and require authorization for requests with any Host header. Disabled by default.
`host`	The **host** option is available if **host-status** is enabled. Specify the HTTP Host header. If Host Status is enabled, the policy matches only if the Host header matches this value. Complete, exact matching is required. For example, `www.example.com` matches `www.example.com` but not `www.example.com.hk`.
`request-url`	The literal URL, such as `/index.php`, or a regular expression, such as `^/*.php` that the HTTP request must contain in order to match the rule. Multiple URLs are supported.

Example

```
config security waf biometrics-based-detection
   edit "Test"
      set ignore-js-check disable
      set monitor-client-events click
      set event-collection-time 10
```

```
      set bot-effective-time 1
      set js-request-url /fadc_client/default_index.js
      set action deny
      set severity low
      set exception IP_exception
      config url-list
         edit 1
            set host-status disable
            set request-url .*
         next
      end
   next
end
```

config security waf fingerprint-based-detection

Use this command to configure Fingerprint Based Detection policies. Fingerprint Based Detection detects the client's fingerprint using multiple characteristics to ascertain whether an access request originates from a human or a bot. Using the Fingerprint Based Detection policy, FortiADC collects and monitors the behavioral fingerprint information, such as WebDriver, WindowsProperties and MimeTypesConsistent, with JavaScript enabled on the client browser. The collected information forms the client fingerprint that is then checked against known automation tools and frameworks to determine whether the requests are generated by automation tools such as Headless Chrome, Selenium or Electron.

After you have configured Fingerprint Based Detection policies, you can select them in WAF profiles.

Before you begin:

- You must have read-write permission for security settings.

Syntax

```
config security waf biometrics-based-detection
   edit <name>
      set ignore-js-check {enable|disable}
      set fingerprint-detectors [chrome_headless|selenium|electron|others]
      set bot-effective-time <integer>
      set js-request-url <string>
      set action <datasource>
      set severity {high|medium|low}
      set exception <datasource>
      config url-list
         edit <No.>
            set host-status {enable/disable}
            set host <string>
            set request-url <string>
         next
      end
   next
end
```

`ignore-js-check`	Enable/disable redirect to a warning page to enable JavaScript. This is disabled by default. • Disable — FortiADC will check if JavaScript is enabled on the client browser. If JavaScript is not enabled, then FortiADC will redirect to a warning page to let the user enable JavaScript. If the client does not enable JavaScript, Fingerprint Based Detection cannot be activated. • Enable — FortiADC will not check if JavaScript is enabled on the client browser. If JavaScript is enabled on the client browser, behavioral fingerprint information can be collected normally and FortiADC can determine if it is a bot or not. But if JavaScript is disabled on the client browser, Fingerprint Based Detection cannot be activated.
`fingerprint-detectors`	Select one or more fingerprint detectors: • chrome_headless — Headless Chrome • selenium — Selenium • electron — Electron • others — includes SlimerJS and CefSharp All fingerprint detectors are preselected by default. If the configuration is saved with no Fingerprint Detectors selected, it will default to the preselected.
`bot-effective-time`	Specify the time interval before FortiADC tests and verifies a bot again, once a bot has been detected. Default: 5 Range: 1-60 minute(s).
`js-request-url`	Specify the URL to use to insert JavaScript code to the client browser. Default: `/fadc_client/fp_detect.js`.
`action`	Select the action profile to apply when a bot is detected. See config security waf action on page 346. The default action is alert.
`severity`	Select the event severity to log when a bot is detected: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is low.
`exception`	Select an exception configuration object. Exceptions identify specific hosts or URL patterns that are not subject to processing by this rule.
config url-list	
`host-status`	If enabled, require authorization only for the specified host. If disabled, ignore hostname in the HTTP request header and require authorization for requests with any Host header. Disabled by default.
`host`	The **host** option is available if **host-status** is enabled. Specify the HTTP Host header. If Host Status is enabled, the policy matches only if the Host header matches this value. Complete, exact matching is required. For example, `www.example.com` matches `www.example.com` but not `www.example.com.hk`.
`request-url`	The literal URL, such as `/index.php`, or a regular expression, such as `^/*.php`

	that the HTTP request must contain in order to match the rule. Multiple URLs are supported.

Example

```
config security waf biometrics-based-detection
   edit "Test"
      set ignore-js-check disable
      set fingerprint-detectors chrome_headless selenium electron others
      set bot-effective-time 1
      set js-request-url /fadc_client/fp_detect.js
      set action deny
      set severity low
      set exception IP_exception
      config url-list
         edit 1
            set host-status disable
            set request-url .*
         next
      end
   next
end
```

config security waf advanced-bot-protection

Use this command to configure the Advanced Bot Protection policy once you have successfully connected to the FortiGuard Advanced Bot Protection server via the Advanced Bot Protection Fabric Connector.

You can configure an Advanced Bot Protection policy for your virtual server to protect your online applications from malicious bots and automated attacks. By incorporating FortiGuard ABP into FortiADC's server policy, client traffic will be directed to the FortiGuard ABP service deployed on Google Cloud where it will be analyzed to identify any malicious bot behavior and initiate appropriate actions in response.

FortiGuard ABP features a multi-dimensional deep learning engine that learns and tracks bot attacks over time, delivering the highest possible accuracy of classification between humans, and good and bad bots.

FortiGuard ABP protects against a wide range of threats, including the following:

- Data harvesting
- Credential stuffing attacks
- Account takeover attempts
- DDoS attacks

After you have configured the Advanced Bot Protection policy, you can reference it in a WAF Profile and apply it in a virtual server policy. However, the Advanced Bot Protection policy does not activate until the FortiGuard ABP Application is fully analyzed and Pre-Provisioned to protect the Application.

Pre-Provisioning is required to identify all URLs that should be protected in your Application domain (such as login URLs), and the locations to which JavaScript need to be inserted to collect client information. Without these resources, FortiADC will not be able to insert the necessary JavaScript for bot detection.

Pre-Provisioning is triggered upon creating the Application, and requires 2 to 3 days to complete. During this process, your FortiGuard ABP Application will be in **Pending** status until Pre-Provisioning is complete. When the Application status is **Ready**, Advanced Bot Protection can be activated in your FortiADC.

When Advanced Bot Protection is initially activated, it is recommended to set the WAF action that would allow you to observe and log any events detected by the FortiGuard Advanced Bot Protection, instead of immediately setting to block.

FortiGuard Advanced Bot Protection uses a multidimensional deep learning engine to learn and track bot attacks over time by using sophisticated AI model training. As FortiGuard ABP builds its training model, it will continue to improve and refine its bot detection capabilities. However, this may mean triggering false positives in the initial stages of the AI model training.

For more information, see the Handbook topic about Advanced Bot Protection.

Before you begin:

- You must have enabled and successfully connected the Advanced Bot Protection connector via `config system global`. See config system global on page 69.
  The `config security waf advanced-bot-protection` command is only available after the Advanced Bot Protection connector is enabled.
- You must have read-write permission for security settings.
- You must have access to the FortiGuard Advanced Bot Protection User Portal to obtain the Application ID from an existing Application or create a new configuration. For more information, see the Handbook topic on how to obtain the FortiGuard ABP Application ID.

Syntax

```
config security waf advanced-bot-protection
   edit <name>
      set bot-detection-status {enable|disable}
      set bot-detection-action {alert|deny|block|silent-deny|captcha|<datasource>}
      set bot-detection-severity {high|medium|low}
      set application-id <string>
      set exception <datasource>
   next
end
```

`bot-detection-status`	Enable/disable the status of this Advanced Bot Protection policy. You must **enable** the **bot-detection-status** to see configuration options for the Advanced Bot Protection policy.
`bot-detection-action`	Specify a WAF action object to apply when a bot is detected. You can specify a predefined or user-defined WAF action profile. (See config security waf action on page 346.) Predefined WAF actions: • alert — WAF policies will allow the traffic to pass and log the event. • block — WAF policies will drop the current attack session by HTTP 403 message and block the attacker (according the attacker's IP address) for 1

	hour, and log the event. • captcha — WAF policies will allow the traffic to pass if the client successfully fulfills the CAPTCHA request, and log the event. • deny — WAF policies will the drop current attack session by HTTP 403 message, and log the event. • silent-deny — WAF policies will drop the current attack session by HTTP 403 message, without logging the event. The default action is alert.
`bot-detection-severity`	Select the event severity to log when a bot is detected: • high — Log as high severity events. • medium — Log as a medium severity events. • low — Log as low severity events. The default is low.
`application-id`	Specify the Application ID assigned to your FortiGuard ABP Application. The Application ID is used to bind this Advanced Bot Protection policy to the FortiGuard ABP Application. For steps on how to obtain the Application ID from the FortiGuard ABP User Portal, see the Handbook topic on how to obtain the FortiGuard ABP Application ID.
`exception`	Select an exception configuration object. Exceptions identify specific hosts or URL patterns that are not subject to processing by this rule.

Example

```
config security waf advanced-bot-protection
   edit "ABP_store"
      set bot-detection-status enable
      set bot-detection-action deny
      set bot-detection-severity low
      set application-id FORTISTOREFORTISTORE
      set exception exception_policy
   next
end
```

config security waf exception

Use this command to create exception configuration objects. WAF exceptions identify specific patterns that are not subject to processing by WAF rules. Use WAF exception rules to reduce false-positives triggered by legitimate HTTP requests that match an attack signature rule.

Before you begin:

- You must have read-write permission for security settings.

FortiADC supports the following exception rule types.

Each exception rule type requires specific parameters to be applied. Use the links above to navigate to the CLI commands for each exception rule type.

After you have created an exception object, you can specify it in WAF profiles and individual WAF feature rules.

For optimal functionality, we recommend keeping the number of WAF exception rules configured to a minimum. If a large number of WAF exception rules are configured, none may work effectively due to limitations of the shared memory (maximum total is 256.0 MBs in the VM platform).

URL

Use this command to create a URL exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type URL
            set host-status {enable|disable}
            set host-pattern <string>
            set url-pattern <string>
         next
      end
   next
end
```

`host-status`	Enable/disable the setting exceptions by host pattern.
`host-pattern`	The **host-pattern** option is configurable if **host-status** is enabled. Specify the matching string. Regular expressions are supported. Maximum length is 128 characters. For example, you can specify `www.example.com`, `*.example.com`, or `www.example.*` to match a literal host pattern or a wildcard host pattern.
`url-pattern`	Specify the matching string. Must begin with a URL path separator (/). Regular expressions are supported. Maximum length is 128 characters. For example, you can specify path names and files with expressions like `\/admin`, `.*\/data\/1.html`, or `\/data.*`.

Example

```
config security waf exception
   edit "url"
      config exception-rule
         edit 1
            set type URL
            set host-status disable
            set url-pattern /url1
         next
      end
   next
end
```

Source IP

Use this command to create a Source IP exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type source-ip
            set ip-netmask <ip&netmask>
         next
      end
   next
end
```

ip-netmask	Specify the IPv4 address with netmask. For example: 192.0.2.5/24

Example

```
config security waf exception
   edit "source-ip"
      config exception-rule
         edit 2
            set type source-ip
            set ip-netmask 192.0.2.0/24
         next
      end
   next
end
```

Source IPv6

Use this command to create a Source IPv6 exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type source-ipv6
            set ipv6-netmask <ip&netmask>
         next
      end
   next
end
```

ipv6-netmask	Specify the IPv6 address with netmask. For example: 2001:0db8:85a3::8a2e:0370:7334/64

Example

```
config security waf exception
   edit "source-ipv6"
      config exception-rule
         edit 3
            set type source-ipv6
            set ipv6-netmask 2001:1::50:1:0:20/128
         next
      end
   next
end
```

HTTP Method

Use this command to create an HTTP method exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type http-method
            set methods [GET|POST|HEAD|TRACE|CONNECT|DELETE|PUT|PATCH|OPTIONS|OTHERS]
         next
      end
   next
end
```

methods	Select the HTTP method(s): • GET • POST

	• HEAD • TRACE • CONNECT • DELETE • PUT • PATCH • OPTIONS • OTHERS

Example

```
config security waf exception
   edit "http-method"
      config exception-rule
         edit 4
            set type http-method
            set methods HEAD TRACE CONNECT DELETE PUT PATCH OPTIONS OTHERS
         next
      end
   next
end
```

HTTP Header

Use this command to create an HTTP Header exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type http-header
            set value-check {enable|disable}
            set name-pattern <string>
            set value-pattern <string>
         next
      end
   next
end
```

`value-check`	Enable/disable value checking for the specified element.
`name-pattern`	Specify the matching string. Regular expressions are supported. Maximum length is 1024 characters. For example: `. Content*`

	Note: Some characters must be escaped to be a valid regular expression or be functional as an exception rule. For details, see Limitations: Escaped Characters on page 332.
`value-pattern`	The **value-pattern** option is **required** if **value-check** is enabled. Specify the matching string. Regular expressions are supported. Maximum length is 1024 characters. For example: `. Content*` **Note**: Some characters must be escaped to be a valid regular expression or be functional as an exception rule. For details, see Limitations: Escaped Characters on page 332.

Example

```
config security waf exception
   edit "http-header"
      config exception-rule
         edit 5
            set type http-header
            set value-check enable
            set name-pattern h_name1
            set value-pattern h_value1
         next
      end
   next
end
```

Cookie

Use this command to create a Cookie exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type cookie
            set value-check {enable|disable}
            set name-pattern <string>
            set value-pattern <string>
         next
      end
   next
end
```

`value-check`	Enable/disable value checking for the specified element.
`name-pattern`	Specify the matching string. Regular expressions are supported. Maximum length is 1024 characters.

	For example: `. Content*` **Note**: Some characters must be escaped to be a valid regular expression or be functional as an exception rule. For details, see Limitations: Escaped Characters on page 332.
`value-pattern`	The **value-pattern** option is **required** if **value-check** is enabled. Specify the matching string. Regular expressions are supported. Maximum length is 1024 characters. For example: `. Content*` **Note**: Some characters must be escaped to be a valid regular expression or be functional as an exception rule. For details, see Limitations: Escaped Characters on page 332.

Example

```
config security waf exception
   edit "cookie"
      config exception-rule
         edit 6
            set type cookie
            set value-check enable
            set name-pattern c_name1
            set value-pattern c_value1
         next
      end
   next
end
```

Parameter

Use this command to create a Parameter exception rule.

Syntax

```
config security waf exception
   edit <name>
      config exception-rule
         edit <No.>
            set type parameter
            set value-check {enable|disable}
            set name-pattern <string>
            set value-pattern <string>
         next
      end
   next
end
```

`value-check`	Enable/disable value checking for the specified element.

`name-pattern`	Specify the matching string. Regular expressions are supported. Maximum length is 1024 characters. For example: `. Content*` **Note**: Some characters must be escaped to be a valid regular expression or be functional as an exception rule. For details, see Limitations: Escaped Characters on page 332.
`value-pattern`	The **value-pattern** option is **required** if **value-check** is enabled. Specify the matching string. Regular expressions are supported. Maximum length is 1024 characters. For example: `. Content*` **Note**: Some characters must be escaped to be a valid regular expression or be functional as an exception rule. For details, see Limitations: Escaped Characters on page 332.

Example

```
config security waf exception
   edit "parameter"
      config exception-rule
         edit 7
            set type parameter
            set value-check enable
            set name-pattern p_name1
            set value-pattern p_value1
         next
      end
   next
end
```

Limitations: Escaped Characters

All **name-pattern** and **value-pattern** support regular expression. However, some characters must be escaped to be a valid regular expression or be functional as an exception rule. If your expression contains characters that require escaping, an error message may be triggered to reject the invalid expression. However, it is also possible that an error may not be triggered by unescaped characters if it is considered syntactically correct — in which case these expressions would not function as exception rules since they will not match any user traffic.

This section lists the most commonly used special characters that need to be escaped to make an input valid or functional as an exception rule.

Brackets: `[]`

Brackets (`[]`) require other characters between the brackets to be a valid exception rule regular expression.

For example:

Invalid: `[]`

Valid: `[123]` — Valid

`[]` is an invalid exception rule regular expression because the input only contain brackets with no other characters in between. Whereas `[123]` is valid because there are number characters between the brackets.

Parentheses: `()`

Parentheses (`()`) require a backslash (`\`) before each parenthesis to be a valid exception rule regular expression — `()` → `\(\)`

For example:

Invalid: `http://x.x.x.x/login?link=mocha:alert('attack%20success')`

Valid: `http://x.x.x.x/login?link=mocha:alert\('attack success'\)`

Focusing on the parameter value, `mocha:alert('attack%20success')` is invalid because there is no backslash before each parenthesis. Whereas `mocha:alert\('attack success'\)` is valid with the backslash inserted before each parenthesis.

Asterisk: `*`

Asterisks (`*`) require a backslash (`\`) before each asterisk to be a valid and functional exception rule — `*` → `\*`

For example:

curl -vv -X POST --cookie "`Cookie123=abcd`"

"`http://x.x.x.x/index.php?n123=v123&p_name1=p_value1`"

Where the cookie name is "`cookie`" and the cookie value is "`a*`"

Invalid: `a*`

Valid: `a\*`"

In this case, both `a*` and `a\*` are both correct in syntax. However, `a*` would not be functional as an exception rule because it would not match any user traffic.

Space: `%20`

Spaces (`%20`) in URLs must be replaced with spaces to be a valid exception rule regular expression.

For example:

Invalid: `http://x.x.x.x/login?link=mocha:alert('attack%20success')`

Valid: `http://x.x.x.x/login?link=mocha:alert(\'attack success'\)`

Focusing on the parameter value, the invalid expression becomes valid when the `%20` is replaced with the space: `mocha:alert('attack%20success')` → `mocha:alert(\'attack success'\)`

Single Quotes: `''`

When the **name-pattern** or **value-pattern** contain single quotes, it will be automatically escaped.

For example:

You may enter the parameter **value-pattern** as: `alert\('attack%20success'\)`

The **value-pattern** will appear as: `alert\\(\'attack success\'\\)`

config security waf heuristic-sql-xss-injection-detection

Use this command to configure SQL injection and cross-site scripting (XSS) detection policies.

In many cases, you can use predefined policies, and you do not need to create them. Table 14 describes the predefined policies.

Predefined SQL injection and XSS detection policies

	SQL Injection			XSS		
Predefined Rules	**Detection**	**Action**	**Severity**	**Detection**	**Action**	**Severity**
High-Level-Security	All except Body SQL Injection Detection	Deny	High	All except Body XSS Injection Detection	Deny	High
Medium-Level-Security	Only SQL URI SQL Injection Detection	Deny	High	None	Alert	Low
Alert-Only	Only SQL URI SQL Injection Detection	Alert	High	None	Alert	Low

The configurations for these policies are shown in the examples that follow. If desired, you can create user-defined policies.

Before you begin:

- You must have read-write permission for security settings.

After you have created an SQL injection/XSS policy, you can specify it in a WAF profile configuration.

Syntax

```
config security waf heuristic-sql-xss-injection-detection
   edit <name>
   set exception <datasource>
   set sql-injection-detection {enable|disable}
   set sql-injection-detection-exception <datasource>
   set sql-injection-action {datasource}
   set sql-injection-severity {high|medium|low}
   set uri-sql-injection-detection {enable|disable}
   set referer-sql-injection-detection {enable|disable}
   set cookie-sql-injection-detection {enable|disable}
   set body-sql-injection-detection {enable|disable}
   set xss-detection {enable|disable}
   set xss-exception <datasource>
```

```
    set xss-action {datasource}
    set xss-severity {high|medium|low}
    set uri-xss-detection {enable|disable}
    set referer-xss-detection {enable|disable}
    set cookie-xss-detection {enable|disable}
    set body-xss-detection {enable|disable}
    next
end
```

exception	Specify an exception configuration object for all modules.
sql-injection-detection	Enable/disable SQL injection detection.
sql-injection-detection-exception	Specify an exception configuration object for the SQL module.
sql-injection-action	Specify a WAF action object.
sql-injection-severity	• high • medium • low
uri-sql-injection-detection	Enable/disable detection in the HTTP request.
referer-sql-injection-detection	Enable/disable detection in the Referer header.
cookie-sql-injection-detection	Enable/disable detection in the Cookie header.
body-sql-injection-detection	Enable/disable detection in the HTTP Body message.
xss-detection	Enable/disable XSS detection.
xss-exception	Specify an exception configuration object for the XSS module.
xss-action	Specify a WAF action object.
xss-severity	• high • medium • low
uri-xss-injection-detection	Enable/disable detection in the HTTP request.
referer-xss-injection-detection	Enable/disable detection in the Referer header.
cookie-xss-injection-detection	Enable/disable detection in the Cookie header.
body-xss-injection-detection	Enable/disable detection in the HTTP Body message.

Example

```
FortiADC-docs # get security waf heuristic-sql-xss-injection-detection High-Level-Security
sql-injection-detection : enable
sql-injection-action : deny
sql-injection-severity : high
uri-sql-injection-detection : enable
referer-sql-injection-detection: enable
cookie-sql-injection-detection: enable
body-sql-injection-detection : disable
xss-detection : enable
xss-action : deny
```

```
xss-severity : high
uri-xss-detection : enable
referer-xss-detection : enable
cookie-xss-detection : enable
body-xss-detection : disable
sql-injection-detection-exception:
xss-exception :
exception :

FortiADC-docs # get security waf heuristic-sql-xss-injection-detection Medium-Level-Security

sql-injection-detection : enable
sql-injection-action : deny
sql-injection-severity : high
uri-sql-injection-detection : enable
referer-sql-injection-detection: disable
cookie-sql-injection-detection: disable
body-sql-injection-detection : disable
xss-detection : disable
xss-action : alert
xss-severity : low
sql-injection-detection-exception:
exception :

FortiADC-docs # get security waf heuristic-sql-xss-injection-detection Alert-Only
sql-injection-detection : enable
sql-injection-action : alert
sql-injection-severity : high
uri-sql-injection-detection : enable
referer-sql-injection-detection: disable
cookie-sql-injection-detection: disable
body-sql-injection-detection : disable
xss-detection : disable
xss-action : alert
xss-severity : low
sql-injection-detection-exception:
exception :
```

config security waf http-protocol-constraint

Use this command to configure HTTP protocol checks: HTTP request parameter lengths, HTTP request method, and HTTP response code.

Table 15 describes the three predefined policies.

Predefined HTTP protocol constraint policies

Predefined Rules	Description
High-Level-Security	Maximum URI length is 2048 characters. Action is set to deny. Severity is set to high.
Medium-Level-Security	Maximum URI length is 2048 characters. Action is set to alert. Severity is set to medium.
Alert-Only	Maximum URI length is 2048 characters. Action is set to alert. Severity is set to low.

The configurations for these rules are shown in the examples that follow. If desired, you can create user-defined rules to filter traffic with invalid HTTP request methods or drop packets with the specified server response codes.

Before you begin:

- You must have read-write permission for security settings.

After you have created an HTTP protocol constraint policy, you can specify it in a WAF profile configuration.

Syntax

```
config security waf http-protocol-constraint
   edit <name>
      set exception <datasource>
      set illegal-host-name-check {enable|disable}
      set illegal-host-name-check-action {datasource}
      set illegal-host-name-check-severity {high|medium|low}
      set illegal-http-version-check {enable|disable}
      set illegal-http-version-check-action {datasource}
      set illegal-http-version-check-severity {high|medium|low}
      set max-cookie-number-in-request <integer>
      set max-cookie-number-in-request-action {datasource}
      set max-cookie-number-in-request-severity {high|medium|low}
      set max-header-number-in-request <integer>
      set max-header-number-in-request-action {datasource}
      set max-header-number-in-request-severity {high|medium|low}
      set max-request-body-length <integer>
      set max-request-body-length-action {datasource}
      set max-request-body-length-severity {high|medium|low}
      set max-request-header-length <integer>
      set max-request-header-length-action {datasource}
      set max-request-header-length-severity {high|medium|low}
      set max-request-header-name-length <integer>
      set max-request-header-name-length-action {datasource}
      set max-request-header-name-length-severity {high|medium|low}
      set max-request-header-value-length <integer>
      set max-request-header-value-length-action {datasource}
      set max-request-header-value-length-severity {high|medium|low}
      set max-uri-length <integer>
      set max-uri-length-action {datasource}
      set max-uri-length-severity {high|medium|low}
      set max-url-parameter-name-length <integer>
      set max-url-parameter-name-length-action {datasource}
      set max-url-parameter-name-length-severity {high|medium|low}
      set max-url-parameter-value-length <integer>
      set max-url-parameter-value-length-action {datasource}
      set max-url-parameter-value-length-severity {high|medium|low}
      config request-method-rule
         edit <No.>
            set exception <datasource>
            set action {datasource}
            set severity {high|medium|low}
            set method {CONNECT DELETE GET HEAD OPTIONS OTHERS POST PUT TRACE }
         next
      end
      config response-code-rule
```

```
        edit <No.>
           set exception <datasource>
           set action {datasource}
           set severity {high|medium|low}
           set code-min <400-599>
           set code-max <400-599>
        next
     end
  next
end
```

exception	Specify an exception configuration object.
illegal-host-name-check	Enable/disable hostname checks. A domain name must consist of only the ASCII alphabetic and numeric characters, plus the hyphen. The hostname is checked against the set of characters allowed by the RFC 2616. Disallowed characters, such as non-printable ASCII characters or other special characters (for example, '<', '>', and the like), are a symptom of an attack.
illegal-host-name-check-action	Specify a WAF action object.
illegal-host-name-check-severity	• high • medium • low
illegal-http-version-check	Enable/disable the HTTP version check. Well-formed requests include the version of the protocol used by the client, in the form of HTTP/v where v is replaced by the actual version number (one of 0.9, 1.0, 1.1). Malformed requests are a sign of traffic that was not sent from a normal browser and are a symptom of an attack.
illegal-http-version-check-action	Specify a WAF action object.
illegal-http-version-check-severity	• high • medium • low
max-cookie-number-in-request	Maximum number of cookie headers in an HTTP request. The default is 16. The valid range is 1-32.
max-cookie-number-in-request-action	Specify a WAF action object.
max-cookie-number-in-request-severity	• high • medium • low
max-header-number-in-request	Maximum number of headers in an HTTP request. The default is 50. Requests with more headers are a symptom of a buffer overflow attack or an attempt to evade detection mechanisms. The valid configuration range is 1-100.
max-header-number-in-request-action	Specify a WAF action object.

max-header-number-in-request-severity	• high • medium • low
max-request-body-length	Maximum length of the HTTP body. The default is 67108864. The valid range is 1-67108864.
max-request-body-length-action	Specify a WAF action object.
max-request-body-length-severity	• high • medium • low
max-request-header-length	Maximum length of the HTTP request header. The default is 8192. The valid range is 1-16384.
max-request-header-action	Specify a WAF action object.
max-request-header-severity	• high • medium • low
max-request-header-name-length	Maximum characters in an HTTP request header name. The default is 1024. The valid range is 1-8192.
max-request-header-name-length-action	Specify a WAF action object.
max-request-header-name-length-severity	• high • medium • low
max-request-header-value-length	Maximum characters in an HTTP request header value. The default is 4096. Longer headers might be a symptom of a buffer overflow attack. The valid configuration range is 1-8192.
max-request-header-value-length-action	Specify a WAF action object.
max-request-header-value-length-severity	• high • medium • low
max-uri-length	Maximum characters in an HTTP request URI. The default is 2048. The valid range is 1-8192.
max-uri-length-action	Specify a WAF action object.
max-uri-length-severity	• high • medium • low
max-url-parameter-name-length	Maximum characters in a URL parameter name. The default is 1024. The valid range is 1-2048.

max-url-parameter-name-length-action	Specify a WAF action object.
max-url-parameter-name-length-severity	• high • medium • low
max-url-parameter-value-length	Maximum characters in a URL parameter value. The default is 4096. The valid range is 1-8192.
max-url-parameter-value-length-action	Specify a WAF action object.
max-url-parameter-value-length-severity	• high • medium • low
config request-method-rule	
exception	Specify an exception configuration object.
action	Specify a WAF action object.
severity	• high • medium • low
method	Specify a space-separated list of methods to match in the HTTP request line: • CONNECT • DELETE • GET • HEAD • OPTIONS • POST • PUT • TRACE • Others **Note**: The first 8 methods are described in RFC 2616. Others contains not commonly used HTTP methods defined by Web Distributed Authoring and Version (WebDAV) extensions.
config response-code-rule	
exception	Specify an exception configuration object.
action	Specify a WAF action object.
severity	• high • medium • low
code-min	Start of the range.
code-max	End of the range.

Example

```
FortiADC-docs # get security waf http-protocol-constraint High-Level-Security
max-uri-length : 2048
max-uri-length-action : deny
max-uri-length-severity : high
max-request-header-name-length: 1024
max-request-header-name-length-action: deny
max-request-header-name-length-severity: high
max-request-header-value-length: 4096
max-request-header-value-length-action: deny
max-request-header-value-length-severity: high
max-url-parameter-name-length : 1024
max-url-parameter-name-length-action: deny
max-url-parameter-name-length-severity: high
max-url-parameter-value-length: 4096
max-url-parameter-value-length-action: deny
max-url-parameter-value-length-severity: high
illegal-http-version-check : enable
illegal-http-version-check-action: deny
illegal-http-version-check-severity: high
illegal-host-name-check : enable
illegal-host-name-check-action: deny
illegal-host-name-check-severity: high
max-cookie-number-in-request : 16
max-cookie-number-in-request-action: deny
max-cookie-number-in-request-severity: high
max-header-number-in-request : 50
max-header-number-in-request-action: deny
max-header-number-in-request-severity: high
max-request-header-length : 8192
max-request-header-length-action: deny
max-request-header-length-severity: high
max-request-body-length : 67108864
max-request-body-length-action: deny
max-request-body-length-severity: high
exception :

FortiADC-docs # get security waf http-protocol-constraint Medium-Level-Security
max-uri-length : 2048
max-uri-length-action : alert
max-uri-length-severity : medium
max-request-header-name-length: 1024
max-request-header-name-length-action: alert
max-request-header-name-length-severity: medium
max-request-header-value-length: 4096
max-request-header-value-length-action: alert
max-request-header-value-length-severity: medium
max-url-parameter-name-length : 1024
max-url-parameter-name-length-action: alert
max-url-parameter-name-length-severity: medium
max-url-parameter-value-length: 4096
max-url-parameter-value-length-action: alert
max-url-parameter-value-length-severity: medium
illegal-http-version-check : enable
illegal-http-version-check-action: alert
illegal-http-version-check-severity: medium
```

```
illegal-host-name-check : enable
illegal-host-name-check-action: alert
illegal-host-name-check-severity: medium
max-cookie-number-in-request : 16
max-cookie-number-in-request-action: alert
max-cookie-number-in-request-severity: medium
max-header-number-in-request : 50
max-header-number-in-request-action: alert
max-header-number-in-request-severity: medium
max-request-header-length : 8192
max-request-header-length-action: alert
max-request-header-length-severity: medium
max-request-body-length : 67108864
max-request-body-length-action: alert
max-request-body-length-severity: medium
exception :

FortiADC-docs # get security waf http-protocol-constraint Alert-Only
max-uri-length : 2048
max-uri-length-action : alert
max-uri-length-severity : low
max-request-header-name-length: 1024
max-request-header-name-length-action: alert
max-request-header-name-length-severity: low
max-request-header-value-length: 4096
max-request-header-value-length-action: alert
max-request-header-value-length-severity: low
max-url-parameter-name-length : 1024
max-url-parameter-name-length-action: alert
max-url-parameter-name-length-severity: low
max-url-parameter-value-length: 4096
max-url-parameter-value-length-action: alert
max-url-parameter-value-length-severity: low
illegal-http-version-check : enable
illegal-http-version-check-action: alert
illegal-http-version-check-severity: low
illegal-host-name-check : enable
illegal-host-name-check-action: alert
illegal-host-name-check-severity: low
max-cookie-number-in-request : 16
max-cookie-number-in-request-action: alert
max-cookie-number-in-request-severity: low
max-header-number-in-request : 50
max-header-number-in-request-action: alert
max-header-number-in-request-severity: low
max-request-header-length : 8192
max-request-header-length-action: alert
max-request-header-length-severity: low
max-request-body-length : 67108864
max-request-body-length-action: alert
max-request-body-length-severity: low
exception :
```

config security waf http-header-security

HTTP response security headers are a set of standard HTTP response headers proposed to prevent or mitigate known XSS, clickjacking, and MIME sniffing security vulnerabilities. These response headers define security policies to client browsers so that the browsers avoid exposure to known vulnerabilities when handling requests.

When the HTTP Security Headers feature is enabled, headers with specified values are inserted into HTTP responses coming from the backend web servers. This is a quick and simple solution to address the security vulnerabilities on user's website without code and configuration changes.

Syntax

```
config security waf http-header-security
   edit <hhs-profile-name>
      set request-status [ enable|disable ]
      set request-url [ URL-string]
      set mode [ add-always | add-replace | add-if-absent ]
      configure http-header-security-list
      edit <name>
         set name [ content-security-policy | x-content-type-options | x-frame-options | x-
              xssprotection
         | http-strict-transport-security ]
         set value [ nosniff | deny | sameorigin | sanitizing-mode | block-mode ]
         set policy <string>
         set report-only [ enable | disable ]
         set max-age <seconds>
         set include-subdomain [ enable | disable ]
         set preload [ enable | disable ]
         next
      end
   end
   config security waf profile
   edit <waf-profile-name>
      set http-header-profile <hhp-profile-name>
end
```

CLI Parameter	Description
request-status	Enable/disable request URL match. • enable — Responses to the request will be processed with the security headers only if the URL of a request matches the specified request URL. • disable — All responses will be processed with the selected security header (s).
request-url	The **request-url** option is available if **request-status** is **enabled**. Specify the URL used to match requests so that security headers can be applied to responses of the matched requests.
mode	Specify header operation mode for the response from the back-end server(s). • add-always — always add the specified header(s). • add-replace —add the specified header(s) if not exist, replace the value of

CLI Parameter	Description
	header(s) which exist already. • add-if-absent — only add the specified header(s) if not exist, do nothing if the same header(s) exist
http-header-security-list	
name	Set the HTTP security header name
value	The directive for the specified header in name. X content type options: nosniff X frame options: deny, sameorigin X XSS protection: sanitizing-mode, block-mode
policy	Only valid if Content-Security-Policy is selected. Enter the header value(s) that setting restrictions on resource types and sources. For example, default-src 'self';script-src 'self';object-src 'self'.
report-only	Enabling report-only switches to “Content-Security-Policy-Report-Only” header, which accepts all directives of CSP. However, “report-only” header only monitors the violations. FortiADC will check the existing of “report-uri” directive once “report-only” selected.
max-age	The time, in seconds, that the browser should remember that a site is only to be accessed using HTTPS. A max-age value of zero (i.e., “max-age=0”) signals the UA cease regarding the host as a Known HSTS Host, including the includeSubDomains directive (if asserted for that HSTS Host).
include-subdomain	Optional. If enabled, rule will apply to all of the site's subdomains as well.
preload	Google maintains an HSTS preload service: https://hstspreload.org/. By following the guidelines and successfully submitting your domain, browsers will never connect to your domain using an insecure connection. While the service is hosted by Google, all browsers have stated an intent to use (or actually started using) the preload list. Most major browsers (Chrome, Firefox, Opera, Safari, IE 11 and Edge) also have HSTS preload lists based on the Chrome list. (See the HSTS compatibility matrix.) However, it is not part of the HSTS specification and should not be treated as official.

Security Header	Description
content security policy	A content security policy (CSP), is an additional layer of security delivered via an HTTP header. This policy helps prevent attacks such as Cross Site Scripting (XSS) and other code injection attacks by defining content sources which are approved thus allowing the browser to load them. Without a CSP, the browser simply loads all files on a page without considering the source which could be harmful. This puts both the site and it’s visitors at risk of malicious activity. There are multiple directives available to website owners who want to implement a content security policy. A server may also define multiple directives within a CSP security header.

Security Header	Description
	For a detailed list of examples and references, visit content-security-policy.com. Additionally, you can use a tool called cspisawesome.com to easily create a CSP specific to your needs. FortiADC also provides a “report-only” flag to switch to “Content-Security-Policy-Report-Only” header, which accepts all directives of CSP, but the difference is that “report-only” header only monitor the violations. FortiADC will check the existing of “report-uri” directive once “report-only” selected.
X content type options	The X-Content-Type-Options response HTTP header is a marker used by the server to indicate that the MIME types advertised in the Content-Type headers should not be changed and be followed. This helps reduce the danger of drive-by downloads and helps treat the content the proper way. There is only one directive that can be used, which is nosniff. An example of the header looks like: `x-content-type-options: nosniff`

X frame options	The x-frame-options header provides clickjacking protection by not allowing iframes to load on your website. It is supported by IE 8+, Chrome 4.1+, Firefox 3.6.9+, Opera 10.5+, Safari 4+.
	There are three directives available for this header: deny, sameorigin and allow-from. But “allow-from” is obsolete and no longer works in modern browers, FortiADC will notsupport it.
	On FortiADC, there are two directive options: deny and sameorigin.
	Once “deny” selected, the header looks like:
	`x-frame-options: DENY`
	Once “sameorigin” selected, the header looks like:
	`x-frame-options: SAMEORIGIN`

X XSS protection	The x-xss-protection header is designed to enable the cross-site scripting (XSS) filter built into modern web browsers. This is usually enabled by default, but using it will enforce it.
	Although these protections are largely unnecessary in modern browsers when sites implement a strong Content-Security-Policy that disables the use of inline JavaScript ('unsafe-inline'), they can still provide protections for users of older web browsers that don't yet support CSP.
	On FortiADC, this function has two modes to be choose: sanitizing-mode and block-mode. Once sanitizing-mode selected (usually default in browsers), an example looks like:
	`x-xss-protection: 1`
	Once block-mode selected, an example looks like:
	`x-xss-protection: 1; mode=block`

HTTP strict transport security	• The HTTP strict-transport-security (HSTS) header is a security enhancement that restricts web browsers to access web servers solely over HTTPS. This ensures the connection cannot be establish through an insecure HTTP connection, would helps to protect websites against protocol downgrade

Security Header	Description
	attacks and cookie hijacking. There are three directives for this header: max-age=<expire-time> • includeSubDomains • preload An example looks like: `strict-transport-security: max-age=31536000; includeSubDomains; preload`

config security waf action

Use this command to configure web application firewall (WAF) actions. A WAF action is referenced by the WAF policies to define which action will be taken when policies detect attacks.

In many cases, you can use predefined profiles to get started.

Predefined actions	Description
alert	WAF policies will allow the traffic to pass and log the event.
block	WAF policies will drop the current attack session by HTTP 403 message and block the attacker (according the attacker's IP address) for 1 hour, and log the event.
captcha	WAF policies will allow the traffic to pass if the client successfully fulfills the CAPTCHA request, and log the event.
deny	WAF policies will the drop current attack session by HTTP 403 message, and log the event.
silent-deny	WAF policies will drop the current attack session by HTTP 403 message, without logging the event.

The configurations for these actions are shown in the examples that follow. If desired, you can create user-defined actions.

Before you begin:

- Usually, predefined actions are enough for normal usage, and most predefined WAF policies reference the predefined actions. After you define your own action, you must specify it in your WAF policies for it to take effect.
- You must have read-write permission for security settings.

Syntax

```
config security waf action
   edit <name>
      set type {deny|pass|period-block|redirect|captcha}
      set log {enable|disable}
```

```
    set deny-code {200|202|204|205|400|403|404|405|406|408|410|500|501|502|503|504}
    set block-period <integer>
    set redirect-url <string>
    set comment <string>
  next
end
```

type	Specify action type from the following: • deny — Blocks the request. This will drop the current session by a HTTP error message. • pass — Allows the request. The current session will be allowed to continue. • period-block — Denies all HTTP requests from a source IP within a period which is specified in the **block-period**. This will drop the current session by a HTTP error message and block the client for a period. • redirect — Sends a redirect. This will drop the current session by a HTTP 302 redirect message and allow the client to redirect to another URL. • captcha — Requires the client to successfully fulfill the CAPTCHA request. The current session will be allowed to continue after the client successfully fulfills the CAPTCHA request.
log	Enable/disable to log the event.
deny-code	The **deny-code** option is available if the **type** is **deny** or **period-block**. Specify HTTP error message code when the action drops the current session. Default: 403. 200, 202, 204, 205, 400, 403, 404, 405, 406, 408, 410, 500, 501, 502, 503, 504
block-period	The **block-period** option is available if the **type** is **period-block**. Specify a time period when action blocks the client. Default: 60 seconds, Range: 1- 3600 seconds.
redirect-url	The **redirect-url** option is available if the **type** is **redirect**. Specify a URL when the action performs a HTTP redirect.

Example

```
FortiADC-docs # get security waf action
== [ alert ]
== [ deny ]
== [ block ]
== [ silent-deny ]
FortiADC-docs # get security waf action alert
type : pass
log : enable
comment :
FortiADC-docs # get security waf action deny
type : deny
log : enable
deny-code : 403
comment :
FortiADC-docs # get security waf action block
type : period-block
log : enable
```

```
deny-code : 403
block-period : 3600
comment :
FortiADC-docs # get security waf action silent-deny
type : deny
log : disable
deny-code : 403
comment :
FortiADC-docs # config security waf action
FortiADC-docs (action) # edit eval
FortiADC-docs (eval) # get
type : deny
log : enable
deny-code : 403
comment : comments
FortiADC-docs (eval) # set type period-block
FortiADC-docs (eval) # set deny-code 200
FortiADC-docs (eval) # set block-period 30
FortiADC-docs (eval) # set log disable
FortiADC-docs (eval) # end
```

config security waf profile

Use this command to configure web application firewall (WAF) profiles. A WAF profile references the WAF policies that are to be enforced.

In many cases, you can use predefined profiles to get started. Table 16 describes the three predefined policies.

Predefined WAF profiles

Predefined Rules	Description
High-Level-Security	HTTP protocol constraints policy: High-Level-Security SQL injection and XSS detection policy: High-Level-Security
Medium-Level-Security	HTTP protocol constraints policy: Medium-Level-Security SQL injection and XSS detection policy: Medium-Level-Security
Alert-Only	HTTP protocol constraints policy: Alert-Only SQL injection and XSS detection policy: Alert-Only

The configurations for these profiles are shown in the examples that follow. If desired, you can create user-defined profiles.

Before you begin:

- You can use predefined WAF profiles, create profiles based on predefined feature options, or create profiles based on user-defined configuration objects. If you want to add user-defined configuration objects, you must create them before using this command to add them to a WAF profile.
- You must have read-write permission for security settings.

After you have created a WAF profile, you can specify it in a virtual server configuration.

Syntax

```
config security waf profile
   edit <name>
      set advanced-bot-protection <datasource>
      set advanced-protection <datasource>
      set api-discovery <datasource>
      set api-gateway <datasource>
      set biometrics-based-detection <datasource>
      set body-decode-length <integer>
      set body-decode-type {xml|html|json}
      set bot-detection <datasource>
      set brute-force-login <datasource>
      set cookie-security <datasource>
      set cors-protection <datasource>
      set csrf-protection <datasource>
      set data-leak-prevention <datasource>
      set description <string>
      set exception <datasource>
      set fingerprint-based-detection <datasource>
      set heuristic-sql-xss-injection-detection <datasource>
      set http-header-cache {enable|disable}
      set http-protocol-constraint <datasource>
      set input-validation-policy <datasource>
      set json-validation <datasource>
      set multiple-decode-loop <integer>
      set openapi-validation <datasource>
      set rule-match-record {enable|disable}
      set threshold-based-detection <datasource>
      set url-protection <datasource>
      set web-attack-signature <datasource>
      set xml-validation <datasource>
   next
end
```

`advanced-bot-protection`	Specify a user-defined configuration object.
`advanced-protection`	Specify a user-defined configuration object.
`api-discovery`	Specify a user-defined configuration object.
`api-gateway`	Specify a user-defined configuration object.
`biometrics-based-detection`	Specify a user-defined configuration object.
`body-decode-length`	Specify a body decode length in byte. (Range: 0 - 4194304 B, default: 1024 B).
`body-decode-type`	Specify the body decode type. **Note**: This only applies when the corresponding validation function is enabled.
`bot-detection`	Specify a user-defined configuration object.
`brute-force-login`	Specify a user-defined configuration object.
`cookie-security`	Specify a user-defined configuration object.

cors-protection	Specify a predefined or user-defined configuration object.
csrf-protection	Specify a user-defined configuration object.
data-leak-prevention	Specify a user-defined configuration object.
description	A string to describe the purpose of the configuration, to help you and other administrators more easily identify its use.
exception	Specify an exception configuration object.
fingerprint-based-detection	Specify a user-defined configuration object.
heuristic-sql-xss-injection-detection	Specify a predefined or user-defined configuration object.
http-protocol-constraint	Specify a predefined or user-defined configuration object.
http-header-cache	Enable/disable caching HTTP headers. Enabled by default. If you experience performance issues, you can disable. However, the cached HTTP headers are used to populate fields in logs resulting from HTTP body scanning. Can only be set with the CLI.
input-validation-policy	Specify a predefined or user-defined configuration object.
json-validation	Specify a predefined or user-defined configuration object.
multiple-decode-loop	Specify the number of times for the multiple decode loop. (Range: 0 - 16, default: 6).
openapi-validation	Specify a predefined or user-defined configuration object.
rule-match-record	Enable to allow the Security Log to display the part of the rule that is matched when the security event is logged. This is disabled by default.
threshold-based-detection	Specify a predefined or user-defined configuration object.
url-protection	Specify a predefined or user-defined configuration object.
web-attack-signature	Specify a predefined or user-defined configuration object.
xml-validation	Specify a predefined or user-defined configuration object.

Example

```
FortiADC-docs # get security waf profile High-Level-Security
web-attack-signature : High-Level-Security
url-protection :
http-protocol-constraint : High-Level-Security
heuristic-sql-xss-injection-detect: High-Level-Security
description :
http-header-cache : enable
exception :

FortiADC-docs # get security waf profile Medium-Level-Security
web-attack-signature : Medium-Level-Security
url-protection :
http-protocol-constraint : Medium-Level-Security
```

```
heuristic-sql-xss-injection-detect: Medium-Level-Security
description :
http-header-cache : enable
exception :


FortiADC-docs # get security waf profile Alert-Only
web-attack-signature : Alert-Only
threshold-based-detection :
url-protection :
http-protocol-constraint : Alert-Only
heuristic-sql-xss-injection-detect : Alert-Only
description :
http-header-cache : enable
exception :

FortiADC-docs # config security waf profile
FortiADC-docs (profile) # edit eval
Add new entry 'eval' for node 3000
FortiADC-docs (eval) # get
web-attack-signature :
url-protection :
http-protocol-constraint :
heuristic-sql-xss-injection-detect:
bot-detection :
biometrics-based-detection :
fingerprint-based-detection :
advanced-bot-protection :
description :
http-header-cache : enable
exception :

FortiADC-docs (eval) # set web-attack-signature Alert-Only
FortiADC-docs (eval) # set http-protocol-constraint Alert-Only
FortiADC-docs (eval) # set heuristic-sql-xss-injection-detect Alert-Only
FortiADC-docs (eval) # set exception exception-group
FortiADC-docs (eval) # set description "evaluate alert-only and exception list"
FortiADC-docs (eval-alert-onl~-) # end
```

config security waf input-validation-policy

Input validation can prevent suspicious HTTP requests. This command allows multiple rules for validations under one validation policy.

Syntax

```
configure security waf input-validation-policy
   edit <name>
      config file-restiction-rule-list
         edit <No.>
            set rule <datasource>
```

```
        next
      end
      config hidden-field-rule-list
        edit <No.>
          set rule <datasource>
        next
      end
      config parameter-validation-rule-list
        edit <No.>
          set rule <datasource>
        next
      end
    next
end
```

file-restriction-rule-list	multiple html file restriction rules
hidden-field-rule-list	multiple html hidden field rules
parameter-validation-rule-list	multiple html parameter validation rules
rule	rule for associated html rule list

Example

```
FortiADC-VM # show security waf input-validation-policy
config security waf input-validation-policy
edit "1"
config parameter-validation-rule-list
edit 1
set rule 1
next
end
config hidden-field-rule-list
end
config file-restriction-rule-list
end
next
end
```

config security waf parameter-validation-rule

The command is used to validate input fields in html forms, and supports data type check in parameter validation.

Syntax

```
config security waf parameter-validation-rule
  edit <name>
    ...
    config parameter_validation_rule_element
      edit <name>
```

```
            set is-type-check {enable | disable}
            set max-length <integer>
            set name <string>
         next
      end
   next
end
```

is-type-check	Enable or disable type check
integer	Maximum character length of value
name	name value

Example

```
FortiADC-VM # show security waf parameter-validation-rule
config security waf parameter-validation-rule
edit "1"
set request-url /
set action deny
config parameter_validation_rule_element
edit 1
set name username
set is-type-check enable
set data-type Email
next
end
next
end
```

config security waf url-protection

Use this command to configure URL protection policies. URL protection policies can filter HTTP requests that match specific character strings and file extensions.

Before you begin:

- You must have read-write permission for security settings.

After you have created a URL protection policy, you can specify it in a WAF profile configuration.

Syntax

```
config security waf url-protection
   edit <name>
      set exception <datasource>
      config url-access-rule
         edit <No.>
            set exception <datasource>
            set action {datasource}
            set severity {high|medium|low}
```

```
            set url-pattern <url-pattern>
         next
      end
      config file-extension-rule
         edit <No.>
            set exception <datasource>
            set action {datasource}
            set severity {high|medium|low}
            set file-extension-pattern <file-extension-pattern>
         next
      end
   next
end
```

exception	Specify an exception configuration object.
action	Specify a WAF action object.
severity	• high • medium • low
url-pattern	Matching string. Regular expressions are supported.
file-extension-pattern	Matching string. Regular expressions are supported.

Example

```
FortiADC-docs # config security waf url-protection
FortiADC-docs (url-protection) # edit url-policy
Add new entry 'url-policy' for node 3050
FortiADC-docs (url-policy) # config url-access-rule
FortiADC-docs (url-access-rule) # edit 1
Add new entry '1' for node 3052
FortiADC-docs (1) # get
url-pattern :
action : alert
severity : low
exception :
FortiADC-docs (1) # set url-pattern tmp
FortiADC-docs (1) # end
FortiADC-docs (url-policy) # config file-extension-rule
FortiADC-docs (file-extension~r) # edit 1
Add new entry '1' for node 3057
FortiADC-docs (1) # get
file-extension-pattern :
action : alert
severity : low
exception :
FortiADC-docs (1) # set file-extension-pattern tmp
FortiADC-docs (1) # end
FortiADC-docs (url-policy) # end
```

config security waf web-attack-signature

Use this command to configure web attack signature policies. The attack signature policy includes rules to enable scanning of HTTP headers and HTTP body content in HTTP requests, HTTP responses, or both.

Table 17 describes the predefined policies. You can select the predefined policies in your WAF profiles, or you can create policies that enable a different set of scan classes or a different action. In this release, you cannot exclude individual signatures or create custom signatures. You can enable or disable the scan classes.

Web Attack Signature predefined policies

Policy	Status	Action
High-Level-Security	• Scan HTTP header—Enabled. • Scan HTTP Request Body—Enabled. • Scan HTTP Response Body—Disabled.	• High Severity Action—Deny. • Medium Severity Action—Deny. • Low Severity Action—Alert.
Medium-Level-Security	• Scan HTTP header—Enabled. • Scan HTTP Request Body—Enabled. • Scan HTTP Response Body—Disabled.	• High Severity Action—Deny. • Medium Severity Action—Alert. • Low Severity Action—Alert.
Alert-Only	• Scan HTTP header—Enabled. • Scan HTTP Request Body—Disabled. • Scan HTTP Response Body—Disabled.	• High Severity Action—Alert. • Medium Severity Action—Alert. • Low Severity Action—Alert.

Before you begin:

- You must have read-write permission for security settings.

After you have created a web attack signature policy, you can specify it in a WAF profile configuration.

Syntax

```
config security waf web-attack-signature
   edit <name>
      set exception <datasource>
      set high-severity-action {datasource}
      set request-body-detection {enable|disable}
      set response-body-detection {enable|disable}
      set medium-severity-action {datasource}
      set low-severity-action {datasource}
      config category
         edit <category-id>
            set action [ alert | deny | block | silent-deny ]
            set status [ enable | disable ]
      end
      config sub-category
         edit <sub-category-id>
            set status {enable|disable}
      next
      end
      config signature
         edit <datasource>
            set status
```

```
          set exception
        next
      end
    next
end
```

exception	Specify an exception configuration object.
request-body-detection	Enable/disable scanning against HTTP request body signatures.
response-body-detection	Enable/disable against HTTP response body signatures.
high-severity-action	Specify a WAF action object.
medium-severity-action	Specify a WAF action object.
low-severity-action	Specify a WAF action object.
config signature	
status	Enable/disable the signature.
exception	Specify an exception configuration object.
config category	
status	Enable/disable the category status.
action	Specify an action configuration object.
config sub-category	
status	Enable/disable the sub-category status.

Example

```
FortiADC-VM # get security waf web-attack-signature High-Level-Security
status : enable
request-body-detection : enable
response-body-detection : disable
high-severity-action : deny
medium-severity-action : deny
low-severity-action : alert
exception:

FortiADC-VM # get security waf web-attack-signature Medium-Level-Security
status : enable
request-body-detection : enable
response-body-detection : disable
high-severity-action : deny
medium-severity-action : alert
low-severity-action : alert
exception:

FortiADC-VM # get security waf web-attack-signature Alert-Only
status : enable
request-body-detection : disable
response-body-detection : disable
high-severity-action : alert
```

```
medium-severity-action : alert
low-severity-action : alert
exception:

FortiADC-docs # config security waf web-attack-signature
FortiADC-docs (web-attack-sig~a) # edit eval
FortiADC-docs (eval) # config signature
FortiADC-docs (signature) # edit 1002010728
FortiADC-docs (1002010728) # get
status : enable
description :
exception :
FortiADC-docs (1002010728) # set status disable
FortiADC-docs (1002010728) # set description "investigate false positive"
FortiADC-docs (1002010728) # end
FortiADC-docs (eval)# config category
FortiADC-docs (category)# edit 1
FortiADC-docs (1)# set action alert
FortiADC-docs (1)# set status enable
FortiADC-docs (1)# end
```

config security waf json-validation-detection

Use this command to set JSON validation detection.

Note: This command only checks HTTP requests with content-type being application/json.

Predefined WAF profiles

Predefined Rules	Required settings
High-Level-Security	format-checks — enable set xss-checks — enable set sql-injection-checks — enable severity —high action — deny
Medium-Level-Security	format-checks — enable set xss-checks — enable set sql-injection-checks — enable severity — medium action — alert
Alert-Only	format-checks — enable set xss-checks — disable set sql-injection-checks — disable severity — low action — alert

Syntax

```
config security waf json-validation-detection
   edit <name>
      set format-checks enable/disable
      set limit-checks enable/disable
      set limit-max-array-value-num <0-4096>
      set limit-max-depth-num <0-4096>
      set limit-max-object-num <0-4096>
      set limit-max-string-len <0-4096>
      set xss-checks enable/disable
      set sql-injection-checks enable/disable
      set exception <datasource>
      set severity low/medium/high
      set action <datasource>
      set schema-checks <enable/disable>
      set json-schema-id <datasource>
   next
end
```

`name`	Specify the name of the JSON detection profile.
`format-checks`	Enable or disable JSON format checks, which are security checks for incoming HTTP requests to determine whether they are well-formed. Note: If enabled, you must specify FortiADC response actions to malformed HTTP requests, as discussed below.
`limit-checks`	Enable or disable parsing limits to protect web servers from attacks, such as DDOS attacks. Note: If enabled, you must change the configuration for the following parameters: • Limit max array value • Limit max depth • Limit max object member • Limit max string
`limit-max-array-value-num`	Specify the maximum value within a single array. The default value is 256. Valid values range from 0 to 4,096. Note: This option is available only when JSON limit-checks is enabled.
`limit-max-depth-num`	Specify the maximum depth in a JSON value. The default value is 16. Valid values range from 0 to 4,096. Note: This option is available only when JSON limit-checks is enabled.
`limit-max-object-num`	Specify the maximum number of members in a JSON object. The default value is 64. Valid values range from 0 fro 4,096. Note: This option is available only when JSON limit-checks is enabled.
`limit-max-string-len`	Specify the maximum length of a string in a JSON request for a name or a value. The default value is 64. Valid values range from 0 to 4,096. Note: This option is available only when JSON limit-checks is enabled.

xss-checks	Enable to examine the bodies of incoming JSON requests that might indicate possible cross-site scripting attacks. Note: If the request contains a positive match, FortiADC will respond with the specified action, as discussed at the end of this table.
sql-injection-checks	Enable to examine the bodies of incoming requests for inappropriate SQL characters and keywords, which may indicate an SQL injection attack. Note: If the request contains a positive match, FortiADC will respond with the specified action, as discussed at the beginning of this table.
exception	*Optional.* Select the exception profile to be applied to the JSON detection profile.
severity	Set the severity level in WAF logs for potential attacks detected by the JSON detection profile by selecting one of the following: • High • Medium • Low
action	Specify the action that FortiADC will take upon detecting a potential attack: You can choose a WAF action object.
schema-checks	Enable or disable JSON schema validation detection. Note:Before enabling JSON schema checks, you must upload a JSON schema file to check whether JSON content is well-formed.
json-schema-id	Select the JSON schema file that you want to use.

Example

```
config security waf json-validation-detection
   edit "all"
      set format-checks enable
      set meta-os-checks disable
      set limit-checks enable
      set limit-max-array-value-num 1
      set limit-max-depth-num 0
      set limit-max-object-num 0
      set limit-max-string-len 0
      set xss-checks enable
      set sql-injection-checks enable
      unset exception
      set severity high
      set action alert
   next
end
```

config security waf json-schema file

Use this command to create an JSON-schema file which defines an JSON schema format for JSON validation detection.

Syntax

```
config security waf json-validation-detection
   edit <name>
```

config security waf xml-schema file

Use this command to create an XML-schema file which defines an XML schema format for XML validation detection.

Syntax

```
config security waf xml-validation-detection
   edit <name>
```

config security waf xml-validation-detection

Use this command to configure XML validation detection.

Note: This command only checks HTTP requests with content type being application/xml and text/xml.

Predefined WAF profiles

Predefined Rules	Required settings
High-Level-Security	format-checks — enable set soap-format-checks— disable set schema-checks — disable set xss-checks — enable set sql-injection-checks — enable severity —high action — deny
Medium-Level-Security	format-checks — enable set soap-format-checks— disable set schema-checks — disable set xss-checks — enable set sql-injection-checks — enable severity — mdeium

Predefined Rules	Required settings
	action — alert
Alert-Only	format-checks — enable set soap-format-checks— disable set schema-checks — disable set xss-checks — disable set sql-injection-checks — disable severity — low action — alert

Syntax

```
config security waf xml-validation-detection
   edit <name>
      set format-checks enable/disable
      set soap-format-checks enable/disable
      set wsdl-checks enable/disable
      set soap_wsdl_id <datasource>
      set schema-checks enable/disable
      set xml-schema-id <datasource>
      set limit-checks enable/disable
      set limit-max-attr-num <1-256>
      set limit-max-attr-name-len <1-2048>
      set limit-max-attr-value-len <1-2048>
      set limit-max-cdata-len <1-65535>
      set limit-max-elem-child-num <1-65535>
      set limit-max-elem-depth-num <1-65535>
      set limit-max-elem-name-len <1-65535>
      set limit-max-namespace-num <0-256>
      set limit-max-namespace-url-len <0-1024>
      set xss-checks enable/disable
      set sql-injection-checks enable/disable
      set exception <datasource>
      set severity low/medium/high
      set action <datasource>
   next
end
```

`name`	Specify the name of the XML detection profile.
`format-checks`	Enable or disable XML format detection.
`schema-checks`	Enable or disable XML schema validation detection. Note:Before enabling XML schema checks, you must upload an XML schema file to check whether XML content is well-formed.
`xml-schema-id`	Select the XML schema file that you want to use.
soap-format-checks	Enable or disable soap-format-checks.
wsdl-checks	Enable or disable WSDL validation detection.

	Note: Before enabling WSDL checks, you must upload an WSDL file to check whether SOAP content is well-formed.
soap_wsdl_id	Select the desired WSDL file.
`limit-checks`	Enable or disable XML limit checks. Note: If enabled, you must can configure the following parameters: • limit-max-attr-num • limit-max-attr-name-len • limit-max-attr-value-len • limit-max-cdata-len • limit-max-elem-child-num • limit-max-elem-depth-num • limit-max-elem-name-len • limit-max-namespace-num • limit-max-namespace-url-len
`limit-max-attr-num`	Specify the maximum number of attributes each individual element is allowed to have. The default value is 256. Valid values range from 1 to 256. Note: This option is available only when XML limit-checks is enabled.
`limit-max-attr-name-len`	Specify the maximum length of each attribute name. The default value is 128. Valid values range from 1 to 2,048. Note: This option is available only when XML limit-checks is enabled.
`limit-max-attr-value-len`	Specify the maximum length of each attribute value. The default value is 128. Valid values range from 1 to 2,048. Note: This option is available only when XML limit-checks is enabled.
`limit-max-cdata-len`	Specify the length of the Cdata for each element. The default value is 65,535. Valid values range from 1 to 65,535. Note: This option is available only when XML limit-checks is enabled.
`limit-max-elem-child-num`	Specify the maximum number of children each element is allowed, including other elements and character information. The default value is 65,535. Valid values range from 1 to 65,535. Note: This option is available only when XML limit-checks is enabled.
`limit-max-elem-depth-num`	Specify the maximum number of nested levels in each element. The default value is 256. Valid values range from 1 to 65,535. Note: This option is available only when XML limit-checks is enabled.
`limit-max-elem-name-len`	Specify the maximum length of the name of each element. The default value is 128. Valid values range from 1 to 65,535. Note: This option is available only when XML limit-checks is enabled.
`limit-max-namespace-num`	Specify the number of namespace declarations in the XML document. The default value is 16. Valid values range from 0 to 256. Note: This option is available only when XML limit-checks is enabled.

limit-max-namespace-url-len	Specify the URL length for each namespace declaration. The default value is 256. Valid values range from 0 to 1,024. Note: This option is available only when XML limit-checks is enabled.
xss-checks	Enable to examine the bodies of incoming XML requests that might indicate possible cross-site scripting attacks. Note: If the request contains a positive match, FortiADC will respond with the specified action, as discussed at the beginning of this table.
sql-injection-checks	Enable to examine the bodies of incoming requests for inappropriate SQL characters and keywords, which may indicate an SQL injection attack. Note: If the request contains a positive match, FortiADC will respond with the specified action, as discussed at the end of this table.
exception	*Optional.* Select the exception profile to be applied to the XML detection profile.
severity	Set the severity level in WAF logs for potential attacks detected by the XML detection profile by selecting one of the following: • High • Medium • Low
action	Specify the action that FortiADC will take upon detecting a potential attack. You can choose a WAF action object.

Example

```
config security waf xml-validation-detection
   edit "all"
      set format-checks enable
      set soap-format-checks enable
      set wsdl-checks enable
      unset soap_wsdl_id
      set schema-checks enable
      unset xml-schema-id
      set limit-checks enable
      set limit-max-attr-num 100
      set limit-max-attr-name-len 100
      set limit-max-attr-value-len 100
      set limit-max-cdata-len 1
      set limit-max-elem-child-num 100
      set limit-max-elem-depth-num 100
      set limit-max-elem-name-len 100
      set limit-max-namespace-num 1
      set limit-max-namespace-url-len 1
      set xss-checks enable
      set sql-injection-checks enable
      unset exception
      set severity medium
      set action alert
   next
end
```

config security waf openapi-schema-file

This command configures the OpenAPI validation.

Syntax

```
configure security waf openapi-schema-file
   edit <name>
      set openapi-schema-file <file>
      next
   end
end
```

`openapi-schema-file`	Specify an OpenAPI schema file. You must first upload an OpenAPI schema file with the command `execute openapi-schema import ftp/tftp <filename> <ip>`.

config security waf openapi-validation-detection

This command sets the OpenAPI validation policy.

Syntax

```
config security waf openapi-validation-detection
   edit <name>
      set action <datasource>
      set exception <datasource>
      set severity {low | medium | high}
      set schema-checks {enable | disable}
      set openapi-schema-id <datasource>
   next
end
```

`action`	Specify the action that FortiADC will take upon detecting a potential attack. The condition `format-checks` is enabled.
`severity`	Set the severity level in WAF logs for potential attacks detected by the OpenAPI detection profile by selecting high, medium, or low. The condition `format-checks` is enabled.
`exception`	Optional. Select the exception profile to be applied to the OpenAPI policy. The condition `format-checks` is enabled.
`schema-checks`	Enable or disable OpenAPI schema validation detection. The condition `format-checks` is enabled.
`openapi-schema-id`	Set OpenAPI schema files datasource

config security waf scanner

Web Application Vulnerability Scanner is a set of automated tools which perform black box test on web applications, to look for security vulnerabilities such as Cross-site scripting, SQL injection, command injection,source code disclosure and insecure server configuration.

To configure the web-vulnerability-scanner, you have to first configure, in the following order, (1) **target-login-option**, (2) **profile**, and (3) **task**. Then, with these completed, you can `config security waf scanner`.

Syntax

```
// first step login
config security web-vulnerability-scanner target-login-option
   edit <name>
   set type <none/basic/advanced>
      // if basic then:
      set username <string>
      set password <string>
   next
end

//optional
config security web-vulnerability-scanner exceptionlist
   edit <name>
      config domain_list
         edit <id>
         set pattern <string>
      next
      end
   next
end

// second step profile
config security web-vulnerability-scanner profile
   edit <name>
      set pool-name <datasource>
      set http-login-option <datasource>
      set mimes-scan <enable/disable>
      set files-scan <enable/disable>
      set messages-scan <enable/disable>
      set apps-scan <enable/disable>
      set context-scan <enable/disable>
      set crawl-limit <integer>
      set exceptionlist <datasource>
      set add_http_cookie <enable/disable>
      set cookie-name <string> //optional
      set cookie-value <string> //optional
      next
end

// third step task
config security web-vulnerability-scanner task
   edit <name>
      set scheduler <datasource>
```

```
      set profile <datasource>
      // uses previously constructed profile
   next
end

// last step: waf scanner
execute web-vulnerability-scan <start/stop> <datasource>
get security scan-report
// datasource uses previously created task
```

WVS Login configuration

Settings	Guidelines
Name	Specify a name for the configuration.
type	Select from two types: • None. Default. • Basic. Basic will tell you to specify a **username** and **password**. For the HTTP login type, you can choose simple HTTP credentials or HTTP-form for authentication. Default is none. • Advanced. HTTP POST HTML Form-based authentication in login. The user needs to provide the username, password, username field name and password field name to FortiADC.
username	Login username.
password	Login password.
username-field	Field name for the username (only for Advanced).
password-field	Field name of the password (only for Advanced).
extend_ parameter	Other parameters in html form to login (only for Advanced).
auth-url	Full url for authentication (only for Advanced).
auth-target-url	Full url to POST for authentication (only for Advanced).
verify-url	Full url to verify the authentication (only for Advanced).

WVS Profile configuration

Settings	Guidelines
Name	Specify a name for the configuration.
pool-name	Select a real sever from the real server pool.
http-login- option	Select an HTTP Login Option.

Settings	Guidelines
mime-scan	The **mime signature**s warn about server responses that have an interesting mime. For example anything that is presented as php-source will likely be interesting.
files-scan	The **files signature**s will use the content to determine if a response is an interesting file. For example, a SVN file.
messages-scan	The **messages signatures** look for interesting server messages. Most are based on errors, such as caused by incorrect SQL queries or PHP execution failures.
apps-scan	The **apps signatures** will help to find pages and applications who's functionality is a security risk by default. For example, phpinfo() pages that leak information or CMS admin interfaces.
context-scan	The **context signatures** are linked to injection tests. They look for strings that are relevant to the current injection test and help to highlight potential vulnerabilities.
crawl-limit	Specify a crawl limit.
exceptionlist	The exception list of scanner.
add_http_cookie	Add cookie to HTTP header.

WVS Task configuration

Settings	Guidelines
name	Specify a name for the configuration.
scheduler	Select a scheduler from the schedule group.
profile	Select a profile.

WAF scanner configuration

Settings	Guidelines
start	Start the web vulnerability task
stop	Stop the web vulnerability task
scan-report	Shows the report of the scans.

Example

```
// first step login
FortiADC-VM (root) # config security web-vulnerability-scanner target-login-option
FortiADC-VM (root) # edit 1
FortiADC-VM (1) # set type basic
FortiADC-VM (1) # set username me
FortiADC-VM (1) # set password dog
```

```
FortiADC-VM (1) # get
type : basic
username : me
password : *

next
end

// second step profile
FortiADC-VM (root) # config security web-vulnerability-scanner profile

FortiADC-VM (profile) # edit test
// Add new entry 'test'
FortiADC-VM (test) # set apps-scan enable
FortiADC-VM (test) # set pool-name Real_Server_Pool

FortiADC-VM (test) # set http-login-option 1
// uses previously created login-option


FortiADC-VM (test) # set crawl-limit 1500


FortiADC-VM (test) # get
pool-name : Real_Server_Pool
http-login-option : 1
mime-scan : disable
files-scan : enable
messages-scan : enable
apps-scan : enable
context-scan : enable
crawl-limit : 1500

next
end

// third step task
FortiADC-VM (root) # config security web-vulnerability-scanner task

FortiADC-VM (task) # edit task1
// Add new entry 'task1'
FortiADC-VM (task1) # get
scheduler :
profile :

FortiADC-VM (task1) # set scheduler 1
// comes from datasource
FortiADC-VM (task1) # profile test
// uses previously created profile
FortiADC-VM (task1) # get
scheduler : 1
profile : test

next
end

// last step: waf scanner
```

```
FortiADC-VM (root) # execute web-vulnerability-scan start
Usage: start/stop <taskname>
Command fail. Return code is -61 (Input is not as expected)

// need to name the task created earlier, 'task1'
FortiADC-VM (root) # execute web-vulnerability-scan start task1
FortiADC-VM (root) # get security scan-report
ID:0 Taskname:task1 Created Time:15:41:16,10-30-18

// example for advanced login

config security web-vulnerability-scanner target-login-option
edit "advanced"
set type advanced
set username username
set password password
set auth-url http://www.example.com
unset auth-target-url
unset verify-url
set username-field userfi
set password-field passfi
unset extend_parameter
next
end
```

config security waf brute-force-login

Brute Force Attack Detection policies can prevent too many login tests. If an HTTP client tries to log into a server via FortiADC and fails too many times, Brute Force Attack Detection policies can stop it.

Syntax

```
configure security waf brute-force-login
   edit <name>
      set description <string>
      set action <string>
config login-page-member
   edit 1
      set access-limit-ip <integer>
      set request-url <regular express string>
      set login-failed-code <HTTP status code>
      set host-status [ enable | disable ]
      set host <regular express string>
   next
end
next
end
```

CLI specification

CLI Parameter	Help message	Type	Scope	Default	Must
set description	HTTP connection limit	string		Null	No
action	the action when reach the limit	object		Null	Yes
access-limit-ip	Login failed times limit	integer	1-65535	1	No
request-url	Type the URL that the HTTP request must match to be included in the brute force login attack's rate calculations.	string	regular express	Null	Yes
login-failed-code	Response code which is used to judge if the login is failed or not.	integer	0-1000	0	No
host-status	Decides to match host name or not.	choice	enable disable	disable	No
host	Host name	string	regular express	Null	No

CLI Parameter	Visible condition	Special value	Effective condition
set description	always visible	N/A	Work through the WAF framework
action	always visible	N/A	
access-limit-ip	always visible	N/A	
request-url	always visible	N/A	
login-failed-code	always visible	0, means not match status code	
host-status	always visible	N/A	
host	always visible	host-status == enable	

Function description

CLI Parameter	Description
set description	Save description message.
action	Brute force attack protect action.
access-limit-ip	When the count of brute force attack reaches the limit, FortiADC will take action based on the source IP.
request-url	This URL is used to identify the login request. If login-failed-code is not set, it will be used to detect the login failed event.
login-failed-code	This code is used to identify the login failed event. If login-failed-code is not set, request-url and host will be used instead.

CLI Parameter	Description
host-status	Decides whether or not the Host field of the HTTP request will take part in the identification of the login request or login failed event together with request-url.
host	After matching url, FortiADC will match the Host.

Example

```
configure security waf brute-force-login
edit brute-login
set description "brute-force-login detection"
set action deny-action
config login-page-member
edit 1
set access-limit-ip 3
set request-url /login*
set login-failed-code 401
set host-status enable
set host www.xxx.com
next
end
edit 2
set access-limit-ip 5
set request-url /aaalogin*
next
end
next
end
```

WAF Profile

```
config security waf profile
edit <name>
set brute-force-login <name>
next
end
```

config security waf advanced-protection

Use this command to configure a security waf advanced-protection profile.

Syntax

```
config security waf profile
   edit<name>
   set advanced-protection <profile name>
end
```

```
config security waf advanced-protection
edit <name>
config advanced-protection-rule
end
end

config advanced-protection-rule
edit<ID>
set type content-scraping
set content-type <file-type-list>
set occurrence-limit <integer>
set occurrence-within <integer>
set percentage-match <integer>
set action <string>
set block-period <integer>
set severity [ high | medium | low ]
end
```

content-type	Text/html, text/plain, text/xml, application/xml, application/soap+xml, application/json.
occurrence-limit	Between 1 and 100000.
occurrence-within	Between 1 and 600 seconds.
percentage-match	Between 0 and 100. 0 means this condition is not needed.
action	String. WAF action profile.
block-period	Between 1 and 3600 seconds.
severity	• High • Medium • Low

Example

```
config security waf advanced-protection
edit "1"
config advanced-protection-rule
edit 1
set content-type text/html
set occurrence-limit 1
set occurrence-within 1
set percentage-match 1
set action alert
next
end
next
end
```

config security waf cookie-security

Use this command to configure waf cookie-security.

Syntax

```
config security waf cookie-security
   edit "test"
      set security-mode <no / encrypted / signed>
      set encrypted_cookie_type <all / list>
         config cookie_list // if the encrypted cookie type is a list
         edit 1
         set cookie_name <name>
         next
      set cookie-replay <enable/disable> // if security mode is encrypted
      set allow-suspicous-cookies < always/ never / custom > //if security mode is encrypted
      set dont_block_until <date> // if allow-suspicous-cookies is custom
      set action <action profile>
      set remove-cookie <enable/disable>
      set severity < high / medium / low >
      set httponly <enable/disable>
      set secure <enable/disable>
      set max_age <integer> //default value is 0, range 0-2147483647
      set exception <waf exception object>
   next
end
```

`config security waf cookie-security`	WAF cookie security policy allows the user to configure features that prevent cookie-based attacks, features such as cookie poisoning detection.
`security-mode < no/ encrypted/ signed >`	**No**—does not apply cookie tampering protection or encrypted cookie. **Signed**—Prevents tampering by tracking the cookie. This option requires that the client supports cookies. When the virtual server receives the first HTTP(S) request from a client, it uses cookie to track the session. After receiving the first response from the back-end server, Forti ADC will append ADC_SIGNED_COOKIE in Set-Cookie and record it into session table. Since the session tracking cookie includes a hash value that FortiADC uses to detect tampering cookie, if FortiADC determines the cookie from the client has changed, it will takes the specified action. **Encrypted**—FortiADC encrypts set-cookie values which have been sent from back-end web server to clients. Clients can only see the encrypted cookies. FortiADC also decrypts cookies which have been submitted by clients before sending them to the back-end server to determine if a cookie attack has been placed.

`encrpyted_cookie_type <all / list>`	Note: only for security-mode encrypted **All**—will encrypt all the cookies. **List**—will encrypt the cookie that match with the cookie-list.
`config cookie_list`	Note: only for security-mode encrypted and if encrypted_ cookie_type is list. The cookie to be encrypted.
`cookie-replay <enable/disable>`	Note: only for security-mode encrypted; optional. Determines whether FortiADC uses the IP address of a request to determine the owner of the cookie. **Enable**—If cookie replay is enabled, the client IP address will be appended to the set-cookie value before encryption. If X-forward header exists, FortiADC will use its IP. Otherwise src IP will become the client IP. Once FortiADC receives it, the cookie will be decrypted and FortiADC will check if the IP address matches with the client IP. Since the public IP of a client is not static in many environments, we recommend that you do not enable cookie-replay.
`allow-suspicous-cookies < always/ never/custom >`	Note: only for security-mode encrypted. Determines whether FortiADC allows requests that contain cookies which FortiADC does not recognize by encrypted cookie function or with missing cookies. When cookie-replay is enabled, the suspicious cookie is a missing cookie that tracks the client IP address. In many cases, when you first introduce the cookie security features, cookies that client browsers have cached earlier generate false positives. To avoid this problem, either select Never, or select Custom and enter an appropriate date on which to start taking the specified action against suspicious cookies. **Never**—never allow suspicious cookies. **Always**—always allow suspicious cookies. **Custom**—Don't Block suspicious cookies Until dont_block_ until specified date.
`dont_block_until <date> // allow-suspicous-cookies is custom`	Note: only for security-mode encrypted.
`action <datasource>`	WAF action.
`severity < high / medium / low >`	Log severity.
`remove-cookie <enable/disable>`	Note: for security-mode encrypted/signed. Accepts the request, but removes the cookie before sending it to the backend web server.
`httponly <enable/disable>`	Note: cookie attribute.

	Enable—add "HTTPOnly" flag to cookies. The HttpOnly attribute limits the scope of the cookie to HTTP requests. In particular, the attribute instructs the user agent to omit the cookie when providing access to cookies via "non-HTTP" APIs (such as a web browser API that exposes cookies to scripts).
`Secure-cookie <enable/disable>`	Note: cookie attribute. Enable—adds the secure flag to cookies. The Secure attribute limits the scope of the cookie to "secure" channels (where "secure" is defined by the user agent). When a cookie has the Secure attribute, the user agent will include the cookie in an HTTP request only if the request is transmitted over a secure channel, typically HTTP over Transport Layer Security (TLS).
`max_age <integer>`	Note: cookie attribute. Default value is 0 (do nothing), range 0- 65535. Add the maximum age (in minutes) if the response from backend server does not have an “Expires” or “Max-Age” attribute.
`exception <waf exception object>`	Exception list for no/encrypted/ signed.

Example

Security-mode no

```
config security waf cookie-security
edit "security-no"
set security-mode no
set action <action waf profile>
set severity < high /medium / low >
set httponly <enable/disable>
set secure <enable/disable>
set max_age <integer> //default value is 0, range 0- 65535
set exception <waf exception object>
next
end
```

Security-mode signed

```
config security waf cookie-security
edit "security-signed"
set security-mode signed
set action <action waf profile>
set remove-cookie <enable/disable>
set severity < high /medium / low >
set httponly <enable/disable>
set secure <enable/disable>
set max_age <integer>//default value is 0, range 0- 65535
set exception <waf exception object>
next
```

Security-mode encrypted

```
config security waf cookie-security
edit "security-enc-list"
set security-mode encrypted
set encrpyted_cookie_type list (<all/list>)
config cookie_list //if the encrypted cookie type is list
edit 1
set cookie_name <name>
next
end
set cookie-replay <enable/disable>
set allow-suspicous-cookies <always/never/ custom>
set dont_block_until <date> // if allow-suspicous-cookies is custom
set action <action profile>
set remove-cookie <enable/disable>
set severity < high / medium / low >
set httponly <enable/disable>
set secure <enable/disable>
set max_age <integer>//default value is 0, range 0- 65535
set exception <waf exception object>
next
end
```

config security waf data-leak-protection

Use this command to configure a DLP policy that can then be applied in a WAF profile. The Data Loss Prevention (DLP) feature allows the Web Application Firewall (WAF) to prevent information leaks, damage and loss. DLP provides desensitization and warning measures for sensitive information leaks on websites, such as SSN numbers and credit card information, as well as the leakage of sensitive keywords.

You can create a DLP Policy to match a sensor based on file content or an HTTP Payload, and the email protocol being used to attach files. It also allows you to choose the action to allow, log, or block the IP address.

Before you begin:

- Configure a virtual server with a WAF Profile.
- Configure a DLP Dictionary object. For details, see config security waf dlp-dictionary on page 381.
- Configure a DLP Sensor object. For details, see config security waf dlp-sensors on page 385.
- Configure a Sensitive Data Type object. For details, see config security waf sensitive-data-type on page 378.

Syntax

```
config security waf data-leak-prevention
   edit <name>
      set status {enable|disable}
      set masking {enable|disable}
      set action {alert|deny|block|silent-deny|captcha|<datasource>}
      set severity {high|medium|low}
      config rule
         edit <name>
            set request-uri-pattern <string>
```

```
            set type {sdt|sensors}
            set sensor <datasource>
            set sensitive-data-type <datasource>
            set threshold <integer>
        next
    end
  next
end
```

status	Enable or disable the profile; default is disable.
masking	Enable masking to replace sensitive data with asterisks (*); default is disable. **Note**: When masking is enabled, all target data will be replaced by asterisks, so the threshold value won't take effect here. Masking only works when Action is Alert, because the connection will reject when action is set as Deny or Block, so no target data will be replaced.
action	Sets the action FortiADC will take if a security check detects a potential attack. This configuration comes from Action in WAF Profile. • alert — Let the request pass when the profile detects a potential attack, only triggering a WAF log. • deny — Drop the incoming request and trigger a WAF log. • block — Block the IP address from incoming requests for 3600 seconds and trigger a WAF log. • silent-deny — Drop the incoming request without triggering a WAF log. • captcha — Allow the traffic to pass if the client successfully fulfills the CAPTCHA request, and trigger a WAF log. **Note**: You can also reference a user-defined WAF action object.
severity	Set the severity in WAF logs for potential attacks detected by DLP Policy. • high • medium • low The default option is **low**.
config rule	
request-uri-pattern	Specify the URI Pattern in the Data Loss Prevention rules. Scanning and receiving an empty value means this rule is not working.
type	Select the DLP data type to match: • sdt — Sensitive Data Type. • sensors — DLP Sensors.
sensor	The **sensor** option is available if **type** is **sensors**. Specify the DLP Sensor you want to apply.
sensitive-data-type	The **sensitive-data-type** option is available if **type** is **sdt**. Specify the Sensitive Data Type you want to apply.
threshold	The **threshold** option is available if **type** is **sdt**. Set a threshold for the Data Loss Prevention rule. The rule will not take effect until the target data exceeds the threshold's specified value. Range 1-10000. Default is 1. This will not work if Masking is enabled.

Example

```
config security waf data-leak-prevention
   edit "dlp-profile-sensors"
      set status enable
      set masking enable
      set action alert
      set severity low
      config rule
         edit 1
            set request-uri-pattern /dir1/
            set type sensors
            set sensor user-defined-sensor1
         next
      end
   next
end

config security waf data-leak-prevention
   edit "dlp-profile-sdt"
      set status enable
      set action alert
      set severity low
      config rule
         edit 1
            set type sdt
            set sensitive-data-type Credit_Card_Number
            set threshold 1
         next
      end
   next
end
```

config security waf sensitive-data-type

Use this command to configure a Sensitive Data Type object for the DLP Policy. A Sensitive Data Type object is referenced as part of the Data Loss Prevention (DLP) policy to prevent information, damage and loss by specifying strings as sensitive data.

Syntax

```
config security waf sensitive-data-type
   edit <name>
      set regex <string>
      set description <string>
   next
end
```

`regex`	Specify the regex string used to match sensitive data. There are two predefined regex strings named Credit_Card_Number and US_Social_Security_Number.

`description`	Comments about this profile. Describe what this profile is used for and what kind of data this regex is used to match.

Example

```
config security waf sensitive-data-type
   edit "Credit_Card_Number"
      set regex "^3(?:[47]\\d([ -]?)\\d{4}(?:\\1\\d{4}){2}|0[0-5]\\d{11}|[68]\\d{12})$|^4
           (?:\\d\\d\\d)?([ -]?)\\d{4}(?:\\2\\d{4}){2}$|^6011([ -]?)\\d{4}(?:\\3\\d{4})
           {2}$|^5[1-5]\\d\\d([ -]?)\\d{4}(?:\\4\\d{4}){2}$|^2014\\d{11}$|^2149\\d
           {11}$|^2131\\d{11}$|^1800\\d{11}$|^3\\d{15}$"
      set description "For credit card numbers from MC, Visa, Amex, Diners/CarteBlanche,
           Discover/Novus, Enroute, and JCB. Matches 341-1111-1111-1111 | 5431-1111-1111-
           1111 | 30569309025904 Non-Matches 30-5693-0902-5904 | 5631-1111-1111-1111 |
           31169309025904."
   next
end
```

Predefined Sensitive Data Type objects

Predefined Sensitive Data Type objects	Description
Credit_Card_Number	For credit card numbers from MC, Visa, Amex, Diners/CarteBlanche, Discover/Novus, Enroute, and JCB. Matches 341-1111-1111-1111 | 5431-1111-1111-1111 | 30569309025904 Non-Matches 30-5693-0902-5904 | 5631-1111-1111-1111 | 31169309025904.
US_Social_Security_Number	This regex validates U.S. social security numbers, within the range of numbers that have been currently allocated. Matches 078-05-1120 | 078 05 1120 Non-Matches 987-65-4320 | 000-00-0000 | (555) 555-5555.
Email	This regex validates email address. Matches example@fortinet.com Non-Matches @fortinet.com.
URL	This regex validates URL. Matches http://www.fortinet.com | https://127.0.0.1/path/example.php?name=test1 | ftp://user:pass@example.com:123 Non-Matches /fortinet.com
Numbers	This regex validates numbers. Matches 65535 Non-Matches a123.
Strings	This regex validates a string. Matches abc Non-Matches abc123.
Date/Time	This regex validates email address. Matches 29/02/1972 | 5-9-98 | 10-11-2002 | February 29, 2004 | 12:15 | 10:26:59 | 22:01:15 Non-Matches 32/12/2019.
IP Address	This regex validates IPv4 or IPv6 address. Matches 127.0.0.1 | FEDC:BA98:7654:3210:FEDC:BA98:7654:3210 | ::FFFF:129.144.52.38 Non-Matches 256.0.0.1 | FEDC:BA98:7654:3210 | ::
GUID	This regex validates a globally unique identifier. Matches 2064d355-c0b9-41d8-9ef7-9d8b26524751 | 2064D355-C0B9-41D8-9EF7-9D8B26524751 Non-Matches 2064D355.

Predefined Sensitive Data Type objects	Description
US Phone	This regex validates a US phone number WITH area code. It is written to all users to enter whatever delimiters they want or no delimiters at all. Matches 111-222-3333 | 111.222.3333 | (111) 222-3333| 1112223333 Non-Matches + 41 111-222-3333 .
US ZIP Code	This regex validates US zip codes. Matches all zip codes of exactly 5 digits except 00000. Optionally, matches zip5+zip4 where zip5 is exactly 5 digits, zip4 is exactly 4 digits, and zip5 and zip4 are, optionally, separated by a single space or hyphen. Captures zip5 and zip4 to named groups to facilitate program manipulation. Matches 12345 | 123456789 | 12345-6789 Non-Matches 123456.
US State Name and Abbrev.	This regex validates 50 US States's Name and Abbrev, case insensitive. Matches California | NewYork | North Carolina | AL.
US Street Address	This regex validates a US Street Address. Matches 123 Lincoln Avenu | 123 West Main St | 12345 Via De La Rosa Non-Matches Lincoln Avenu.
UK Vehicle Registration	This regex validates a UK vehicle registration system currently in use (as defined by the DVLA and put into effect from September 2001, and therefore does not allow registrations prior to this date). Matches AB51DVL | AB 51 DVL Non-Matches AB-51-DVL.
UK Bank Sort Code	This regex validates the format of a UK bank sort code. Matches 20-40-36 | 50-25-48 | 45-85-66 Non-Matches 204036.
Post Office Box	This regex validates a Post Office Box. Matches P. O. Box | p.o. box | PO Box | po box Non-Matches office box.
Chinese ID card	This regex validates a Chinese ID card number. Matches 2064d355-c0b9-41d8-9ef7-9d8b26524751 | 2064D355-C0B9-41D8-9EF7-9D8B26524751 Non-Matches 2064D355.
Chinese phone	This regex validates a Chinese telphone number. Matches 86 13512341234 | +86 15812341234 | 86 13612341234 Non-Matches 14012341234.
Australian Phone	This regex validates a Australian telephone number, most Australian telephone numbers including 13, 1300, 1800, 1900, std and international +61- format numbers. It allows optional spaces, dashes and brackets in most cases. Matches 1300 123 123 | 1300123123 | +61212341234 | (02) 1234-1234 | 02 1234 1234 Non-Matches 1400123123.
Canadian Postal Code	Canadian Postal Code format is (A1A 1X1) or (a1a 1x1). Its made up of two parts. Forward Sortation Area (FSA) and Local Delivery Unit (LDU). Read more on wikipedia. The letters D, F, I, O, Q, or U are not used on postal Code. Matches M1R 4B0 | L0R 1B1 | L0R1B9 Non-Matches MDR 4B0.

config security waf dlp-dictionary

Use this command to configure a DLP Dictionary object to use in the DLP Sensor. A DLP dictionary defines the patterns of data. The term "pattern" denotes a set of attributes specific to a given data type. For example, credit card numbers constitute numeric data that follow either the 14-digit or 16-digit patterns associated with credit cards. If the data adheres to these patterns, FortiADC will identify it as a match.

Before you begin:

- You must have a valid FortiGuard DLP service license and have enabled the service on FortiADC.

Syntax

```
config security waf dlp-dictionary
   edit <name>
      set match-type {any|all}
      set description <string>
      config entries
         edit <name>
            set status {enable|disable}
            set fg-data-type {uk-iban|can-natl_id-sin|luhn-algo|can-natl_id-prox|can-
                 pass|usa-pass-1|usa-pass-2|uk-pass|aus-pass|fra-pass|jpn-pass|can-health_
                 service|can-phin|glb-cc-amex|glb-cc-bcgl|glb-cc-cabl|glb-cc-dinr|glb-cc-
                 inst|glb-cc-jcb|glb-cc-kloc|glb-cc-lasr|glb-cc-maes|glb-cc-solo|glb-cc-
                 disc|glb-cc-mc|glb-cc-visa|glb-cc-vsmc|usa-natl_id-ssn|can-dl-ab|can-dl-
                 bc|can-dl-mb|can-dl-nb|can-dl-nl-2|can-dl-nl-1|can-dl-nt|can-dl-nu|can-dl-
                 pe-1|can-dl-pe-2|can-dl-qc|can-dl-sk|can-dl-yt|usa-dl-al|usa-dl-ak|usa-dl-
                 az|usa-dl-ar|usa-dl-co|usa-dl-ct|usa-dl-de|usa-dl-dc|usa-dl-fl|usa-dl-
                 ga|usa-dl-hi|usa-dl-id|usa-dl-il|usa-dl-in|usa-dl-ia|usa-dl-ks|usa-dl-
                 ky|usa-dl-la|usa-dl-me|usa-dl-md|usa-dl-ma|usa-dl-mi|usa-dl-mn|usa-dl-
                 ms|usa-dl-mo|usa-dl-ne|usa-dl-nv|usa-dl-nh|usa-dl-nj|usa-dl-nm|usa-dl-
                 ny|usa-dl-nc|usa-dl-oh|usa-dl-ok|usa-dl-or|usa-dl-pa|usa-dl-ri|usa-dl-
                 sc|usa-dl-sd|usa-dl-tn|usa-dl-tx|usa-dl-ut|usa-dl-vt|usa-dl-va|usa-dl-
                 wv|usa-dl-wi|usa-dl-wy|can-bank_account|usa-natl_id-prox|can-dl-ns|can-dl-
                 on|usa-dl-ca|jpn-swift|usa-swift|usa-dl-nd|usa-dl-wa|uk-swift|deu-swift|fra-
                 swift|aus-swift|chn-swift|can-sin}
            set repeat {enable|disable}
         next
      end
   next
end
```

`match-type`	Select the match type: • any — Data meeting the criteria specified by any one of the dictionary entries will be identified as a match. • all — Data meeting the criteria specified by all dictionary entries will be identified as a match.
`description`	Comments about this DLP Dictionary object.
config entries	
`status`	Enable the **Status** if you intend to apply this data type.

`fg-data-type`	Select a FortiGuard Data Type: • uk-iban • can-natl_id-sin • luhn-algo • can-natl_id-prox • can-pass • usa-pass-1 • usa-pass-2 • uk-pass • aus-pass • fra-pass • jpn-pass • can-health_service • can-phin • glb-cc-amex • glb-cc-bcgl • glb-cc-cabl • glb-cc-dinr • glb-cc-inst • glb-cc-jcb • glb-cc-kloc • glb-cc-lasr • glb-cc-maes • glb-cc-solo • glb-cc-disc • glb-cc-mc • glb-cc-visa • glb-cc-vsmc • usa-natl_id-ssn • can-dl-ab • can-dl-bc • can-dl-mb • can-dl-nb • can-dl-nl-2 • can-dl-nl-1 • can-dl-nt • can-dl-nu • can-dl-pe-1 • can-dl-pe-2 • can-dl-qc • can-dl-sk

- can-dl-yt
- usa-dl-al
- usa-dl-ak
- usa-dl-az
- usa-dl-ar
- usa-dl-co
- usa-dl-ct
- usa-dl-de
- usa-dl-dc
- usa-dl-fl
- usa-dl-ga
- usa-dl-hi
- usa-dl-id
- usa-dl-il
- usa-dl-in
- usa-dl-ia
- usa-dl-ks
- usa-dl-ky
- usa-dl-la
- usa-dl-me
- usa-dl-md
- usa-dl-ma
- usa-dl-mi
- usa-dl-mn
- usa-dl-ms
- usa-dl-mo
- usa-dl-ne
- usa-dl-nv
- usa-dl-nh
- usa-dl-nj
- usa-dl-nm
- usa-dl-ny
- usa-dl-nc
- usa-dl-oh
- usa-dl-ok
- usa-dl-or
- usa-dl-pa
- usa-dl-ri
- usa-dl-sc
- usa-dl-sd
- usa-dl-tn

	• usa-dl-tx • usa-dl-ut • usa-dl-vt • usa-dl-va • usa-dl-wv • usa-dl-wi • usa-dl-wy • can-bank_account • usa-natl_id-prox • can-dl-ns • can-dl-on • usa-dl-ca • jpn-swift • usa-swift • usa-dl-nd • usa-dl-wa • uk-swift • deu-swift • fra-swift • aus-swift • chn-swift • can-sin
`repeat`	Enable this option if you want to match data exclusively when it appears multiple times. With this option enabled, you can specify the times of occurrence in the DLP Sensor settings.

Example

```
config security waf dlp-dictionary
   edit "user-defined-dict1"
      set match-type any
      set description "User Defined Dictionary 1"
      config entries
         edit 1
            set status enable
            set fg-data-type can-natl_id-sin
            set repeat disable
         next
      end
   next
end
```

Predefined DLP Sensor objects

You can use the following predefined DLP Dictionary objects in Data Loss Prevention rules.

Predefined DLP Dictionary object	Match Type	Description
EICAR-TEST-FILE	Any	EICAR Test File for DLP
can-natl_id-pk	Any	
can-natl_id-sin-dict	Any	Canadian SIN Card Number Dictionary
glb-pass-pk	Any	
can-pass-dict	Any	Canadian Passport Dictionary
usa-pass-dict	Any	USA Passport Dictionary
uk-pass-dict	Any	UK Passport Dictionary
aus-pass-dict	Any	Australia Passport Dictionary
fra-pass-dict	Any	France Passport Dictionary
jpn-pass-dict	Any	Japan Passport Dictionary
can-health_service-pk	Any	
can-phin-pk	Any	
can-phin-dict	Any	Canadian Personal Health Identification Number Dictionary
can-health_service-dict	Any	Canadian Health Service Dictionary
glb-cc-pk	Any	
glb-cc-dict	Any	Global Credit Card Dictionary
usa-natl_id-pk	Any	
glb-dl-pk	Any	
can-dl-dict	Any	Canadian Driver's License Dictionary
can-bank_account-pk	Any	
can-bank_account-dict	Any	Canadian Bank Account Dictionary
usa-natl_id-ssn-dict	Any	USA SSN Card Number Dictionary
glb-swift-pk	Any	
source_code-python	Any	Python Source Code Dictionary
source_code-c	Any	C Source Code Dictionary
source_code-java	Any	Java Source Code Dictionary

config security waf dlp-sensors

Use this command to configure a DLP Sensor object to use in a DLP Policy. A DLP Sensor defines which dictionaries to check. You can match any dictionary or all dictionaries. It can also count the number of dictionary matches to trigger the

sensor.

Before you begin:

- You must have a valid FortiGuard DLP service license and have enabled the service on FortiADC.
- Configure a DLP Dictionary object. For details, see config security waf dlp-dictionary on page 381.

Syntax

```
config security waf dlp-sensors
   edit <name>
      set match-type {any|all}
      set description <string>
      config entries
         edit <name>
            set status {enable|disable}
            set dlp-dictionary <datasource>
            set count <integer>
         next
      end
   next
end
```

`match-type`	Select the match type: • any — Data meeting the criteria specified by any one of the dictionaries will be identified as a match. • all — Data meeting the criteria specified by all dictionaries will be identified as a match.
`description`	Comments about this DLP Sensor object.
config entries	
`status`	Enable the **Status** if you intend to apply this sensor.
`dlp-dictionary`	Specify a DLP Dictionary object.
`count`	Specify the occurrence threshold for the dictionary match. The sensor will be triggered when the dictionary match reaches the specified number of times. Default: 1 Range: 1-255. For example, if the dictionary applies to credit card numbers and the count is set to 4, the sensor will be triggered when credit card number occurs four times in the HTTP request or response.

Example

```
config security waf dlp-sensors
   edit "user-defined-sensor1"
      set match-type any
      set description "User Defined DLP Sensor 1"
      config entries
         edit 1
            set status enable
```

```
            set dlp-dictionary can-natl_id-pk
            set count 1
         next
      end
   next
end
```

Predefined DLP Sensor objects

You can use the following predefined DLP Sensor objects in Data Loss Prevention rules.

Predefined DLP Sensor object	Match Type	Description	Dictionaries
can-hia	Any	Canadian Health Information Act (HIA) Sensor	• can-pass-dict • can-natl_id-sin-dict • can-phin-dict • can-health_service-dict
can-pii	Any	Canadian Personal Identifiable Information (PII) Sensor	• can-dl-dict • can-natl_id-sin-dict • can-pass-dict • can-health_service-dict • can-bank_account-dict • can-phin-dict
source_code	Any	Source Code Sensor	• source_code-python • source_code-c • source_code-java

config security waf csrf-protection

Use this command to configure waf csrf-protection.

Syntax

```
config security waf csrf-protection
     edit <csrf-protection-name> // csrf protection name
        set action [alert | deny | block | silent-deny | <datasource> // default value:
             alert
        set severity [low | medium | high] // default value: low
        set status [enable | disable] // default value: disable
     config csrf-page-list
     edit 1
        set url-pattern [url] // URL of page, it supports regular expression
        set parameter-filter [ enable | disable] // default value: disable
        set parameter-filter-name <name> // parameter name
        set parameter-filter-pattern <value> // parameter value, it supports regular
             expression
```

```
      next
end
config csrf-url-list
   edit 1
      set url-pattern [url]
      set parameter-filter [ enable | disable] // default value: disable.
      set parameter-name <name> // parameter name
      set parameter-pattern <value> // parameter value, it supports regular expression
   next
end
end
end

config security waf profile
edit "waf"
set csrf-protection <csrf-protection-name> // csrf protection name
next
end
```

csrf-protection-name	CSRF protection name.
action	Default value: alert.
severity	Default: low.
status	Default value: disable.
csrf-page-list	When FortiADC receives a request for a web page in the page list, it inserts a javascript in the web page. The script runs in the client's web browser and automatically appends a anti-csrf token.
url-pattern	Page URL, supports regular expression.
parameter-filter	Enable or disable. Default is disable. In some cases, a request for a web page and the requests generated by its links have the same URL. FortiADC cannot distinguish between requests to add javascript to and requests to check for the anti-CSRF parameter. To avoid this issue, you create unique Page List and URL List items by adding a parameter filter to them. The parameter filter allows you to add additional criteria to match in the URL or HTTP body of a request.
parameter-name	Parameter name.
parameter-pattern	Parameter value, supports regular expression.
csrf-url-list	The URL list contains all the URLs that you want to protect. FortiADC will verify the anti-csrf token when you access the URL.

Example

```
config security waf csrf-protection
edit "csrf"
set status enable
set action deny
config csrf-page-list
edit 1
```

```
set url-pattern /csrf/csrf-all-in-one.php
next
end
config csrf-url-list
edit 1
set url-pattern /csrf/csrf-all-in-one.php
set parameter-filter enable
set parameter-filter-name say
set parameter-filter-value .*
next
end
next
end
```

config security waf allowed-origin

Use this command to configure the Allowed Origin List for Cross-Origin Resource Sharing (CORS) Protection.

The Allowed Origin List specifies the allowed domains using the HTTP response header. The header can contain either a * to indicate that all domains are allowed OR a specified domain to indicate the specified allowed domain.

Allowed Origin can only take effect in the CORS Protection rule when the **Apply to All CORS Traffic** is **disabled**. In the CORS Protection Rule List configuration, the Apply to All CORS Traffic option is disabled by default, which then requires you to apply an Allowed Origin List for the CORS Protection rule. If the Allowed Origin List is not applied, the CORS Protection rule would not work as the empty list would not match the condition.

Enabling the Apply to All CORS Traffic option hides the Allowed Origin option, making it inapplicable to the CORS Protection rule.

Syntax

```
config security waf allowed-origin
   edit <name>
      config allowed-origin-list
         edit <name>
            set protocol {HTTP|HTTPS|ANY}
            set origin-name <string>
            set port <integer>
            set include-sub-domains {enable|disable}
         next
      end
   next
end
```

protocol	Specify which type of protocols are allowed for the connections between foreign applications and your application. • HTTP • HTTPS

	• ANY The default is HTTP.
origin-name	Enter the foreign application's domain name or IP address. Wildcards are supported. (Range: 1-128 characters).
port	Specify the TCP port number for the CORS connections. (Range: 0-65535; default: 80).
include-sub-domains	Enable/disable to allow/disallow the Origin Value to match with the domains of its sub level. This is disabled by default.

Example

```
config security waf allowed-origin
   edit "1"
      config allowed-origin-list
         edit 1
            set protocol ANY
            set origin-name *
            set port 0
            set include-sub-domains enable
         next
      end
   next
end
```

config security waf cors-headers

Use this command to configure the CORS Headers List for the Cross-Origin Resource Sharing (CORS) Protection.

The HTTP headers on this list may be "allowed" or "exposed" in the CORS Protection Rule List. If allowed, FortiADC will use the headers list to verify whether the headers used in the CORS requests are legitimate. If exposed, FortiADC will expose the headers in the headers list in JavaScript and share with foreign applications.

Syntax

```
config security waf cors-headers
   edit <name>
      config headers-list
         edit <name>
            set header <string>
         next
      end
   next
end
```

header	Specify the HTTP header as a string. (Range: 1-63 characters).

Example

```
config security waf cors-headers
   edit <header-list1>
      config headers-list
         edit 1
            set header header1
         next
      end
   next
end
```

config security waf cors-protection

Use this command to configure Cross-Origin Resource Sharing (CORS) Protection.

Cross-Origin Resource Sharing (CORS) is a browser mechanism which enables controlled access to resources located outside of a given domain The CORS standard works by adding new HTTP headers that allow servers to describe which origins are permitted to read that information from a web browser. It extends and adds flexibility to the same-origin policy so that websites would not be restricted to accessing resources from the same origin.

However, in the process of enabling information sharing between sites, the significance of CORS configuration may be overlooked and allow for vulnerabilities. One such example is the Cross-Origin Request Site, an OWASP TOP10 Security Misconfiguration vulnerability.

To protect your applications against CORS vulnerabilities, use the CORS Protection feature to ensure that only legitimate CORS requests from allowed web applications can reach your application.

Syntax

```
config security waf cors-protection
   edit <name>
      set status {enable|disable}
      config cors-rule-list
         edit 1
            set action {alert|deny|block|silent-deny}
            set host-status {enable|disable}
            set request-url <string>
            set remove-other-headers {enable|disable}
            set allowed-methods {enable|disable}
            set allowed-headers {enable|disable}
            set exposed-headers {enable|disable}
            set allowed-origin <datasource>
            set methods {GET,POST,HEAD,TRACE,CONNECT,DELETE,PUT,PATCH}
            set allowed-headers-list <datasource>
            set exposed-headers-list <datasource>
            set insert-allowed-credentials {enable|disable}
            set allowed-credentials {true|false|none}
            set insert-max-age {enable|disable}
            set allowed-maximum-age <integer>
         next
      end
```

```
  next
end
```

`status`	Enable/disable CORS protection. This is disabled by default.
config cors-rule-list	
`action`	Specify the WAF action: • alert • deny • block • silent-block The default action is **block**.
`host-status`	Enable/disable to allow this rule to protect a specific domain name or IP address. This is disabled by default.
`request-url`	Specify the request URL as a regular expression. The maximum length is 8192 characters.
`remove-other-headers`	Enable/disable to remove the other headers that are excluded in the **exposed-headers-list**. This is disabled by default.
`allowed-methods`	Enable/disable to allow FortiADC to use the **methods** specified to verify whether the methods used in the CORS requests are legitimate. This is disabled by default.
`allowed-headers`	Enable/disable to allow FortiADC to use the **allowed-headers-list** to verify whether the headers used in the CORS requests are legitimate. This is disabled by default.
`exposed-headers`	Enable/disable to allow FortiADC to expose the specified headers in the **exposed-headers-list** in JavaScript and share with foreign applications. This is disabled by default.
`allowed-origin`	Specify the name of the Allowed Origin List (previously configured through config security waf allowed-origin). The allowed origin list ensures only the CORS traffic from the specified applications are allowed.
`methods`	If **allowed-methods** is enabled, specify the method(s): • GET • POST • HEAD • TRACE • CONNECT • DELETE • PUT • PATCH

`allowed-headers-list`	If **allowed-headers** is enabled, specify the name of the CORS Headers List to allow. (This is previously configured through config security waf cors-headers). FortiADC uses the allowed-headers-list to verify whether the headers used in the CORS requests are legitimate.
`exposed-headers-list`	If **exposed-headers** is enabled, specify the name of the CORS Headers List to expose. (This is previously configured through config security waf cors-headers). FortiADC will expose the headers in the exposed-headers-list in JavaScript and share with foreign applications.
`insert-allowed-credentials`	Enable/disable to allow whether the CORS requests from foreign applications can include user credentials. This is disabled by default.
`allowed-credentials`	If **insert-allowed-credentials** is enabled, select one of the following options: • true • false • none The default option is **none**.
`insert-max-age`	Enable/disable to specify a maximum time period before the result of the preflight request expires.
`allowed-maximum-age`	If **insert-max-age** is enabled, specify the maximum time period in seconds. (Range: 0-86400, default: 0).

Example

```
config security waf cors-protection
   edit "test"
      set status enable
      config cors-rule-list
         edit 2
            set action block
            set host-status disable
            set request-url /test
            set remove-other-headers disable
            set allowed-methods enable
            set allowed-headers enable
            set exposed-headers enable
            set allowed-origin test
            set methods GET
            set allowed-headers-list test1
            set exposed-headers-list test2
            set insert-allowed-credentials enable
            set allowed-credentials false
            set insert-max-age enable
            set allowed-max-age 0
         next
      end
   next
end
```

configure security ztna-profile

Use this command to create a ZTNA profile.

The ZTNA profile is the ZTNA policy used to enforce access control to Layer 7 HTTPS and TCPS virtual servers. ZTNA profiles consist of one or more ZTNA rule that determine the Source IP and ZTNA tags that are allowed access, and the resulting action to take.

After you have created a ZTNA profile, you can apply the Security ZTNA profile to a Layer 7 HTTPS or TCPS virtual server to activate ZTNA for server load balancing. Ensure the corresponding Client SSL profile is enabled for client certificate verification. For details, see config load-balance virtual-server on page 230 and config load-balance client-ssl-profile on page 133.

The ZTNA profile is an integral part of the Zero Trust Network Access (ZTNA) functionality. For more information, see the FortiADC Handbook on ZTNA.

Before you begin:

- You must have registered the FortiADC device through the FortiClient EMS connector. This can be done through CLI (for details, see config endpoint-control fctems on page 45 and execute fctems on page 599). However, it is recommended to configure the FortiClient EMS connector from the GUI. For more information, see the FortiADC Handbook on the FortiClient EMS Connector.
- You must have Read-Write permission for System settings.

Syntax

```
configure security ztna-profile
   edit <name>
      set log {enable|disable}
      config rule-list
         edit <id>
            set source-ip <address1> <address2> … <addressn>
            set ztna-tags <tags-name1> <tags-name2> … <tags-name3>
            set action {pass|deny}
         next
      end
   next
end
```

`log`	Enable/disable logging.
config rule-list	
`source-ip`	Specify the source IPs.
`ztna-tags`	Specify the ZTNA tags.
`action`	Select either of the following actions: • pass • deny The default action is **deny**.

Example

```
config security ztna profile
   edit "low-pass"
      set log enable
      config rule-list
         edit 1
            set source-ip Any
            set ztna-tags FCTEMS8822003242_Low
            set action pass
         next
      end
   next
end
```

config system

The `config system` commands configure system settings.

This chapter is a reference for the following commands:

config system accprofile

Use this command to manage access profiles.

Access profiles provision permissions to roles. The following permissions can be assigned:

- Read (view access)
- Read-Write (view, change, and execute access)
- No access

When an administrator has only read access to a feature, the administrator can access the web UI page for that feature, and can use the `get` and `show` CLI command for that feature, but cannot make changes to the configuration.

In larger companies where multiple administrators divide the share of work, access profiles often reflect the specific job that each administrator does (“role”), such as account creation or log auditing. Access profiles can limit each administrator account to their assigned role. This is sometimes called role-based access control (RBAC).

Table 20 lists the administrative areas that can be provisioned. If you provision read access, the role can view the web UI menu (or issue a CLI get command). If you provision read-write access, the role can save configuration changes (or issue a CLI set command).

For complete access to *all* commands and abilities, you must log in with the administrator account named **admin**.

Areas of control in access profiles

Web UI Menus	CLI Commands

System	`config system`
	`diagnose hardware`
	`diagnose netlink`
	`diagnose sniffer`
	`diagnose system`
	`execute date`
	`execute ping`
	`execute ping-options`
	`execute traceroute`

Networking	`config router`
Server Load Balance	`config load-balance`
Link Load Balance	`config link-load-balance`
Global Load Balance	`config global-dns-server`
Security	`config firewall`
Log & Report	`config log` `execute formatlogdisk`
* For each `config` command, there is an equivalent `get`/`show` command. The `config` commands require write permission. The `get`/`show` commands require read permission.	

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system accprofile
   edit <name>
      set firewall {none|read|read-write}
      set global-load-balance {none|read|read-write}
      set link-load-balance {none|read|read-write}
      set load-balance {none|read|read-write}
      set log {none|read|read-write}
      set router {none|read|read-write}
      set security {none|read|read-write}
      set system {none|read|read-write}
   next
end
```

`firewall`	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
`global-load-balance`	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
`link-load-balance`	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
`load-balance`	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
`log`	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
`router`	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
`security`	Set the permission: • none—Do not provision access for the menu.

	• read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.
system	Set the permission: • none—Do not provision access for the menu. • read—Provision ready-only access. • read-write—Enable the role to make changes to the configuration.

Example

```
FortiADC-docs # config system accprofile
FortiADC-docs (accprofile) # edit doc-user
Add new entry 'doc-user' for node 772

FortiADC-docs (doc-user) # get
system : none
router : none
firewall : none
load-balance : none
log : none
link-load-balance : none
global-load-balance : none
security : none

FortiADC-docs (doc-user) # set system read-write
FortiADC-docs (doc-user) # end
```

config system address

Use this command to create the IPv4 address objects that you use to specify matching source and destination addresses in policies.

The following policies use address objects:

- Connection limit policies
- Firewall policies
- Link Load Balance policies
- QoS policies

Basic Steps

1. Create address objects.
2. Specify them when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system address
   edit <name>
      set type {ip-netmask | ip-range}
      set ip-netmask <ip&netmask>
      set ip-min <class_ip>
      set ip-max <class_ip>
   next
end
```

type	• ip-netmask: address block • ip-range: address range
ip-netmask	Specify a subnet using the address/mask notation.
ip-min	Specify the start of an address range.
ip-max	Specify the end of an address range.

Example

```
FortiADC-docs # config system address
FortiADC-docs (address) # edit TEST-NET-1
Add new entry 'TEST-NET-1' for node 3800
FortiADC-docs (TEST-NET-1) # get
type : ip-netmask
ip-netmask : 0.0.0.0/0
FortiADC-docs (TEST-NET-1) # set ip-netmask 192.0.2.0/24
FortiADC-docs (TEST-NET-1) # next
FortiADC-docs (address) # edit TEST-NET-2
Add new entry 'TEST-NET-2' for node 3800
FortiADC-docs (TEST-NET-2) # set ip-netmask 198.51.100.0/24
FortiADC-docs (TEST-NET-2) # next
FortiADC-docs (address) # edit TEST-NET-3
Add new entry 'TEST-NET-3' for node 3800
FortiADC-docs (TEST-NET-3) # set ip-netmask 203.0.113.0/24
FortiADC-docs (TEST-NET-3) # end
FortiADC-docs #
```

config system address6

Use this command to create the IPv6 address objects that you use in firewall rules.

You create address objects to specify matching source and destination addresses in policies.

The following policies use address objects:

- Connection limit policies
- Firewall policies
- Link Load Balance policies
- QoS policies

Basic Steps

1. Create address objects.
2. Specify them when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system address6
   edit <No.>
      set type {ip6-network | ip6-range}
      set ip6-network <ip&netmask>
      set ip6-min <class_ip>
      set ip6-max <class_ip>
   next
end
```

type	• ip6-network: address block • ip6-range: address range
ip6-network	Specify a subnet using the address/mask notation.
ip6-min	Specify the start of an address range.
ip6-max	Specify the end of an address range.

Example

```
FortiADC-docs # config system address6
FortiADC-docs (address6) # edit WAN
Add new entry 'WAN' for node 3811
FortiADC-docs (WAN) # set ip6-network 2001:DB8::/32
FortiADC-docs (WAN) # end
```

config system addrgrp

Use this command to create the IPv4 address groups that you use to specify matching source and destination addresses in policies.

The following policies use address groups:

- Link Load Balance policies

Basic Steps

1. Create address objects.
2. Configure address group objects.
3. Select the address groups when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.
- You must have created IPv4 address objects.

Syntax

```
config system addrgrp
   edit <name>
      config member
         edit <name>
            set address <datasource>
         next
      end
   next
end
```

address	Specify an IPv4 address object.

Example

```
FortiADC-docs # config system addrgrp
FortiADC-docs (addrgrp) # edit WAN
Add new entry 'WAN' for node 3806
FortiADC-docs (WAN) # config member
FortiADC-docs (member) # edit 1
Add new entry '1' for node 3808
FortiADC-docs (1) # set address TEST-NET-3
FortiADC-docs (1) # end
FortiADC-docs (WAN) # next
FortiADC-docs (addrgrp) # edit LAN
Add new entry 'LAN' for node 3806
FortiADC-docs (LAN) # config member
FortiADC-docs (member) # edit 1
Add new entry '1' for node 3808
FortiADC-docs (1) # set address TEST-NET-1
FortiADC-docs (1) # next

FortiADC-docs (member) # edit 2
Add new entry '2' for node 3808
FortiADC-docs (2) # set address TEST-NET-2
FortiADC-docs (2) # end
FortiADC-docs (LAN) # end
```

config system addrgrp6

Use this command to create the IPv6 address groups that you use to specify matching source and destination addresses in policies.

The following policies use address groups:

- Link Load Balance policies

Basic Steps

1. Create address objects.
2. Configure address group objects.
3. Select the address groups when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.
- You must have created IPv4 address objects.

Syntax

```
config system addrgrp6
   edit <name>
      config member
         edit <name>
            set address <datasource>
         next
      end
   next
end
```

address	Specify an IPv6 address object.

Example

```
FortiADC-docs # config system addrgrp6
FortiADC-docs (addrgrp6) # edit WAN-6
Add new entry 'WAN-6' for node 3817
FortiADC-docs (WAN-6) # config member
FortiADC-docs (member) # edit 1
Add new entry '1' for node 3819
FortiADC-docs (1) # set address WAN
FortiADC-docs (1) # end
FortiADC-docs (WAN-6) # end
```

config system admin

Use this command to manage administrator and REST API administrator accounts.

Administrator user accounts can be created and configured through the CLI. For details, see Administrator accounts on page 405.

REST API administrator accounts can only be edited through the CLI but not created. You can only create a REST API administrator account through the GUI. However, once the account is created in the GUI, the REST API administrator can be edited in the CLI. For details, see REST API administrator accounts on page 407.

Administrator accounts

Use `config system admin` to create and manage administrator accounts.

We recommend that only network administrators—and if possible, only a single person—use the **admin** account. You can configure accounts that provision different scopes of access. For example, you can create an account for a security auditor who must only be able to view the configuration and logs, but *not* change them.

Before you begin:

- If you want to use RADIUS, LDAP or TACACS+ authentication, you must have already have created the RADIUS server, LDAP server or TACACS+ server configuration.
- You must have read-write permission for system settings.

Syntax

```
config system admin
   edit <name>
      set access-profile <datasource>
      set auth-strategy {local|ldap|radius|tacacs_plus}
      set ldap-server <datasource>
      set radius-server <datasource>
      set tacacs-plus-server <datasource>
      set is-system-admin {no|yes}
      set password <passwd>
      set trusted-hosts <ip&netmask>
      set vdom <datasource>
      set wildcard {disable|enable}
   next
end
```

`<name>`	Name of the administrator account, such as `admin1` or `admin@example.com`. Do not use spaces or special characters except the 'at' symbol (`@`) or dot (`.`). The maximum length is 35 characters. **Note:** This is the user name that the administrator must provide when logging in to the CLI or web UI. After you initially save the configuration, you cannot edit the name.

`access-profile`	Specify a user-defined or predefined profile. The predefined profile named **super_admin_prof** is a special access profile used by the **admin** account. However, specifying this access profile will *not* confer all permissions of the **admin** account. For example, the new administrator would not be able to reset lost administrator passwords. *Note*: This option does not appear for the **admin** administrator account, which by definition always uses the **super_admin_prof** access profile.
`auth-strategy`	• local — Use the local authentication server. • ldap — Use an LDAP authentication server. • radius — Use a RADIUS authentication server. • tacacs_plus — Use a TACACS+ authentication server.
`ldap-server`	If using LDAP, specify the LDAP server configuration.
`radius-server`	If using RADIUS, specify the RADIUS server configuration.
`tacacs-plus-server`	If using TACACS+, specify the TACACS+ server configuration.
`is-system-admin`	Select whether or not to allow the administrator account to have Global access, which is required to access all virtual domains, configure Automation Stitches, and create system backup files. • No — This is the default option. The administrator account will only have access to the virtual domain specified in this configuration. Administrators with no Global Admin permission cannot configure Automation Stitches or create system backup files. • Yes — The administrator account will have access to all virtual domains. With Global Admin access, the administrator can configure Automation Stitches and create system backup files.
`password`	Set a strong password for all administrator accounts. The password should be at least eight characters long, be sufficiently complex, and be changed regularly.
`wildcard`	Enable/disable user wildcard for remote server authentication.

`trusted-hosts`	Source IP address and netmask from which the administrator is allowed to log in. For multiple addresses, separate each entry with a space. You can specify up to three trusted areas. They can be single hosts, subnets, or a mixture.
	Configuring trusted hosts hardens the security of the system. In addition to knowing the password, an administrator must connect only from the computer or subnets you specify.
	Trusted host definitions apply both to the web UI and to the CLI when accessed through Telnet, SSH, or the CLI console widget. Local console access is *not* affected by trusted hosts, as the local console is by definition not remote, and does not occur through the network.
	If ping is enabled, the address you specify here is also a source IP address to which the system will respond when it receives a ping or traceroute signal.
	To allow logins only from *one* computer, enter only its IP address and 32- or 128-bit netmask:
	`192.0.2.2/32`
	`2001:0db8:85a3::8a2e:0370:7334/128`
	To allow login attempts from any IP address (not recommended), enter:
	`0.0.0.0/0.`

	Caution: If you restrict trusted hosts, do so for *all* administrator accounts. Failure to do so means that all accounts are still exposed to the risk of brute force login attacks. This is because if you leave even *one* administrator account unrestricted (i.e. `0.0.0.0/0`), the system must allow login attempts on all network interfaces where remote administrative protocols are enabled, and wait until *after* a login attempt has been received in order to check that user name's trusted hosts list. **Tip:** If you allow login from the Internet, set a longer and more complex New Password, and enable only secure administrative access protocols. We also recommend that you restrict trusted hosts to IPs in your administrator's geographical area. **Tip:** For improved security, restrict all trusted host addresses to single IP addresses of computer(s) from which *only* this administrator will log in.
`vdom`	If you have enabled the virtual domain feature, specify the virtual domain that this administrator can view and manage. Note: You can create multiple VDOMs separated by space.

Example

```
FortiADC-VM # config system admin
FortiADC-VM (admin) # edit doc-admin
Add new entry 'doc-admin' for node 78
FortiADC-VM (doc-admin) # set access-profile doc-admin
FortiADC-VM (doc-admin) # end

FortiADC-VM # get system admin doc-admin
is-system-admin : no
vdom : root
password : *
trusted-hosts : 0.0.0.0/0 ::/0
auth-strategy : local
access-profile : doc-admin
theme :
role-list :
privilege-map :
access-token : 3p6RgrzT21ciDMdwgowh9Lwd303SoSsrhygy0Or0PDhrnuXBQrRZdnagne
     6K6y9o5qU5el31WkqiMmRANIy04IfpWl91SjnXHh0TA1SukjM6DCFoidnmVCKQVRRN8cIP
```

REST API administrator accounts

Use `config system admin` to edit the configurable attributes of an existing REST API administrator account.

Before you begin:

- You must have read-write permission for system settings.
- Created a REST API administrator account in the GUI.

Although users can use an API request to create a REST API administrator account, the resulting token would not be properly assigned to the user. Without an assigned user this authorization token would be invalid and would not be able to access the supported FortiADC REST APIs.

Syntax

```
config system admin
   edit <name>
      set is-system-admin {no|yes}
      set trusted-hosts <ip&netmask>
      set access-profile <datasource>
      set comments <string>
      set cors-allow-origin <string>
   next
end
```

<name>	Enter the login name of the REST API administrator account.
is-system-admin	Select either of the following global admin access options: • no — The account can access the virtual domain specified in this configuration only. This is the default option. • yes — The account can access all virtual domains.
trusted-hosts	If restricted to trusted hosts is enabled, specify the trusted host IP address and netmask allowed to log in to the REST API. For multiple addresses, separate each entry with a space. You can specify up to three trusted areas. They can be single hosts, subnets, or a mixture.
access-profile	The **access-profile** option is configurable if **is-system-admin** is **no**. Specify a user-defined or predefined profile. The predefined profile named **super_admin_prof** is a special access profile used by the **admin** account. However, specifying this access profile will *not* confer all permissions of the **admin** account. For example, the new administrator would not be able to reset lost administrator passwords. *Note*: This option does not appear for the **admin** administrator account, which by definition always uses the **super_admin_prof** access profile.
comments	(Optional) Enter comments about the administrator account.
cors-allow-origin	If CORS Allow Origin is enabled, then specify the URL that can access the REST API.

Example

```
config system admin
   edit "restapi_admin"
      set is-system-admin no
      set trusted-hosts 0.0.0.0/0 ::/0
      set access-profile super_admin_prof
      set comments test
      set cors-allow-origin https://fndn.fortinet.net
   next
end
```

config system auto backup

Use this command to configure scheduled system backup.

Before you begin:

- You must have Global Administrator access. Ensure that your admin account settings has **Global Admin** set to **Yes**.

Syntax

```
config system auto-backup
   set address <ip>
   set folder <string>
   set overwrite-config {enable|disable}
   set password <string>
   set port <integer>
   set scheduled-backup-day {Sunday|Monday|Tuesday|Wednesday|Thursday|Friday|Saturday}
   set scheduled-backup-frequency {daily|weekly|every}
   set scheduled-backup-status {enable|disable}
   set scheduled-backup-time <hh:mm>
   set storage {disk|sftp}
   set username <string>
end
```

`address`	The IP address of the SFTP server.
`folder`	Specify the folder path on the SFTP server.
`overwrite-config`	Enable or disable "overwrite-config" when "storage" is "disk". If overwrite-config is disabled, FortiADC will stop backing up configurations when the maximum size or files is met. By default, this setting is disabled.
`port`	Specify the listening port on the SFTP server.
`scheduled-backup-day`	Specify one day of the week (i.e., Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday.)
`scheduled-backup-frequency`	Specify a scheduled backup frequency: • Daily—Specify the time of day for daily backup. • Weekly—Specify the day and time for weekly backup. • Every—Specify the time for periodic backup.
`scheduled-backup-status`	Enable or disable scheduled backup.
`scheduled-backup-time`	Specify the backup time in the format of <hh:mm> hour and minute, hh: 0-23, mm: {00|15|30|45}.
`storage`	Specify where to save the backup configuration files: • disk—Hard disk. • sftp—SFTP server.

username	The user name used to log into the SFTP server.
password	The password used to log into the SFTP server.

Example

The following example for a scenario where the storage is on a local disk:

```
FortiADC-VM # config system auto-backup
FortiADC-VM (auto-backup) # get
scheduled-backup-status : disable
FortiADC-VM (auto-backup) # set scheduled-backup-status enable
FortiADC-VM (auto-backup) # set scheduled-backup-day Monday
FortiADC-VM (auto-backup) # set scheduled-backup-time 03:30
FortiADC-VM (auto-backup) # set overwrite-config enable
FortiADC-VM (auto-backup) # show full

config system auto-backup
   set scheduled-backup-status enable
   set scheduled-backup-frequency weekly
   set scheduled-backup-day Monday
   set scheduled-backup-time 03:30
   set storage disk
   set overwrite-config enable
end

FortiADC-VM (auto-backup) # get
scheduled-backup-status : enable
scheduled-backup-frequency : weekly
scheduled-backup-day : Monday
scheduled-backup-time : 03:30
storage : disk
overwrite-config : enable
```

The following example is for a scenario where the storage is on an SFTP server:

```
FortiADC-VM # config sys auto-backup
FortiADC-VM (auto-backup) # set scheduled-backup-status enable
FortiADC-VM (auto-backup) # set storage sftp
FortiADC-VM (auto-backup) # set scheduled-backup-frequency daily
FortiADC-VM (auto-backup) # set scheduled-backup-time 12:00
FortiADC-VM (auto-backup) # set address 10.0.100.2
FortiADC-VM (auto-backup) # set username test
FortiADC-VM (auto-backup) # set password test
FortiADC-VM (auto-backup) # set folder /backup
FortiADC-VM (auto-backup) # show full

config system auto-backup
   set scheduled-backup-status enable
   set scheduled-backup-frequency daily
   set scheduled-backup-time 12:00
   set storage sftp
   set address 10.0.100.2
   set port 22
   set username test
```

```
   set password ENC
        FQ21TuLJKNMzW3AD3GkVQX2yUyeMiC8251rM6XFbNuZkE80xcV+miFjCymkvA+LS8211ZJ8trQFF7RI1NnAN
        dFOlyiABjZ2ZB/YqsbOETL/Ckywe
   set folder /backup
end

FortiADC-VM (auto-backup) # get
scheduled-backup-status : enable
scheduled-backup-frequency : daily
scheduled-backup-time : 12:00
storage : sftp
address : 10.0.100.2
port : 22
username : test
password : *
folder : /backup
```

config system azure

Use this command to configure the FortiADC VM that is deployed on the Azure Cloud Platform.

Before you begin:

- You must have read-write permission for system settings.
- You must have deployed a FortiADC VM on the Azure Cloud Platform.

Syntax

```
config system azure
   set tenant-id <tenant-id>
   set subscription-id <subscription-id>
   set client-id <client-id>
   set client-secret <client-secret>
   set azure-region {china|germany|global|usgov}
   set resource-group <resource-group>
end
```

`tenant-id`	Specify the Azure Tenant ID (directory ID).
`subscription-id`	Specify the Azure subscription ID.
`client-id`	Specify the Azure client ID (application ID).
`client-secret`	Specify the Azure client secret (application key).
`azure-region`	Select one of the following regions: • china • germany • global • usgov
`resource-group`	Specify the Azure resource group.

Example

```
FAD-HA-vm1 # config system azure
FAD-HA-vm1 (azure) # set tenant-id XXXXXXXXXXXXXXXXXXXX
FAD-HA-vm1 (azure) # set client-id XXXXXXXXXXXXXXXXXXXXXXXXXXXX
FAD-HA-vm1 (azure) # set client-secret XXXXXXXXXXXXXXXXXXXXXXXXXXXX
FAD-HA-vm1 (azure) # set azure-region global
FAD-HA-vm1 (azure) # end

FAD-HA-vm1 # get sys azure
tenant-id : XXXXXXXXXXXXXXXXXXXX
subscription-id :
client-id : XXXXXXXXXXXXXXXXXXXXXXXXXXXX
client-secret : *
azure-region : global
resource-group :
```

config system azure-lb-backend-ip

Use this command to configure the Azure Load Balancer backend IP to associate which can then be used to associate as a virtual server IP.

Before you begin:

- You must have read-write permission for system settings.
- You must have deployed a FortiADC VM on the Azure Cloud Platform.

Syntax

```
config system azure-lb-backend-ip
   edit <azure-lb-backend-name>
      set ip <class_ip>
   next
end
```

azure-lb-backend-name	Specify the ALB backend name.
class_ip	Specify the ALB backend IP.

Example

```
FAD-HA-vm1 # config system azure-lb-backend-ip
FAD-HA-vm1 (azure-lb-backe~p) # edit "ext-FADHaLBBackendAddrPool-1"
FAD-HA-vm1 (ext-FADHaLBBac~1) # set ip 10.2.0.5
FAD-HA-vm1 (ext-FADHaLBBac~1) # next
FAD-HA-vm1 (azure-lb-backe~p) # end

FAD-HA-vm1 # show sys azure-lb-backend-ip
config system azure-lb-backend-ip
```

```
  edit "ext-FADHaLBBackendAddrPool-1"
    set ip 10.2.0.5
  next
end
```

config system certificate ca

Use this command to configure CA certificates. An alternative to execute certificate ca.

Before you begin:

- You must have read-write permission for system settings.
- You must import a ca certificate in system.

Syntax

```
config system certificate ca
   edit <name>
      set certificate-file <certificate-filename>
   next
end
```

`certificate`	Paste the name of a CA certificate file between quotation marks as shown in the example.

Example

```
FortiADC-VM # config system certificate ca
FortiADC-VM (ca) # edit "ca-new"
FortiADC-VM (ca-new) # set ca-new.cer
FortiADC-VM (ca-new) # end
```

config system certificate ca_group

Use this command to manage CA groups.

Create CA groups to facilitate the configuration of the certificate validator that is associated with a virtual server.

Include in the CA group all of the CAs for the pool of backend servers to be associated with a single virtual server.

Before you begin:

- You must have already added the CAs to the CA certificate store.
- You must have read-write permission for system settings.

Syntax

config system certificate ca_group

edit <name>

config group_member

edit <No.>

set ca <datasource>

next

end

next

end

ca	Specify the CA to add to the group.

config system certificate certificate_verify

Use this command to manage certificate validation rules.

To be valid, a client certificate must meet the following criteria:

- Must not be expired or not yet valid
- Must not be revoked by either certificate revocation list (CRL) or, if enabled, online certificate status protocol (OCSP)
- Must be signed by a certificate authority (CA) whose certificate you have imported into the FortiADC appliance
- Must contain a CA field whose value matches a CA's certificate
- Must contain an Issuer field whose value matches the Subject field in a CA's certificate

Certificate validation rules specify the CA certificates to use when validating client certificates, and they specify a CRL and/or OCSP server, if any, to use for certificate revocation checking.

You select a certificate validation configuration object in the profile configuration for a virtual server. If the client presents an invalid certificate during the authentication phase of a SSL/TLS session initiation, the FortiADC system will not allow the connection.

Before you begin:

- You must have already created a CA group and OCSP or CRL configuration.
- You must have read-write permission for system settings.

Syntax

```
config system certificate certificate_verify
   edit "verify"
      set verify-depth <integer>
      set customize-error-ignore <enable/disable>
```

```
      set ca-ignore-errors <ca_errors>
      set cert-ignore-errors <cert_errors>
      config group_member
         edit 1
            set ca-certificate <ca>
            set ocsp <ocsp rule>
            set crl <crl rule>
         next
      end
   next
end
```

verify-depth	Specify the depth from the last intermediate CA to the root CA.
customize-error-ignore	Enable or disable "ignore errors".
ca-ignore-errors	Specify the errors on the CA to be ignored. Applicable only when "customize-error-ignore" is enabled.
cert-ignore-errors	Specify the errors on the certificate to be ignored. Applicable only when "customize-error-ignore" is enabled.

Example

```
FortiADC-VM # config system certificate certificate_verify
FortiADC-VM (certificate_ve~i) # edit "verify"
FortiADC-VM (verify) # set verify-depth
<integer> Verify depth
FortiADC-VM (verify) # set customize-error-ignore
enable enable option
disable disable option
FortiADC-VM (verify) # set ca-ignore-errors
UNABLE_TO_GET_ISSUER_CERT OPENSSL 2
UNABLE_TO_GET_CRL OPENSSL 3
CERT_NOT_YET_VALID OPENSSL 9
CERT_HAS_EXPIRED OPENSSL 10
CRL_NOT_YET_VALID OPENSSL 11
CRL_HAS_EXPIRED OPENSSL 12
DEPTH_ZERO_SELF_SIGNED_CERT OPENSSL 18
SELF_SIGNED_CERT_IN_CHAIN OPENSSL 19
UNABLE_TO_GET_ISSUER_CERT_LOCALLY OPENSSL 20
UNABLE_TO_VERIFY_LEAF_SIGNATURE OPENSSL 21
CERT_CHAIN_TOO_LONG OPENSSL 22
INVALID_CA OPENSSL 24
INVALID_PURPOSE OPENSSL 26
CERT_UNTRUSTED OPENSSL 27
CERT_REJECTED OPENSSL 28
FortiADC-VM (verify) # set cert-ignore-errors
UNABLE_TO_GET_ISSUER_CERT OPENSSL 2
UNABLE_TO_GET_CRL OPENSSL 3
CERT_NOT_YET_VALID OPENSSL 9
CERT_HAS_EXPIRED OPENSSL 10
CRL_NOT_YET_VALID OPENSSL 11
CRL_HAS_EXPIRED OPENSSL 12
DEPTH_ZERO_SELF_SIGNED_CERT OPENSSL 18
SELF_SIGNED_CERT_IN_CHAIN OPENSSL 19
```

```
UNABLE_TO_GET_ISSUER_CERT_LOCALLY OPENSSL 20
UNABLE_TO_VERIFY_LEAF_SIGNATURE OPENSSL 21
CERT_CHAIN_TOO_LONG OPENSSL 22
INVALID_CA OPENSSL 24
INVALID_PURPOSE OPENSSL 26
CERT_UNTRUSTED OPENSSL 27
CERT_REJECTED OPENSSL 28
FortiADC-VM (verify) #
```

config system certificate crl

Use this command to manage certificate revocation lists (CRL). You can enable CRL by importing a CRL file or specifying a CRL URL.

A CRL is a file that contains a list of revoked certificates, their serial numbers, and their revocation dates. The file also contains the name of the issuer of the CRL, the effective date, and the next update date. By default, the shortest validity period of a CRL is one hour.

Some potential reasons for certificates to be revoked include:

- A CA server was hacked and its certificates are no longer trustworthy.
- A single certificate was compromised and is no longer trustworthy.
- A certificates has expired and is not supposed to be used past its lifetime.

You can upload a CRL file or specify a URL for the CRL file.

Online certificate status protocol (OCSP) is an alternative to CRL. OCSP is useful when you do not want to deploy CRL files, for example, or want to avoid the public exposure of your PKI structure even if it is only invalid certificates.

Before you begin:

- You must know the URL of a CRL server or have downloaded the CRL file and be able to browse to it so that you can upload it.
- You must have read-write permission for system settings.

Syntax

```
config system certificate crl
   edit <name>
      set crl <certificate-filename>
      set http-url <string>
      set scep-url <string>
      set host-header <string>
   next
end
```

crl	Paste thename of a CRL certificate file between quotation marks as shown in the example.
http-url	Specify an HTTP URL.
scep-url	Specify a SCEP URL.
host-header	Specify a hostname in the HTTP request header.

Example

```
FortiADC-VM # config system certificate crl
FortiADC-VM (crl) # edit "crl"
FortiADC-VM (crl) # set crl-file global_crl.cer
FortiADC-VM (crl) # end
```

See also

- execute certificate crl

config system certificate intermediate_ca

Use this command to configure intermediate CAs.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system certificate intermediate_ca
   edit <name>
      set certificate <certificate-filename>
   next
end
```

certificate	Paste the name of an intermediate CA file between quotation marks as shown in the example.

Example

```
FortiADC-VM # config system certificate intermediate_ca
FortiADC-VM (intermediate_ca) # edit "intermediate_ca"
FortiADC-VM (intermediate_ca) # set certificate-file intermediate_ca.cer
FortiADC-VM (intermediate_ca) # end
```

config system certificate intermediate_ca_group

Use this command to manage intermediate CA groups.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system certificate intermediate_ca_group
   edit <name>
      config group_member
         edit <No.>
            set ca <datasource>
         next
      end
   next
end
```

`ca`	Specify a CA configuration object.

config system certificate local

In order for FortiADC to authenticate client certificates, you can either generate a certificate signing request or upload trusted CA certificates to FortiADC. This configuration is automatically created after you have successfully imported an automated local certificate or have generated/regenerated a certificate signing request (CSR) file. As the records are automatically generated, editing is not recommended.

The information recorded in `config system certificate local` differs depending on the method used to authenticate the client certificate.

- Record from generating/regenerating a certificate signing request on page 418
- Record from importing an automated local certificate on page 419

For the command to generate/regenerate a CSR file, see execute certificate local on page 555. For the command to import an automated local certificate, see execute certificate local import automated on page 556.

Record from generating/regenerating a certificate signing request

This configuration is automatically created after you have successfully generated/regenerated a CSR file.

Syntax

```
config system certificate local
     edit <name>
```

```
            set certificate-file <certificate-filename>
            set comments <string>
            set csr <csr>
            set password <passwd>
            set private-key-file <key-filename>
        next
end
```

certificate	Modify "contents" in certificate and private-key to "file".
comments	Optional administrator note.
csr	Paste the contents of a CSR file between quotation marks as shown in the example.
password	Password that was used to encrypt the file. The FortiADC system uses the password to decrypt and install the certificate.
private-key	Paste the contents of a key file between quotation marks as shown in the example.

Example

```
FortiADC-VM # config system certificate local
FortiADC-VM (local) # edit "csr"
FortiADC-VM (csr) # set private-key-file csr.key
FortiADC-VM (csr) # set csr-file csr.csr
FortiADC-VM (csr) # end
FortiADC-VM # config system certificate local
FortiADC-VM (local) # edit "new-local"
FortiADC-VM (new-local) # set private-key-file new-local.key
FortiADC-VM (new-local) # set certificate-file new-local.cer
FortiADC-VM (new-local) # end
```

Record from importing an automated local certificate

This configuration is automatically created after you have successfully imported an automated local certificate.

Syntax

```
config system certificate local
   edit <name>
      set acme_status <not_set|pending|ok|failed>
      set acme-domain <domain_name>
      set acme-server-url <URL>
      set acme-email <Email>
      set acme-key-type <RSA-2048|RSA-3072|RSA-4096|ECDSA-P256|ECDSA-P384|ECDSA-P521>
      set acme-renew-win <Integer>
      set acme-server-timeout <Integer>
      set acme-ca-group <CA Group>
      set private-key-file <key-filename>
      set certificate-file <certificate-filename>
   next
end
```

acme_status	The status of the ACME certificate: • not_set • pending • ok • failed The default status is **not_set**.
acme-domain	The web server domain to be protected by the certificate.
acme-server-url	The ACME server URL.
acme-email	The email address to be used by ACME for renewal fail notices.
acme-key-type	The certificate type based on the key type: • RSA-2048 • RSA-3072 • RSA-4096 • ECDSA-P256 • ECDSA-P384 • ECDSA-P521
acme-renew-win	The time (in minutes) to renew the certificate before it is expired.
acme-server-timeout	The ACME server connection timeout (in seconds).
acme-ca-group	The CA certificate group to verify the API server's certificate.

Example

```
config system certificate local
   edit "acme_cert"
      set acme_status ok
      set acme-domain test01
      set acme-server-url https://10.3.0.204:14000/dir
      set acme-email test@example.com
      set acme-key-type RSA-2048
      set acme-renew-win 1
      set private-key-file test1-TLS-RSA-2048.key
      set certificate-file test1-TLS-RSA-2048.cer
   next
end
```

config system certificate local_cert_group

Use this command to manage local certificate groups.

Create local groups to facilitate the configuration of profiles that are associated with a virtual server.

Include in the local certificate group all of the server certificates and intermediate CAs for the pool of backend servers to be associated with a single virtual server.

Before you begin:

- You must have already added the certificates to the local certificate store and Intermediate CA certificate store.
- You must have read-write permission for system settings.

Syntax

```
config system certificate local_cert_group
   edit <name>
      config group_member
         edit <No.>
            set default {enable|disable}
            set intermediate-ca-group <datasource>
            set local-cert <datasource>
         next
      end
   next
end
```

`default`	Specify one certificate to be the default for the group.
`intermediate-ca-group`	Specify an Intermediate CA group configuration.
`local-cert`	Specify a local certificate configuration.

config system certificate remote

Use this command to configure a remote certificate. You can enable OCSP by importing an OCSP CA or specifying an OSCP URL. If you want to use the configuration in a certificate verify configuration, you must add both an OCSP CA and URL.

OCSP enables you to validate or revoke certificates by query, rather than by importing certificate revocation list (CRL) files. Since distributing and installing CRL files can be a considerable burden in large organizations, and because delay between the release and install of the CRL represents a vulnerability window, this can often be preferable.

To use OCSP queries, you must first install the certificates of trusted OCSP/CRL servers.

Before you begin:

- You must know the URL of an OCSP server or have downloaded the certificate and key files and be able to browse to them so that you can upload them.
- You must have read-write permission for system settings.

Syntax

```
config system certificate remote
   edit "cert"
        set certificate-file cert.cer
```

```
   next
end
```

cert	Paste the contents of a CA file between the quotation marks (" "), as shown in the example below.

Example

```
FortiADC-VM # config system certificate remote
FortiADC-VM (remote) # edit new-remote-ca
FortiADC-VM (new-remote-ca) # set certificates-file new-remote-ca.cer
FortiADC-VM (new-remote-ca) # end
```

See also

- execute certificate remote

config system certificate ocsp_stapling

Use this command to configure Online Certificate Status Protocol Stapling. You can enable OCSP stapling by importing an OCSP response or quote an OCSP profile.

In a stapling scenario, the certificate holder queries the OCSP server themselves at regular intervals, obtaining a signed time-stamped OCSP response. When the site's visitors attempt to connect to the site, this response is included ("stapled") with the TLS/SSL Handshake via the Certificate Status Request extension response. Note that the TLS client must explicitly include a Certificate Status Request extension in its Client Hello TLS/SSL handshake message.

OCSP_staping could be used in a `local_certificate_group`, and the local certificate in OCSP stapling must be the local certificate in the local certificate group.

Syntax

```
config system certificate OCSP_stapling
edit <name>
   set OCSP <datasource>
   set OCSP-response-file <OCSP-response-filename>
   set issuer-certificate <datasource>
   set local-certificate <datasource>
   set response-update-ahead-time <integrate>
   set response-update-interval <integrate>
end
```

ocsp	Quote from system certificate OCSP.

ocsp-response	A certificate containing the OCSP response from the OCSP server.
issuer-certificate	The issuer CA of the local certificate.
local-certificate	The certificate used by FortiADC.
response-update-ahead-time	The default is 1h (1 hour). Valid values are Xh (hour), Xm (minute), and Xs (second). For example, 5m, 30s (=5 minute and 30 seconds).
response-update-interval	The number of seconds (200 ms by default) that FortiADC waits for a response from the OCSP responder. FortiADC will block the link once it times out.

Example

```
config system certificate OCSP_stapling
     edit "ocsp_staping"
          set local-certificate cert
          set issuer-certificate cacert
          set OCSP-response-file ocsp_staping.cer
     next
end
```

config system certificate ocsp_stapling

Use this command to configure Online Certificate Status Protocol Stapling. You can enable OCSP stapling by importing an OCSP response or quote an OCSP profile.

In a stapling scenario, the certificate holder queries the OCSP server themselves at regular intervals, obtaining a signed time-stamped OCSP response. When the site's visitors attempt to connect to the site, this response is included ("stapled") with the TLS/SSL Handshake via the Certificate Status Request extension response. Note that the TLS client must explicitly include a Certificate Status Request extension in its Client Hello TLS/SSL handshake message.

OCSP_staping could be used in a `local_certificate_group`, and the local certificate in OCSP stapling must be the local certificate in the local certificate group.

Syntax

```
config system certificate OCSP_stapling
edit <name>
   set OCSP <datasource>
   set OCSP-response-file <OCSP-response-filename>
   set issuer-certificate <datasource>
   set response-update-ahead-time <integrate>
   set response-update-interval <integrate>
end
```

ocsp	Quote from system certificate OCSP.
ocsp-response	A certificate containing the OCSP response from the OCSP server.
issuer-certificate	The issuer CA of the local certificate.
response-update-ahead-time	The default is 1h (1 hour). Valid values are Xh (hour), Xm (minute), and Xs (second). For example, 5m, 30s (=5 minute and 30 seconds).
response-update-interval	The number of seconds (200 ms by default) that FortiADC waits for a response from the OCSP responder. FortiADC will block the link once it times out.

Example

```
config system certificate OCSP_stapling
     edit "ocsp_staping"
          set issuer-certificate cacert
          set OCSP-response-file ocsp_staping.cer
     next
end
```

config system certificate remote

Use this command to configure a remote certificate. You can enable OCSP by importing an OCSP CA or specifying an OSCP URL. If you want to use the configuration in a certificate verify configuration, you must add both an OCSP CA and URL.

OCSP enables you to validate or revoke certificates by query, rather than by importing certificate revocation list (CRL) files. Since distributing and installing CRL files can be a considerable burden in large organizations, and because delay between the release and install of the CRL represents a vulnerability window, this can often be preferable.

To use OCSP queries, you must first install the certificates of trusted OCSP/CRL servers.

Before you begin:

- You must know the URL of an OCSP server or have downloaded the certificate and key files and be able to browse to them so that you can upload them.
- You must have read-write permission for system settings.

Syntax

```
config system certificate remote
   edit "cert"
        set certificate-file cert.cer
   next
end
```

cert	Paste the contents of a CA file between the quotation marks (" "), as shown in the example below.

Example

```
FortiADC-VM # config system certificate remote
FortiADC-VM (remote) # edit new-remote-ca
FortiADC-VM (new-remote-ca) # set certificates-file new-remote-ca.cer
FortiADC-VM (new-remote-ca) # end
```

See also

- execute certificate remote

config system dns

Use this command to configure DNS.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system dns
   set primary <class_ip>
   set secondary <class_ip>
end
```

primary	Specify the IP address for the primary DNS server.
secondary	Specify the IP address for the secondary DNS server.

Example

```
FortiADC-VM # get system dns
primary : 8.8.8.8
secondary : 0.0.0.0

FortiADC-VM # config system dns
FortiADC-VM (dns) # set secondary 8.8.4.4
FortiADC-VM (dns) # end
```

```
FortiADC-VM # get system dns
primary : 8.8.8.8
secondary : 8.8.4.4
```

config system dns-vdom

Use this command to configure the System DNS resolver for non-root VDOMs and override Global DNS settings to set a DNS server IP per VDOM for more flexibility.

This functionality is only available in the non-root VDOM. The root VDOM defaults to inherit the Global DNS settings.

Note: The DNS settings per VDOM is also not available when VDOMs are in Share Network mode (ADOM mode) as all ADOMs will share the same network namespace.

Before you begin:

- You must have VDOM enabled and have access to a non-root VDOM.
- You must have a good understanding of DNS and knowledge of the remote DNS servers that can be used to communicate with Internet domain servers.
- You must have Read-Write permission for System settings.

Syntax

```
config system dns-vdom
   set dns-override {enable|disable}
   set primary <class_ip>
   set secondary <class_ip>
end
```

`dns-override`	Enable to apply the DNS settings configured for this non-root VDOM instead of inheriting from the Global DNS settings.
`primary`	Specify the IP address for the primary DNS server.
`secondary`	Specify the IP address for the secondary DNS server.

Example

```
config system dns-vdom
   set dns-override enable
   set primary 1.1.1.1
   set secondary 1.1.1.2
end
```

config system external-resource

Use this command to create an IP Address connector that allows you to dynamically import an external block list from an HTTP/HTTPS server in the form of a plain text file. Block lists can be used to enforce special security requirements, such as long term policies to always block access to certain websites, or short term requirements to block access to known compromised locations. The lists are dynamically imported, so that any changes are immediately imported by FortiADC.

After you have imported your external block list through the IP Address connector, you can apply the IPs as the source or destination address for IPv4 and IPv6 firewall policies.

- You cannot delete an IP Address connector or modify its status if the external resource is being used in an IPv4 or IPv6 firewall policy.
- Up to 512 external resources can be supported, however, large numbers of external resources may affect system performance.

Requirements:

- The external block list must be accessible from an HTTP/HTTPS server.
- The import file must be in plain text and each line must contain an IP, IP Range, or Subnet in the below formats:

IP/ IP Range/ Subnet	Example
IPv4	192.168.2.100
IPv4 Range	172.200.1.4/16
IPv4 Subnet	172.16.8.1-172.16.8.100
IPv6	2001:0db8::eade:27ff:fe04:9a01
IPv6 Range	2001:0db8::eade:27ff:fe04:9a01/120
IPv6 Subnet	2001:0db8::eade:27ff:fe04:aa01-2001:0db8::eade:27ff:fe04:ab01

- The maximum import file size is 1 MB (which is about 5000 line entries).

Syntax

```
config system external-resource
   edit <name>
      set resource <string>
      set type address
      set username <string>
      set password <string>
      set refresh-rate <integer>
      set comments <string>
      set status {enable|disable}
   next
end
```

`resource`	Specify the URI of the HTTP/HTTPS server where the IP address list is stored.

username	Specify the username to be used to access this IP address list.
password	Specify the password to be used to access this IP address list.
refresh-rate	Specify the refresh rate in minutes. (Default: 5. Range: 1-43200 minutes). FortiADC will retrieve the data from the HTTP/HTTPS server periodically according to the refresh rate.
comments	Optionally, enter comments about the IP Address connector.
status	Enable/disable the IP Address connector.

Example

```
config system external-resource
   edit "my_ip_list"
      set resource http://10.106.206.243/1.txt
      set type address
      set username test1
      set password passwd
      set refresh-rate 5
      set status enable
   next
end
```

config system fortiguard

Use this command to configure how the FortiADC system receives scheduled updates from FortiGuard services.

FortiGuard periodically updates the WAF Signature Database, IP Reputation Database, and Geo IP Database.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system fortiguard
   set override-server-status {enable|disable}
   set override-server-address <string>
   set tunneling-status {enable|disable}
   set tunneling-dns {enable|disable}
   set tunneling-address <proxy_address>
   set tunneling-password <password>
   set tunneling-port <proxy_port>
   set tunneling-username <string>
   set anycast {enable|disable}
   set anycast-source {fortinet|aws}
   set scheduled-update-day {Sunday | Monday | Tuesday | Wednesday | Thursday | Friday |
        Saturday}
   set scheduled-update-frequency {daily|weekly|every}
```

```
    set scheduled-update-status {enable|disable}
    set scheduled-update-time <hh:mm>
    set update-dldb {enable|disable}
end
```

override-server-status	Enable/disable connection to the override server address.
override-server-address	Override server IP address.
tunneling-status	Enable/disable Web proxy tunneling for FDN. Disabled by default.
tunneling-dns	Enable/disable DNS via web proxy tunneling for FDN.
tunneling-address	Web proxy IP address.
tunneling-password	The password for Web proxy authentication.
tunneling-port	Web proxy port.
tunneling-username	The username for Web proxy authentication.
anycast	Enable/disable anycast fortiguard server. Enabled by default.
anycast-source	Anycast FortiGuard server source. Choose the source between FDS hosted by Fortinet or a mirror hosted by AWS for better performance.
scheduled-update-day	Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday.
scheduled-update-frequency	• Every—Schedule periodic updates. Specify the time to perform the update. • Daily—Schedule daily updates. Specify the time of day to perform the update. • Weekly—Schedule weekly updates. Specify the day and time to perform the update.
scheduled-update-status	Enable/disable scheduled updates.
scheduled-update-time	<hh:mm> hour and minute, hh: 0-23, mm: {00|15|30|45}.
update-dldb	Enable to allow the DLP database to be updated. This is enabled by default. When **update-dldb** is disabled, the scheduled FortiGuard service updates will skip the DLP package updates. In addition, if **update-dldb** is disabled, the `execute update-now` and `execute update-dldb` commands will not trigger the DLP package update.

Example

```
FortiADC-VM # get system fortiguard
scheduled-update-status: enable
scheduled-update-frequency: weekly
scheduled-update-day: Sunday
scheduled-update-time: 04:00
override-server-status: disable
push-update-status : enable
push-update-override-status: disable
tunneling-status : disable
```

```
FortiADC-VM # config system fortiguard
FortiADC-VM (fortiguard) # set scheduled-update-time 23:45
FortiADC-VM (fortiguard) # end

FortiADC-VM # get system fortiguard
scheduled-update-status: enable
scheduled-update-frequency: weekly
scheduled-update-day: Sunday
scheduled-update-time: 23:45
override-server-status: disable
push-update-status : enable
push-update-override-status: disable
tunneling-status : disable
```

See also

- config system web-filter

config system ha

Use this command to configure high availability (HA) settings.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system ha
   set mode {active-active | active-active-vrrp | active-passive | standalone}
   set arps <integer>
   set arps-interval <integer>
   set auto-config-sync {enable|disable}
   set datadev <datasource>
   set group-id <integer>
   set group-name <string>
   set ha-eth-type <4 digit hex>
   set hatrans-eth-type <4 digit hex>
   set hb-interval <integer>
   set hb-lost-threshold <integer>
   set hb-type {multicast|broadcast|unicast}
   set hbdev <datasource>
   set l2ep-eth-type (4 digit hex)
   set http-persistence-pickup {enable|disable}
   set local-node-id <integer>
   set l4-persistence-pickup {enable|disable}
   set l4-session-pickup {enable|disable}
```

```
    set mgmt-status {enable | disable}
    set mgmt-interface <interface>
    set mgmt-ip <ip address>
    set mgmt-ip-allowaccess {https ping ssh snmp http telnet}
    set mgmt-trust-ip {enable | disable}
       config mgmt-trust-ip-list
          edit <name>
             set type {ip-netmask | ip-range}
             set ip-network <ip&netmask>
             set start-ip <ip address>
             set end-ip <ip address>
          next
    set monitor <datasource>
    set node-list {0 1 2 3 4 5 6 7}
    set override {enable|disable}
    set priority <integer>
    set config-priority <integer>
    set remote-ip-monitor {enable|disable}
    set remote-ip-failover-hold-time <integer>
    set remote-ip-failover-threshold <integer>
       config remote-ip-monitor-list
          edit <name>
             set health-check-interval <integer>
             set health-check-retry <integer>
             set health-check-timeout <integer>
             set interface <datasource>
             set remote-address <class_ip>
          next
    end
end
```

mode	• active-active • active-active-vrrp • active-passive • standalone **Note**: If you change this setting, you are logged out of the CLI, and you can log in again if permitted by the new configuration.
arps	Number of times that the cluster member broadcasts extra address resolution protocol (ARP) packets when it takes on the primary role. (Even though a new NIC has not actually been connected to the network, the member does this to notify the network that a new physical port has become associated with the IP address and virtual MAC of the HA cluster.) This is sometimes called “using gratuitous ARP packets to train the network,” and can occur when the primary node is starting up, or during a failover. Also configure ARP Packet Interval. Normally, you do not need to change this setting. Exceptions include: Increase the number of times the primary node sends gratuitous ARP packets if an active-passive cluster takes a long time to fail over or to train the network. Sending more gratuitous ARP packets may help the failover to happen faster.

	Decrease the number of times the primary node sends gratuitous ARP packets if the cluster has a large number of VLAN interfaces and virtual domains. Because gratuitous ARP packets are broadcast, sending them might generate a large amount of network traffic. As long as the active-passive cluster fails over successfully, you can reduce the number of times gratuitous ARP packets are sent to reduce the amount of traffic produced by a failover. The valid range is 1 to 60. The default is 5.
`arps-interval`	Number of seconds to wait between each broadcast of ARP packets. Normally, you do not need to change this setting. Exceptions include: Decrease the interval if an active-passive cluster takes a long time to fail over or to train the network. Sending ARP packets more frequently may help the failover to happen faster. Increase the interval if the cluster has a large number of VLAN interfaces and virtual domains. Because gratuitous ARP packets are broadcast, sending them might generate a large amount of network traffic. As long as the active-passive cluster fails over successfully, you can increase the interval between when gratuitous ARP packets are sent to reduce the rate of traffic produced by a failover. The valid range is from 1 to 20. The default is 6 seconds.
`auto-config-sync`	Enable/disable automatic configuration synchronization. When enabled, synchronization occurs immediately when an appliance joins the cluster, and thereafter every 30 seconds. Disable if you prefer to manage synchronization manually.
`datadev`	Set the network interface to be used for data synchronization among cluster nodes. You can configure up to two data ports. If one data port fails, its traffic fails over to the next data port. If all data ports fail, data synchronization traffic fails over to the heartbeat port. If you do not configure a data port, the heartbeat port is used for synchronization. Use the same port numbers for all cluster members. For example, if you select port3 on the primary node, select port3 as the data port interface on the other member nodes.
`group-id`	Number that identifies the HA cluster. Nodes with the same group ID join the cluster. If you have more than one HA cluster on the same network, each cluster must have a different group ID. The group ID is used in the virtual MAC address that is sent in broadcast ARP messages. The valid range is 0 to 31. The default value is 0.
`group-name`	Name to identify the HA cluster if you have more than one. This setting is optional, and does not affect HA function. The maximum length is 63 characters.
`ha-eth-type`	A Layer-3 protocol number for the HA data channel. It is used for heartbeat packets type, and is also used for Layer-7/Layer-4 session persistence sync.

`hatrans-eth-type`	A Layer-3 protocol number for the HA data channel. It works in active-active (AA) mode, and is used for traffic relay between HA nodes in AA mode.
`hb-interval`	Number of 100-millisecond intervals at which heartbeat packets are sent. This is also the interval at which a node expects to receive heartbeat packets. This part of the configuration is pushed from the primary node to member nodes. The valid range is 1 to 20 (that is, between 100 and 2,000 milliseconds). **Note:** Although this setting is pushed from the primary node to member nodes, you should initially configure all nodes with the same Detection Interval to prevent inadvertent failover from occurring before the initial synchronization.
`hb-type`	Specify whether the destination MAC address of HA message is broadcast, multicast, or unicast (this is only supported in Active-Active-VRRP mode).
`hb-lost-threshold`	Number of times a node retries the heartbeat and waits to receive HA heartbeat packets from the other nodes before concluding the other node is down. This part of the configuration is pushed from the primary node to member nodes. Normally, you do not need to change this setting. Exceptions include: Increase the failure detection threshold if a failure is detected when none has actually occurred. For example, in an active-passive deployment, if the primary node is very busy during peak traffic times, it might not respond to heartbeat packets in time, and a standby node might assume that the primary node has failed. Decrease the failure detection threshold or detection interval if administrators and HTTP clients have to wait too long before being able to connect through the primary node, resulting in noticeable down time. The valid range is from 1 to 60. **Note:** Although this setting is pushed from the primary node to member nodes, you should initially configure all nodes with the same HB Lost Threshold to prevent inadvertent failover from occurring before the initial synchronization.
`hbdev`	Set the network interface to be used for heartbeat packets. You can configure one or two heartbeat ports. Use the same port number for all cluster members. For example, if you select port3 on the primary node, select port3 as the heartbeat interface on the other member nodes. **Note**: If a switch is used to connect the heartbeat interfaces, the heartbeat interfaces must be reachable by Layer 2 multicast.
`l2ep-eth-type`	A Layer-3 protocol number for the HA data channel. It is used for configuration sync, HC result sync, and applications dynamic data.
`http-persistence-pickup`	Enable to synchronize Layer 7 session data used for persistence to backend servers. When enabled, the Source Address Persistence table is synchronized between HA members. When not enabled, a node that receives traffic due to failover would not know that a session had been created already, so it will be treated as a new session.

	Synchronization of the persistence table is not required for cookie-based or hash-based persistence methods to get the desired result. Client traffic will be routed to the same backend server. Synchronization of the persistence table is not possible for SSL session ID. When the session via the first node is terminated, the client must re-establish an SSL connection via the second node. When a client requests a new SSL connection with an SSL server, the initial TCP connection has an SSL Session ID of 0. This zero value tells the server that it needs to set up a new SSL session and to generate an SSL Session ID. The server sends the new SSL Session ID in its response to the client as part of the SSL handshake.
`l4-persistence-pickup`	Enable to synchronize Layer 4 session data used for persistence to backend servers. When enabled, the Source Address Persistence table is synchronized between HA members. When not enabled, a node that receives traffic because of load balancing or failover would not know that a session had been created already, so it will be treated as a new session. Synchronization of the persistence table is not required for hash-based persistence methods to get the desired result. Client traffic will be routed to the same backend server.
`l4-session-pickup`	Enable to synchronize Layer 4 connection state data. When enabled, the TCP session table is synchronized. If subsequent traffic for the connection is distributed through a different cluster node because of failover, the TCP sessions can resume without interruption. When not enabled, a node that receives traffic because of failover would not know that a session had been created already, and the client will be required to re-initialize the connection.
`local-node-id`	A number that uniquely identifies the member within the cluster. The valid range is 0-7. In an active-active deployment, this number is used in the virtual MAC address that is sent in ARP responses. In an active-passive deployment, this number is not used.
`mgmt-status`	This setting must be enabled before other management options can be set.
`mgmt-interface`	Set a management interface.
`mgmt-ip`	Set a management IP address.
`mgmt-ip-allowaccess`	Set which methods are allowed access to the management IP.
`mgmt-trust-ip`	Enable/disable the Trust IPs Access Control (TIAC) feature to restrict access to the management interface according to the Trust IP Address List. If the source IP is not on the Management Trust IP Address List, the device will refuse the client directly. To add IP addresses to the Management Trust IP Address List, enable the feature and save the configuration. After you have saved it the first time, you can edit it to add IPs to the list. **Note**: The Management Trust IP Address List will not be synchronized to peer nodes.

`monitor`	One or more network interfaces that correlate with a physical link. These ports will be monitored for link failure. Port monitoring (also called interface monitoring) monitors physical network ports to verify that they are functioning properly and linked to their networks. You can monitor physical interfaces and 802.3ad aggregated interfaces. **Note:** To prevent an unintentional failover, do not configure port monitoring *until* you configure HA on all appliances and have plugged in the cables to link the physical network ports that will be monitored.
`node-list`	Specify the node IDs for the nodes in the cluster. An active-active cluster can have up to eight members.
`override`	Enable to make Device Priority a more important factor than uptime when selecting the primary node.
`priority`	Number indicating priority of the member node when electing the cluster primary node. The smaller the number, the higher the priority. The valid range is 0 to 9. The default is 5. This setting is optional and does not affect the election of the configuration source. **Note**: By default, unless you enable Override, uptime is more important than this setting.
`config-priority`	Allows you to determine which configuration the system uses when synchronizing the configuration between the HA nodes. It is highly recommended that you use this option to manually set different HA configuration priority values on the nodes. Otherwise, you'll have no control over the system's primary-secondary configuration sync behavior. When the configuration priority values are identical on both nodes (whether by default or by configuration), the system uses the configuration of the appliance with the larger serial number to override that of the appliance with the smaller serial number. When the configuration priority values on the nodes are different, the configuration of the appliance with the lower configuration priority will prevail. Range of acceptable values is 0 to 255. Default is 100.
`remote-ip-monitor`	Enable/disable active monitoring of a beacon remote IP address.
`remote-ip-failover-hold-time`	If failover occurs due to a remote IP monitor test, and this node's role changes (to primary or secondary), it cannot change again until the holdtime elapses. Holdtime can be used to prevent looping.The default holdtime is 120 seconds. The valid range is 60-86400.
`remote-ip-failover-threshold`	Number of unreachable remote-ip-monitor-list to indicate failure. The default is 5. The valid range is 1-64.
config remote-ip-monitor-list	
`health-check-interval`	Seconds between each health check. Should be more than the timeout to prevent overlapping health checks. The default is 10.
`health-check-retry`	Number of retries to confirm up or down. The default is 3 retries. The valid range is 1-10.

health-check-timeout	Seconds to wait for a reply before assuming that the health check has failed. The default is 5.
interface	Interface to send the health check ping.
remote-address	Remote address to ping.
config mgmt-trust-ip-list	
type	Select the IP address type from the following: • ip-netmask • ip-range
ip-network	If ip-netmask is selected as the address type, specify the IP address and CIDR-formatted subnet mask, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.
start-ip	If ip-range is selected as the address type, specify the start of a range of IP addresses and CIDR-formatted subnet masks, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.
end-ip	If ip-range is selected as the address type, specify the end of a range of IP addresses and CIDR-formatted subnet masks, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.

Example

```
FortiADC-VM # get system ha
mode : standalone
hbdev :
datadev :
group-id : 0
group-name :
priority : 5
config-priority : 100
override : disable
hb-interval : 2
arps : 5
hb-lost-threshold : 6
arps-interval : 6
l7-persistence-pickup : disable
l4-persistence-pickup : disable
l4-session-pickup : disable
auto-config-sync : enable
monitor :
remote-ip-monitor : disable
boot-time : 30
ha-eth-type : 8890
hatrans-eth-type : 8892
l2ep-eth-type : 8893
hb-type : multicast

FortiADC-VM # config system ha
FortiADC-VM (ha) # set hbdev port2
FortiADC-VM (ha) # set datadev port3
```

```
FortiADC-VM (ha) # set group-name dc1-pair
FortiADC-VM (ha) # set priority 1
FortiADC-VM (ha) # set mode active-passive
FortiADC-VM (ha) # end

(M) FortiADC-VM # get system ha
mode : active-passive
hbdev : port2
datadev : port3
group-id : 0
group-name : dc1-pair
priority : 1
config-priority : 100
override : disable
hb-interval : 2
arps : 5
hb-lost-threshold : 6
arps-interval : 6
l7-persistence-pickup : disable
l4-persistence-pickup : disable
l4-session-pickup : disable
auto-config-sync : enable
monitor :
remote-ip-monitor : disable
boot-time : 30
ha-eth-type : 8890
hatrans-eth-type : 8892
l2ep-eth-type : 8893
hb-type : multicast
```

config system health-check

Use this command to create health check configuration objects.

In server load balancing deployments, the system uses health checks to poll the members of the real server pool to test whether an application is available. You can also configure additional health checks to poll related servers, and you can include results for both in the health check rule. For example, you can configure an HTTP health check test and a RADIUS health check test. In a web application that requires user authentication, the web server is deemed available only if the web server and the related RADIUS server pass the health check.

In link load balancing deployments, the health check can poll either the ISP link group member itself or a “beacon” server that is deployed on the other side of the ISP link. A beacon is an IP address that must be reachable in order for the link to be deemed available. A beacon can be any IP address, such as a main office, core router, or virtual server at another data center.

If a pool member fails a health check and retries also fail, it is deemed unavailable. The ADC does not send it connections until it is deemed available.

If you expect a backend server is going to be unavailable for a long period, such as when it is undergoing hardware repair, it is experiencing extended down time, or when you have removed it from the server farm, you can improve the performance of the FortiADC system by setting the status of the pool member to Disabled, rather than allowing the system to continue to attempt health checks.

Table 21 describes the predefined health checks. You can get started with these or create custom objects.

Predefined health check configuration objects

Predefined	Description
LB_HLTHCK_HTTP	Sends a HEAD request to the server port 80. Expects the server to return an HTTP 200.
LB_HLTHCK_HTTPS	Sends a HEAD request to the server port 443. Expects the server to return an HTTP 200.
LB_HLTHCK_ICMP	Pings the server.
LB_HLTHCK_TCP_ ECHO	Sends a TCP echo to server port 7. Expects the server to respond with the corresponding TCP echo.

Before you begin:

- You must have a good understanding of TCP/IP and knowledge of the services running on your backend servers.
- You must know the IP address, port, and configuration details for the applications running on backend servers. For some application protocol checks, you must specify user credentials.
- You must have read-write permission for load balancing settings.

After you have configured a health check, you can select it in the server load balacing real server configuration or in the link-load-balancing gateway link configuration.

Syntax

```
config system health-check
   edit <name>
      set type {diameter | dns | ftp | http | https | icmp | imap4 | l2-detection | ldap |
           ldaps | mssql | mysql | oracle | pop3 | radacct | radius | rtsp | script | sip |
           sip-tcp | smtp | snmp | snmp-custom | ssh | tcp | tcp-echo | tcphalf | tcpssl |
           udp}
      set dest-addr-type {ipv4 | ipv6 | fqdn}
      set dest-addr <ipv4 or ipv6>
      set fqdn <fqdn>
      set hostname <string>
      set interval <integer>
      set retry <integer>
      set timeout <integer>
      set up-retry <integer>
      set port <integer>
      set method-type {http_get | http_head}
      set send-string <string>
      set receive-string <string>
      set status-code <integer>
      set match-type {match_all | match_status | match_string}
```

```
    set http-connect {local_connect | no_connect | remote_connect}
    set http-version {http_1.0 | http_1.1}
    set additional-string <additional string for http header content>
    set remote-host <string>
    set remote-port <integer>
    set addr-type {ipv4 | ipv6}
    set domain-name <string>
    set host-addr <class_ip>
    set nas-ip <string>
    set username <username>
    set password <password>
    set password-type {user-password | chap-password}
    set secret-key <string>
    set sip-request-type {register|options}
    set folder <string>
    set file <string>
    set passive {enable|disable}
    set agent-type {UCD|WIN2000}
    set community <string>
    set cpu <integer>
    set disk <integer>
    set mem <integer>
    set version {v1|v2c}
    config snmp-custom
       edit <name>
          set oid <string>
          set value-type {ASN_COUNTER | ASN_INTEGER | ASN_OCTET_STR | ASN_UINTEGER}
          set compare-type {equal | greater | less }
          set counter-value <integer>
          set name <string>
          set weight <integer>
       next
    set origin-host <string>
    set origin-realm <string>
    set vendor-id <integer>
    set product-name <string>
    set host-ip-addr <class_ip>
    set host-ip6-addr <class_ip>
    set auth-appid <integer>
    set acct-appid <integer>
    set connect-data-type {connect_string | service_name | sid}
    set service_name <string>
    set sid <string>
    set connect-string <string>
    set oracle-send-string <string>
    set oracle-receive-string <string>
    set script {<datasource> | CURL_HTTP_CODE | ICMP | PORT_STATUS }
    set attribute <string>
    set baseDN <string>
    set bindDN <string>
    set filter <string>
    set database <string>
    set mssql-send-string <string>
    set mssql-receive-string <string>
    set verify-host-cert {enable | disable}
    set ca <datasource>
next
```

```
end
```

Health check configuration

Settings	Guidelines
General	
`<name>`	Configuration name. No spaces or special characters. After you initially save the configuration, you cannot edit the name.
`type`	Specify the health check type. After you have specified the type, the CLI commands are constrained to the ones that are applicable to the specified type, not all of the settings described in this table. • diameter • dns • ftp • http • https • icmp • imap4 • l2-detection • ldap • ldaps • mssql • mysql • oracle • pop3 • radacct • radius • rtsp • script • sip • sip-tcp • smtp • snmp • snmp-custom • ssh • tcp • tcp-echo • tcphalf • tcpssl • udp
`dest-addr-type`	Specify the destination address type. • ipv4 • ipv6

Settings	Guidelines
	• fqdn — destination FQDN type is only supported for LDAP and LDAPS health check types. **Note**: For the LDAP or LDAPS health check types, if Verify Host Certificate is enabled, the destination address type must match the CN in the LDAP/S server certificate as either IP address or FQDN. For example, if the CN in the LDAP/S server certificate is FQDN, then the destination address in the health check configuration must be FQDN as well.
`dest-addr`	The **dest-addr** option is available if **dest-addr-type** is **ipv4** or **ipv6**. Optional. If no destination IP address is specified, the real server health check is sent to the real server IP address and the gateway link health check is sent to the ISP link IP address. If you are creating rules that test related servers or a test to a “beacon” server, specify the destination IP address. If testing an HTTP proxy, specify the proxy address, not the remote server address.
`fqdn`	The **fqdn** option is available if **dest-addr-type** is **fqdn**. Specify the destination FQDN (Fully Qualified Domain Name).
`hostname`	For HTTP or HTTPS health checks, you can specify the hostname (FQDN) instead of the destination IP address. This is useful in VM environments where multiple applications have the same IP address.
`interval`	Seconds between each health check. Should be more than the timeout to prevent overlapping health checks. The default is 10.
`retry`	Attempts to retry the health check to confirm availability. The default is 1.
`timeout`	Seconds to wait for a reply before assuming that the health check has failed. The default is 5.
`up-retry`	Attempts to retry the health check to confirm availability. The default is 1.
ICMP	
No specific options	Simple ping to test connectivity.
TCP / TCP Half Open / TCP SSL / UDP	
`port`	Listening port number of the backend server. Usually HTTP is 80, FTP is 21, DNS is 53, POP3 is 110, IMAP4 is 143, RADIUS is 1812, and SNMP is 161 or 162.
HTTP/HTTPS	
`port`	Listening port number of the backend server. Usually HTTP is 80. If testing an HTTP proxy server, specify the proxy port.
`method-type`	HTTP method for the test traffic: • HTTP GET—Send an HTTP GET request to the server. A response to an HTTP GET request includes HTTP headers and HTTP body. • HTTP HEAD—Send an HTTP HEAD request. A response to an HTTP

Settings	Guidelines
	HEAD request includes HTTP headers only.
`send-string`	The request URL, such as /contact.php.
`receive-string`	A string expected in return when the HTTP GET request is successful.
`status-code`	The health check sends an HTTP request to the server. Specify the HTTP status code in the server reply that indicates a successful test. Typically, you use status code 200 (OK). Other status codes indicate errors.
`match-type`	What determines a failed health check? • Match String • Match Status • Match All (match both string and status) Not applicable when using HTTP HEAD. HTTP HEAD requests test status code only.
`http-connect`	If the real server pool members are HTTP proxy servers, specify an HTTP CONNECT option: • local_connect—Use HTTP CONNECT to test the tunnel connection through the proxy to the remote server. The member is deemed available if the request returns status code 200 (OK). • remote_connect—Use HTTP CONNECT to test both the proxy server response and remote server application availability. If you select this option, you can configure an HTTP request within the tunnel. For example, you can configure an HTTP GET/HEAD request to the specified URL and the expected response. • no_connect—Do not use the HTTP CONNECT method. This option is the default. The HTTP CONNECT option is useful to test the availability of proxy servers only. See the FortiADC Deployment Guide for FortiCache for an example that uses this health check.
`http-version`	Specify the version of HTTP
`additional-string`	attach some string to HTTP header content
`remote-host`	If you use HTTP CONNECT to test proxy servers, specify the remote server IP address.
`remote-port`	If you use HTTP CONNECT to test proxy servers, specify the remote server port.
DNS	
`addr-type`	IPv4 or IPv6
`domain-name`	The FQDN, such as www.example.com, to use in the DNS A/AAAA record health check.
`host-addr`	IP address that matches the FQDN, indicating a successful DNS health check.

Settings	Guidelines
RADIUS / RADIUS Accounting	
`port`	Listening port number of the backend server. Usually RADIUS is 1812 and RADIUS accounting is 1813.
`nas-ip`	NAS IP address.
`username`	User name of an account on the backend server.
`password`	The corresponding password.
`password-type`	• User—If the backend server does not use CHAP, select this option. • CHAP—If the backend server uses CHAP and does not require a secret key, select this option.
`secret-key`	The secret set on the backend server.
SIP / SIP TCP	
`sip-request-type`	Specify the SIP request type to be used for health checks: • register • options
`status-code`	The expected response code. If not set, response code 200 is expected. Specify `0` if any reply should indicate the server is available.
SMTP	
`port`	Listening port number of the backend server. Usually SMTP is 25.
`domain-name`	The FQDN, such as www.example.com, to use in the SMTP health check.
POP3	
`port`	Listening port number of the backend server. Usually POP3 is 110.
`username`	User name of an account on the backend server.
`password`	The corresponding password.
IMAP4	
`port`	Listening port number of the backend server. Usually IMAP4 is 143.
`username`	User name of an account on the backend server.
`password`	The corresponding password.
`folder`	Specify a mail folder name. The default is INBOX.
FTP	
`port`	Listening port number of the backend server. Usually FTP is 21.
`username`	User name of an account on the backend server.
`password`	The corresponding password.

Settings	Guidelines
file	Specify a file that exists on the backend server. Path is relative to the initial login path. If the file does not exist or is not accessible, the health check fails.
passive	Select this option if the backend server uses passive FTP.
SNMP	
port	Listening port number of the backend server. Usually SNMP is 161.
agent-type	UCD or Windows 2000
community	Must match the SNMP community string set on the backend server. If this does not match, all SNMP health checks fail.
cpu	Maximum normal CPU usage. If overburdened, the health check fails.
disk	Maximum normal disk usage. If the disk is too full, the health check fails.
mem	Maximum normal RAM usage. If overburdened, the health check fails.
version	SNMP v1 or v2c.
SNMP Custom	
port	Listening port number of the backend server. Usually SNMP is 161 or 162.
community	Must match the SNMP community string set on the backend server. If this does not match, all SNMP health checks fail.
version	SNMP v1 or v2c.
config snmp-custom	
oid	String specifying the OID to query.
value-type	Abstract syntax notation (ASN) value type: • ASN_COUNTER • ASN_INTEGER • ASN_OCTET_STR • ASN_UINTEGER
compare-type	• equal • greater • less The default option is **less**.
counter-value	Specify the value for the evaluation. The range is 1-2147483647.
name	Specify the SNMP custom name.
weight	Specify the SNMP custom weight.
SSH	
port	Listening port number of the backend server. Usually SSH is 22.
username	Username for test login.

Settings	Guidelines
password	Corresponding password.
L2 Detection	
No specific options	Link Layer health checker. Sends ARP (IPv4) or NDP (IPv6) packets to test whether a physically connected system is available.
MySQL	
username	Specify the user name of the MySQL database.
password	Specify the password corresponding to the MySQL database user name.
dest-addr	Specify the IP address of the MySQL database server.
port	Listening port number of the backend server.
Diameter	
origin-host	Specify the FortiADC appliance that originates the Diameter message. The value is in FQDN format and used to uniquely identify a Diameter node for duplicate connection and routing loop detection. Note: Some Diameter servers do not accept multiple connections from the same origin host. If you set the origin host the same as the origin host (Identity) of the Diameter load-balance profile and use the health check and Diameter load balance profile in the same virtual server, the health check or the Diameter load-balance profile may run into certain undefined problems.
origin-realm	Specify the realm of the FortiADC appliance that originates the Diameter message. The value is in FQDN format.
vendor-id	Specify the type Unsigned32 vendor ID which contains the IANA "SMI Network Management Private Enterprise Codes" value assigned to the vendor of a Diameter application. The default is 12356.
product-name	Specify the type UTF8String product name which contains the vendor assigned name for the product.
host-ip-addr	Specify the type IPv4 address used to inform a Diameter peer of the sender's IP address when the destination address type is IPv4. The default is blank, meaning that it is the address of the FortiADC's outgoing interface.
host-ip6-addr	Specify the type IPv6 address used to inform a Diameter peer of the sender's IP address when the destination address type is IPv6. The default is blank, meaning that it is the address of the FortiADC's outgoing interface.
auth-appid	Specify the type Unsigned32 authentication application ID used to advertise support of the authentication and authorization portion of an application. This filed is optional; the default is 0 (zero).
acct-appid	Specify the type Unsigned32 accounting application ID used to advertise support of the accounting portion of an application. This field is optional; the default is 0 (zero).

Settings	Guidelines
Oracle	
port	Listening port number of the OracleDB server
username	Specify the database username
password	Specify the database password
connect-data-type	Select either of the following: • service_name • sid • connect_string Setting these configurations depends on the configuration of the server.
service_name	When you select a Service name, use this to specify the Service name.
sid	When you select an SID, use this to specify the SID.
connect-string	When you select a service name, use this to specify connect string.
oracle-send-string	Send a string (command) to OracleDb server.
oracle-receive-string	The string we expect to receive.
row	The row in which the send string (command) takes effect.
column	The column in which the send string (command) takes effect.
Script	
port	Specify the port that is used by the script.
script	Specify the script we create or pre-define. Predefined scripts: • CURL_HTTP_CODE • ICMP • PORT_STATUS
LDAP	
port	Port Listening port number of the backend server. Usually LDAP is 389.
password	The corresponding password.
attribute	Attributes for the LDAP health check object.
baseDN	The distinguished name where a LDAP server will search from.
bindDN	The distinguished name used to bind to a LDAP server.
filter	Criteria to use in selecting results.

Settings	Guidelines
MSSQL	
port	Specify the listening port number of the MSSQL server. Valid values range from 0 to 65535.
username	Specify the database user name. (Optional)
password	Specify the database password, if applicable.
database	Specify the name of the MSSQL database.
mssql-send-string	Specify the MSSQL send string.
mssql-receive-string	Specify the MSSQL receive string.
row	The row in which the send string (command) takes effect.
column	The column in which the send string (command) takes effect.
LDAPS	
port	Port Listening port number of the backend server. The default port is 636 for LDAPS.
password	The corresponding password.
bindDN	The distinguished name used to bind to a LDAPS server.
baseDN	The distinguished name where a LDAPS server will search from.
filter	Criteria to use in selecting results.
attribute	Attributes for the LDAPS health check object.
verify-host-cert	Enable to verify the LDAPS server certificate. This is disabled by default.
ca	The **CA** option is available if **Verify Host Certificate** is **enabled**. Specify the CA certificate.

In SLB deployments, a health check port configuration specifying port 0 acts as a wildcard.The port for health check traffic is imputed from the real server pool member.

In LLB and GLB deployments, specifying port 0 is invalid because there is no associated configuration to impute a proper port. If your health check port configuration specifies port 0, you will not be able to use it in an LLB or GLB configuration.

Example

The following is an example of an HTTP health check for HTTP proxy servers:

```
FortiADC-VM # config system health-check
FortiADC-VM (health-check) # edit HTTP-CONNECT-TEST
Add new entry 'HTTP-CONNECT-TEST' for node 2763
```

```
FortiADC-VM (HTTP-CONNECT-T~S) # set type http
FortiADC-VM (HTTP-CONNECT-T~S) # set http-connect remote_connect

FortiADC-VM (HTTP-CONNECT-T~S) # get
type : http
interval : 10
timeout : 5
retry : 1
up-retry : 1
port : 0
dest-addr-type : ipv4
dest-addr : 0.0.0.0
method-type : http_head
send-string : /
status-code : 200
http-connect : remote_connect
remote-host :
remote-port : 0

FortiADC-VM (HTTP-CONNECT-T~S) # set remote-host 10.1.1.1
FortiADC-VM (HTTP-CONNECT-T~S) # set remote-port 113
FortiADC-VM (HTTP-CONNECT-T~S) # set send-string /myapp/index.html
FortiADC-VM (HTTP-CONNECT-T~S) # end
FortiADC-VM #
```

The following is an example of a SIP health check:

```
FortiADC-VM # config system health-check
FortiADC-VM (health-check) # edit sip-health-check
Add new entry 'sip-health-check' for node 2763
FortiADC-VM (sip-health-check) # set type sip
FortiADC-VM (sip-health-check) # get
type : sip
interval : 10
timeout : 5
retry : 1
up-retry : 1
port : 0
dest-addr-type : ipv4
dest-addr : 0.0.0.0
status-code : 200
sip-request-type : register
FortiADC-VM (sip-health-check) # set interval 15
FortiADC-VM (sip-health-check) # set retry 2
FortiADC-VM (sip-health-check) # set timeout 3
FortiADC-VM (sip-health-check) # set status-code 403
FortiADC-VM (sip-health-check) # end
```

The following is an example of an SNMP health check for a server running the UCD agent:

```
FortiADC-VM # config system health-check
FortiADC-VM (health-check) # edit lb-health-check
Add new entry 'lb-health-check' for node 2763
FortiADC-VM (lb-health-check) # set type snmp
FortiADC-VM (lb-health-check) # get
type : snmp
interval : 10
```

```
timeout : 5
retry : 1
up-retry : 1
port : 0
dest-addr-type : ipv4
dest-addr : 0.0.0.0
cpu : 96
mem : 96
disk : 96
agent-type : UCD
community :
version : v1

FortiADC-VM (lb-health-check) # set community company-string
FortiADC-VM (lb-health-check) # set port 161
FortiADC-VM (lb-health-check) # set cpu 50
FortiADC-VM (lb-health-check) # set mem 50
FortiADC-VM (lb-health-check) # set disk 50
FortiADC-VM (lb-health-check) # set version v2c

FortiADC-VM (lb-health-check) # get
type : snmp
interval : 10
timeout : 5
retry : 1
up-retry : 1
port : 161
dest-addr-type : ipv4
dest-addr : 0.0.0.0
cpu : 50
mem : 50
disk : 50
agent-type : UCD
community : company-string
version : v2c

FortiADC-VM (lb-health-check) # end
```

The following example configures a custom SNMP health check for a server that does not support the UCD or Windows 2000 agent type.

```
FortiADC-VM # config system health-check
FortiADC-VM (health-check) # edit snmp-linux
Add new entry 'snmp-linux' for node 2763
FortiADC-VM (snmp-linux) # set type snmp-custom
FortiADC-VM (snmp-linux) # get
type : snmp-custom
interval : 10
timeout : 5
retry : 1
up-retry : 1
port : 0
dest-addr-type : ipv4
dest-addr : 0.0.0.0
community :
version : v1
oid :
```

```
value-type :
FortiADC-VM (snmp-linux) # set version v2c
FortiADC-VM (snmp-linux) # set oid ".1.3.6.1.4.1.2021.10.1.3.1"
FortiADC-VM (snmp-linux) # set value-type ASN_INTEGER
FortiADC-VM (snmp-linux) # set compare-type greater
FortiADC-VM (snmp-linux) # set counter-value 80
FortiADC-VM (snmp-linux) # end
FortiADC-VM #
```

config system health-check-script

This command is deprecated. You must use the web UI to upload a script file.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system health-check-script
```

Example

```
FortiADC-VM # config system health-check-script


FortiADC-VM (health-check-script) # edit ?
name health check script name

FortiADC-VM (health-check-script) # edit hcscript_name
Add new entry 'hcscript_name' for node 2800

FortiADC-VM (hcscript_name) # set
Parsing error at 'set'. err=1
```

config system interface

Use this command to configure network interfaces.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system interface
   edit port1
      set floating {enable|disable}
      set floating-ip <string>
      set traffic-group <string>
      set allowaccess {http https ping snmp ssh telnet}
      set ip <ip&netmask>
      set ip6 <ip&netmask>
      set mac-addr <xx:xx:xx:xx:xx:xx>
      set mode {static|pppoe|DHCP}
      set disc-retry-timeout <integer>
      set dns-server-override {enable|disable}
      set idle-timeout <integer>
      set lcp-echo-interval <integer>
      set lcp-max-echo-fails <integer>
      set pppoe-default-gateway {enable|disable}
      set username <string>
      set password <passwd>
      set mtu <integer>
      set retrieve_physical_hwaddr {enable|disable}
      set speed {10full | 10half | 100full | 100half | 1000full | 1000half | auto}
      set status {down | up}
      set vdom <datasource>
      set type {vlan|aggregate}
      set retrieve_dhcp_gateway {enable | disable}
      set dhcp-gateway-distance <integer>
      set vlanid <integer>
      set interface <datasource>
      set aggregate-algorithm {layer2 | layer2-3 | layer3-4}
      set aggregate-mode {802.3ad | balance-alb | balance-rr | balance-tlb | balance-xor|
           broadcast}
      set member <datasource>
      set secondary-ip {enable|disable}
      config secondary-ip-list
         edit 1
            set allowaccess {http https ping snmp ssh telnet}
            set ip <ip&netmask>
            set floating {enable|disable}
            set floating-ip <string>
            set traffic-group <string>
         next
      end
      config ha-node-ip-list
         edit <No.>
            set ip <ip&netmask>
            set node <integer>
            set allowaccess {http https ping snmp ssh telnet}
         next
      end
      set ha-node-secondary-ip {enable|disable}
      config ha-node-secondary-ip-list
         edit <No.>
            set ip <ip&netmask>
            set node <integer>
            set allowaccess {http https ping snmp ssh telnet}
```

```
      next
    end
    set trust-ip {enable|disable}
    config trust-ip-list
      edit <name>
        set type {ip-netmask|ip-range}
        set ip-network <ip&netmask>
        set start-ip <ip>
        set end-ip <ip>
      next
      edit <name>
        set type {ip6-netmask|ip6-range}
        set ip6-network <ip6&netmask>
        set start-ip6 <ip6>
        set end-ip6 <ip6>
      next
    end
  next
end
```

Note: Since the 4.7.0 release, two new interface types (i.e., loop-back and soft-switch) have been supported. When setting the interface type to soft-switch, be sure to set the member ports, as illustrated in the commands below:

```
config system interface
  edit "testint"
    set type loopback| aggregate| soft-switch | vlan
    set member port8 port9
    … …
  next
end
```

Note: In the 6.2.0 release, the inter-VDOM routing feature was introduced, allowing the traffic to be sent between VDOMs without additional physical interfaces that was previously required for multiple VDOM setups. You can create a VDOM-link pair using the `config system vdom-link` command. This will create a VDOM-link pair in the system interface. However, by default, these VDOM links will not be assigned an IP address or `allowaccess` options, so you would not be able to route traffic between the VDOM links until these settings are configured.

Use the following commands to configure the interface settings for the VDOM links:

```
config system interface
  edit <vdom-link-name0>
    set type vdom-link
    set vdom <vdom-name>
    set ip <ip&netmask>
    set allowaccess {http https ping snmp ssh telnet}
    set description "***"
  next
  edit <vdom-link-name1>
    set type vdom-link
    set vdom <vdom-name>
    set ip <ip&netmask>
    set allowaccess {http https ping snmp ssh telnet}
    set description "***"
  next
end
```

allowaccess	Allow inbound service traffic. Select from the following options: • HTTP—Enables connections to the web UI. We recommend this option only for network interfaces connected to a trusted private network, or directly to your management computer. • HTTPS—Enables secure connections to the web UI. We recommend this option instead of HTTP. • Ping—Enables ping and traceroute to be received on this network interface. When it receives an ECHO_REQUEST (“ping”), FortiADC will reply with ICMP type 0 (ECHO_RESPONSE or “ping”). • SNMP—Enables SNMP queries to this network interface. • SSH—Enables SSH connections to the CLI. We recommend this option instead of Telnet. • Telnet—Enables Telnet connections to the CLI. We recommend this option only for network interfaces connected to a trusted private network, or directly to your management computer.
mac-addr	The MAC address is read from the interface. If necessary, you can set the MAC address.
retrieve_physical_hwaddr	Enable or disable.
mtu	The default is 1500. We recommend you maintain the default.
speed	Select one of the following speed/duplex settings: • Auto—Speed and duplex are negotiated automatically. Recommended. • 10half—10 Mbps, half duplex. • 10full—10 Mbps, full duplex. • 100half—100 Mbps, half duplex. • 100full—100 Mbps, full duplex. • 1000half—1000 Mbps, half duplex. • 1000full—1000 Mbps, full duplex.
status	This Status column is not the detected physical link status; it is the administrative status (Up/Down) that indicates whether you permit the network interface to receive and/or transmit packets.
vdom	If applicable, select the virtual domain to which the configuration applies.
mode	• Static—Specify a static IP address. The IP address must be on the same subnet as the network to which the interface connects. Two network interfaces cannot have IP addresses on the same subnet (i.e. overlapping subnets). • PPPoE—Use PPPoE to retrieve a configuration for the IP address, gateway, and DNS server. For example, if this interface uses a DSL connection to the Internet, your ISP may require this option.
type	If you are editing the configuration for a physical interface, you cannot set the type. If you are configuring a logical interface, you can select from the following options: • Aggregate—A logical interface you create to support the aggregation of multiple physical interfaces. • VLAN—A logical interface you create to VLAN subinterfaces on a single physical

	interface.
set mode static	
ip	Specify the IP address and CIDR-formatted subnet mask, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.
ip6	Specify the IP address and CIDR-formatted subnet mask, separated by a forward slash (/), such as 2001:0db8:85a3:::8a2e:0370:7334/64. Dotted quad formatted subnet masks are not accepted.
floating	Enable/Disable floating IP.
floating-ip	Enter the floating IP. Available only if **floating** is enabled. **Note**: Ensure the Floating IP is different from the Interface IP, otherwise network issues will occur due to the interface/port conflict.
traffic-group	Specify the traffic group object.
set mode pppoe	
disc-retry-timeout	Seconds the system waits before it retries to discover the PPPoE server.
dns-server-override	Use the DNS addresses retrieved from the PPPoE server instead of the one configured in the FortiADC system settings.
idle-timeout	Disconnect after idle timeout in seconds. The default is 0. The valid range is 0 to 32,000.
lcp-echo-interval	LCP echo interval in seconds. The default is 5. The valid range is 1 to 255.
lcp-max-echo-fails	Maximum missed LCP echo messages before disconnect. The default is 3. The valid range is 1 to 255.
pppoe-default-gateway	Use the default gateway retrieved from the PPPoE server instead of the one configured in the FortiADC system settings.
username	PPPoE account user name.
password	PPPoE account password.
set type vlan	
vlanid	VLAN ID of packets that belong to this VLAN. If one physical network port (that is, a VLAN trunk) will handle multiple VLANs, create multiple VLAN subinterfaces on that port, one for each VLAN ID that will be received. If multiple different physical network ports will handle the same VLANs, on each of the ports, create VLAN subinterfaces that have the same VLAN IDs. The valid range is between 1 and 4094. The value you specify must match the VLAN ID added by the IEEE 802.1q-compliant router or switch connected to the VLAN subinterface.
interface	Physical interface associated with the VLAN; for example, port2.

set type aggregate	
`aggregate-algorithm`	Connectivity layers that will be considered when distributing frames among the aggregated physical ports: • Layer 2 • Layer 2-3 • Layer 3-4
`aggregate-mode`	Link aggregation type: • 802.3ad • Balance-alb • Balance-rr • Balance-tlb • Balance-xor • Broadcast
`member`	Specify the physical interfaces that are included in the aggregation.
set type loopback	Set as the loopback interface, which is used by other features, such as VS,1-1 NAT, GLB, VT, OSPF, BGP, etc.
set type soft-switch	Set the interface type used for transparent mode. All interfaces that belong to the same soft-switch will be in the same broadcast domain. Use of a soft-switch can greatly simplify customer deployment because they do not have to change their network topologies when adding new FortiADC devices to their environment.
config secondary-ip-list	
`allowaccess`	Allow inbound service traffic. Specify a space-separated list of the following options: • HTTP—Enables connections to the web UI. We recommend this option only for network interfaces connected to a trusted private network, or directly to your management computer. • HTTPS—Enables secure connections to the web UI. We recommend this option instead of HTTP. • Ping—Enables ping and traceroute to be received on this network interface. When it receives an ECHO_REQUEST (“ping”), FortiADC will reply with ICMP type 0 (ECHO_RESPONSE or “pong”). • SNMP—Enables SNMP queries to this network interface. • SSH—Enables SSH connections to the CLI. We recommend this option instead of Telnet. • Telnet—Enables Telnet connections to the CLI. We recommend this option only for network interfaces connected to a trusted private network, or directly to your management computer.
`ip`	Secondary IP addresses can be used when you deploy the system so that it belongs to multiple logical subnets. If you assign multiple IP addresses to an interface, you must assign them static addresses. To add secondary IP addresses, enable the feature and save the configuration. After you have saved it the first time, you can edit it to add secondary IP addresses and enable inbound traffic to that address.

config ha-node-ip-list	
allowaccess	Enable inbound service traffic on the IP address for the specified services.
ip	You use the HA node IP list configuration in an HA active-active deployment. For each HA cluster node, configure an HA node IP list that includes an entry for each cluster node. When the appliance is in standalone mode, it uses the physical port IP address; when it is in HA mode, it uses the HA node IP address. For each address, specify an IP address using the CIDR-formatted subnet mask, separated by a forward slash (/), such as 192.0.2.5/24.
node	ID of the corresponding node.
config ha-node-secondary-ip-list	
allowaccess	Enable inbound service traffic on the IP address for the specified services.
ip	You use the HA node secondary IP list configuration if the interfaces of the nodes in an HA active-active deployment are configured with secondary IP addresses. For each address, specify an IP address using the CIDR-formatted subnet mask, separated by a forward slash (/), such as 192.0.2.5/24.
node	ID of the corresponding node.
config trust-ip-list	
type	Select the IP address type from the following: • ip-netmask • ip-range • ip6-netmask • ip6-range
ip-network	If ip-netmask is selected as the address type, specify the IP address and CIDR-formatted subnet mask, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.
ip6-network	If ip6-netmask is selected as the address type, specify the IP address and CIDR-formatted subnet mask, separated by a forward slash (/), such as 2001:0db8:85a3:::8a2e:0370:7334/64. Dotted quad formatted subnet masks are not accepted.
start-ip	If ip-range is selected as the address type, specify the start of a range of IP addresses and CIDR-formatted subnet masks, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.
end-ip	If ip-range is selected as the address type, specify the end of a range of IP addresses and CIDR-formatted subnet masks, separated by a forward slash (/), such as 192.0.2.5/24. Dotted quad formatted subnet masks are not accepted.
start-ip6	If ip6-range is selected as the address type, specify the start of a range of IP addresses and CIDR-formatted subnet mask, separated by a forward slash (/), such as 2001:0db8:85a3:::8a2e:0370:7334/64. Dotted quad formatted subnet masks are not accepted.

end-ip6	If ip6-range is selected as the address type, specify the end of a range of IP addresses and CIDR-formatted subnet mask, separated by a forward slash (/), such as 2001:0db8:85a3:::8a2e:0370:7334/64. Dotted quad formatted subnet masks are not accepted.

Example

The following example configures port1 (the management interface):

```
FortiADC-VM # get system interface port1
type : physical
mode : static
vdom : root
redundant-primary :
ip : 192.168.1.99/24
ip6 : ::/0
allowaccess : https ping ssh snmp http telnet
mtu : 1500
speed : auto
status : up
mac-addr : 00:0c:29:e8:a0:86
secondary-ip : enable

FortiADC-VM # config system interface
FortiADC-VM (interface) # edit port1
FortiADC-VM (port1) # set ip 192.0.2.5/24
FortiADC-VM (port1) # end

FortiADC-VM # get system interface port1
type : physical
mode : static
vdom : root
redundant-primary :
ip : 192.0.2.5/24
ip6 : ::/0
allowaccess : https ping ssh snmp http telnet
mtu : 1500
speed : auto
status : up
mac-addr : 00:0c:29:e8:a0:86
secondary-ip : enable
trust-ip: enable

config system interface
   edit port1
      set floating enable
      set floating-ip 172.1.1.1
      set traffic-group traffic-group-1
      set secondary-ip enable
      config secondary-ip list
         edit 1
            set allow ping icmp http https
            set floating enable
            set floating-ip 67.1.1.1
```

```
            set traffic-group traffic-group-2
         next
      end
      config trust-ip-list
         edit 1
            set type ip-netmask
            set ip-network 192.1.1.1/32
         next
         edit 2
            set type ip6-netmask
            set ip6-network 2001:0db8:85a3::8a2e:0370:7334/64
         next
         edit 3
            set type ip-range
            set start-ip 192.1.1.1
            set end-ip 255.255.255.255
         next
         edit 4
            set type ip6-range}
            set start-ip6 ::
            set end-ip6 FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
         next
      end
```

The following example configures vlan interfaces on port7:

```
FortiADC-VM # config system interface
FortiADC-VM (interface) # edit vlan102
Add new entry 'vlan102' for node 1
FortiADC-VM (vlan102) # set type vlan
FortiADC-VM (vlan102) # set vlanid 102
FortiADC-VM (vlan102) # set ip 10.10.100.102/32
FortiADC-VM (vlan102) # set interface port7
FortiADC-VM (vlan102) # next

FortiADC-VM (interface) # edit vlan103
Add new entry 'vland103' for node 1
FortiADC-VM (vland103) # set type vlan
FortiADC-VM (vland103) # set vlanid 103
FortiADC-VM (vland103) # set ip 10.10.103.102/32
FortiADC-VM (vland103) # set interface port7
FortiADC-VM (vland103) # end

FortiADC-VM # get system interface

== [ vlan102 ]
type: vlan
vdom: root
redundant-primary: 0
ip: 10.10.100.102/32
ip6: ::/0
allowaccess:
status: up
interface: port7
== [ vlan103 ]
type: vlan
vdom: root
```

```
redundant-primary: 0
ip: 10.10.103.102/32
ip6: ::/0
allowaccess:
status: up
interface: port7
```

config system isp-addr

Use this command to amend the predefined and restored ISP address books, or to configure new ISP address books.

The following policies use the ISP address book objects:

- ISP routes
- LLB proximity routes
- LLB policies
- GLB data center configuration

ISP address books contain IP subnet addresses and associated province location settings for ISP links. The province setting is used in GLB deployments in China to enable location awareness that is province-specific. For example, a user can be directed to a datacenter in Beijing or Guangdong rather than simply China.

Figure 4 shows the three types of address book entries:

- Predefined—Addresses and associated province location settings for China Mobile, China Telecom, and China Unicom. The IP subnet addresses in the predefined address books are not exposed in the user interface. The predefined package is provided to make it easier for you to configure a route when all you know and all you need to know is the name of the ISP that hosts the link.
- Restored—Addresses imported from a text file. The IP subnet addresses in the restored address books are not exposed in the user interface. “Restored” addresses can help you rapidly build an ISP address book configuration.
- User-defined—In the ISP address configuration, you can modify the predefined and restored address books by specifying subnets to add or exclude from them. This gives you flexibility in case you encounter address conflicts or the ISP instructs you to add a subnet address manually. You can also create new user-defined entries for other ISPs.

In systems with multiple VDOMs, these commands apply to the current VDOM only. In other words, if you configure an exclusion, it is applicable to the current VDOM only; it does not change the predefined address book.

You can use the execute isplookup command to see whether an IP address belongs to any of the address books. If an address is can be found in more than one address book, the results are returned in the following priority: user-defined, restored, predefined.

ISP address book types

The text file for the Restored entries has the following format:

```
#this is a comment line
ISP name:ABC
Province:Beijing
```

```
1.1.1.0/24
Province:Unknown
2.2.0.0 255.255.0.0
#this is a comment line too
3.3.3.3/32
ISP name:DEF
Province:Shanghai
4.4.4.0 255.255.255.0
5.5.0.0/16
```

You use the execute restore command to import the file and the execute backup command to export it.

You use the execute clean command to erase entries that were imported from the text file. The clean operation does not affect the predefined addresses or user-configured entries. If a restored entry has user-configured elements (for example, an exclude list), the clean operation clears the addresses but preserves the configuration and converts it to a user-defined type.

Basic Steps

1. Create address objects.
2. Specify them when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system isp-addr
   edit china-mobile
      config exclude-address
         edit <No.>
            set ip-netmask <ip&netmask>
         next
      end
      config address
         edit <No.>
            set ip-netmask <ip&netmask>
            set province <datasource>
         next
      end
   next
   edit china-telecom
      config exclude-address
         edit <No.>
            set ip-netmask <ip&netmask>
         next
      end
      config address
         edit <No.>
            set ip-netmask <ip&netmask>
            set province <datasource>
         next
      end
   next
   edit china-unicom
```

```
      config exclude-address
         edit <No.>
            set ip-netmask <ip&netmask>
         next
      end
      config address
         edit <No.>
            set ip-netmask <ip&netmask>
            set province <datasource>
         next
      end
   next
   edit <name>
      config address
         edit <No.>
            set ip-netmask <ip&netmask>
            set province <datasource>
         next
      end
   next
end
```

ip-netmask	Specify addresses to exclude or add using the address/mask notation.
province	Specify the associated province location. The configuration supports the following selections:

Anhui	Henan	Shanxi(taiyuan)
Beijing	Hubei	Shanxi(xian)
Chongqing	Hunan	Sichuan
Fujian	Jiangsu	Tianjin
Gansu	Jiangxi	Xianggang
Guangdong	Jilin Liaoning	Xinjiang
Guangxi	Neimenggu	Xizang
Guizhou	Ningxia	Yunnan
Hainan	Qinghai	Zhejiang
Hebei	Shandong	Unknown
Heilongjiang	Shanghai	

Note: Each VDOM can have up to 32 main entries.

Example

```
FortiADC-VM # config system isp-addr
FortiADC-VM (isp-addr) # edit china-mobile
FortiADC-VM (china-mobile) # get
type : predef

FortiADC-VM (china-mobile) # config address

FortiADC-VM (address) # edit 1
```

```
Add new entry '1' for node 2739

FortiADC-VM (1) # get
ip-netmask : 0.0.0.0/0
province :

FortiADC-VM (1) # set ip-netmask 192.168.1.0/24
FortiADC-VM (1) # set province Beijing
FortiADC-VM (1) # end
FortiADC-VM (china-mobile) # end
```

See also

- execute isplookup
- execute backup
- execute clean
- execute restore
- config system setting

config system mailserver

Use this command to configure an SMTP email server if you want to send notifications by email.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system mailserver
   set address <string>
   set port <integer>
   set security {starttls|none}
   set smtp-auth {enable|disable}
   set username <string>
   set password <passwd>
end
```

address	IP address or FQDN of an SMTP server (such as FortiMail) or email server that the appliance can connect to in order to send alerts and/or generated reports.
port	Listening port number of the server. Usually, SMTP is 25.
security	STARTTLS is an extension to plain text communication protocols. It enables a plain text connection to be upgraded to an encrypted (TLS or SSL) connection instead of using a separate port for encrypted communication. Specify this option if you have implemented STARTTLS for your mailserver; otherwise, leave unset or specify none.

smtp-auth	Enable if the SMTP server requires authentication.
username	Username for authentication to the SMTP server.
password	Password for authentication to the SMTP server.

Example

```
FortiADC-VM # get system mailserver
address :
port : 25
security:
smtp-auth : enable
username :
password : *

FortiADC-VM # config system mailserver
FortiADC-VM (mailserver) # set address 192.168.1.125
FortiADC-VM (mailserver) # set username admin
FortiADC-VM (mailserver) # set password strongpass
FortiADC-VM (mailserver) # end

FortiADC-VM # get system mailserver
address : 192.168.1.125
port : 25
security:
smtp-auth : enable
username : admin
password : *
```

config system one-click-glb-server

Use this command to configure the Fabric Device Settings for the FortiGSLB Connector.

Syntax

```
config system one-click-glb-server
   set status {enable|disable}
   set interval <integer>
   set cloud-server-url <URL>
end
```

status	Enable/disable the FortiGSLB function.
interval	Specify how often FortiADC should try to connect to the FortiGSLB. (Range: 10-1800; default: 15).
cloud-server-url	Specify the URL of the cloud server.

config system overlay-tunnel

Use this command to configure an overlay tunnel.

FortiADC support two types of overlay protocols—VXLAN and NVGRE.

- Virtual Extensible LAN (VXLAN) is a network virtualization technology used in large cloud-computing deployments. It encapsulates OSI Layer-2 Ethernet frames within Layer-3 IP packets using the standard destination port 4789. VXLAN endpoints that terminate VXLAN tunnels are known as VXLAN tunnel endpoints (VTEPs), and can be virtual or physical switch ports. For more information, see RFC 7348.
- Network Virtualization using Generic Routing Encapsulation (NVGRE) is a network virtualization technology that attempts to alleviate the scalability problems associated with large cloud-computing deployments. It uses Generic Routing Encapsulation (GRE) to tunnel Layer-2 packets over Layer-3 networks.

Before you begin, make sure that you have read-write permission to configure system settings.

Syntax

```
config system vxlan
   edit <name> <string>
      set type {vxlan|nvgre}
      set interface <datasource>
      set vni <integer>
      set vsid <integer
      set ip-version {ipv4-unicast|ipv4-multicast}
      set dstport <integer>
      set multicast-ttl <integer>
      set destination-ip-addresses <class_ip>
      config remote-host
         edit <No.>
            set host-mac-address <xx:xx:xx:xx:xx:xx>
            set vtep <class_ip>
         next
      end
   next
end
```

type	Select a virtual overlay networking protocol: • VXLAN (default) • NVGRE
interface	The outing interface for VXLAN encapsulated traffic.
dstport	The VXLAN destination port (number). The default is 4789. The valid range is 1–6553.
vni	The VXLAN network ID. The valid range is 1–16777215.
vsid	The NVGRE ID. The valid range is 1–16777215.
ip-version	The IP version to use for the VXLAN interface and for communication over VXLAN. • ipv4-unicast—Use IPv4 unicast addressing over VXLAN or NVGRE.

	• ipv4-multicast—Use IPv4 multicast addressing over VXLAN.
destination-ip-address	Specify the destination IP address. Note: For IPv4 unicast, specify an IPv4 address of the VXLAN interface on the device at the remote end of the VXLAN. You can set multiple VTEP IP addresses, splitting with space char; for IPv4 multicast, specify one multicast IP address only.
multicast-ttl	The option applies to IPv4 multicast IP type only. Specify the multicast TTL. Valid values are from 0 (default) to 255.
remote-host	Add static MAC_to_VTEP to VXLAN mapping table.
host-mac-address	Set the remote host MAC address. The format is xx:xx:xx:xx:xx:xx
vtep	Set the remote VTEP IP address.

Example

The following commands create a VXLAN interface with two VTEP peers:

```
config system overlay-tunnel
   edit "vxlan1"
      set type vxlan
      set interface port2
      set ip-version ipv4-unicast
      set destination-ip-addresses 10.75.0.202 10.75.0.88
      set dstport 4789
      set vni 1122
      config remote-host
      end
next
```

The following commands create a VXLAN interface with a multicast IP:

```
config system overlay-tunnel
   edit "vxlan1"
      set type vxlan
      set interface vlan249
      set ip-version ipv4-multicast
      set destination-ip-addresses 239.1.1.1
      set dstport 4789
      set vni 1122
      config remote-host
         edit 1
            set host-mac-address 22:22:22:22:22:22
            set vtep 3.2.2.2
      end
next
```

The following commands create an NVGRE interface with two remote gateway IPs:

```
config system overlay-tunnel
   edit "nvgre1"
      set type nvgre
```

```
    set interface vlan249
    set ip-version ipv4-unicast
    set destination-ip-addresses 10.75.0.202 10.75.0.88
    set dstport 4789
    set vsid 1122
    config remote-host
    end
  next
```

After creating a VXLAN/NVGRE tunnel, the system will create one interface automatically accordingly.

To diagnose your VXLAN configuration, use the following command:

```
diagnose sys vxlan fdb list vxlan1
(M) FortiADC-VM# diagnose system vxlan-fdb vxlan1
ff:ff:ff:ff:ff:ff dst 10.249.100.31 via vlan249 self permanent
ff:ff:ff:ff:ff:ff dst 10.249.100.38 via vlan249 self permanent
22:22:22:22:22:22 dst 3.2.2.2 via vlan249 self permanent
```

config system password-policy

Use this command to set requirements for administrator passwords.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system password-policy
  set status {enable|disable}
  set apply-to admin-user
  set minimum-length <integer>
  set must-contain {lower-case-letter non-alphanumeric number upper-case-letter}
end
```

`status`	Enable/disable password requirements.
`apply-to admin-user`	Apply the policy to all admin users.
`minimum-length`	Specify a minimum length. The default is 8.
`must-contain`	Specify character requirements.

Example

```
FortiADC-VM # get system password-policy
status : disable

FortiADC-VM # config system password-policy
```

```
FortiADC-VM (password-policy) # set status enable
FortiADC-VM (password-policy) # end

FortiADC-VM # get system password-policy
status : enable
apply-to : admin-user
minimum-length : 8
must-contain :
```

config system schedule-group

Use this command to create schedule objects to use in link load balancing policies. A policy rule can be time-bound: one time, daily, weekly, or monthly.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system schedule-group
   edit <name>
      config schedule-member <No.>
         edit <name>
            set type {daily-recurring | monthly-recurring | one-time | weekly-recurring}
            set endtime-of-enddate <string>
            set starttime-of-startdate <string>
            set day-of-month <integer>
            set enddate <string>
            set startdate <string>
            set day-of-week {friday | monday | saturday | thursday | tuesday | wednesday}
         next
      end
   next
end
```

type	• One Time • Daily • Weekly • Monthly
endtime-of-enddate	HH:MM. Minutes must be 00, 15, 30, or 45.
startime-of-startdate	HH:MM.
day-of-month	1 - 31.
enddate	YYYY/MM/DD.

startdate	YYYY/MM/DD.
day-of-week	Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday.

config system scripting

This command is deprecated. You must use the web UI to upload a script file.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system scripting
```

Example

```
FortiADC-VM # config system scripting


FortiADC-VM (scripting) # edit ?
name scripting name

FortiADC-VM (scripting) # edit script_name
Add new entry 'script_name' for node 2800

FortiADC-VM (script_name) # set
Parsing error at 'set'. err=1
```

config system sdn-connector

Use this command to create a Cloud SDN connector. Cloud SDN connectors provide integration and orchestration of Fortinet products with public and private cloud solutions. In a typical cloud environment, resources are dynamic and often provisioned and scaled on-demand. By using an SDN connector, you can ensure that changes to cloud environment attributes are automatically updated in the Security Fabric.

Syntax

Kubernetes Connector

```
config system sdn-connector
   edit <name>
```

```
      set type kubernetes
      set status {enable | disable}
      set server <server address>
      set server-port <port number>
      set secret-token <string>
      set update-interval <seconds>
   next
end
```

OCI Connector

```
config system sdn-connector
   edit <name>
      set type oci
      set tenant-id <string>
      set compartment-id <string>
      set user-id <string>
      set oci-region <string>
      set oci-cert <string>
      set use-metadata-iam {enable | disable}
      set ha-status {enable | disable}
   next
end
```

AWS Connector

```
config system sdn-connector
   edit <name>
      set type aws
      set status {enable | disable}
      set aws-region <string>
      set aws-accesskey <string>
      set aws-secretkey <string>
      set update-interval <seconds>
      set use-metadata-iam {enable | disable}
   next
end
```

SAP Connector

```
config system sdn-connector
   edit <name>
      set status {enable | disable}
      set server <server address>
      set sap-ms-http-port <port number>
      set sap-icm-http-port <port number>
      set sap-sidadm <string>
      set sap-password <string>
      set update-interval <seconds>
   next
end
```

CLI Parameter	Description
type	The type of SDN. • kubernetes • aws

CLI Parameter	Description
	• oci • sap
Kubernetes connector	
status	Enable/disable SDN connector
server	Server IP address.
server-port	Port number. Default is 6443. Range is 1 to 65535.
secret-token	Specify a secret token. **Note**: Versions of Kubernetes before v1.22 automatically created long term credentials for accessing the Kubernetes API. This older mechanism was based on creating token Secrets that could then be mounted into running Pods. In more recent versions, including Kubernetes v1.28, API credentials are obtained directly by using the TokenRequest API, and are mounted into Pods using a projected volume. The tokens obtained using this method have bounded lifetimes, and are automatically invalidated when the Pod they are mounted into is deleted. You can still manually create a service account token Secret; for example, if you need a token that never expires. However, using the TokenRequest subresource to obtain a token to access the API is recommended instead.
update-interval	Specify an update interval in seconds. Default is 30. Range is 30 to 3600.
OCI connector	
tenant-id	Specify the OCI region type.The tenant ID to log in to OCI.
compartment-id	The Compartment ID in which your compute instances are deployed.
user-id	The user ID to log in to OCI.
oci-region	Specify the OCI region where your compute instances are located.
oci-cert	The certificate that FortiADC uses to build connections with OCI.
use-metadata-iam	When FortiADC is deployed on OCI, you can assign IAM role for it to access OCI objects.
ha-status	Enable this option if your OCI instances are deployed in HA mode.
AWS connector	
status	Enable or disable the AWS connector.
aws-region	Specify the region where your instances are deployed.
aws-accesskey	Specify the access key ID.
aws-secretkey	Specify the secret access key.
update-interval	Specify the update interval for the connector to get AWS objects and dynamically populates the information in the server pool configuration.

CLI Parameter	Description
use-metadata-iam	When FortiADC is deployed on AWS, you can assign IAM role for it to access EC2 instances and EKS objects.
SAP Connector	
status	Enable or disable the SAP connector.
server	Type the IP address of the SAP server.
sap-ms-http-port	Specify the SAP MS HTTP port that FortiADC uses to communicate with the SAP server.
sap-icm-http-port	Specify the ICM HTTP Port.
sap-sidadm	Specify the SID admin account that FortiADC uses to access the resources in this account.
sap-password	Specify the password.
update-interval	Specify the update interval for the connector to get SAP objects and dynamically populates the information in the server pool configuration.

config system servicegrp

Use this command to create the service group objects that you use to specify matching services in policies.

The following policies use service group objects:

- Link Load Balance policies

Basic Steps

1. Create service group objects.
2. Specify them when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.
- You must have created service objects.

Syntax

```
config system servicegrp
   edit "servicegrp-name"
      set member-list <A> <B> <C>
end
```

servicegrp-name	The name of the service-group.
member-list	List of supported system services, which include the following: • HTTP • HTTPS • ICMP • TELNET • SSH • FTP • SMTP • SMTPS • IMAP • IMAPS • POP3 • POP3S • DHCP • DNS • NTP • SNMP • SNMP_TRAP • SYSLOG • LDAP • LDAPS • RADIUS • RADIUS_OLD • KERBEROS • SMB • SAMBA • MYSQL • GRE • ALL • service1

Example

```
config system service
   edit "http"
      set protoco-type tcp
      set specify-source-port enable
      set source-port-min 1
      set source-port-max 65535
      set destination-port-min 80
      set destination-port-max 80
   next
   edit "icmp"
      set protoco-type icmp
   next
end
```

```
config system servicegrp
     edit "servicegrp_test"
        set member-list HTTP ICMP
     next
   end

config system addrgrp
     edit "1"
        set member-list 10_10 10_20
     next
   end

config system addrgrp6
     edit "v6_1"
        set member-list v6_10 v6_20
     next
end
```

config system servicegrp

Use this command to create the service group objects that you use to specify matching services in policies.

The following policies use service group objects:

- Link Load Balance policies

Basic Steps

1. Create service group objects.
2. Specify them when you configure your policies.

Before you begin:

- You must have read-write permission for system settings.
- You must have created service objects.

Syntax

```
config system servicegrp
   edit "servicegrp-name"
      set member-list <A> <B> <C>
end
```

servicegrp-name	The name of the service-group.
member-list	List of supported system services, which include the following: • HTTP

- HTTPS
- ICMP
- TELNET
- SSH
- FTP
- SMTP
- SMTPS
- IMAP
- IMAPS
- POP3
- POP3S
- DHCP
- DNS
- NTP
- SNMP
- SNMP_TRAP
- SYSLOG
- LDAP
- LDAPS
- RADIUS
- RADIUS_OLD
- KERBEROS
- SMB
- SAMBA
- MYSQL
- GRE
- ALL
- service1

Example

```
config system service
   edit "http"
      set protoco-type tcp
      set specify-source-port enable
      set source-port-min 1
      set source-port-max 65535
      set destination-port-min 80
      set destination-port-max 80
   next
   edit "icmp"
      set protoco-type icmp
   next
end

config system servicegrp
      edit "servicegrp_test"
         set member-list HTTP ICMP
      next
```

```
   end

config system addrgrp
     edit "1"
        set member-list 10_10 10_20
     next
   end

config system addrgrp6
     edit "v6_1"
        set member-list v6_10 v6_20
     next
end
```

config system setting

Use this command to configure log database behavior when disk utilization reaches its capacity.

Before you begin:

You must have read-write permission for system settings.

Syntax

```
config system setting
   set statistics-db-full {overwrite | nowrite}
   set log-db-full {overwrite | nowrite}
   set predefine-isp {enable|disable}
end
```

`statistics-db-full`	Specify whether to overwrite stats or stop writing stats when the database disk allocation (10% of total disk space) is full. The default is overwrite the earliest stats.
`log-db-full`	Specify whether to overwrite logs or stop writing logs when the database disk allocation (40% of total disk space) is full. The default is overwrite the earliest logs.
`predefine-isp`	Enable/disable the predefined ISP address book. Enabled by default. You can use this setting to disable if you experience address conflicts that you cannot resolve using the ISP address book exceptions list.

Example

```
FortiADC-VM # get system setting
statistics-db-full : overwrite
log-db-full : overwrite
predefine-isp: enable

FortiADC-VM # config system setting
```

```
FortiADC-VM (setting) # set statistics-db-full nowrite

FortiADC-VM (setting) # end
```

config system snmp community

Use this command to configure SNMP community settings.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system snmp community
   edit <No.>
      set name <string>
      set queryportv1 <integer>
      set queryportv2c <integer>
      set queryv1-status {enable|disable}
      set queryv2c-status {enable|disable}
      set status {enable|disable}
         config host
            edit <No.>
               set host-type <query>
               set ip <subnet>
            next
         end
   next
end
```

`name`	Name of the SNMP community to which the FortiADC system and at least one SNMP manager belongs, such as `management`. You must configure the FortiADC system to belong to at least one SNMP community so that community's SNMP managers can query system information and receive SNMP traps. You can add up to three SNMP communities. Each community can have a different configuration for queries and traps, and the set of events that trigger a trap. You can also add the IP addresses of up to eight SNMP managers to each community to designate the destination of traps and which IP addresses are permitted to query the FortiADC system.
`queryportv1`	Port number on which the system listens for SNMP queries from the SNMP managers in this community. The default is 161.
`queryportv2c`	Port number on which the system listens for SNMP queries from the SNMP managers in this community. The default is 161.

queryv1-status	Enable/disable SNMP v1 queries.
queryv2c-status	Enable/disable SNMP v2c queries.
status	Enable/disable the configuration.
config host	
host-type	• query—Accept queries from this host.
ip	Specify a subnet address for the SNMP manager to receive traps and be permitted to query the FortiADC system. SNMP managers have read-only access. You can add up to 8 SNMP managers for a user. To allow any IP address using this SNMP username to query the FortiADC system, enter `0.0.0.0/0`. For security best practice reasons, however, this is not recommended. **Caution:** The system sends security-sensitive traps, which should be sent only over a trusted network, and only to administrative equipment. **Note:** If there are no other host IP entries, entering only `0.0.0.0/0` effectively disables traps because there is no specific destination for trap packets. If you do not want to disable traps, you must add at least one other entry that specifies the IP address of an SNMP manager.

Example

```
FortiADC-VM # config system snmp community

FortiADC-VM (community) # edit 1
Add new entry '1' for node 318

FortiADC-VM (1) # get
name :
status : enable
queryv1-status : enable
queryportv1 : 161
queryv2c-status : enable
queryportv2c : 161
trapv1-status : enable

FortiADC-VM (1) # set name community1

FortiADC-VM (1) # config host
<Enter>

FortiADC-VM (1) # config host
FortiADC-VM (host) # edit 1
Add new entry '1' for node 333

FortiADC-VM (1) # get
ip : 0.0.0.0
host-type : any

FortiADC-VM (1) # set ip 192.0.2.1/32

FortiADC-VM (1) # end
```

```
FortiADC-VM (1) # end
```

config system snmp sysinfo

Use this command to configure SNMP settings.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system snmp sysinfo
   set contact <string>
   set description <string>
   set location <string>
   set status {enable|disable>
end
```

`contact`	Contact information for the administrator or other person responsible for this system, such as a phone number (`555-5555`) or name (`jdoe`). The contact information can be up to 35 characters long, and can contain only letters (a-z, A-Z), numbers, hyphens (-) and underscores (_).
`description`	A description or comment about the system, such as `dont-reboot`. The description can be up to 35 characters long, and can contain only letters (a-z, A-Z), numbers, hyphens (-) and underscores (_).
`location`	Physical location of the appliance, such as `floor2`. The location can be up to 35 characters long, and can contain only letters (a-z, A-Z), numbers, hyphens (-) and underscores (_).
`status`	Enable/disable the SNMP agent, so that the system can send traps and receive queries.

Example

```
FortiADC-VM # get system snmp sysinfo
status : disable
description :
location :
contact :

FortiADC-VM # config system snmp sysinfo
FortiADC-VM (sysinfo) # set status enable
FortiADC-VM (sysinfo) # end

FortiADC-VM # get system snmp sysinfo
status : enable
description :
```

```
location :
contact :
```

config system snmp user

Use this command to manage SNMP settings.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system snmp user
   edit <name>
      set query-status {enable|disable}
      set queryport <integer>
      set security-level {authnopriv | authpriv | noauthnopriv}
      set auth-proto {sha1|md5}
      set auth-pwd <passwd>
      set priv-proto {aes|des}
      set priv-pwd <passwd>
      set status {enable|disable}
      config host
         edit <name>
            set ip <subnet>
         next
      end
   next
end
```

query-status	Enable/disable SNMP queries.
queryport	Port number on which the system listens for SNMP queries from the SNMP managers in this community. The default is 161.
security-level	• authnopriv—Authenticated but unencrypted. • authpriv—Authenticated and encrypted. • noauthnopriv—Unauthenticated and unencrypted.
auth-proto	• SHA1 • MD5
auth-pwd	Passphrase used to generate the key.
priv-proto	• AES • DES
priv-pwd	Passphrase used to generate the key.

status	Enable/disable the user configuration.
config host	
ip	Specify a subnet address for the SNMP manager to receive traps and be permitted to query the FortiADC system. SNMP managers have read-only access. You can add up to 8 SNMP managers for a user. To allow any IP address using this SNMP username to query the FortiADC system, enter `0.0.0.0/0`. For security best practice reasons, however, this is not recommended. **Caution:** The system sends security-sensitive traps, which should be sent only over a trusted network, and only to administrative equipment. **Note:** If there are no other host IP entries, entering only `0.0.0.0/0` effectively disables traps because there is no specific destination for trap packets. If you do not want to disable traps, you must add at least one other entry that specifies the IP address of an SNMP manager.

Example

```
FortiADC-VM # config system snmp user

FortiADC-VM (user) # edit docs
Add new entry 'docs' for node 1152
FortiADC-VM (docs) # set status enable
FortiADC-VM (docs) # end

FortiADC-VM # get system snmp user docs
status : enable
security-level :
query-status : disable
queryport : 161
trap-status : disable
trapport-local : 162
trapport-remote : 162
trapevent : cpu mem logdisk system raid ha remote-storage
```

config system tcpdump

This configuration is for the tcpdump utility in the Web UI. The configuration saves TCP dump commands and filter expressions so that they can be re-run from the Web UI. The CLI supports its own tcpdump service. See execute packet-capture/packet-capture6.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system tcpdump
   edit <No.>
```

```
    set host <ip&netmask>
    set interface <datasource>
    set logtraffic {enable|disable}
    set max-packet-count <integer>
    set port <integer>
    set protocol {arp icmp tcp udp}
    set specified-protocol {enable|disable}
    set status {enable|disable}
end
```

host	IP address for the interface used for tcpdump.
interface	Interface to use for tcpdump.
logtraffic	Enable/disable event logs about using tcpdump.
max-packet-count	Maximum number of packets to capture.
port	Port to use for tcpdump.
protocol	Specify the protocol traffic to capture.
specified-protocol	Enable/disable the protocol option.
status	Enable/disable the configuration.

Example

```
FortiADC-VM # config system tcpdump
FortiADC-VM (tcpdump) # edit 1
Add new entry '1' for node 2725
FortiADC-VM (1) # set interface port1
FortiADC-VM (1) # set status enable
FortiADC-VM (1) # set max-packet-count 5
FortiADC-VM (1) # end

FortiADC-VM # get system tcpdump 1
interface : port1
status : enable
logtraffic : enable
ipv6 : disable
host :
port :
specified-protocol : disable
max-packet-count : 5
```

config system time manual

Use this command to manage system time.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system time manual
   set daylight-saving-time {enable|disable}
   set zone <0-71>
   next
end
```

daylight-saving-time	Enable if you want the system to adjust its own clock when its time zone changes between daylight saving time (DST) and standard time.
zone	Specify the code number for the time zone where the appliance is located.

Example

```
FortiADC-VM # get system time manual
daylight-saving-time: enable
zone : 4
```

See also

- execute date

config system time ntp

Use this command to manage the connection to an NTP server.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system time ntp
   set ntpsync {enable|disable}
   set ntpserver <string>
   set syncinterval <integer>
end
```

ntpsync	Enable/disable use of NTP.

ntpserver	Specify the IP address or domain name of an NTP server or pool, such as pool.ntp.org. To find an NTP server, go to http://www.ntp.org.
syncinterval	Specify how often the system synchronizes its time with the NTP server. The default is 60 minutes. The valid range is 1-1440.

Example

```
FortiADC-VM # get system time ntp
ntpsync : disable

FortiADC-VM # config system time ntp
FortiADC-VM (ntp) # set ntpsync enable
FortiADC-VM (ntp) # end

FortiADC-VM # get system time ntp
ntpsync : enable
ntpserver : pool.ntp.org
syncinterval : 60
```

config system web-filter

Use this command to manage FortiGuard web filter category updates. FortiGuard maintains massive lists of web sites classified into categories so that you can enforce categorical decisions in your rules, like "do not do SSL forward proxy for sites belonging to the Personal Privacy category."

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system web-filter
   set cache-status {enable|disable}
   set cache-ttl <integer>
   set fds-port <integer>
end
```

cache-status	Enable/disable caching of the categorical lists of websites.
cache-ttl	Specify cache expiration. The default is 3600. The valid range is 10 to 86,400. When the cache expires, FortiADC initiates an update from FortiGuard.
fds-port	Specify the port to receive updates. The default is 53. An alternative is 8888.

Example

```
FortiADC-VM # config system web-filter
FortiADC-VM (web-filter) # set cache-status enable
FortiADC-VM (web-filter) # end
```

See also

- config system fortiguard

config system fortisandbox

Use this command to configure FortiSandbox settings.

Syntax

```
config system fortisandbox
   set type {fsa}
   set status {enable | disable}
   set server <server_ip>
   set email <email_address>
   set source-ip <ip_address>
   set type <fsa|cloud>
   set enc-algorithm <default|high|low|disable>
end
```

type	FSA—FortiSandbox appliance. Cloud—FortiCLoud Sandbox appliance.
status	Click the button to enable or disable FortiSandbox service. Note: FortiSandbox is disabled by default.
server	Enter the IP address of the FortiSandbox appliance. Note: This option applies if you want to use a on-premise FortiSandbox appliance for service.
email	The email address of the party to be notified.
source-ip	The IP address of the source interface on the FortiADC appliance.
enc-algorithm	Configure the level of SSL protection for secure communication with FortiSandbox

config system alert

This command can be used to configure the alert (`alert-name`) that is referenced in `config system alert-policy`. However, automation stitches CLI commands are recommended to be used only for debug purposes. To configure the stitches through GUI, refer to the FortiADC Handbook on Automation.

There are two types of alerts:

- Event alert on page 485 — These alerts are system predefined, corresponding to the automation trigger types Security Events, HA Failover, and System Events.
- Metric alert on page 486 — These alerts are user-defined, corresponding to the automation trigger types for SLB Metrics, System Metrics, and Interface Metrics.

Before you begin:

- You must have Global Administrator access. Ensure that your admin account settings has **Global Admin** set to **Yes**.

Event alert

You can use this command to create an event alert. Automation alert events are predefined, corresponding to the Security Events, HA Failover, and System Events automation triggers.

It is not recommended to configure automation alert events through the CLI.

Automation alert events are system predefined. However, the CLI does not restrict users from defining their own events which could result in invalid configurations or unexpected behavior.

For details on how to configure automation alert events in the GUI, see the FortiADC Handbook on Configure Automation Triggers.

Syntax

```
config system alert
   edit <alert_name>
      set alert-source-type event
      set priority {high|low|middle}
      set event <alert_event_name>
   next
end
```

The `<alert_event_name>` is defined in `config system alert-event`.

```
config system alert-event
   edit <alert_event_name>
      set alias <string>
      set comments <string>
   next
end
```

Metric alert

You can use the following command to create a metric alert. Automation alert metrics are user-defined, corresponding to the SLB Metrics, System Metrics, and Interface Metrics automation triggers.

Syntax

```
config system alert
   edit <alert_name>
      set alert-source-type metric
      set metric-object-type slb-virtual-server
      set metric-object-instance <virtual-server-name>
      set duration 600
      config alert-metric-expr-member
         edit <alert-metric-expr-member_name>
            set metric <metric_name>
            set metric-comparator {le | eq | ge}
            set value <int>
         next
      end
   next
end
```

Note: Only one metric member is supported.

The <metric_name> is defined in config system alert-metric.

```
config system alert-metric
   edit <metric_name>
      set type {if | slb |sys}
      set alias <string>
      set comments <string>
   next
end
```

Related topics:

- config system alert-action on page 486
- config system alert-policy on page 487

config system alert-action

Use this command to configure automation stitches actions.

Before you begin:

- You must have Global Administrator access. Ensure that your admin account settings has **Global Admin** set to **Yes**.

Syntax

```
config system alert-action
   edit <action_name>
      config action_list
         edit <index_number>
            set type {email | fortigate-ip-ban | script | snmp-trap | syslog | webhook}
         next
      end
   next
end
```

Use the following commands to configure the specific action.

- `config system alert-email`
- `config system alert-snmp-trap`
- `config system alert-script`
- `config system alert-webhook`
- `config system alert-fortigate-ip-ban`
- `config system alert-syslog`

The automation stitches CLI commands are recommended to use only for debug purpose. To configure the stitches through GUI, refer to "Chapter 23: Security Fabric" in *FortiADC Handbook*.

Related topics:

- config system alert-policy on page 487
- config system alert-action on page 486

config system alert-policy

Use this command to configure automation stitches.

Before you begin:

- You must have Global Administrator access. Ensure that your admin account settings has **Global Admin** set to **Yes**.

Syntax

```
config system alert-policy
   edit "alert_policy_name"
      set status {enable|disable}
      set action <action_name>
      set action_interval <int>
      set egress_vdom {local | root}
```

```
      set type {ha-failover | interface-metric | period-block-ip | schedule | security-event
           | slb-metric | system-event | system-metric}
      set comments comments
      config alert-member
         edit "alert_member_name"
            set alert-name <system_alert_name>
         next
      end
   next
end
```

The <action_name> is defined in config system alert-action, and the <system_alert_name> in config system alert.

The automation stitches CLI commands are recommended to be used only for debug purposes. To configure the stitches through GUI, refer to "Chapter 23: Security Fabric" in *FortiADC Handbook*.

Related topics:

- config system alert-action on page 486
- config system alert-policy on page 487

config system alert-email

Use this command to configure email alert objects.

Syntax

```
config system alert-email
   edit "email_name"
      set from <string_email_format>
      set to <string_email_format>
      set subject <string>
      set body <string>
   next
end
```

email_name	The name of an email alert object.
from	The sender's email address, e.g., fortiadc_send@fortinet.com.
to	The recipient's email address, e.g., receiver@fortinet.com.
subject	The subject of the email.
body	The body of the email.

Example

```
config system alert-syslog
   edit "1"
      set server 10.0.11.16
      set port 514
   next
end
config system alert-email
   edit "email1"
      set from fortiadc_send@fortinet.com
      set to receiver@fortinet.com
   next
end
```

config system alert-snmp-trap

Use this command to configure SNMP alert traps.

Syntax

```
config system alert-snmp-trap
   edit "snmp_trap_name"
      set ip <ipv4 address>
      set version {version1|version2c|version3}
      set trapport-local <integer>
      set trapport-remote <integer>
   next
end
```

snmp_trap_name	Add or edit a table value
ip	The IP address of the SNMP trap server.
version	The version of the SNMP trap server.
trapport-local	The local trap port number of the SNMP server.
trapport-remote	The remote trap port number of the SNMP server.

Example

```
config system alert-snmp-trap
   edit "1"
      set ip 10.0.11.16
      set version version2c
      set trapport-local 162
      set trapport-remote 162
```

```
   next
end
```

config system central-management

Use this command to configure central management settings for connecting CM (central management) server.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config system central-management
   set mgmt-type {none|FortiADC-Manager}
   set mgmt-addr <string>
   set interval <integer>
   set status {disable|enable }
end
```

mgmt-type	Specify the CM Server type.
none	Central management is not being used.
FortiADC-Manager	A customized CM server is being used.
mgmt-addr	IP address or FQDN of CM Server
interval	Retry Interval for connecting to CM Server.
status	Enable to start connecting to CM Server. Disable to stop connecting to CM Server.

Example

```
FortiADC-VM # config system central-management
FortiADC-VM (central-manage~t) # set mgmt-type FortiADC-Manager
FortiADC-VM (central-manage~t) # set mgmt-addr 10.0.100.166
FortiADC-VM (central-manage~t) # set status enable
FortiADC-VM (central-manage~t) # end
FortiADC-VM # get system central-management
mgmt-type : FortiADC-Manager
mgmt-addr : 10.0.100.166
interval : 10
status : enable
FortiADC-VM # get sys sta
Version: FortiADC-VM v5.2.0,build0423,181220
VM Registration: Valid: License has been successfully authenticated with registration
     servers.
VM License File: License file and resources are valid.
VM Resources: 1 CPU/8 allowed, 3743 MB RAM/16384 MB allowed, 29 GB Disk/1024 GB allowedVM
     Resources: 1 CPU/8 allowed, 3814 MB RAM/16384 MB allowed, 29 GB Disk/1024 GB allowed
Serial-Number: FADV080000146968
```

```
WAF Signature DB: 00001.00020
IP Reputation DB: 00001.00094
Regular Virus DB: 00062.00475
Extended Virus DB: 00062.00467
Extreme Virus DB: 00062.00323
AV Engine: 00006.00006
Bootloader Version: n/a
Hard Disk: Capacity 29 GB, Used 5 GB (19.67%), Free 23 GB
Log Size: 1 GB, 5%
Hostname: FortiADC-VM
HA Configured Mode: standalone
HA Effective Mode: Standalone
Distribution: International
CM Agent State: (Enabled) ONLINE
Uptime: 0 days 0 hours 10 minutes
Last Reboot: Wed Sep 26 00:23:20 PDT 2018
System Time: Wed Sep 26 00:33:36 PDT 2018
```

config user

The `config user` commands configure the authentication framework for administrator accounts and user accounts.

This chapter is a reference for the following commands:

config user ldap

Use this command to configure a connection to an LDAP server that can authenticate administrator or user logins.

Basic steps:

1. Create an LDAP authentication server configuration.
2. Select the LDAP server configuration when you add administrator users or create user groups.

Before you begin:

- You must know the IP address and port used to access the LDAP server. You must know the CN and DN where user credentials are stored on the LDAP server.
- You must have read-write permission for system settings.

Syntax

```
config user ldap
   edit <name>
      set cnid <string>
      set dn <string>
      set port <integer>
      set server <string>
      set vdom <datasource>
   next
end
```

cnid	Common name (cn) attribute for the LDAP record. For example: `cn`
dn	Distinguished name (dn) attribute for the LDAP record. For example: `cn=John%20Doe,dc=example,dc=com`
port	Port number for the server. The commonly used port for LDAP is 389.
server	IP address for the server.
vdom	Reserved for future use.

config user local

Use this command to configure user accounts in the local authentication server. You can add or delete accounts, or change the password, but you cannot edit usernames.

Before you begin:

- You must have read-write permission for system settings.

Syntax

```
config user local
   edit <name>
      set password <password>
   next
end
```

<name>	Name of the user account, such as `user1` or `user1@example.com`. Do not use spaces or special characters except the 'at' symbol (`@`) or dot (`.`). The maximum length is 35 characters. After you initially save the configuration, you cannot edit the name.
password	Specify a password. The stored password will be encrypted.

config user radius

Use this command to configure a connection to a RADIUS server that can authenticate administrator or user logins.

Basic steps:

1. Create a RADIUS authentication server configuration.
2. Select the RADIUS server configuration when you add administrator users or user groups.

Before you begin:

- You must know the IP address, port, authentication protocol, and shared secret used to access the RADIUS server.
- You must have read-write permission for system settings.

Syntax

```
config user radius
   edit <name>
      set auth-type {chap|ms_chap|ms_chapv2|pap}
      set port <integer>
      set secret <passwd>
      set server <string>
      set timeout <integer>
      set vdom <datasource>
   next
end
```

auth-type	• chap—Challenge-Handshake Authentication Protocol. • ms_chap—Microsoft version of CHAP. • ms_chapv2—Microsoft version of CHAP, version 2. • pap—Password authentication protocol.
port	Port number for the server. The commonly used port for RADIUS is 1812.
secret	Shared secret string used when connecting to the server.
server	IP address for the server.
timeout	Remote Authentication Timeout (seconds)
vdom	Reserved for future use.

config user tacacs+

Use this command to configure the Terminal Access Controller Access-Control System Plus (TACACS+) authentication server.

Basic steps:

1. Configure a connection to a TACACS+ server that can authenticate administrator or user logins.
2. Select the TACACS+ server configuration when you add administrator users or user groups.

Before you begin:

- You must know the IP address, port, authentication protocol, and shared secret used to access the TACACS+ server.
- You must have read-write permission for system settings.

Syntax

```
config user tacacs+
   edit <name>
      set server <string>
      set secret <passwd>
      set auth-type {auto|ms_chap|chap|pap|ascii}
      set port <integer>
      set timeout <integer>
      set vdom <datasource>
   next
end
```

server	Enter the IP address or FQDN of the TACACS+ server.
secret	Shared secret string used when connecting to the TACACS+ server. The shared secret can be a maximum of 16 characters in length.
auth-type	Specify the authentication protocol used for the TACACS+ server: • auto — FortiADC tries all authentication protocols in order: MS-CHAP → CHAP → PAP → ASCII. • ms_chap — Microsoft version of CHAP (Challenge Handshake Authentication Protocol). • chap — Challenge Handshake Authentication Protocol (defined in RFC 1994). • pap — Password Authentication Protocol. • ascii — American Standard Code for Information Interchange. The default option is **auto**.
port	Port number for the server. The commonly used port for TACACS+ is 49.
timeout	Specify the amount of time that FortiADC must wait for responses from the remote TACACS+ server before it times out the connection. Valid values are from 5 to 60 seconds. The default is 5 seconds.
vdom	Reserved for future use.

config user user-group

Use this command to configure user groups. User groups are authorized by the virtual server authorization policy. The user group configuration references the authentication servers that contain valid user credentials.

Suggested steps:

1. Configure LDAP, RADIUS and TACACS+ servers, if applicable.
2. Configure local users.
3. Configure user groups (reference servers and local users).

4. Configure an authorization policy (reference the user group).
5. Configure the virtual server (reference the authorization policy).

Before you begin:

- You must have created configuration objects for any LDAP, RADIUS and/or TACACS+ server you want to use, and you must have created user accounts for local users.
- You must have read-write permission for system settings.

After you have created user groups, you can specify them in the `load-balance auth-policy` configuration.

Syntax

```
config user user-group
   edit <name>
      set auth-log {none|fail|success|all}
      set auth-session-timeout <integer>
      set auth-timeout <integer>
      set user-cache {enable|disable}
      set user-cache-timeout <integer>
      set client-auth-method {html_form_auth|http_auth|ntlm_auth}
      set use-default-form {enable|disable}
      set auth_form_profile <datasource>
      set group-type {normal|SSO}
      set authentication-relay <datasource>
      set sso-support {enable|disable}
      set sso-domain <string>
      set logoff-path <string>
      config member
         edit <No.>
            set type {local|ldap|radius|tacacs_plus}
            set local-user {<name> <name> ...}
            set ldap-server <datasource>
            set radius-server <datasource>
            set tacacs-plus-server <datasource>
         next
      end
      config user cust_auth_form
         edit <name>
            set auth_form-file <file>
            set username_field <username field name>
            set password_field <password field name>
            set virtual_path <virtual path>
         next
      end
   next
end
```

`auth-log`	Specify one of the following logging options for authentication events: • none — No logging • fail — Log failed attempts • success — Log successful attempts • all — Log all (both failed and successful attempts)

auth-session-timeout	Specify the authentication session timeout. Valid values range from 1 to 180 minutes. The default is 3 (minutes).
auth-timeout	Timeout for query sent from FortiADC to a remote authentication server. The default is 2000 milliseconds. The valid range is 1-60,000 milliseconds.
user-cache	Enable to cache the credentials for the remote users (LDAP, RADIUS, TACACS+) once they are authorized.
user-cache-timeout	The **user-cache-timeout** option is available if **user-cache** is enabled. Timeout for cached user credentials. The default is 300 seconds. The valid range is 1-86,400 seconds.
client-auth-method	• html_form_auth • http_auth • ntlm_auth (only if you want to use NTLM server as a authentication server)
use-default-form	The **use-default-form** option is available if **client-auth-method** is **html_form_auth**. Enabled by default to use the default authentication form. Disable to use a customized authentication form.
auth_form_profile	The **auth_form_profile option** is available if **client-auth-method** is **html_form_auth** and **use-default-form** is **disabled**. Set profile of authentication form. You can use the default or the profile name in cust_auth_form.
group-type	• normal — Default. No action is needed. • sso — enables Single Sign-On (SSO).
authentication-relay	The **authentication-relay** option is available if **group-type** is **sso**. Set an authentication relay profile.
sso-support	The **sso-support** option is available if **group-type** is **sso**. Enable/disable SSO Cross Domain Support. This is disabled by default. When enabled, you must specify the SSO domain. **Note**: Authentication policies cannot be applied to multiple virtual servers. Due to security reasons, such as protection against XSS attacks, there is no shared mechanism between virtual servers to decrypt cookies. As a result, you cannot log into a second virtual server while already logged into the first virtual server as the virtual servers are independent from each other. SSO Cross Domain Support allows you to have multiple domain names on the same virtual server (the virtual host), where you can specify a first-level domain name to enable the second-level domain names on the virtual server to decrypt cookies at the same time.
sso-domain	The **sso-domain** option is available if **group-type** is **sso** and **sso-support** is **enabled**. Specify the SSO domain.
logoff-path	The **logoff-path** option is available if **group-type** is **sso**. Specify the log-off URL.

config member	
type	Authentication server type.
local-user	To add local users, specify the local usernames.
ldap-server	To add LDAP users, specify the LDAP server configuration name.
radius-server	To add RADIUS users, specify the RADIUS server configuration name.
tacacs-plus-server	To add TACACS+ users, specify the TACACS+ server configuration name.
config user cust_auth_form	
auth_form-file	Profile name of authentication form
username_field	Username field name in customized form
password_field	The password field name in customized form
virtual_path	The virtual path to redirect

Example

```
config user user-group
   edit "normal-group"
      set client_auth_method html_form_auth
      set auth_form_profile <default/profile_name>
      config member
         edit 1
               set local-user local-user-1
         next
         edit 2
               set type ldap
               set ldap-server ldap-server
         next
         edit 3
               set type radius
               set radius-server radius-server
         next
      end
   next
   config user cust_auth_form
      edit "test"
         set auth_form-file local-user-1_tst.zip
         set username_field user-1
         set password_field pw-1
         set virtual_path <virtual_path>
      next
   end
   edit "SSO-Kerbros-Group"
      set group-type SSO
      set authentication-relay auth-relay-1
      set logoff-path logoff.html
      set sso-support enable
      set sso-domain kfor.com
      config member
         edit 1
```

```
            set local-user local-user-1
        next
        edit 2
            set type ldap
            set ldap-server ldap-server
        next
        edit 3
            set type radius
            set radius-server radius-server
        next
      end
   next
   edit "SSO-HTTPBasic-Group"
      set group-type SSO
      set authentication-relay auth-relay-2
      set logoff-path logoff
      set sso-support enable
      set sso-domain sss.com
      config member
      end
   next
end
```

config user authentication-relay

Use this command to configure the authentication relay, which includes Kerberos and HTTP basic SSO configurations.

Syntax

```
config user authentication-relay
   edit <authentication-relay name>
      set authorization HTTPError401 | always
      set delegation-type Kerberos | http-basic
      set kdc-ip <string> FQDN/ip of kdc
      set kdc-port <integer> the port number of kdc server
      set realm <string> realm (upper case)
      set domain-prefix-support enable/disable
      set domain-prefix <string> domain to prefix
      set delegator-account <string> KCD delegator principal
      set delegator-password <passwd> KCD delegator password
      set delegated-spn <string> KCD delegated service principal
   next
end
```

The following table describes parameters used for configuring authentication relay using Kerberos SSO.

delegation-type	Select Kerberos or HTTP Basic. Note: You MUST select Kerberos when configuring authentication relay for Kerberos SSO.

authorization	Can select HTTPError401 or always. After a client account authenticates successfully, FortiADC first sends the request to the server and waits for the server's response before performing authentication on its part. If `HTTPErr401` is set, FortiADC will do the authentication only when it has received the 401 response. Furthermore, if the client requests for more information from the web after FortiADC has gotten the authentication service ticket, FortiADC will send the request without the ticket. FortiADC will send another request with the service ticket only when the server returns the 401 unauthorized response. When `always` is set, FortiADC always does the authentication no matter what response it receives from the server. If the client requests for more information from the web after FortiADC has gotten the Kerberos service ticket, FortiADC will always send the request with the service ticket.
kdc-ip	The KDC server IP address.
kdc-port	The port on which the KDC server listens for Kerberos authentication.
realm	The realm which supports Kerberos authentication. Note: You must use uppercase letters and '.' in the string.
delegated-spn	The identification which shows the service running on the server. The SPN uses this format: `HTTP/sharepoint.ft3.local@FT3.LOCAL` Where • HTTP— Refers to the service running on the server. • The string between / and @ —Refers to the host, which supports regexp. • The string after @ — Refers to the realm that supports the service. It MUST be in upper-case letters.
delegator-account	The FortiADC proxy Kerberos authentication account.
delegator-password	The delegator account password.
domain-prefix-support	Domain prefix support: This is a switch to enable or disable the default domain prefix function. Sometimes the domain controller requires the user to log in with the user name format "`domain\username`" such as '`KFOR\user1`' When this option is enabled, the user can also successfully log in by only entering '`user1`' because FortiADC is able to automatically add the prefix '`KFOR\`'and then send '`KFOR\user1`'to the server. Domain prefix: The value will be added as the domain prefix when the switch above is enabled and when the user inputs the username without the domain. The value of this domain prefix MUST be a valid NetBIOS domain name.

Example 1: Configure Kerberos authentication relay:

```
config user authentication-relay
   edit "auth-relay-1"
      set kdc-ip 2.2.1.202
      set realm KFOR.COM
```

```
      set delegator-account test
      set delegator-password ENC
      set delegated-spn http/server11202.kfor.com@kfor.com
   next
end
```

Example 2: Configure HTTP-basic authentication relay:

```
config user authentication-relay
   edit "auth-relay-2"
      set delegation-type http-basic
      set authorization always
      set domain-prefix-support enable
      set domain-prefix SSS
   next
end
```

config user oauth

Use this command to configure the OAuth policy for OAuth 2.0 authentication.

Syntax

```
config user oauth
   edit <name>
      set auth-url <string>
      set token-url <string>
      set client-id <string>
      set client-secret <string>
      set http-method {GET|POST}
      set redirect-url <string>
      set relay-mode {enable|disable}
      set include-granted-scopes {true|false}
      set prompt {disable|none|consent|select_account}
      set token-timeout <integer>
      set scope-logout-url /logout
      config scope-list
         edit 1
            set scope-url <string>
         next
      end
   next
end
```

CLI Parameter	Description
auth-url	The URL of the authorization server.
token-url	The URL of the token server

CLI Parameter	Description
`client-id`	The client ID for your application.
`client-secret`	The secret used to apply for the access token.
`http-method`	The HTTP method used for the OAuth transaction. Select from the following values: • POST • GET
`redirect-url`	The URL of the redirected server.
`relay-mode`	Enable/disable relay mode allows FortiADC to add an Authorization Header to the HTTP request after verifying the token.
`include-granted-scopes`	Select from the following values: • true • false • none This enables applications to use incremental authorization to request access to additional scopes in context. If you set this parameter's value to `true` and the authorization request is granted, then the new access token will also cover any scopes to which the user previously granted the application access.
`prompt`	A space-delimited, case-sensitive list of prompts to present the user. If you do not specify this parameter, the user will only be prompted the first time your project requests access. Possible values are: • disable — Disable prompts. • none — Do not display any authentication or consent screens. • consent — Prompt the user for consent. • select_account — Prompt the user to select an account.
`token-timeout`	The amount of time in seconds the token will be valid. (Range: 120-86,400, default = 3600). The client will not be allowed to access the scope after this time has elapsed.
`scope-logout-url`	Access to this URL will trigger a logout event. FortiADC will delete the cookie, so in the next access you will need to repeat the OAuth setup process.
`scope-list`	A space-delimited list of scopes that identify the resources that your application could access on the user's behalf. These values inform the consent screen that the resource server displays to the user. Scopes enable your application to only request access to the resources that it needs while also enabling users to control the amount of access that they grant to your application. This results in an inverse relationship between the number of scopes requested and the likelihood of obtaining user consent.

CLI Parameter	Description
scope-url	This URL specifies the location of the resource that your application could access on the user's behalf and will be shown to the user to obtain their consent when they access the resource server. The relative path of a URL is permitted.

Example

```
config user oauth
   edit "oauth"
      set auth-url https://accounts.google.com/o/oauth2/v2/auth
      set token-url https://www.googleapis.com/oauth2/v4/token
      set client-id 49178883990-conasjq8hiero0rtc5olhk7c5719i36i.apps.googleusercontent.com
      set client-secret ENC
           VSiVjX6ZdFjBoDSjmOHBCYNeTAij3tbIR/4+kRF5g0U/B40FDbIGgDI/ZzrEmStXe0SG7GuYYizXOCyrX
           vncJHO5IX1hsX4WQXr/raBq6fe6Y0+rx74PXhUeGBdfLZsPMTrhPAx17Yncwq14Ry6pJnHclh8Lk3vMBY
           1kGQ==
      set http-mode POST
      set relay-mode enable
      set include-granted-scopes true
      set prompt consent
      set token-timeout 8888
      set scope-logout-url /logout
      config scope-list
         edit 1
            set scope-url https://www.googleapis.com/auth/blogger
         next
      end
   next
end
```

config user saml-idp

Security Assertion Markup Language (SAML) defines an XML-based framework for describing and exchanging security information among online business entities. It is the most popular protocol for implementing Web SSO.

The SAML protocol has two components—the Service Provider (SP) and the Identify Provider (IDP). They use SAML-defined formatted XML to talk to each other and deliver the identity information called Authentication Assertion.

Use this command to configure a saml-idp user.

Syntax

```
config user saml-idp
   edit <name>
      set comments <string>
      set idp-file <datasource>
   next
end
```

name	Specify a unique name for the SAML service provider.
comments	Set a string for comments.
idp-file	Select a preexisting idp-file.

Example

```
adc-3-228 (root) # config user saml-idp
adc-3-228 (saml-idp) # edit 1
adc-3-228 (1) # set comments "hello"
adc-3-228 (1) # get
comments : hello
idp-file : fortiauth-idp-666 (available)
adc-3-228 (saml-idp) # end
```

config user saml-sp

Use this command to configure a saml-sp user.

Syntax

```
config user saml-sp
   edit <name>
      set entity-id <string>
      set service-url <string>
      set assertion-consuming-service-path <string>
      set assertion-consuming-service-binding <string>
      set metadata-path <string>
      set logoff-path <string>
      set logoff-binding {post|binding}
      set local-cert <datasource>
      set auth-session-lifetime <integer>
      set auth-session-timeout <integer>
      set idp-metadata <datasource>
      set assertion-require-sign {enable|disable}
      set authnrequest-sign-algorithm {rsa-sha1|rsa-sha256|rsa-sha512}
      set sso-export {enable|disable}
      set export-assertion {enable|disable}
      set export-assertion-path <string>
      set export-cookie {enable|disable}
      config export-assertion-acl
         edit <name>
            set ip-mask <integer>
```

```
        next
      end
    next
end
```

entity-id	Specify the SAML service provider's entity ID, which is the SAML service provider's URL.
service-url	Specify the SAML service URL. The default value is **/SSO**.
assertion-consuming-service-path	Specify the Assertion Consuming Service Path. The default value is **/SAML2/Post**
assertion-consuming-binding	Specify the Assertion Consuming Service Binding Type. The default value is **post**.
metadata-path	Specify the Metadata Export Service Location. The default value is **/Metadata**.
logoff-path	Specify the Single Logout Path. The default value is **/SLO/Logout**.
logoff-binding	Select either of the following Single Logout Binding Type: • post • redirect The default value is **post**.
local-cert	Specify a local certification. The default is **Factory**.
auth-session-lifetime	Specify the Authentication Session Lifetime in seconds. (Range: 1-2592000, Default: 28800)
auth-session-timeout	Specify the Authentication Session Timeout in seconds. (Range: 1-86400, Default: 3600)
idp-metadata	Specify an IDP metadata file. Note: You must have the IDP metadata file imported into FortiADC ahead of time.
assertion-require-sign	Enable/disable the AuthNRequest algorithm to allow FortiADC to sign the SAML authentication request. This is enabled by default.
authnrequest-sign-algorithm	Select either of the following AuthNRequest algorithm: • rsa-sha1 • rsa-sha256 • rsa-sha512 The default value is **rsa-sha1**.
sso-export	Enable(d) by default, which allows FortiADC to forward SSO information to the real server, which in turn gets the authentication information and implements the SSO function.
export-assertion	Enable(d) by default, which allows FortiADC to send to the real server the URL where the Authentication Assertion (i.e., identity information) can be fetched.
export-assertion-path	Specify the Export Assertion Path. The default value is **/GetAssertion**.
export-cookie	Enable(d) by default, which allows FortiADC to send to the real server the cookie of a site that the user last visited.

config export-assertion-acl	
ip-mask	Enter the IP address of the real server (or the IP Netmask if the real server is one of a group of real servers) that requests authentication assertions.

Example

```
config user saml-sp
   edit "sp-example"
      set entity-id foradc221-7170
      set service-url /SSO
      set assertion-consuming-service-path /SAML2/Post
      set assertion-consuming-service-binding post
      set metadata-path /Metadata
      set logoff-path /SLO/Logout
      set logoff-binding post
      set local-cert Factory
      set auth-session-lifetime 28800
      set auth-session-timeout 3600
      set idp-metadata idp-example
      set assertion-require-sign enable
      set authnrequest-sign-algorithm rsa-sha512
      set sso-export enable
      set export-assertion enable
      set export-assertion-path /GetAssertion
      set export-cookie {enable|disable}
      config export-assertion-acl
         edit 1
            set ip-mask 192.168.0.2/31
         next
      end
   next
end
```

diagnose

The `diagnose` commands display diagnostic information that can help you troubleshoot problems. These commands do not have an equivalent in the web UI.

This chapter is a reference for the following commands:

diagnose antivirus quarantine

Syntax

```
diagnose antivirus quarantine delete <checksum>
diagnose antivirus quarantine list { all | infected | http | https | smtp }
diagnose antivirus quarantine purge
```

`delete`	Delete the files which checksum is as specified.
`list`	List quarantine files by filters. all - list all files. infected - list the files which status is 'infected'. http - list the files which service is 'http'. https - list the files which service is 'https'. smtp - list the files which service is 'smtp'.
`purge`	Delete all quarantine files.

Example

```
FortiADC-VM # diagnose antivirus quarantine list all
Quarantine List (Count = 6)
----------------------------
CHECKSUM SIZE FIRST-TIMESTAMP LAST-TIMESTAMP SERVICE STATUS DC TTL FILENAME DESCRIPTION
4c9bf9c5 22528 2018-12-05 17:54 2018-12-05 17:54 HTTP Infected 0 335:56 '4c9bf9c5.HTTP'
     'W32/Bika.1910'
4c9bf9c5 22528 2018-12-05 17:54 2018-12-05 17:54 HTTPS Infected 0 335:56 '4c9bf9c5.HTTPS'
     'W32/Bika.1910'
4c9bf9c5 22528 2018-12-05 17:54 2018-12-05 17:54 SMTP Infected 0 335:56 '4c9bf9c5.SMTP'
     'W32/Bika.1910'
b2c5aca8 8192 2018-12-05 17:54 2018-12-05 17:54 HTTP Infected 0 335:56 'b2c5aca8.HTTP'
     'W32/Borges.8192.B'
b2c5aca8 8192 2018-12-05 17:54 2018-12-05 17:54 HTTPS Infected 0 335:56 'b2c5aca8.HTTPS'
     'W32/Borges.8192.B'
b2c5aca8 8192 2018-12-05 17:54 2018-12-05 17:54 SMTP Infected 0 335:56 'b2c5aca8.SMTP'
     'W32/Borges.8192.B'
FortiADC-VM # diagnose antivirus quarantine delete b2c5aca8
FortiADC-VM # diagnose antivirus quarantine list all
```

```
Quarantine List (Count = 3)
----------------------------
CHECKSUM SIZE FIRST-TIMESTAMP LAST-TIMESTAMP SERVICE STATUS DC TTL FILENAME DESCRIPTION
4c9bf9c5 22528 2018-12-05 17:54 2018-12-05 17:54 HTTP Infected 0 335:55 '4c9bf9c5.HTTP'
     'W32/Bika.1910'
4c9bf9c5 22528 2018-12-05 17:54 2018-12-05 17:54 HTTPS Infected 0 335:55 '4c9bf9c5.HTTPS'
     'W32/Bika.1910'
4c9bf9c5 22528 2018-12-05 17:54 2018-12-05 17:54 SMTP Infected 0 335:55 '4c9bf9c5.SMTP'
     'W32/Bika.1910'
```

diagnose debug cmdb

Use this command to set the debug level for CLI commands. The debug messages are returned when you enter CLI commands.

Syntax

```
diagnose debug cmdb [<level>] <Enter>
```

<Enter>	If you do not specify a debug level and press Enter, the command displays the current debug level.
<level>	Valid range is 0 to 8, where 0 disables debug logs and 8 generates the most verbose logging.

Example

```
FortiADC-VM # diagnose debug cmdb 8
```

After you set the debug level, messages are written to the CLI when you enter commands:

```
FortiADC-VM # config system interface

FortiADC-VM (interface) # edit 2
Add new entry '2' for node 1

FortiADC-VM (2) # set ip 123
set attribute [4]ip
invalid ip address/mask 123.
data converting failed 4
Command fail. Return code is -39

FortiADC-VM (2) # end
attribute 'type' must be set
fgtlog: added a new entry '2' failed (-56) for "system interface"
Command fail. Return code is -56

FortiADC-VM #
```

diagnose debug enable/disable

Use this command to turn debug log output on or off.

Debug logging can be very resource intensive. To minimize the performance impact on your system, use debugging only during periods of minimal traffic, with a local console CLI connection rather than a Telnet or SSH CLI connection. Disable debugging when you are finished.

By default, the most verbose logging that is available from the web UI for any log type is the **Information** severity level. Due to their usually unnecessary nature, logs at the severity level of **Debug** are disabled and hidden. They can only be enabled and viewed from the CLI. Typically this is done only if your configuration seems to be correct, you cannot diagnose the problem without more information, and possibly suspect that you may have found either a hardware failure or software bug.

To use debug logs, you must:

1. Set the verbosity level for the specific module whose debugging information you want to view, via a debug log command such as:
   ```
   debug application hasyncd 5
   ```
2. Enable debug logs overall. To do this, enter:
   ```
   diagnose debug enable
   ```
3. View the debug logs. For convenience, debugging logs are immediately output to your local console display or terminal emulator, but debug log files can also be uploaded to a server. For more complex issues or bugs, this may be required in order to send debug information to Fortinet Technical Support.

   Debug logs will be generated only if the application is running. To verify the application is running, use diagnose system top .
4. The CLI displays debug logs as they occur until you disable it by entering:
   ```
   diagnose debug disable
   ```

- Close your terminal emulator, thereby ending your administrative session.
- Send a termination signal to the console by pressing Ctrl+C.
- Reboot the appliance. To do this, you can use the command:
  ```
  execute reboot
  ```

Syntax

```
diagnose debug {enable|disable}
```

`debug {enable|disable}`	Select whether to enable or disable recording of logs at the debug severity level.

diagnose debug flow

Use this command to debug particular traffic flows. Debug messages for traffic matching the filter and mask are displayed to the terminal screen.

Syntax

```
diagnose debug flow filter {addr <addr>|saddr <addr>|daddr <addr>|proto <integer>|virtual-
     server <VS-name>|clear|negate <addr|saddr|daddr|proto>|show}
diagnose debug flow mask {packet|session|persist|drop|layer4-server-loadbalance|all|custom
     <mask>}
diagnose debug flow show
diagnose debug flow start [<count>]
diagnose debug flow stop
```

`filter`	Specify filters. Issue multiple commands to add filters. Use the negate option to define "not in" matching. Filters determine the traffic flows for which the debug logs are written. You can match flows based on host address, source address, destination address, and protocol.
`mask`	Specify a mask that sets the type of data written to the screen.
`show`	Show current status, filters, and mask options.
`start`	Start debugging. The `[<count>]` option specifies a number of debug lines to output.
`stop`	Stop debugging.

Example

```
FortiADC-docs # diagnose debug flow ?
filter filter
mask mask
show Stop trace.
start Start trace.
stop Stop trace.
```

```
FortiADC-docs # diagnose debug flow stop
```

```
FortiADC-VM # diagnose debug flow filter ?
addr IP address.
clear Clear filter.
daddr Destination IP address.
negate negate
proto Protocol number.
saddr Source IP address.
show Show filter configuration.
virtual-server virtual server
```

```
FortiADC-docs # diagnose debug flow filter saddr 3.3.3.3
FortiADC-docs # diagnose debug flow filter daddr 4.4.4.4
FortiADC-docs # diagnose debug flow filter proto 1
FortiADC-docs # diagnose debug flow filter virtual-server VS1

FortiADC-VM # diagnose debug flow mask ?
all all debug info.
custom custom flow mask.
ddos ddos protection info.
drop drop packet info.
ips ips protection info.
layer4-server-loadbalance l4 loadbalance debug info.
packet packet info(default is on).
persist-cache persistence cache info.
session session info.

FortiADC-docs # diagnose debug flow mask all

FortiADC-VM # diagnose debug flow start
Start flow debug, set debug info count to 1000000000

FortiADC-VM # diagnose debug flow show
---------running status && config-----------
----flow debug is running, remain count 1000000000
----flow filter-------------
proto: any
host addr: 50.1.0.100-50.1.0.100
Host saddr: any
Host daddr: any
Virtual server : VS1
----flow mask---------------
layer4-server-loadbalance
---------current terminal config-----------
----flow filter-------------
proto: any
host addr: 50.1.0.100-50.1.0.100
Host saddr: any
Host daddr: any
Virtual server: VS1
----flow mask---------------
layer4-server-loadbalance

FortiADC-VM # [03-15 12:56:56] [trace id:1]ip_vs_out: packet continues traversal as normal
[03-15 12:56:56] Create session fwd:M c:50.1.0.1:57028 v:50.1.0.100:80 l:50.1.0. 1:57028
d:50.1.2.3:80 conn->flags:80140 conn->refcnt:2
[03-15 12:56:56] Incoming packet: TCP 50.1.0.1:57028->50.1.0.100:80
[03-15 12:56:56] TCP source port 57028 dst port 80 flag [S...]
[03-15 12:56:56] TCP input [S...] 50.1.0.1:57028->50.1.2.3:80 state: NONE->SYN_ RECV conn-
>refcnt:2
[03-15 12:56:56] After DNAT: TCP 50.1.0.1:57028->50.1.2.3:80
[03-15 12:56:56] NAT xmit, send packet to client
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] Outgoing packet: TCP 50.1.2.3:80->50.1.0.1:57028
[03-15 12:56:56] TCP source port 80 dst port 57028 flag [S.A.]
[03-15 12:56:56] After SNAT: TCP 50.1.0.100:80->50.1.0.1:57028
[03-15 12:56:56] Fast response xmit, send packet to client
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
```

```
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] [trace id:2]ip_vs_out: packet continues traversal as normal
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] Incoming packet: TCP 50.1.0.1:57028->50.1.0.100:80
[03-15 12:56:56] TCP source port 57028 dst port 80 flag [..A.]
[03-15 12:56:56] TCP input [..A.] 50.1.0.1:57028->50.1.2.3:80 state: SYN_RECV-> ESTABLISHED
conn->refcnt:2
[03-15 12:56:56] After FNAT-IN: TCP 50.1.0.1:57028->50.1.2.3:80
[03-15 12:56:56] Fast xmit, send packet to RS
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] [trace id:3]ip_vs_out: packet continues traversal as normal
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] Incoming packet: TCP 50.1.0.1:57028->50.1.0.100:80
[03-15 12:56:56] TCP source port 57028 dst port 80 flag [..A.]
[03-15 12:56:56] After FNAT-IN: TCP 50.1.0.1:57028->50.1.2.3:80
[03-15 12:56:56] Fast xmit, send packet to RS
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] Outgoing packet: TCP 50.1.2.3:80->50.1.0.1:57028
[03-15 12:56:56] TCP source port 80 dst port 57028 flag [..A.]
[03-15 12:56:56] After SNAT: TCP 50.1.0.100:80->50.1.0.1:57028
[03-15 12:56:56] Fast response xmit, send packet to client
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] Outgoing packet: TCP 50.1.2.3:80->50.1.0.1:57028
[03-15 12:56:56] TCP source port 80 dst port 57028 flag [..A.]
[03-15 12:56:56] After SNAT: TCP 50.1.0.100:80->50.1.0.1:57028
[03-15 12:56:56] Fast response xmit, send packet to client
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] [trace id:4]ip_vs_out: packet continues traversal as normal
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] Incoming packet: TCP 50.1.0.1:57028->50.1.0.100:80
[03-15 12:56:56] TCP source port 57028 dst port 80 flag [..A.]
[03-15 12:56:56] After FNAT-IN: TCP 50.1.0.1:57028->50.1.2.3:80
[03-15 12:56:56] Fast xmit, send packet to RS
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] [trace id:5]ip_vs_out: packet continues traversal as normal
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] Incoming packet: TCP 50.1.0.1:57028->50.1.0.100:80
[03-15 12:56:56] TCP source port 57028 dst port 80 flag [.FA.]
[03-15 12:56:56] TCP input [.FA.] 50.1.0.1:57028->50.1.2.3:80 state: ESTABLISHE D->CLOSE_
WAIT conn->refcnt:2
[03-15 12:56:56] After FNAT-IN: TCP 50.1.0.1:57028->50.1.2.3:80
[03-15 12:56:56] Fast xmit, send packet to RS
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] lookup TCP 50.1.2.3:80->50.1.0.1:57028 hit
[03-15 12:56:56] Outgoing packet: TCP 50.1.2.3:80->50.1.0.1:57028
[03-15 12:56:56] TCP source port 80 dst port 57028 flag [.FA.]
[03-15 12:56:56] TCP output [.FA.] 50.1.0.1:57028->50.1.2.3:80 state: CLOSE_WAI T->TIME_WAIT
conn->refcnt:2
[03-15 12:56:56] After SNAT: TCP 50.1.0.100:80->50.1.0.1:57028
[03-15 12:56:56] Fast response xmit, send packet to client
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
```

```
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] [trace id:6]ip_vs_out: packet continues traversal as normal
[03-15 12:56:56] lookup TCP 50.1.0.1:57028->50.1.0.100:80 hit
[03-15 12:56:56] Incoming packet: TCP 50.1.0.1:57028->50.1.0.100:80
[03-15 12:56:56] TCP source port 57028 dst port 80 flag [..A.]
[03-15 12:56:56] After FNAT-IN: TCP 50.1.0.1:57028->50.1.2.3:80
[03-15 12:56:56] Fast xmit, send packet to RS
[03-15 12:56:59] Expire session TCP c:50.1.0.1:57028 v:50.1.0.100:80 d:50.1.2.3: 80 fwd:M
s:5 conn->flags:80100 conn->refcnt:0 dest->refcnt:2

FortiADC-docs # diagnose debug flow stop
```

diagnose debug info

Use this command to display a list of debug log settings.

Syntax

```
diagnose debug info
```

Example

```
FortiADC-VM # diagnose debug info
debug output: disable
kernel debug level: 0 (0x0)
cli/cmdb debug level: 0 (0x0)
cmdb_event debug level: 0 (0x0)
gdns debug level: 0 (0x0)
kernelconfd debug level: 0 (0x0)
info_centerd debug level: 0 (0x0)
hasyncd debug level: 0 (0x0)
updated debug level: 0 (0x0)
miglogd debug level: 0 (0x0)
sshd debug level: 0 (0x0)
healthcheckd debug level: 2 (0x2)
netd debug level: 0 (0x0)
lb debug level: 0 (0x0)
udproxyd debug level: 0 (0x0)
httproxyd debug level: 0 (0x0)
dnsproxyd debug level: 0 (0x0)
alertmaild debug level: 0 (0x0)
synconf debug level: 0 (0x0)
ntpd debug level: 0 (0x0)
crlupdated debug level: 0 (0x0)
snmpd debug level: 0 (0x0)
flg_indexd debug level: 0 (0x0)
flg_reportd debug level: 0 (0x0)
flg_accessd debug level: 0 (0x0)
rtmd debug level: 0 (0x0)
```

```
ospfd debug level: 0 (0x0)
llb debug level: 0 (0x0)
```

diagnose debug module

Use this command to set the debug level for module daemons.

Syntax

```
diagnose debug module {adfs | alertd | apisecd | authd | av | awsd | bfd | bgpd | cm_client
      | cmdb | cmdb_event | crlupdated | dnsproxy | fast_statis | fcnacd | flg_accessd |
      flg_indexd | flg_reportd | fnginx | forticldd | gdns | hasyncd | healthcheckd |
      httproxy | httproxy3 | infod | ips-engine | ips-monitor | kernel | kernelconfd | kubed
      | lb | lbdns | llb | miglogd | named | netd | ntpd | ocid | opsips | ospfd | quard |
      rtmd | sapd | scanner_integration | shell-access | snmpd | sshd | ssl-of-httproxy |
      synconf | udproxy | updated | wadd | waf | wafmonitor | wassd | wccpd | wvs}
      [<level>]<Enter>
```

adfs	Get/set the debug level for AD FS daemon.
alertd	Get/set the debug level for automation stitches.
apisecd	Get/set the debug level for apisecd.
authd	Get/set the debug level for authd.
av	Get/set the debug level for anti-virus daemon.
awsd	Get/set the debug level for awsd.
bfd	Get/set the debug level for bfd daemon.
bgpd	Get/set the debug level for bgpd daemon.
cm_client	Get/set the debug level for cm_client daemon.
cmdb	Get/set the debug level for CLI and CMDB.
cmdb_event	Get/set the debug level for cmdb event.
crlupdated	Get/set the debug level for crlupdated daemon.
dnsproxy	Get/set the debug level for dnsproxy daemon.
fast_statis	Get/set the debug level for fast_statis events.
fcnacd	Get/set the debug level for fcnacd
flg_accessd	Get/set the debug level for flg_accessd daemon.
flg_indexd	Get/set the debug level for flg_indexd daemon.
flg_reportd	Get/set the debug level for flg_reportd daemon.

fnginx	Get/set the debug level for fnginx daemon.
forticldd	Get/set the debug level for forticldd daemon.
gdns	Get/set the debug level for gdns daemon.
hasyncd	Get/set the debug level for hasyncd daemon.
healthcheckd	Get/set the debug level for healthcheckd daemon.
httproxy	Get/set the debug category for httproxy daemon.
httproxy3	Get/set the debug level for httproxy3 daemon.
infod	Get/set the debug level for infod events.
ips-engine	Get/set the debug level for ips engine daemon.
ips-monitor	Get/set the debug level for ips monitor daemon.
kernel	Get/set the debug level for kernel.
kernelconfd	Get/set the debug level for L4 kernelconf daemon.
kubed	Get/set the debug level for kubed.
lb	Get/set the debug level for lb daemon.
lbdns	Get/set the debug level for lbdns daemon.
llb	Get/set the debug level for llb daemon.
miglogd	Get/set the debug level for miglogd daemon.
named	Get/set the debug level for named daemon.
netd	Get/set the debug level for netd daemon.
ntpd	Get/set the debug level for ntpd daemon.
ocid	Get/set the debug level for ocid.
opsips	Get/set the debug level for opsips daemon.
ospfd	Get/set the debug level for ospfd daemon.
quard	Get/set the debug level for quarantine daemon.
rtmd	Get/set the debug level for rtmd daemon.
sapd	Get/set the debug level for sapd.
scanner_integration	Get/set the debug level for scanner integration daemon.
shell-access	Get/set the debug level for shell access daemon.
snmpd	Get/set the debug level for snmpd daemon.
sshd	Get/set the debug level for sshd daemon.
ssl-of-httproxy	Get/set the debug level for httproxy ssl module.

synconf	Get/set the debug level for sysconf daemon.
udproxy	Get/set the debug level for udproxy daemon.
updated	Get/set the debug level for updated daemon.
wadd	Get/set the debug level for Web Anti Defacement daemon.
waf	Get/set the debug level for waf module in httproxy daemon.
wafmonitor	Get/set the debug level for wafmonitor.
wassd	Get/set the debug level for wassd.
wccpd	Get/set the debug level for wccpd events.
wvs	Get/set the debug level for wvs daemon.
<Enter>	If you do not specify a debug level and press Enter, the command displays the current debug level.
<level>	<level> is a mask. Valid levels are the following values added together: 1 - error message, 2 - main event, 4 - config event, 8 - file sync message, 16 - hb message, 31 - start all. For example, 3 means error messages and main events.

Example

```
FortiADC-VM # diagnose debug module lb ?
<level> set/get debug level for lb daemon

FortiADC-VM # diagnose debug module lb
lb debug level is 0

FortiADC-VM # diagnose debug module lb 3

FortiADC-VM # diagnose debug module lb
lb debug level is 3
```

Httproxy is not only organized by level.

```
diagnose debug module httproxy {submodule}/
     {all/alert/warning/confi/verbose}/{show/set-filter/show-
     filter/unset-filter}

show: Shows httproxy debug status
set-filter: Set debug filter for httproxy
unset-filter: Unset debug filter for httproxy
show-filter: Show debug filter of httproxy
all: All
alert: Httproxy Alert Message
warning: Httproxy Warning Message
conf: Httproxy config debug info
verbose: Httproxy traffic verbose debug info
ssl_major: Httproxy sll major debug info
```

diagnose debug module fcnacd

Use this command to view information about your FortiClient NAC daemon (fcnacd), which handles FortiADC to FortiClient EMS connectivity.

Syntax

```
diagnose debug module fcnacd {error | debug | info | all} {set | unset}
diagnose debug module fcnacd show
```

Example

```
ADC# diagnose debug module fcnacd info set
ADC# diagnose debug module fcnacd show

ocid debug switch status:
error is off
debug is off
info is on
```

diagnose debug module fnginx

Use this command to view load balancing debug information for the following modules:

fnginx modules	fnginx_new modules
• MySQL • Diameter • RTSP • RTMP	• SMTP • FTP • MSSQL • RADIUS • ISO8583

Debug filtering is supported for fnginx_new modules. For details, see Fnginx debug filter on page 520.

The `diagnose debug module fnginx` and `set-filter` commands can be executed on the root and non-root VDOMs. However, debug logs cannot be isolated from VDOMs.

Syntax

```
diagnose debug module fnginx {show|all|conf|ssl_ae_
     info|stat|rtsp|mysql|smtp|rtmp|diameter|ftp|radius|iso8583|mssql|av|scripting}
```

`show`	Show the fnginx debug status.
`all`	View the debug information for all fnginx modules.

conf	View the debug information for the configuration.
ssl_ae_info	View the debug information for the SSL Authenticated Encryption (AE) cryptography.
stat	View the statistic debug information.
rtsp	View the RTSP load balancing debug information.
mysql	View the MySQL load balancing debug information.
smtp	View the SMTP load balancing debug information.
rtmp	View the RTMP load balancing debug information.
diameter	View the Diameter load balancing debug information.
ftp	View the FTP load balancing debug information.
radius	View the RADIUS load balancing debug information.
iso8583	View the ISO8583 load balancing debug information.
mssql	View the MSSQL load balancing debug information.
av	View the Antivirus debug information.
scripting	View the stream scripting debug information.

Example

```
FortiADC-VM # diagnose debug module fnginx mysql set
profile type is mysql.
addr type 1.
make pool member conf, ip addr 20.6.2.1, port 80.
make pool member conf, ip addr 20.6.2.2, port 80.
make pool member conf, ip addr 20.6.2.3, port 80.
add vdom rlimit, vdom id: 1, ip: 1.1.1.1, port: 80, ssl: 0
test temp config success
dump configure data:
adc {
upstream mysql {
server 20.6.2.1:80 weight=1 up group_id=0 rs_name=pool1-1 id=3200;
server 20.6.2.2:80 weight=1 up group_id=0 rs_name=pool1-2 id=3201;
server 20.6.2.3:80 weight=1 up group_id=0 rs_name=pool1-3 id=3202;
mysql;
}
server mysql {
listen 1.1.1.1:80;
proxy_pass mysql;
fngx_log off;
persistence none;
source_address off;
mysql;
proxy_mode transaction;
mysql_mode 0;
}
}
```

Fnginx debug filter

You can set filters to specify the type of information to view to more easily troubleshoot and locate bugs. The set-filter/unset-filter options are currently only supported for **fnginx_new modules**. Both IPv4 and IPv6 are supported.

Any updates and changes to the debug filter will only take effect on new traffic flow. If the current connection is established, the following flow information will still be printed even if the filter does not match.

You can use the following keywords and operators to specify the debug filter:

Supported filter	Guideline
Keywords	• vsname — Virtual Server name • rsname — Real Server name • srcip — Source IP • dstip — Destination IP • srcport — Source port • dstport — Destination port Maximum length of a vsname/rsname is 63 characters.
Conditional operator	=, !=, in
Logical operator	&, |, () Maximum total number of logical operators and brackets is 32.

The maximum length of a filter expression is 1023 characters.

To better focus the debug log to only print the specified debug information, it is recommended to first enable debug for the specific fnginx module and set the debug filter, and then enable the debug output.

For example:

```
FADC # diagnose debug module fnginx smtp
FADC # diagnose debug module fnginx set-filter "(srcip in
     10.65.1.0/24) & vsname=vs-smtp-25"
FADC # diagnose debug enable
```

Syntax

```
diagnose debug module fnginx {set-filter|unset-filter|show-filter}
```

set-filter	Set the debug filter for fnginx modules. Set-filter only supports fnginx_new modules: SMTP, FTP, MSSQL, RADIUS, and ISO8583.
unset-filter	Unset the debug filter for fnginx modules.
show-filter	Show the debug filter for fnginx modules.

Example

```
diagnose debug module fnginx set-filter "srcip=10.65.1.62 & vsname=vs-smtp-25"
diagnose debug module fnginx set-filter "(srcip in 10.65.1.0/24) & vsname=vs-smtp-25"
diagnose debug module fnginx set-filter "(srcip=10.65.1.62 | srcip=2001:1234::a41:3e) &
(vsname=vs-smtp-25 | vsname=vs-smtp-IPv6) "
diagnose debug module fnginx set-filter "((srcip in 10.65.1.0/24) & vsname=vs-smtp-25) |
(srcip=10.65.1.62 & vsname=vs-radius)"
diagnose debug module fnginx set-filter "srcip=10.65.1.62 & srcport=20001 & vsname=vs-smtp-
25 & rsname=rs66 & dstport=25"
```

diagnose debug module httproxy scripting

Use this command to view debug information for HTTP Scripting. Set the `scripting` subcommand to check your Lua script. The `scripting_minor` subcommand outputs further information of the backend code for more in-depth troubleshooting.

Syntax

```
diagnose debug module httproxy scripting {set|unset}
diagnose debug module httproxy scripting_minor {set|unset}
```

diagnose debug module httproxy ssl

Use this command to view information about ssl.

Syntax

```
diagnose debug module httproxy ssl_major set/unset
diagnose debug module httproxy ssl_minor set/unset
diagnose debug module httproxy ssl_error set/unset
diagnose debug module httproxy ssl_ae_info set/unset
```

Example

```
FortiADC-VM # diagnose debug module httproxy ssl_major set
Thu Oct 5 2017 18:01:21.262797 16456 ssl_sock_init@(src/ssl_sock.c:3492) [sess id:4 vs:vs
     clt:5.1.1.1:44498] fd=2:client-side:common:ssl_init: Initing SSL
Thu Oct 5 2017 18:01:21.262797 16456 ssl_sock_handshake@(src/ssl_sock.c:5191) [sess id:4
     vs:vs clt:5.1.1.1:44498] fd=2:client-side:common:handshake: start calling SSL_do_
     handshake
Thu Oct 5 2017 18:01:21.262797 16456 ssl_sock_handshake@(src/ssl_sock.c:5220) [sess id:4
     vs:vs clt:5.1.1.1:44498] fd=2:client-side:common:handshake: Enable FD Read Poll
```

```
Thu Oct 5 2017 18:01:21.482538 16456 ssl_sock_handshake@(src/ssl_sock.c:5191) [sess id:4
     vs:vs clt:5.1.1.1:44498] fd=2:client-side:common:handshake: start calling SSL_do_
     handshake
Thu Oct 5 2017 18:01:21.482538 16456 ssl_sock_handshake@(src/ssl_sock.c:5220) [sess id:4
     vs:vs clt:5.1.1.1:44498] fd=2:client-side:common:handshake: Enable FD Read Poll
Thu Oct 5 2017 18:01:21.487098 16456 ssl_sock_handshake@(src/ssl_sock.c:5191) [sess id:4
     vs:vs clt:5.1.1.1:44498] fd=2:client-side:common:handshake: start calling SSL_do_
     handshake
Thu Oct 5 2017 18:01:21.487098 16456 shctx_new_cb@(src/shctx.c:435) [sess id:4 vs:vs
     clt:5.1.1.1:44498] fd=2:client-side:common:client_cache: new session
Thu Oct 5 2017 18:01:21.487098 16456 shsess_store@(src/shctx.c:357) [sess id:4 vs:vs
     clt:5.1.1.1:44498] fd=2:client-side:common:client_cache: trying to store
Thu Oct 5 2017 18:01:21.487098 16456 shsess_get_next@(src/shctx.c:312) [sess id:4 vs:vs
     clt:5.1.1.1:44498] fd=2:client-side:common:client_cache: trying to get vacuum space
Thu Oct 5 2017 18:01:21.487098 16456 shsess_store@(src/shctx.c:417) [sess id:4 vs:vs
     clt:5.1.1.1:44498] fd=2:client-side:common:client_cache: succeed
```

diagnose debug module httproxy ztna

Use this command to view information about your Layer 7 HTTPS virtual server that has referenced a ZTNA Profile.

Syntax

```
diagnose debug module httproxy ztna {set | unset}
```

diagnose debug module kernel

Use this command to set the debug log level for kernel debugging. When enabled, kernel errors are printed to the screen.

Syntax

```
diagnose debug module kernel [<level>] <Enter>
```

<Enter>	If you do not specify a debug level and press Enter, the command displays the current debug level.
<level>	Valid range is 0 to 8, where 0 disables debug logs and 8 generates the most verbose logging.

Example

```
FortiADC-VM # diagnose debug module kernel ?
<Integer> debug level (0-8).

FortiADC-VM # diagnose debug module kernel 5
```

```
FortiADC-VM # diagnose debug module kernel
Kernel debug level is 5
```

diagnose debug module miglogd syslog

Use this command to view the debug information for the miglogd syslog.

In functionality that require FortiADC to send logs to servers that uses TCP SSL (such as FortiWeb Cloud, Rsyslog, or other open source servers with SSL protocols), the TCP SSL certificates and negotiations is complex, involving several steps. In the case where communication issues arise between FortiADC and the server, you can use the `diagnose debug module miglogd syslog` command to print out the key information about these communications to find the cause.

Syntax

```
diagnose debug module miglogd syslog
```

diagnose debug module named

Use this command to view information about the named daemon for debugging purposes.

The named daemon debug logs will be saved into individual files for each VDOM:

- `/tmp/root_named.log` for root VDOM
- `/tmp/vdom_named.log` for non-root VDOM

Syntax

```
diagnose debug module named {traffic | dnssec | statistics | event | all}
diagnose debug module named show
```

`traffic`	View the named daemon traffic messages.
`dnssec`	View the named daemon DNSSEC messages.
`statistics`	View the named daemon statistics messages.
`event`	View the named daemon event messages.
`all`	View all named daemon messages.
`show`	Shows the debug status of the named daemon.

diagnose debug module waf

Use this command to view information about your Web Application Firewall.

Syntax

```
diagnose debug module waf { show | all | mem_detail | mem | rule | ac | ac_detail | dfa |
     sql_xss | http_cons | url_protect | rulescan | framework | framework_scan | framework_
     http | db | log | config | exception | bot | multipart_decoder | urlencoded_decoder |
     chunk_decoder | decompress_decoder | html_decoder | json_decoder | xml_decoder |
     openapi_decoder | cookie_security | brute_protection | html_input_validation |
     advanced_protection | csrf_protect | credential_stuff | dlp | http_header_sec | api_
     gateway | cors_protect | biometrics | threshold | fingerprint | api_security |
     advanced_bot}
```

`show`	Shows the WAF debug status.
`all`	View all WAF debug information.
`mem_detail`	View the WAF memory debug information details.
`mem`	View the WAF memory debug information.
`rule`	View the debug information for the WAF signature rules operation.
`ac`	View the debug information for the AC (Aho–Corasick) string matching of WAF signatures.
`ac_detail`	View the debug information details for the AC (Aho–Corasick) string matching of WAF signatures.
`dfa`	View the debug information for the DFA (deterministic finite automaton) string matching of WAF signatures.
`sql_xss`	View the SQL/XSS Injection Detection debug information.
`http_cons`	View the HTTP Protocol Constraint debug information.
`url_protect`	View the URL Protection debug information.
`rulescan`	View the Web Attack Signature scanning debug information.
`framework`	View the WAF framework debug information.
`framework_scan`	View the WAF framework scanner debug information.
`framework_http`	View the WAF framework HTTP protocol parser debug information.
`db`	View the WAF database debug information.
`log`	View the WAF attack log debug information.
`config`	View the WAF Profile configuration debug information.
`exception`	View the WAF Exception rule configuration debug information.
`bot`	View the Bot Detection debug information.
`multipart_decoder`	View the WAF multipart decoder debug information.

`urlencoded_decoder`	View the WAF URL-encoded decoder debug information.
`chunk_decoder`	View the WAF chunk decoder debug information.
`decompress_decoder`	View the WAF decompress decoder debug information.
`html_decoder`	View the WAF HTML decoder debug information.
`json_decoder`	View the WAF JSON decoder debug information.
`xml_decoder`	View the WAF XML decoder debug information.
`openapi_decoder`	View the OpenAPI decoder debug information.
`cookie_security`	View the Cookie Security debug information.
`brute_protection`	View the Brute Force Attack Detection debug information.
`html_input_ validation`	View the Input Validation Policy debug information.
`advanced_ protection`	View the Advanced Protection debug information.
`csrf_protect`	View the CSRF Protection debug information.
`credential_stuff`	View the Credential Stuffing Defense debug information.
`dlp`	View the Data Leak Prevention debug information.
`http_header_sec`	View the HTTP Header Security debug information.
`api_gateway`	View the API Gateway debug information.
`cors_protect`	View the CORS Protection debug information.
`biometrics`	View the Biometrics Based Detection debug information.
`threshold`	View the Threshold Based Detection debug information.
`fingerprint`	View the Fingerprint Based Detection debug information.
`api_security`	View the API Security debug information.
`advanced_bot`	View the Advanced Bot Protection debug information.

diagnose debug module wassd

Use this command to set the wassd debug log level. When you enable debug, the output will reflect the debug log level set with `diagnose debug module wassd` command.

The wassd daemon forms the connection between FortiADC and FortiWeb Cloud and performs several integral functions when AI Threat Analytics is enabled. This includes the following:

- Establishing a web socket connection with the FortiWeb Cloud using a token. The wassd identifies whether a CA exists before registering to the FortiWeb Cloud. If a CA does exist, then the wassd will send the issue date of the CA certificate to the FortiWeb Cloud.

- Updating FortiWeb Cloud with FortiADC configuration changes, such as HA status changes, member updates, or mode modification.
- Updating device certificates received from the FortiWeb Cloud. If wassd registered to the FortiWeb Cloud without the issue date of the CA or that the certificate has expired, then FortiWeb Cloud will send new certificates (including the certificate, key, and CA) to wassd. The wassd will update to the local certificate and CA table, and register to FortiWeb Cloud again with the latest CA issue date.
- Starting the forwarding of FortiADC attack logs to FortiWeb Cloud. If wassd has successfully registered to FortiWeb Cloud, then it will start the action with the log server and port from the FortiWeb Cloud.

Note:

The wassd daemon is create for AI Threat Analytics and executes the `wassd_ws` Python script when AI Threat Analytics is enabled. The backend log for the Python script is stored in `/var/log/wassd.log`.

Syntax

```
diagnose debug module wassd {show | all}
diagnose debug module wassd {error | info | debug}|{set | unset}
```

`show`	Shows the wassd debug switch status.
`all`	Set the debug switch status to all option to **on**.
`error`	Turn on/off the wassd debug switch for errors logged in the wassd. If the **error** debug switch status is **on**, when you print all wassd debug information, the output will include wassd errors.
`info`	Turn on/off the wassd debug switch for general information logged about wassd system operations. If the **info** debug switch status is **on**, when you print all wassd debug information, the output will include wassd general system operation information.
`debug`	Turn on/off the wassd debug switch for detailed information about wassd that can be used to troubleshoot unexpected behavior. If the **debug** debug switch status is **on**, when you print all wassd debug information, the output will include wassd debug logs.
`set`	Set the debug switch **on**.
`unset`	Set the debug switch **off**.

Example

```
FortiADC-VM # diagnose debug module wassd show
wassd debug switch status:
error is off
debug is off
info is off

FortiADC-VM # diagnose debug module wassd all

FortiADC-VM # diagnose debug module wassd show
wassd debug switch status:
```

```
error is on
debug is on
info is on

FortiADC-VM # diagnose debug module wassd error
set    Set switch on.
unset  Set switch off.
```

diagnose debug timestamp

Use this command to timestamp debug messages.

Syntax

```
diagnose debug timestamp {enable|disable}
```

Example

```
FortiADC-VM (root) # diagnose debug timestamp enable
FortiADC-VM (root) # 2016-01-11 18:10:03 [trace common]Destroy conntrack:protocol 1, In if 0
     3.3.3.3:24104 -> 4.4.4.4:2048 Reverse:In if 0 4.4.4.4:24104 -> 3.3.3.3:0
2016-01-11 18:10:03 [trace id:13]recv a ip packet, MAC 00:0c:29:4d:fe:84 ->
     00:0c:29:b2:41:f2 3.3.3.3 -> 4.4.4.4 iif port2 proto 1dent 0 flags 0x40 length 84 ttl
     64
2016-01-11 18:10:03 [trace id:13]record reverse route info into session: iif port2 mac
     00:0c:29:4d:fe:84
2016-01-11 18:10:03 [trace id:13]No session matched, create new session
2016-01-11 18:10:03 [trace common]tuple src 0x3030303 sport 0, dst 0x4040404 dport 0, proto
2016-01-11 18:10:03 [trace common]use dest address hash, len=1
2016-01-11 18:10:03 [trace common]iif 7 oif 0 tuple src 0x3030303 dst 0x4040404 proto 1
     sport 0 dport 0
2016-01-11 18:10:03 [trace common]matched policy 1
2016-01-11 18:10:03 [trace common]llb route table id 4097
2016-01-11 18:10:03 [trace id:13]find input route interface port3 nexthop 1.1.1.1
2016-01-11 18:10:03 [trace id:13]ip output by if port3
2016-01-11 18:10:03 [trace id:13]DSTCACHE: save dst dir 0, nexthop 1.1.1.1 dev port3 filled
     into SESSION prot 1 [3.3.3.3:24106, 4.4.4.4:2048] -> [4.4.4.4:24106, 3.3.3.3:0]
2016-01-11 18:10:03 [trace id:13]Confirm conntrack:protocol 1, In if 0 3.3.3.3:24106 ->
     4.4.4.4:2048 Reverse:In if 0 4.4.4.4:24106 -> 3.3.3.3:0
2016-01-11 18:10:03 [trace id:13]ip finish output2 nexthop by route 0x1010101 if port3
2016-01-11 18:10:03 [trace id:14]recv a ip packet, MAC 00:0c:29:44:93:be ->
     00:0c:29:b2:41:fc 4.4.4.4 -> 3.3.3.3 iif port3 proto 1dent 6852 flags 0x0 length 84
     ttl 63
2016-01-11 18:10:03 [trace id:14]Session found
2016-01-11 18:10:03 [trace id:14]find input route interface port2 nexthop 0.0.0.0
2016-01-11 18:10:03 [trace id:14]ip output by if port2
2016-01-11 18:10:03 [trace id:14]DSTCACHE: save dst dir 1, nexthop 0.0.0.0 dev port2 filled
     into SESSION prot 1 [3.3.3.3:24106, 4.4.4.4:2048] -> [4.4.4.4:24106, 3.3.3.3:0]
2016-01-11 18:10:03 [trace id:14]Transmit packet by reverse route, dev port2 dest mac
     00:0c:29:4d:fe:84
```

diagnose endpoint-control client list

Use this command to view information about the FortiClient endpoints synchronized to FortiADC from FortiClient EMS.

Syntax

```
diagnose endpoint-control client list
```

diagnose endpoint-control tag list

Use this command to view information about the ZTNA tags synchronized to FortiADC from FortiClient EMS.

Syntax

```
diagnose endpoint-control tag list
```

diagnose firewall-session clear

Use this command to clear all firewall sessions in the connection tracking table.

Syntax

```
diagnose firewall-session clear
```

diagnose hardware deviceinfo

Use this command to display hardware information that might be useful in debugging.

Syntax

```
diagnose hardware {get|set} deviceinfo nic [<port>] <Enter>
diagnose hardware {get|set} deviceinfo nic-detail [<port>] <Enter>
```

`nic`	Displays port settings. If you do not specify a port and press Enter, the command displays output for all ports.
`nic-detail`	Displays detailed port settings and statistics. If you do not specify a port and press Enter, the command displays output for all ports.

Example

```
FortiADC-VM # diagnose hardware get deviceinfo ?
nic display network interface controller status
nic-detail display detailed network interface controller status

FortiADC-VM # diagnose hardware get deviceinfo nic-detail port1
Interface: port1
driver: vmxnet3
version: 1.1.29.0-k-NAPI
firmware-version:
bus-info: 0000:03:00.0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no

Settings for port1:
Supported ports: [ TP ]
Supported link modes: 1000baseT/Full
10000baseT/Full
Supported pause frame use: No
Supports auto-negotiation: No
Advertised link modes: Not reported
Advertised pause frame use: No
Advertised auto-negotiation: No
Speed: 10000Mb/s
Duplex: Full
Port: Twisted Pair
PHYAD: 0
Transceiver: internal
Auto-negotiation: off
MDI-X: Unknown
Supports Wake-on: uag
Wake-on: d
Link detected: yes

Pause parameters for port1:
Cannot get device pause settings: Operation not supported
Inter-| Receive | Transmit
face |bytes packets errs drop fifo frame compressed multicast|bytes packe ts errs drop fifo
     colls carrier compressed
port10: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

diagnose hardware ioport

Use this command to display I/O information that might be useful in debugging.

Syntax

```
diagnose hardware {get|set} ioport {byte|word|long} <address_hex>
```

ioport	Specify whether to read byte, word, or long from the port.
address_hex	The hexadecimal address of the I/O port.

Example

First, use the `diagnose hardware sysinfo` command to find the address hex number for the port you want to diagnose:

```
FortiADC-VM # diagnose hardware get sysinfo ioports
0000-0cf7 : PCI Bus 0000:00
 0000-001f : dma1
0020-0021 : pic1
0040-0043 : timer0
0050-0053 : timer1
0060-0060 : keyboard
0064-0064 : keyboard
0070-0077 : rtc
0080-008f : dma page reg
00a0-00a1 : pic2
00c0-00df : dma2
00f0-00ff : fpu
0170-0177 : 0000:00:07.1
0170-0177 : piix
01f0-01f7 : 0000:00:07.1
01f0-01f7 : piix
02f8-02ff : serial
0376-0376 : 0000:00:07.1
0376-0376 : piix
03c0-03df : vga+
03f6-03f6 : 0000:00:07.1
03f6-03f6 : piix
03f8-03ff : serial
0cf0-0cf1 : pnp 00:00
0cf8-0cff : PCI conf1
0d00-feff : PCI Bus 0000:00
1000-103f : 0000:00:07.3
1000-103f : pnp 00:00
1000-1003 : ACPI PM1a_EVT_BLK
1004-1005 : ACPI PM1a_CNT_BLK
1008-100b : ACPI PM_TMR
100c-100f : ACPI GPE0_BLK
1010-1015 : ACPI CPU throttle
1040-104f : 0000:00:07.3
1040-104f : pnp 00:00
1060-107f : pnp 00:0b
1080-10bf : 0000:00:07.7
10c0-10cf : 0000:00:07.1
10c0-10cf : piix
10d0-10df : 0000:00:0f.0
1400-14ff : 0000:00:10.0
```

```
2000-3fff : PCI Bus 0000:02
4000-4fff : PCI Bus 0000:03
4000-400f : 0000:03:00.0
5000-5fff : PCI Bus 0000:0b
5000-500f : 0000:0b:00.0
6000-6fff : PCI Bus 0000:13
6000-600f : 0000:13:00.0
7000-7fff : PCI Bus 0000:1b
7000-700f : 0000:1b:00.0
8000-8fff : PCI Bus 0000:04
8000-800f : 0000:04:00.0
9000-9fff : PCI Bus 0000:0c
9000-900f : 0000:0c:00.0
a000-afff : PCI Bus 0000:14
a000-a00f : 0000:14:00.0
b000-bfff : PCI Bus 0000:1c
b000-b00f : 0000:1c:00.0
c000-cfff : PCI Bus 0000:05
c000-c00f : 0000:05:00.0
d000-dfff : PCI Bus 0000:0d
d000-d00f : 0000:0d:00.0
e000-efff : PCI Bus 0000:15
```

Then, use the `diagnose hardware ioport` command to display the ioport value:

```
FortiADC-VM # diagnose hardware get ioport long 001f
inl(001f)=ffffffff
```

diagnose hardware pciconfig

Use this command to display PCI registers that might be useful in debugging.

Syntax

```
diagnose hardware {get|set} pciconfig [bus <bus> | id <id> | option <option>] <Enter>
```

`bus`	Display registers for the specified bus.
`id`	Display registers for the specified id.
`option`	Options for displaying the register.

Example

```
FortiADC-VM # diagnose hardware get pciconfig ?
bus list devices on the specified bus
id list devices with the specified vendor and device ID
option v n t x H1
<Enter>
```

```
FortiADC-VM # diagnose hardware get pciconfig
00:00.0 Class 0600: 8086:7190 (rev 01)
00:01.0 Class 0604: 8086:7191 (rev 01)
00:07.0 Class 0601: 8086:7110 (rev 08)
00:07.1 Class 0101: 8086:7111 (rev 01)
00:07.3 Class 0680: 8086:7113 (rev 08)
00:07.7 Class 0880: 15ad:0740 (rev 10)
00:0f.0 Class 0300: 15ad:0405
00:10.0 Class 0100: 1000:0030 (rev 01)
00:11.0 Class 0604: 15ad:0790 (rev 02)
00:15.0 Class 0604: 15ad:07a0 (rev 01)
00:15.1 Class 0604: 15ad:07a0 (rev 01)
00:15.2 Class 0604: 15ad:07a0 (rev 01)
00:15.3 Class 0604: 15ad:07a0 (rev 01)
00:15.4 Class 0604: 15ad:07a0 (rev 01)
00:15.5 Class 0604: 15ad:07a0 (rev 01)
00:15.6 Class 0604: 15ad:07a0 (rev 01)
00:15.7 Class 0604: 15ad:07a0 (rev 01)
00:16.0 Class 0604: 15ad:07a0 (rev 01)
00:16.1 Class 0604: 15ad:07a0 (rev 01)
00:16.2 Class 0604: 15ad:07a0 (rev 01)
00:16.3 Class 0604: 15ad:07a0 (rev 01)
00:16.4 Class 0604: 15ad:07a0 (rev 01)
00:16.5 Class 0604: 15ad:07a0 (rev 01)
00:16.6 Class 0604: 15ad:07a0 (rev 01)
00:16.7 Class 0604: 15ad:07a0 (rev 01)
00:17.0 Class 0604: 15ad:07a0 (rev 01)
00:17.1 Class 0604: 15ad:07a0 (rev 01)
00:17.2 Class 0604: 15ad:07a0 (rev 01)
00:17.3 Class 0604: 15ad:07a0 (rev 01)
00:17.4 Class 0604: 15ad:07a0 (rev 01)
00:17.5 Class 0604: 15ad:07a0 (rev 01)
00:17.6 Class 0604: 15ad:07a0 (rev 01)
00:17.7 Class 0604: 15ad:07a0 (rev 01)
00:18.0 Class 0604: 15ad:07a0 (rev 01)
00:18.1 Class 0604: 15ad:07a0 (rev 01)
00:18.2 Class 0604: 15ad:07a0 (rev 01)
00:18.3 Class 0604: 15ad:07a0 (rev 01)
00:18.4 Class 0604: 15ad:07a0 (rev 01)
00:18.5 Class 0604: 15ad:07a0 (rev 01)
00:18.6 Class 0604: 15ad:07a0 (rev 01)
00:18.7 Class 0604: 15ad:07a0 (rev 01)
03:00.0 Class 0200: 15ad:07b0 (rev 01)
04:00.0 Class 0200: 15ad:07b0 (rev 01)
05:00.0 Class 0200: 15ad:07b0 (rev 01)
0b:00.0 Class 0200: 15ad:07b0 (rev 01)
0c:00.0 Class 0200: 15ad:07b0 (rev 01)
0d:00.0 Class 0200: 15ad:07b0 (rev 01)
13:00.0 Class 0200: 15ad:07b0 (rev 01)
14:00.0 Class 0200: 15ad:07b0 (rev 01)
1b:00.0 Class 0200: 15ad:07b0 (rev 01)
1c:00.0 Class 0200: 15ad:07b0 (rev 01)

FortiADC-VM # diagnose hardware get pciconfig option ?
v verbose information
n display number id
```

```
t tree view of bus
x dump configuration space data in hexadecimal
H1 direct access hardware

FortiADC-VM # diagnose hardware get pciconfig option t
-[00]-+-00.0
+-01.0-[01]--
+-07.0
+-07.1
+-07.3
+-07.7
+-0f.0
+-10.0
+-11.0-[02]--
+-15.0-[03]----00.0
+-15.1-[04]----00.0
+-15.2-[05]----00.0
+-15.3-[06]--
+-15.4-[07]--
+-15.5-[08]--
+-15.6-[09]--
+-15.7-[0a]--
+-16.0-[0b]----00.0
+-16.1-[0c]----00.0
+-16.2-[0d]----00.0
+-16.3-[0e]--
+-16.4-[0f]--
+-16.5-[10]--
+-16.6-[11]--
+-16.7-[12]--
+-17.0-[13]----00.0
+-17.1-[14]----00.0
+-17.2-[15]--
+-17.3-[16]--
+-17.4-[17]--
+-17.5-[18]--
+-17.6-[19]--
+-17.7-[1a]--
+-18.0-[1b]----00.0
+-18.1-[1c]----00.0
+-18.2-[1d]--
+-18.3-[1e]--
+-18.4-[1f]--
+-18.5-[20]--
+-18.6-[21]--
`-18.7-[22]--
```

diagnose hardware sysinfo

Use this command to display system information that might be useful in debugging.

Syntax

```
diagnose hardware {get|set} sysinfo {cpu | interrupts | iomen | ioports | memory | mtrr |
     slab | stream | df>
```

cpu	Display detailed information for all CPU.
interrupts	Display system interrupt information.
iomem	Display the memory map of I/O ports.
ioports	Display the address list of I/O ports.
memory	Display system memory information.
mttr	Display the memory type range register.
slab	Display memory allocation information.
stream	Display STREAM benchmark results.
df	Display disk free information.

Example

```
FortiADC-VM # diagnose hardware get sysinfo ?
cpu display detailed information for all installed CPU(s)
interrupts display system interrupts information
iomem display the memory map of I/O ports
ioports display the address list of I/O ports
memory display system memory information
mtrr display the memory type range register
slab display memory allocation information
stream display STREAM benchmark results
df display disk free information

FortiADC-VM # diagnose hardware get sysinfo df
Filesystem Size Used Available Use% Mounted on
/dev/root 193.7M 141.6M 52.1M 73% /
none 0 0 0 0% /proc
none 0 0 0 0% /sys
none 0 0 0 0% /sys/kernel/debug
none 256.0M 7.6M 248.4M 3% /tmp
none 0 0 0 0% /dev/pts
none 256.0M 0 256.0M 0% /dev/shm
/dev/sda2 96.8M 68.8M 23.1M 75% /data
/dev/sdb1 23.5G 1.3G 21.0G 6% /var/log
/dev/sda3 378.3M 10.1M 348.7M 3% /home
/dev/loop0 984.3M 35.2M 899.1M 4% /var/log/debug
```

diagnose llb policy list

Use this command to display diagnostic information about link load balancing policies.

Syntax

```
diagnose llb policy list
```

Example

```
FortiADC-docs # diagnose llb policy list
------------------------------------------------------------
policy index 1, route table id 4097
flag (0):
ingress if(1): 7
dest(0):
service(0):
```

diagnose netlink backlog

Use this command to set the backlog length.

Syntax

```
diagnose netlink backlog [get] [<integer>]
```

[get]	Specify the get option to display the current setting. Otherwise, the command sets the backlog length.
<integer>	Backlog length.

Example

```
FortiADC-VM # diagnose netlink backlog ?
get see current backlog length
<backlog> set new backlog length

FortiADC-VM # diagnose netlink backlog get
Current backlog is 1000
```

```
FortiADC-VM # diagnose netlink backlog 2000

FortiADC-VM # diagnose netlink backlog get
Current backlog is 2000
```

diagnose netlink device

Use this commad to display network interface RX/TX statistics.

Syntax

```
diagnose netlink device
```

Example

```
FortiADC-VM # diagnose netlink device
Inter-| Receive | Transmit
face |bytes packets errs drop fifo frame compressed multicast|bytes packe ts errs drop fifo
     colls carrier compressed
vtb0: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
vtb1: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
port3: 418337774 4267852 0 168 0 0 0 363608 260 2 0 0 0 0 0 0
port10: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
port8: 418337474 4267847 0 163 0 0 0 363608 260 2 0 0 0 0 0 0
vsport-101010A: 0 0 0 0 0 0 0 0 2 60 2 0 0 0 0 0 0
port5: 418337654 4267850 0 166 0 0 0 363608 260 2 0 0 0 0 0 0
gre0: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
gre1: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
port2: 418334234 4267793 0 169 0 0 0 363608 2910 63 0 0 0 0 0 0
bond0: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
imq0: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
port7: 418337534 4267848 0 164 0 0 0 363608 260 2 0 0 0 0 0 0
lo: 123360587 775740 0 0 0 0 0 0 123360587 7 75740 0 0 0 0 0 0
port4: 418337714 4267851 0 167 0 0 0 363608 260 2 0 0 0 0 0 0
port9: 418337474 4267847 0 162 0 0 0 363609 1034285 1 2167 0 0 0 0 0 0
port1: 491225752 5104578 0 170 0 0 0 363608 174736576 15 03116 0 0 0 0 0 0
sit0: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
port6: 418337594 4267849 0 165 0 0 0 363608 260 2 0 0 0 0 0 0
haport0: 0 0 0 0 0 0 0 0 1034025 12 165 0 5 0 0 0 0
```

diagnose netlink interface

Use this command to display detailed network interface information, such as family, type, MTU, flags. It is similar to the shell command `ifconfig`.

Syntax

```
diagnose netlink interface list [<interface>] <Enter>
```

Example

```
FortiADC-VM # diagnose netlink interface ?
list list interface

FortiADC-VM # diagnose netlink interface list ?
<interface-name> interface name
port1 physical
port2 physical
port3 physical
port4 physical
port5 physical
port6 physical
port7 physical
port8 physical
port9 physical
port10 physical


FortiADC-VM # diagnose netlink interface list port1

if=port1 family=00 type=1 index=4 mtu=1500 link=0 primary=0
flags=up broadcast run multicast
Qdisc=pfifo_fast hw_addr=00:09:0f:09:00:01: broadcast_addr=ff:ff:ff:ff:ff:ff:
stat: rxp=6453991526227804 txp=749850502384418443 rxb=0 txb=170 rxe=363546 txe=0 rxd=0 txd=0
     mc=0 collision=0
re: rxl=0 rxo=6474731918196736 rxc=5103452 rxf=1502687 rxfi=491093643 rxm=174588 175
te: txa=0 txc=0 txfi=170 txh=0 txw=363546
```

diagnose netlink ip/ipv6

Use these commands to list interface details, or to add or delete a physical network interface.

Back up the configuration before deleting a network interface table entry.

Syntax

```
diagnose netlink {ip|ipv6} add <interface_name> <ipaddress> <netmask>
diagnose netlink {ip|ipv6} delete <interface_name> <ipaddress>
diagnose netlink {ip|ipv6} flush
diagnose netlink {ip|ipv6} list
```

<interface_name>	Name of the interface to add or delete from the network interface table.
<ipaddress>	IP address of the network interface.
<netmask>	Subnet mask.

Example

```
FortiADC-VM # FortiADC-VM # diagnose netlink ip ?
add add netlink ip address
delete delete netlink ip address
flush flush netlink ip address
list list netlink ip address

FortiADC-VM # diagnose netlink ip list
IP=127.0.0.1 MASK=255.255.255.0 index=1 devname=lo
IP=127.129.1.1 MASK=255.255.255.255 index=1 devname=lo
IP=172.30.144.100 MASK=255.255.252.0 index=4 devname=port1
IP=10.1.1.1 MASK=255.255.255.255 index=4 devname=port1
IP=7.7.7.7 MASK=255.255.255.255 index=7 devname=port2
IP=5.5.5.5 MASK=255.255.255.255 index=7 devname=port2
IP=11.11.11.11 MASK=255.255.255.255 index=7 devname=port2
IP=12.12.12.12 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.9 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.8 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.7 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.6 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.5 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.4 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.3 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.2 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.1 MASK=255.255.255.255 index=7 devname=port2
IP=172.0.0.0 MASK=255.255.255.255 index=7 devname=port2
IP=1.1.100.1 MASK=255.255.255.255 index=7 devname=port2
IP=1.1.100.2 MASK=255.255.255.255 index=7 devname=port2
IP=169.254.160.134 MASK=255.255.0.0 index=17 devname=haport0
```

diagnose netlink neighbor/neighbor6

Use these commands to list the neighbor table (ARP cache), or to add or delete neighbors.

Syntax

```
diagnose netlink {neighbor|neighbor6} add <interface_name> <ipaddress> <macaddress>
diagnose netlink {neighbor|neighbor6} delete <interface_name> <ipaddress>
diagnose netlink {neighbor|neighbor6} flush
diagnose netlink {neighbor|neighbor6} list
```

<interface_name>	Name of the interface to add or delete from the neighbors table.
<ipaddress>	IP address of the network interface.
<macaddress>	MAC address.

Example

```
FortiADC-VM # diagnose netlink neighbor list
ifindex=1 ifname=lo 127.0.0.1 00:00:00:00:00:00 state=00000040 use=2255 confirm=8255
     update=2255 ref=0
```

diagnose netlink route/route6

Use this command to display the route table.

Syntax

```
diagnose netlink {route|route6} [list | flush]
```

Example

```
FortiADC-VM # diagnose netlink route ?
list list routing table
flush flush routing table

FortiADC-VM # diagnose netlink route list
tab=252 type=local protocol=boot flag=00000000 oif=1(lo) prio=400
tab=0 type=unreachable protocol=kernel flag=00000000 oif=1(lo) prio=ffffffff
tab=254 type=unicast protocol=kernel flag=00000000 oif=17(haport0) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=4(port1) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=7(port2) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=10(port3) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=12(port4) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=5(port5) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=8(port6) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=11(port7) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=13(port8) dst:fe80::/64 prio=100
tab=254 type=unicast protocol=kernel flag=00000000 oif=6(port9) dst:fe80::/64 prio=100
```

```
tab=254 type=unicast protocol=kernel flag=00000000 oif=9(port10) dst:fe80::/64 prio=100
tab=0 type=unreachable protocol=kernel flag=00000000 oif=1(lo) prio=ffffffff
tab=255 type=local protocol=unspec flag=00000000 oif=1(lo) dst:::1/128 gwy::: prio=0
tab=255 type=local protocol=unspec flag=00000000 oif=1(lo) dst:fe80::/128 gwy::: prio=0

[...]
```

diagnose netlink tcp

Use this command to view a list of TCP raw socket details, including:

- `sl` — Kernel socket hash slot.
- `local_address` — IP address and port number pair of the network interface in hexadecimal, such as `DD01010A:0050`.
- `rem_address` — Remote host network interface and port number pair. If not connected, this will contain `00000000:0000`.
- `st` — TCP state code (e.g. `OA` for listening, `01` for established, or `06` for timeout wait)
- `tx_queue` — Kernel memory usage by the transmission queue.
- `rx_queue` — Kernel memory usage by the retransmission queues.
- `tr`, `tm-> when`, `retrnsmt` — Kernel socket state debugging information.
- `uid` — User ID of the socket's creator (on FortiADC, always `0`).
- `timeout` — Connection timeout.
- `inode` — Pseudo-file system i-node of the process.

Syntax

```
diagnose netlink tcp
```

Example

```
FortiADC-VM # diagnose netlink tcp
sl local_address rem_address st tx_queue rx_queue tr tm->when retrnsmt ui d timeout inode
0: 86A0FEA9:0015 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2455 1
     ffff88005ad16f40 100 0 0 10 0
1: 0100007F:0035 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2852 1
     ffff88005c6acd80 100 0 0 10 0
2: 64901EAC:0035 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2855 1
     ffff88005c6ad440 100 0 0 10 0
3: 64901EAC:0016 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 38004000 1
     ffff88005f4ce880 100 0 0 10 0
4: 86A0FEA9:0016 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 38004001 1
     ffff88005f4cc6c0 100 0 0 10 0
5: 0100007F:0016 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 38004003 1
     ffff88005f4ce1c0 100 0 0 10 0
6: 64901EAC:0017 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2451 1
     ffff88005ad15b00 100 0 0 10 0
7: 86A0FEA9:0017 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2452 1
     ffff88005ad161c0 100 0 0 10 0
```

```
8: 0100007F:0017 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2453 1
     ffff88005ad16880 100 0 0 10 0
9: 0100007F:03B9 00000000:0000 0A 00000000:00000000 00:00000000 00000000 0 0 2959 1
     ffff88005c6adb00 100 0 0 10 0

[...]
```

diagnose netlink udp

Use this command to view a list of UDP raw socket details, including:

- `sl` — Kernel socket hash slot.
- `local_address` — IP address and port number pair of the network interface in hexadecimal, such as `DD01010A:0050`.
- `rem_address` — Remote host network interface and port number pair. If not connected, this will contain `00000000:0000`.
- `st` — TCP state code in hexadecimal (e.g. `0A` for listening, `01` for connection established, or `06` for waiting for data)
- `tx_queue` — Kernel memory usage by the transmission (Tx) queue.
- `rx_queue` — Kernel memory usage by the retransmission (Rx) queues. (This is not used by UDP, since the protocol itself does not support retransmission.)
- `tr`, `tm-> when`, `retrnsmt` — Kernel socket state debugging information. (These are not used by UDP, since the protocol itself does not support retransmission.)
- `uid` — User ID of the socket's creator (on FortiADC, always `0`).
- `timeout` — Connection timeout.
- `inode` — Pseudo-file system inode of the process.
- `ref`, `pointer` — Pseudo-file system references.

Syntax

```
diagnose netlink udp
```

Example

```
FortiADC-VM # diagnose netlink udp
sl local_address rem_address st tx_queue rx_queue tr tm->when retrnsmt ui d timeout inode
     ref pointer drops
171: 0100007F:0FA0 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 1165 2
     ffff88006bf90000 0
202: 00000000:87BF 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 4962 2
     ffff88006bf91500 0
223: 00000000:F7D4 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 38534860 2
     ffff88005f319180 0
318: 00000000:3033 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 38504036 2
     ffff88005f318700 0
319: 00000000:D034 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 3279 2
     ffff88006bf90e00 0
320: 64901EAC:0035 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 2854 2
     ffff88006bf90a80 0
```

```
320: 0100007F:0035 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 2851 2
     ffff88006bf90700 0
475: 00000000:ECD0 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 24123242 2
     ffff88005f318000 0
494: 00000000:24E3 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 38500439 2
     ffff88005f318a80 0
546: 00000000:2D17 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 20533867 2
     ffff88005f318380 0
610: 00000000:9957 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 37907911 2
     ffff88005f319500 0
1010: 00000000:52E7 00000000:0000 07 00000000:00000000 00:00000000 00000000 0 0 3576 2
     ffff88006bf90380 0
```

diagnose server-load-balance dns-clients

Use this command to display the virtual servers.

Syntax

```
diagnose server-load-balance dns-clients virtual-server
```

Example

```
FortiADC-VM (root) # diagnose server-load-balance dns-clients virtual-server
virtual-server virtual server name
L4VS load-balance.virtual-server
L7VS load-balance.virtual-server
1 load-balance.virtual-server
```

diagnose server-load-balance persistence

Use this command to filter and display the persistence table (current sessions).

Syntax

```
diagnose server-load-balance persistence filter {'<option>'|show|clear}
diagnose server-load-balance persistence list
diagnose server-load-balance persistence clear [l4]
```

`filter`	Create, show, or clear session list filters.

	Use multiple commands to add filters to the filter list. For example, one command to add a source-ip filter and another to add a vs-name filter. Put the filter expression in single quotes. Filter options include: • `source-ip` — Single IP address or specify start and end addresses of a range. • `source-port` — Single port number or start and end port numbers of a range. • `dest-ip` — Single IP address or specify start and end addresses of a range. • `dest-port` — Single port number or start and end port numbers of a range. • `vs-name` — Specify a space-separated list of up to 8 virtual server configuration names.
`list`	List matching sessions.
`clear`	Clear the list of matching sessions. Clear options include: • `l4` — l4 (lowercase "L" four) is the abbreviated form of Layer 4. Use the l4 option to clear the Layer 4 persistence table.

Example

```
FortiADC-VM # diagnose server-load-balance persistence filter 'source-ip 10.1.1.1
     10.1.1.100'
FortiADC-VM # diagnose server-load-balance persistence filter 'vs-name vs1 vs2'
FortiADC-VM # diagnose server-load-balance persistence filter show
filter=[flag:1000
source ip range: :: - :: port range: 0 - 0
dest ip range: :: - :: port range: 0 - 0
virtual server: vs1 vs2 ]
FortiADC-VM # diagnose server-load-balance persistence list
client-ip/port virtual-server-ip/port local-ip/port real-server-ip/port protocol service
     state in-bytes out-bytes expire virtual-server-name real-server-name
FortiADC-VM #
```

diagnose server-load-balance session

Use this command to filter and display the session table (current sessions).

Syntax

```
diagnose server-load-balance session filter {'<option>'|show|clear}
diagnose server-load-balance session list
diagnose server-load-balance session clear [l4]
```

`filter`	Create, show, or clear session list filters. Use multiple commands to add filters to the filter list. For example, one command to add a source-ip filter and another to add a vs-name filter. Put the filter expression in single quotes. Filter options include: • `source-ip`—Single IP address or specify start and end addresses of a range. • `source-port`—Single port number or start and end port numbers of a range. • `dest-ip`—Single IP address or specify start and end addresses of a range. • `dest-port`—Single port number or start and end port numbers of a range. • `trans-source-ip`—Single IP address or specify start and end addresses of a range. • `trans-source-port`—Single port number or start and end port numbers of a range. • `trans-dest-ip`—Single IP address or specify start and end addresses of a range. • `trans-dest-port`—Single port number or start and end port numbers of a range. • `type`—Specify `ipv4`, `ipv6`, `ipv4v6`, or `ipv6v4`. • `protocol`—Specify `tcp` or `udp`. • `vs-name`—Specify a space-separated list of up to 8 virtual server configuration names. • `rs-name`—Specify a space-separated list of up to 8 real server configuration names.
`list`	List matching sessions.
`clear`	Clear the list of matching sessions. Clear options include: • `l4` — l4 (lowercase "L" four) is the abbreviated form of Layer 4. Use the l4 option to clear the Layer 4 session table.

Example

```
FortiADC-VM # diagnose server-load-balance session filter 'source-ip 10.1.1.1 10.1.1.100'
FortiADC-VM # diagnose server-load-balance session filter 'vs-name vs1 vs2'
FortiADC-VM # diagnose server-load-balance session filter show
filter=[flag:1000 type:0 protocol:0 service:0
source ip range: :: - :: port range: 0 - 0
dest ip range: :: - :: port range: 0 - 0
trans source ip range: :: - :: port range: 0 - 0
trans dest ip range: :: - :: port range: 0 - 0
virtual server: vs1 vs2
real server:]
FortiADC-VM # diagnose server-load-balance session list
client-ip/port virtual-server-ip/port local-ip/port real-server-ip/port protocol service
     state in-bytes out-bytes expire virtual-server-name real-server-name
FortiADC-VM #
```

diagnose server-load-balance slb_load

Use this command to display the vdoms and relevant information.

Syntax

```
diagnose server-load-balance slb_load
```

Example

```
FortiADC-VM (root) # diagnose server-load-balance slb_load
<Enter>

FortiADC-VM (root) # diagnose server-load-balance slb_load
vdom name is root
     virtual-server L4VS
          real-server 1
          id:1, status 1, weight 0
     virtual-server L7VS
          real-server 2
          id:1, status 1, weight 0
     virtual-server 1
          real-server 1
          id:1, status 1, weight 0
vdom name is vd1
     virtual-server VS
          real-server self
          id:1, status 0, weight 0
     virtual-server vs2
          real-server self
          id:1, status 0, weight 0
```

diagnose sniffer packet

Use this command to perform a packet trace on one or more network interfaces.

Packet capture, also known as sniffing or packet analysis, records some or all of the packets seen by a network interface (that is, the network interface is used in promiscuous mode). By recording packets, you can trace connection states to the exact point at which they fail, which may help you to diagnose some types of problems that are otherwise difficult to detect.

FortiADC appliances have a built-in sniffer. Packet capture on FortiADC appliances is similar to that of FortiGate appliances. Packet capture output appears on your CLI display until you stop it by pressing Ctrl+C, or until it reaches the number of packets that you have specified to capture.

Packet capture can be very resource intensive. To minimize the performance impact on your FortiADC appliance, use packet capture only during periods of minimal traffic, with a local console CLI connection rather than a Telnet or SSH CLI connection, and be sure to stop the command when you are finished.

For additional information on the packet sniffer utility, see the Fortinet Knowledge Base article Using the FortiOS built-in packet sniffer.

Syntax

```
diagnose sniffer packet [{any | <interface_name>} [{none | '<filter_str>'} [{1 | 2 | 3}
     [<packets_int>]]]]
```

`{any | <interface_name>}`	Type the name of a network interface whose packets you want to capture, such as `port1`, or type `any` to capture packets on all network interfaces. If you omit this and the following parameters for the command, the command captures all packets on all network interfaces.

`{none | '<filter_str>'}`	Type either `none` to capture all packets, or type a filter that specifies which protocols and port numbers that you do or do not want to capture, such as `'tcp port 25'`. Surround the filter string in quotes (`'`).
	Filters use tcpdump syntax:
	`'[[src|dst] host {<host1_fqdn> | <host1_ipv4>}] [and|or]`
	`[[src|dst] host {<host2_fqdn> | <host2_ipv4>}] [and|or]`
	`[[arp|ip|gre|esp|udp|tcp] port <port1_int>] [and|or]`
	`[[arp|ip|gre|esp|udp|tcp] port <port2_int>]'`
	To display only the traffic between two hosts, specify the IP addresses of both hosts. To display only forward or reply packets, indicate which host is the source, and which is the destination.
	For example, to display UDP port 1812 traffic between 1.example.com and either 2.example.com or 3.example.com, you would enter:
	`'udp and port 1812 and src host 1.example.com and dst \`
	`( 2.example.com or 2.example.com \)'`

`{1 | 2 | 3}`	Type one of the following integers indicating the depth of packet headers and payloads to capture: `1` — Display the packet capture timestamp, plus basic fields of the IP header: the source IP address, the destination IP address, protocol name, and destination port number. Does *not* display all fields of the IP header; it omits: • IP version number bits • Internet header length (`ihl`) • type of service/differentiated services code point (`tos`) • explicit congestion notification • total packet or fragment length • packet ID • IP header checksum • time to live (`TTL`)

	• IP flag • fragment offset • options bits 2 — All of the output from 1, plus the packet payload in both hexadecimal and ASCII. 3 — All of the output from 2, plus the the link layer (Ethernet) header. For troubleshooting purposes, Fortinet Technical Support may request the most verbose level (3).
<packets_int>	Type the number of packets to capture before stopping. If you do not specify a number, the command will continue to capture packets until you press Ctrl+C.

Example

The following example captures three packets of traffic from any port number or protocol and between any source and destination (a filter of none), which passes through the network interface named port1. The capture uses a low level of verbosity (indicated by 1).

```
FortiADC-VM # diagnose sniffer packet port1 none 1 3
interfaces=[port1]
filters=[none]
0.000000 172.30.144.20.53800 -> 172.30.144.100.22: ack 202368347
0.000000 172.30.144.100.22 -> 172.30.144.20.53800: psh 202368415 ack 2508304372
0.000000 172.30.144.100.22 -> 172.30.144.20.53800: psh 202368531 ack 2508304372
```

If you are familiar with the TCP protocol, you might notice that the packets are from the middle of a TCP connection. Because port 22 is used (highlighted above in bold), which is the standard port number for SSH, the packets might be from an SSH session.

Example

The following example captures packets traffic on TCP port 80 (typically HTTP) between two hosts, 192.168.0.1 and 192.168.0.2. The capture uses a low level of verbosity (indicated by 1). Because the filter does not specify either host as the source or destination in the IP header (src or dst), the sniffer captures both forward and reply traffic.

```
FortiADC# diagnose sniffer packet port1 'host 192.168.0.2 or host 192.168.0.1 and tcp port
     80' 1
```

A specific number of packets to capture is not specified. As a result, the packet capture continues until the administrator presses Ctrl+C. The sniffer then confirms that five packets were seen by that network interface. Below is a sample output.

```
192.168.0.2.3625 -> 192.168.0.1.80: syn 2057246590
192.168.0.1.80 -> 192.168.0.2.3625: syn 3291168205 ack 2057246591
192.168.0.2.3625 -> 192.168.0.1.80: ack 3291168206
192.168.0.2.3625 -> 192.168.0.1.80: psh 2057246591 ack 3291168206
192.168.0.1.80 -> 192.168.0.2.3625: ack 2057247265
5 packets received by filter
0 packets dropped by kernel
```

diagnose system top

Use this command to view a list of the most system-intensive processes and to change the refresh rate.

Syntax

```
diagnose system top [delay <integer>]
```

`delay`	Refresh interval (seconds).

Once you execute this command, it continues to run and display in the CLI window until you enter `q` (quit).

While the command is running, you can press `Shift + P` to sort the five columns of data by CPU usage (the default) or `Shift + M` to sort by memory usage.

Example

This example displays a list of the top system processes and sets the update interval at 10 seconds.

```
FortiADC-VM # diagnose system top ?
delay refresh display period

FortiADC-VM # diagnose system top delay ?
<delay> delay in seconds

FortiADC-VM # diagnose system top delay 30
Run Time: 13 days, 5 hours and 9 minutes
0U, 1S, 97I; 1620T, 613F
php-fpm 635 S 1.9 0.7
php-fpm 636 S 1.9 0.7
mysqld 528 S 0.0 5.8
named 1238 S 0.0 1.0
alertemail 522 S 0.0 0.9
php-fpm 13467 S 0.0 0.6
php-fpm 525 S 0.0 0.6
cmdbsvr 86 S 0.0 0.6
cli 13065 S 0.0 0.5
snmpd 536 S 0.0 0.2
miglogd 523 S 0.0 0.2
nginx 524 S 0.0 0.2
updated 512 S 0.0 0.2
cli 21276 R 0.0 0.2
flg_indexd 10367 S 0.0 0.2
lb 520 S 0.0 0.2
sshd 515 S 0.0 0.2
scheduled 506 S 0.0 0.2
info_cente 533 S 0.0 0.2
crlupdated 535 S 0.0 0.2
hasyncd 518 S 0.0 0.2
flg_access 10370 S 0.0 0.2
llbd 507 S 0.0 0.1
netd 511 S 0.0 0.1
```

```
lvs 517 S 0.0 0.1
gdns 516 S 0.0 0.1
llbr_hcd 509 S 0.0 0.1
keepalived 519 S 0.0 0.1
getty 513 S 0.0 0.1
```

The first line indicates the up time. The second line lists the processor and memory usage, where the parameters from left to right mean:

- U — Percent of user CPU usage (in this case 0%)
- S — Percent of system CPU usage (in this case 1%)
- I — Percentage of CPU idle (in this case 97%)
- T — Total memory in kilobytes (in this case 1620 KB)
- F — Available memory in kilobytes (in this case 613 KB)

The five columns of data provide the process name (such as `updated`), the process ID (`pid`), the running status, the CPU usage, and the memory usage. The status values are:

- S — Sleeping (idle)
- R — Running
- Z — Zombie (crashed)
- < — High priority
- N — Low priority

diagnose system vm

Use this command to view information about a virtual appliance.

Syntax

```
diagnose system vm
```

Example

```
FortiADC-VM # diagnose system vm
UUID: 564d2ec7705469089699f1852ce8a086
File: License file and resources are valid.
Resources: 1 CPU/1 allowed, 1620 MB RAM/2048 MB allowed, 23 GB Disk/1024 GB allowed
Registered: 1 (True)
Status: 1 (Valid: License has been successfully authenticated with registration servers.)
FDS code: 200
Warn count: 0
Copy count: 0
Received: 113788700
Warning: 0
Recv: 201503092104
Dup:
```

diagnose system threat-analytics info

Use this command to view the system information for Threat Analytics.

Syntax

```
diagnose system threat-analytics info
```

Example

```
FortiADC-VM # diagnose system threat-analytics info
User ID: 1168325

WS Connection: Connected
Log_forward: disable
License status: Trial started
Trial ends in 14 days
```

diagnose tech-report

Use this command to run a batch of commands that Fortinet support can use to troubleshoot issues you have reported. You might be directed to copy and paste the screen output into an email or email attachment.

Syntax

```
diagnose tech-report
```

Example

```
FortiADC-VM # diagnose tech-report
```

diagnose waf api-security memory

Use this command to view the current memory consumption from WAF API Security processes in the VDOM. Each VDOM is allocated a maximum of 32 MB to be used by API Security.

Syntax

```
diagnose waf api-security memory
```

Example

```
FADC (root) # diagnose waf api-security memory
Total: 32M
Block count: 99864
Block size: 336
Used: 0.02K
Free: 32767.98K
```

execute

The `execute` commands have an immediate and decisive effect on your FortiADC appliance and, for that reason, should be used with care. Unlike `config` commands, most `execute` commands do not result in any configuration change.

This chapter is a reference for the following commands:

execute autoupdate

Use this command to display information for connecting FortiADC to FortiGuard for the purpose of updating the FortiGuard license or services.

Syntax

```
execute autoupdate {connection <ip>| contract | log | registration | servers}
```

`connection <ip>`	Checks the connection with a specific FortiGuard server.
`contract`	Displays the FortiGuard contract information.
`log`	Displays the debugging log.

registration	Displays the FortiGuard registration information.
servers	Displays the FortiGuard servers that FortiADC can connect to.

execute caching

Use this command to show information about a virtual server cache or to clear the cache.

Syntax

```
execute caching {show|clean} <vsname>
```

show	Show cache statistics.
clean	Clear the cache.
<vsname>	Name of the virtual server.

Example

```
FortiADC-VM # execute caching ?
show show
clean clean
FortiADC-VM # execute caching show vs1
Warning: ram caching is not enabled on vs1
```

execute certificate ca

Use this command to import or export a certficate file. This command will create ca configuration automatically. Please see details in config system certificate ca.

Syntax

```
execute certificate ca import tftp <filename> <ip>
execute certificate ca export tftp <cert> <filename> <ip>
```

<cert>	Local (FortiADC) certificate name.
<filename>	Name of the certificate file.
<ip>	IP address of the TFTP server.

Example

```
FortiADC-VM # execute certificate ca import tftp ca.crt 192.168.1.23
Done.

FortiADC-VM # execute certificate ca export tftp ca ca-export.crt 192.168.1.23
#
Done.
```

execute certificate crl

Use this command to import or export a certficate file. This command will create ca configuration automatically. Please see details in config system certificate crl.

Syntax

```
execute certificate crl import tftp <filename> <ip>
```

<filename>	Name of the certificate file.
<ip>	IP address of the TFTP server.

Example

```
FortiADC-VM # execute certificate crl import tftp crl.r0 192.168.1.23

Done.
```

execute certificate local

Use this command to import/export a certficate file or to generate/regenerate a CSR file. When you generate a CSR, you can create an RSA or ECDSA private key. This command will create ca configuration automatically. Please see details in config system certificate local.

Note: Importing a local certificate with pfx format is not supported, unless you have first used FortiADC to generate the CSR.

Syntax

```
execute certificate local import tftp <filename> <ip>
execute certificate local export tftp <cert> <filename> <ip>
```

```
execute certificate local generate <cert_name> <keytype> {<curve_name>|<keysize>} <subject>
     <country> <state> <city> <org> <unit> <email>
execute certificate local regenerate
```

<cert>	Local (FortiADC) certificate name.
<filename>	Name of the certificate file.
<ip>	IP address of the TFTP server.

Example

```
FortiADC-VM # execute certificate local import tftp fortiadc.crt 192.168.1.23

FortiADC-VM # execute certificate local export tftp Factory fortiadc.crt 192.168.1.23
#
Done.

FortiADC-VM # execute certificate local generate csr-test ECDSA secp521r1 example null ca
     sunnyvale fortinet fadc root
Generating a 512 bit ECDSA private key with curve name secp521r1 and message digest
     algorithm SHA-512
Generating X.509 certificate request
Done.

FortiADC-VM # execute certificate local regenerate
self certificate regenerated!
```

execute certificate local import automated

Use this command to import local certificates using the ACME protocol to get SSL/TLS certificates from Let's Encrypt or other ACME providers.

As part of the certificate importing functionality, FortiADC supports the Automatic Certificate Management Environment (ACME) protocol for automating the interactions between certificate authorities (CAs) and their users' web servers.

Certificates imported through Let's Encrypt have a ninety-day lifetime (which may differ from other ACME providers). These certificates must be renewed prior to expiration. FortiADC supports the TLS-ALPN-01 and DNS-01 challenge types. The TLS-ALPN-01 challenge supports automatic certificate renewal. The DNS-01 challenge requires manual certificate renewal, however, only the DNS-01 challenge can issue certificates containing wildcard domain names.

Before you begin:

- You must have Read-Write permission for System settings.

Syntax

```
execute certificate local import automated <cert_name> <domain> <email> <key_type>
     {RSA|ECDSA} <key_size> {<key_size>|<curve_name>} <password> <server_url> <ca_group>
```

```
<challenge_type> {tls-alpn-01|dns-01} {<renew_win>|<challenge_wait>}
```

Execute Parameter	Description
<cert_name>	Specify the certificate name that can be referenced by other parts of the configuration, such as www_example_com. The maximum length is 35 characters. Do not use spaces or special characters. **Note**: If the **challenge_type** is **tls-alpn-01**, the **cert_name** must match the name of the "placeholder" certificate that is linked to the HTTPS virtual server. For details, see Fulfilling the ACME TLS-ALPN-01 challenge on page 560.
<domain>	Specify the web server domain to be protected by the certificate. **Note**: If the **challenge_type** is **tls-alpn-01**, the **domain** must be from the HTTPS virtual server that is linked to the "placeholder" certificate. For details, see Fulfilling the ACME TLS-ALPN-01 challenge on page 560.
<email>	Enter the email address that will receive notifications regarding the status of the certificate. Depending on which ACME service provider you use, you may receive notification for when the certificate request has been approved through the Certificated Services or when the certificate is due to expire.
<key_type>	Select either of the following key types: • RSA • ECDSA **Note**: If the **challenge_type** is **tls-alpn-01**, the **key_type** must match the key type of the "placeholder" certificate that is linked to the HTTPS virtual server. For details, see Fulfilling the ACME TLS-ALPN-01 challenge on page 560.
<key_size>	Specify the **key_size** if the **key_type** is **RSA**. Select one of the following key sizes: • 2048 • 3072 • 4096
<curve_name>	Specify the **curve_name** if the **key_type** is **ECDSA**. Select one of the following curve names: • P256 • P384 • P521
<password>	Specify the password to decrypt the file. If the file was encrypted by a password when generated, the same password must be provided when the file is imported to FortiADC. If the file was generated without a password, there is no need to specify a password when importing the file to FortiADC. Enter null if there is no password.
<server_url>	To use Let's Encrypt as the ACME provider, enter null as the server_url. To use other ACME providers, such as Buypass AS, specify the URL of the ACME server. The ACME request URL must begin with "https://".

Execute Parameter	Description
	After you have obtained the ACME certificate from your chosen ACME service provider, you will need to provide the ACME server URL to connect to FortiADC. This will enable FortiADC to act as the ACME client to send the ACME request and receive the ACME certificate/key. **Note**: The ACME server URL is unique to the ACME service provider. Please refer to the documentation from your ACME provider for further information.
`<ca_group>`	Specify the name of the CA Group. FortiADC will use the CA certificate in the CA Group to verify the certificate sent by the ACME provider. Enter `null` to not verify.
`<challenge_type>`	The ACME server requires validation that you control the domain names in the certificate using "challenges" as defined by the ACME standard. FortiADC supports the TLS-ALPN-01 and DNS-01 challenge types. Select either of the following challenge types: • tls-alpn-01 — The TLS-ALPN-01 supports automatic certificate renewal. However, this method cannot be used to validate wildcard domains. To use this challenge type, you will need to make preparations to fulfill the challenge **before** completing the certificate import configurations (for details, see Fulfilling the ACME TLS-ALPN-01 challenge on page 560). • dns-01 — The DNS-01 challenge can be used to issue certificates containing wildcard domain names. To use this challenge type, you will need to take steps to fulfill the challenge after completing the certificate import configurations (for details, see Fulfilling the ACME DNS-01 challenge on page 559). Certificates imported using the DNS-01 challenge need to be manually renewed.
`<renew_win>`	Specify the **renew_win** if the **challenge_type** is **tls-alpn-01**. Specify a renew window (in minutes) to automatically renew the certificate before it expires. (Range: 0-43200 minutes). Setting the renew window to 0 will disable the automatic certificate renewal.
`<challenge_wait>`	Specify the **challenge_wait** if the **challenge_type** is **dns-01**. Specify the ACME DNS-01 challenge wait time in minutes. (Range: 1-1440 minutes). The ACME DNS-01 challenge wait time refers to the amount of time you will have to fulfill the DNS-01 challenge. A longer challenge wait time is recommended to ensure enough time is allotted to perform the required Public DNS configuration changes and for the changes to take effect. For more information, see Fulfilling the ACME DNS-01 challenge on page 559.

Example

```
FortiADC # execute certificate local import automated ACME-test test.com test@fortinet.com
     RSA 2048 null null null dns-01 3
Done.

FortiADC # execute certificate local import automated ACME-test test.com test@fortinet.com
     ECDSA P521 null null null tls-alpn-01 15
Done.
```

Fulfilling the ACME DNS-01 challenge

The DNS-01 challenge asks you to prove that you control the DNS for your domain name by putting a specific value in a TXT record under that domain name.

After you have executed the CLI command to import your automated local certificate, the ACME DNS challenge information is generated. With this information, you will configure your Public DNS Service to create the TXT record.

Certificates generated by the ACME DNS-01 challenge cannot be renewed automatically. Please manually renew the certificate before it expires.

To add the record the DNS challenge information to the Public DNS Service:

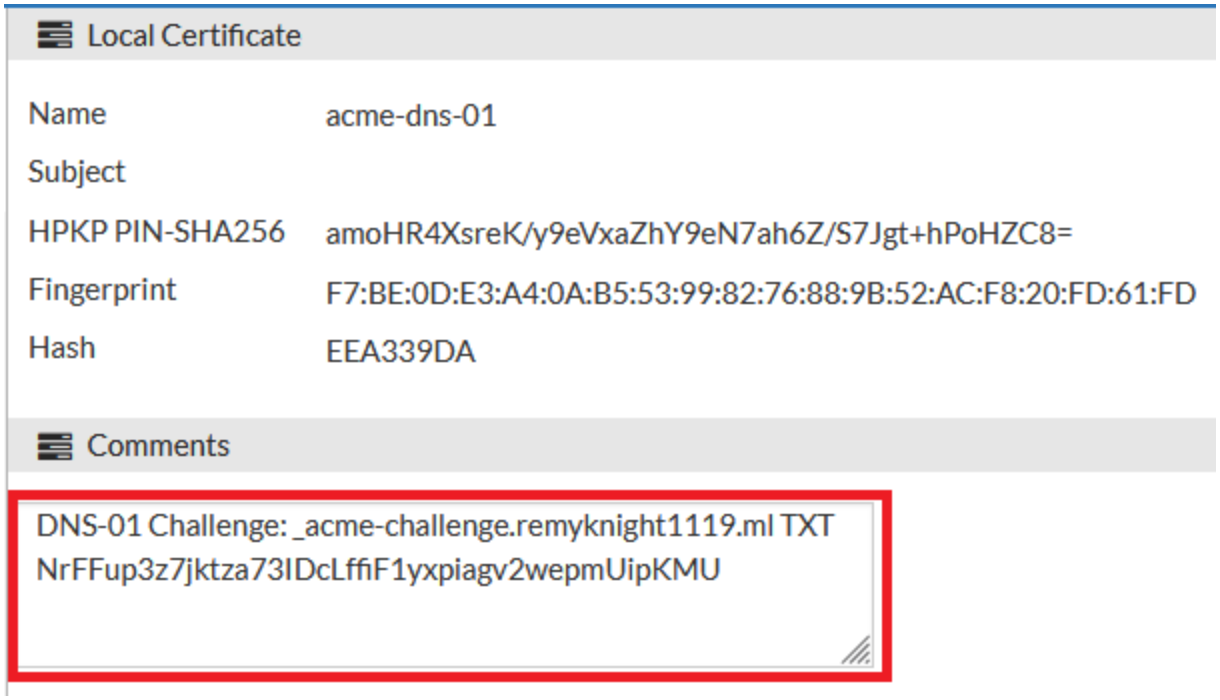
1. Obtain the ACME DNS challenge information using either of the following methods.
   - After you have executed the CLI command to import your automated local certificate, you will be shown the challenge information. Save this information for use later.

```
FortiADC # execute certificate local import automated acme-dns-01-2 remyknight1119.ml a@gmail.com RSA 2048 null null null 22
Done.

FortiADC # Domain:         remyknight1119.ml
Record:         _acme-challenge.remyknight1119.ml
Type:           TXT
Content:        "o_uQuc1Hb_L3zrm-eoGXcOHEk2KOrUyaiiTcE8yluIc"
Note:           Some DNS managers add quotes automatically, A single set is needed
```

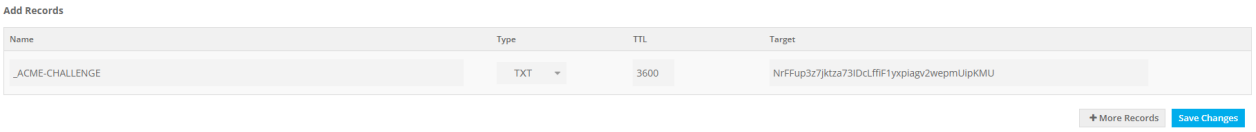
   - If you missed the above information in the CLI, then you can view the information in the GUI.

     In the Local Certificate page, locate the local certificate record and click the (View icon) to see the details.



2. Login to your DNS service provider and go to your DNS Domain management page.

3. Add a record and input the challenge information into the corresponding fields.



Name	`_ACME-CHALLENGE` is a fixed value.
Type	Set the record type as `TXT`.
TTL	Set this to the default value.
Target	Paste the content from your ACME DNS-01 challenge information.

4. Save the changes.
   The DNS configuration changes may take several minutes to take effect.

The ACME provider will then query the DNS system for that record to find a match. If there is a match, the ACME certificate passes validation (certificate status will progress from Pending → OK). However, if the record is not found within the specified challenge wait time then the certificate validation fails (certificate status is Fail).

If the certificate validation fails, then you will need to delete the record and import a new automated local certificate to try again.

It is recommended to set a longer challenge wait time to allow enough time for the DNS configuration changes to take effect. If the DNS configuration changes has not taken effect at the time the ACME provider queries the DNS system for the TXT record, then the validation will fail. Various factors may influence the speed of the DNS (such as the DNS service provider, network speed, network traffic), so the DNS configuration changes may take as long as 20 minutes to take effect.

Fulfilling the ACME TLS-ALPN-01 challenge

In FortiADC, to fulfill the TLS-ALPN-01 challenge, the ACME server validates control of the domain name by connecting to the Virtual Server at one of the addresses resolved for the domain name. This is achieved by linking a certificate to an HTTPS virtual server to allow the ACME server resolving domain to point to its IP. Then FortiADC generates a temporary certificate to fulfill the validation.

Before configuring an automated certificate using the TLS-ALPN-01 challenge, you must set up the following:

- A valid local certificate that functions as a placeholder
- An HTTPS virtual server to link the placeholder certificate

Once the placeholder certificate has been linked to the HTTPS virtual server, you will then use the placeholder certificate name and the domain name from the virtual server to import the automated certificate using the TLS-ALPN-01 challenge. This certificate then replaces the placeholder certificate so that it will be linked to the HTTPS virtual server to fulfill the TLS-ALPN-01 challenge.

To prepare the placeholder certificate and HTTPS virtual server for the ACME TLS-ALPN-01 challenge:

1. Generate or import a local certificate. This certificate must be valid (Status is OK). Ensure the Key Type of this placeholder certificate matches the automated certificate you intend to import. For example, if the placeholder certificate is RSA, then the automated certificate you will be importing must also be RSA. Record the certificate name for use in later steps. For details, see execute certificate local on page 555 or execute certificate local import

automated on page 556.
Note: If importing a local certificate, you should only import the following certificate types: Certificate, PKCS12 Certificate and Local CSR Certificate. As the placeholder certificate must be valid, it is not recommended to use an Automated certificate type for this purpose since this type of certificate cannot be valid until the ACME challenge is fulfilled.

2. Create a local certificate group and add the placeholder certificate you have created previously under this certificate group. Specify the placeholder certificate as the local certificate configuration. Record the certificate group name for use in later steps. For details, see config system certificate local_cert_group on page 420.
3. Create a Client SSL profile and add the certificate group you have created previously as the local certificate group. Record the Client SSL profile name for use in later steps. For details, see config load-balance client-ssl-profile on page 133.
4. Create an HTTPS virtual server. Apply the Client SSL profile you have created previously. For details, see config load-balance virtual-server on page 230.
The Address of this HTTPS virtual server must be associated to a domain to ensure it can be reached by the ACME provider. It is recommended that this domain be registered at a DNS service provider so you can set the domain to point to a specific IP address. Record the domain for use in later steps.
5. Import the automated certificate using the TLS-ALPN-01 challenge type.
Input the information for the following settings according to the guidelines below. For detailed steps, see execute certificate local import automated on page 556.

Setting	Guideline
<cert_name>	The name must match the name of the placeholder certificate. Once this automated certificate configuration is completed, it will replace the placeholder certificate.
<domain>	Input the domain of the HTTPS virtual server that has been linked to the placeholder certificate. The ACME provider will reach this domain that points to the HTTPS virtual server IP address.
<key_type>	The Key Type must match the placeholder certificate.

execute certificate remote

Use this command to import or export a remote certfication file. This command will create ca configuration automatically. Please see details in config system certificate remote.

Syntax

```
execute certificate remote import tftp <filename> <ip>
execute certificate remote export tftp <cert> <filename> <ip>
```

{import|export}	Whether to import or export the file.
<cert>	Local (FortiADC) certificate name.

<filename>	Name of the certificate file.
<ip>	IP address of the TFTP server.

Example

```
FortiADC-VM # execute certificate remote import tftp ca.crt 192.168.1.23

Done.

FortiADC-VM # execute certificate remote export tftp ca remote.crt 192.168.1.23

Done.
```

execute certificate config

Use this command to verify the certficate file is a supported type.

Syntax

```
execute certificate config verify
```

Example

```
FortiADC-VM # execute certificate config verify
```

execute checklogdisk

Use this command to run diagnostics on the hard disk. If the command reports issues, you can run execute fixlogdisk to resolve them.

Note: The command name is a misnomer. The pair of commands troubleshoots all hard disk issues, not just issues relating to the log partition.

Syntax

```
execute checklogdisk
```

Example

```
FortiADC-docs # execute checklogdisk
This operation will temporarily pause the system, check and autofix log disk!
Do you want to continue? (y/n)y
System is checking ...
```

execute clean

Use this command to restore the factory default ISP address book definitions. In systems with multiple VDOMs, the command applies to the current VDOM only.

Syntax

```
execute clean isp-address
```

Example

```
FortiADC-VM # execute clean isp-address
This operation will clean the current restored ISP address-books and related ISP/proximity
     routes!
Do you want to continue? (y/n)y
```

execute config-sync

Execute a configuration from config sync-list.

Syntax

```
execute config sync {get|put} <ip> <port> <password>
```

See also

- config config sync-list on page 66

execute date

Use this command to display or set the system date and time.

Syntax

```
execute date [<mm/dd/yyyy> [hh:mm:ss]] <Enter>
```

<Enter>	If you do not specify a date, the command returns the current system date.
<mm/dd/yyyy>	Current date where the FortiADC appliance is located. MM/DD/YY format.
[hh:mm:ss]	HH:MM:SS format.

Example

```
FortiADC-VM # execute date ?
date <mm/dd/yyyy> [hh:mm:ss]
<mm/dd/yyyy> mm/dd/yyyy, mm: 1-12, dd: 1-31, yyyy: 2001-2100

FortiADC-VM # execute date
Tue Mar 10 10:00:47 PDT 2015

FortiADC-VM # execute date 03/10/2015
send buff to ha. pid=31876, buff=
exec date
end
```

execute discovery-glb-virtual-server

Use this command to populate the global load balancing server configuration virtual server list for the specified virtual server.

Syntax

```
execute discovery-glb-virtual-server {server|override-server} <servername>
```

{server|override-server} <servername>	Use `server <servername>` to populate the virtual server member list with virtual servers from the local FortiADC configuration. <servername> is the name of the global-load-balance servers configuration. After the list had been populated, you can edit the configuration to add a gateway health check. Use `override-server <servername>` to discover the virtual server configuration and overwrite any local configuration information for those servers.

Example

```
FortiADC-VM # execute glb-dprox-lookup 172.30.144.100
Searching Address 172.30.144.100
```

```
get error sendmsg = Connection refused
Matched nothing!
```

execute dumpsystem

Use this command to generate a system dump file. System dump files can help Fortinet support engineers analyze an issue for you.

Syntax

```
execute dumpsystem [console <enable|disable>]
```

`[console <enable|disable>]`	Enable/disable writing debug information to the console during the dump.

Example

```
FortiADC-VM # execute dumpsystem console ?
enable debug info will output to console
disable debug info will not output to console
FortiADC-VM # execute dumpsystem console enable
FortiADC-VM # execute dumpsystem
This operation will reboot the system!
Do you want to continue? (y/n)y
Begins to dump userspace information
Failed to open /proc/1185/comm, No such file or directory
Failed to open /proc/1186/comm, No such file or directory
Failed to open /proc/1187/comm, No such file or directory
Failed to open /proc/1188/comm, No such file or directory
Failed to open /proc/1189/comm, No such file or directory
Failed to open /proc/1190/comm, No such file or directory
Failed to open /proc/1191/comm, No such file or directory
Failed to open /proc/1192/comm, No such file or directory
Failed to open /proc/1193/comm, No such file or directory
Failed to open /proc/1194/comm, No such file or directory
Begins to dump kernel information
```

See also

- execute dumpsystem-file

execute dumpsystem-file

You use this command to manage system dump files. System dump files can help Fortinet support engineers analyze an issue for you.

Syntax

```
execute dumpsystem-file {delete <filename>|list|upload {ftp|tftp} <filename> <ip>}
```

`delete <filename>`	Delete the specified file.
`list`	List all system dump files.
`upload {ftp|tftp} <filename> <ip>`	Upload the specified file to the specified TFTP server.

Example

```
FortiADC-VM # execute dumpsystem-file list
-rw------- 1 0 0 96719189 Mar 15 13:35 coredump-2016-03-15-13_35
-rw-r--r-- 1 0 0 16654391 Mar 15 13:34 user_coredump_2016_03_15_13_34_46.tar.bz2
FortiADC-VM # execute dumpsystem-file upload tftp coredump-2016-03-15-13_35 172.30.184.77
coredump-2016-03-15- 7% |**  | 7152k 0:09:58 ETA
```

See also

- execute dumpsystem

execute factoryreset

Use this command to reset the system to its default settings for the currently installed firmware version. If you have not upgraded or downgraded the firmware, this restores factory default settings.

Back up your configuration first. This command resets all changes that you have made to the configuration file and reverts the system to the default values for the firmware version. Depending on the firmware version, this could include factory default settings for the IP addresses of network interfaces.

Syntax

```
execute factoryreset
```

Example

```
FortiADC-VM # execute factoryreset
This operation will change all settings to factory defaults!
Do you want to continue? (y/n)y

System is resetting to factory defaults...
```

execute factoryreset2

Use this command to partially reset the device so that all settings are returned to factory default except for the VDOM, interface, and static route settings.

Syntax

```
execute factoryreset2
```

Example

```
FortiADC-VM # execute factoryreset2
This operation will change all settings to factory defaults, except system
     global/VDOMs/system interface/router static!
Do you want to continue? (y/n)y

System is resetting to factory defaults...
```

execute fixlogdisk

Use this command to fix hard disk issues reported by the execute checklogdisk command.

Note: The command name is a misnomer. The pair of commands troubleshoots all hard disk issues, not just issues relating to the log partition.

Syntax

```
execute fixlogdisk
```

Example

```
FortiADC-docs # execute fixlogdisk
This operation will temporarily pause the system, check and fix the log disk!
Do you want to continue? (y/n)
```

execute formatlogdisk

Use this command to clear the logs from the hard disk and reformat the disk.

This operation deletes all locally stored log files.

Syntax

```
execute formatlogdisk
```

Example

```
FortiADC-VM # execute formatlogdisk
This operation will erase all data on the log disk!
Do you want to continue? (y/n)
```

execute geolookup

Use this command to look up the country for the specified IP address.

Syntax

execute geolookup <ip>

<ip>	IP address to look up.

Example

execute geolookup 8.8.8.8

8.8.8.8 "United States"

execute glb-dprox-lookup

Use this command to query the dynamic proximity RTT record for the specified IP address.

Syntax

```
execute glb-dprox-lookup <class_ip>
```

<ip>	Lookup the specified IPv4 or IPv6 address.

Example

```
FortiADC-docs # execute glb-dprox-lookup 192.168.0.1
Searching Address 192.168.0.1
get error sendmsg = Connection refused
Matched nothing!
FortiADC-docs #
```

execute glb-persistence-lookup

Use this command to query the GSLB persistence table to see if an IP address has an entry in it.

Syntax

```
execute glb-persistence-lookup <classip>
```

<classip>	IP address you want to look up.

Example

```
FortiADC-VM # execute glb-dprox-lookup 172.30.144.100
Searching Address 172.30.144.100
get error sendmsg = Connection refused
Matched nothing!
FortiADC-VM #
```

execute ha force fetch-peers-info

Use this command to fetch the HA peer information for debugging.

Syntax

```
execute ha force fetch-peers-info
```

Example

```
FortiADC # execute ha force fetch-peers-info
This operation will fetch peers info !
Do you want to continue? (y/n)
```

execute ha force standby

Use this command to force the current HA node into standby status for the specified traffic-group. This is only supported in HA VRRP mode.

Syntax

```
execute ha force standby traffic-group <traffic-group name>
```

`<traffic-group name>`	Specify the traffic group name.

execute ha force sync-config

Use this command to manually sync the configuration from the primary to secondary nodes.

Syntax

```
execute ha force sync-config
```

Example

```
(M) FortiADC-VM # execute ha force sync-config
```

```
This operation will overwrite secondary's config!
Do you want to continue? (y/n)y
(M) FortiADC-VM #
```

execute ha force transfer-file

Use this command to transfer files between HA nodes.

The files to be transferred must be in “/var/log/crash/”.

Use the command `diag crashlog list` to get the name of the files stored in “/var/log/crash/”.

Syntax

```
execute ha force transfer-file <file-name> <node-id>
```

`<file-name>`	The name of the file to be transferred. The length of the absolute file name (file name plus “/var/log/crash/”) can’t be more than 64-byte characters.
`<node-id>`	The node ID of the HA member.

execute ha manage

Use this command to telnet to the command-line interface of a peer HA cluster node. This is useful when you want to configure node-specific settings, like HA priority. Most settings are pushed from the primary node to member nodes.

Syntax

```
execute ha manage <index>
```

Example

```
FortiADC-VM # execute ha manage 0
```

Note: `<index>` represents an individual ADC member that has already joined the HA cluster. The index number starts from 0.

You can check your index number using the CLI command: `execute ha manage ?`

For example:

```
FortiADC-VM # execute ha manage ?
           <0> FADV020000190xxx
```

execute hardware-ssl list-ciphers

Use this command to print the supported hardware SSL cipher suite.

This command is only available if the FortiADC hardware version supports hardware SSL.

Hardware SSL is supported in FortiADC 400F, 1200F, 2200F, 4200F and 5000F.

Syntax

```
execute hardware-ssl list-ciphers
```

Example

```
FortiADC-400F # execute hardware-ssl list-ciphers
["ECDHE-ECDSA-AES256-SHA","ECDHE-ECDSA-AES128-SHA256","ECDHE-ECDSA-AES128-SHA","ECDHE-RSA-
     AES256-SHA","DHE-RSA-AES256-SHA256","DHE-RSA-AES256-SHA","AES256-SHA256","AES256-
     SHA","ECDHE-RSA-AES128-SHA256","ECDHE-RSA-AES128-SHA","DHE-RSA-AES128-SHA256","DHE-
     RSA-AES128-SHA","AES128-SHA256","AES128-SHA"]
```

execute health-check-verify

Use this command to use the specified health check to check the status of any IP address.

Syntax

execute health-check-verify <ip address> <hc name> count <int> interval <int> timeout <int> <port>|<enter>

<ip address>	The IP address of the health check object.
<hc name>	The name of the health check object. **Note:** The health check MUST be an existing Health Check in FortiADC.
count <int>	**Optional:** Specify the count. Range is 1 - 1000 (default = 1).
interval <int>	**Optional:** Specify the amount of time after the previous health check before this health check executes, in seconds (1 - 3600, default = 5).

timeout <int>	**Optional:** Specify the timeout period between health checks, in seconds (1 - 3600, default = 3).
<port>|<enter>	If you set a port value, health check traffic will use the port. If you do not set the port value but press Enter instead, health check traffic will use the port value from the specified health check name.

Example

```
FortiADC-VM # execute health-check-verify LB_HLTHCK_ICMP 10.0.0.1
recv hc state is UP
FortiADC-VM # execute health-check-verify LB_HLTHCK_HTTP 10.0.0.1 8080
recv hc state is DOWN
```

execute hwmon

Use this command to view the hardware components status, including the CPU, System Fan, System temperature, and PSU (is supported).

Syntax

```
execute hwmon
```

Example

```
FortiADC-5000F # execute hwmon
CPU Temperature:
        CPU0 Temp: 42.000000 C
        CPU1 Temp: 37.000000 C
        CPU2 Temp: 41.000000 C
        CPU3 Temp: 44.000000 C

Fan Speed:
        FAN1 Speed: 6400.000000 RPM
        FAN2 Speed: 6400.000000 RPM

System Temperature:
        Sys Temp1: 21.000000 C

PSU Status:
        PSU 1 [detected]
            Speed : 5472.000000 RPM
            Temp : 23.000000 C
            Voltage: 12.199219 V
        PSU 2 [detected]
            Speed : 5536.000000 RPM
            Temp : 31.000000 C
            Voltage: 12.298828 V
```

execute isplookup

Use this command to query whether an IP address belongs to an ISP address book.

Syntax

```
execute isplookup <ip>
```

<ip>	Lookup the specified IP address.

Example

```
FortiADC-VM # execute isplookup 1.1.1.1
ISP: china-mobile, province Beijing, subnet 1.1.1.0/24
```

execute log delete-file

Use this command to delete a log file.

Syntax

```
execute log delete-file <filename>
```

<filename>	Log filename.

execute log delete-type

Use this command to delete a log files for a specified log type.

Syntax

```
execute log delete-type {elog|tlog|alog|all}
```

elog	Delete event logs.
tlog	Delete traffic logs.
alog	Delete securty logs.
all	Delete logs for all types.

execute log list-type

Use this command to list log files for a specified log type.

Syntax

execute log list-type {elog|tlog|alog|all}

elog	List event logs.
tlog	List traffic logs.
alog	List securty logs.
all	List logs for all types.

Example

```
FortiADC-VM # execute log list-type ?
<type|all> list all log file by <type>(elog|tlog|alog|all)

FortiADC-VM # execute log list-type all
1.admin.elog 31440 Tue Mar 10 10:01:36 2015
1.app.elog 30578 Tue Feb 24 08:59:09 2015
1.config.elog 23239 Tue Mar 10 13:26:06 2015
1.system.elog 2291 Tue Mar 10 13:50:08 2015
1.dns.tlog 0 Tue Dec 9 12:10:52 2014
1.fw.tlog 0 Tue Dec 9 12:10:52 2014
1.slb_http.tlog 0 Tue Dec 9 12:10:52 2014
1.slb_layer4.tlog 0 Tue Dec 9 12:10:52 2014
1.slb_radius.tlog 0 Tue Dec 9 12:10:52 2014
1.slb_tcps.tlog 0 Tue Dec 9 12:10:52 2014
1.ip_reputation.alog 0 Tue Dec 9 12:10:52 2014
1.synflood.alog 0 Tue Dec 9 12:10:52 2014
FortiADC-VM # execute log rebuild-db
You need to wait 2 minutes at least until log rebuild completes
```

execute log rebuild-db

Use this command to rebuild the log database.

Syntax

```
execute log rebuild-db
```

Example

```
FortiADC-VM # execute log rebuild-db
You need to wait 2 minutes at least until log rebuild completes
```

execute nslookup

Use this command to perform nslookup queries.

Syntax

execute nslookup name {<fqdn>|<ip>}

<fqdn>	Lookup the IP address for the specified host.
<ip>	Lookup the FQDN for the specified IP address.

Example

```
FortiADC-VM # execute nslookup name example.com

Non-authoritative answer:
Name: example.com
Address: 93.184.216.34
```

execute oas-file import

Use this command to manually import OAS files for API Discovery. FortiADC supports OAS files compliant with OpenAPI 3.0 and Swagger 2.0 standard to parse and discover as internal API endpoints that can also be matched by incoming API requests or responses.

JSON and YAML file types are supported.

Syntax

```
execute oas-file import {ftp|tftp} <filename> <ip>
```

import	Select the transfer protocol to use for the OAS file import: • ftp • tftp The file should be in JSON or YAML format, and should follow OpenAPI 3.0 or Swagger 2.0 specifications.
<filename>	Specify the OAS file name.
<ip>	Specify the IP address.

execute packet-capture/packet-capture6

You use these commands to capture packets using tcpdump.

Syntax

```
execute {packet-capture|packet-capture6} <interface> ["Expression"] [<count>]
     [pcap|text] [<filename>]
```

<interface>	Network interface to listen for traffic, such as port1 or port2.
["Expression"]	Specify a filter expression to determine the packets that are captured. Only packets that match the expression are captured. If no expression is specified, all packets received at the interface are captured. For information on filter expressions, see the TCP dump man page: http://www.tcpdump.org/manpages/pcap-filter.7.html
[<count>]	Specify the number of packets to capture and then exit. The valid range is 1 to 10,000. If you do not specify a count, you can terminate the capture by pressing Ctrl-C.
[pcap|text]	Specify pcap or text. If you do not specify a file type, the results are printed to the screen and not to a file.
[<filename>]	Specify the filename for the saved capture. Do not specify a filename extension. The extension .pcap or .txt is added automatically.

Example

The following examples show the tcpdump commands:

```
FortiADC-VM # execute packet-capture port1 "tcp port 80" 5 text test1
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on port1, link-type EN10MB (Ethernet), capture size 65535 bytes
5 packets captured
5 packets received by filter
0 packets dropped by kernel
```

```
FortiADC-VM # execute packet-capture-file list
-rw-r--r-- 1 0 0 577 Sep 3 14:31 test1.txt

FortiADC-VM # execute packet-capture-file upload tftp test1.txt 192.168.1.23
```

See also

execute packet-capture-file

execute packet-capture-file

You use this command to manage tcpdump files.

Syntax

```
execute packet-capture-file {cat <filename>|delete <filename>|list|upload tftp <filename>
     <ip>}
```

`cat <filename>`	Display file contents to the screen.
`delete <filename>`	Delete the specified file.
`list`	List all packet capture files.
`upload tftp <filename> <ip>`	Upload the specified file to the specfied TFTP server.

Example

```
FortiADC-VM # execute packet-capture-file ?
cat show one file
delete delete one file
list list all files
upload upload

FortiADC-VM # execute packet-capture-file list
-rw-r--r-- 1 0 0 802 Mar 10 14:17 test1.txt

FortiADC-VM # execute packet-capture-file cat test1.txt
14:16:58.073847 IP 1.1.1.2.80 > 172.30.144.100.27361: Flags [R.], seq 3807765751, ack
     1748607346, win 2896, options [nop,nop,TS val 836272587 ecr 1224723070], length 0
14:16:58.599663 IP 172.30.144.100.27363 > 1.1.1.2.80: Flags [R.], seq 504059189, ack
     4210316583, win 2920, options [nop,nop,TS val 1224738073 ecr 836272140], length 0
14:16:58.599684 IP 172.30.144.100.32792 > 1.1.1.1.80: Flags [R.], seq 802377254, ack
     4202724881, win 2920, options [nop,nop,TS val 1224738073 ecr 836272140], length 0
14:17:01.723398 IP 1.1.1.1.80 > 172.30.144.100.32792: Flags [R.], seq 1, ack 0, win
     2896, options [nop,nop,TS val 836272952 ecr 1224733072], length 0
```

```
14:17:01.723872 IP 1.1.1.2.80 > 172.30.144.100.27363: Flags [R.], seq 1, ack 0, win
     2896, options [nop,nop,TS val 836272952 ecr 1224733072], length 0

FortiADC-VM # execute packet-capture-file upload tftp test1.txt 192.168.1.23
```

execute ping-option/ping6-option

Use these commands to configure the behavior of the execute ping/ping6 command.

Syntax

```
execute ping-option data-size <bytes_int>
execute ping-option df-bit {yes | no}
execute ping-option pattern <bufferpattern_hex>
execute ping-option repeat-count <repeat_int>
execute ping-option source {auto | <interface_ipv4>}
execute ping-option timeout <seconds_int>
execute ping-option tos {<service_type>}
execute ping-option ttl <hops_int>
execute ping-option validate-reply {yes | no}
execute ping-option view-settings
```

`data-size`	Datagram size in bytes.The default is 56. This option enables you to send out packets of different sizes for testing the effect of packet size on the connection. If you want to configure the pattern that will be used to buffer small datagrams to reach this size, also configure pattern <bufferpattern_hex>.
`df-bit`	Enter either `yes` to set the DF bit in the IP header to prevent the ICMP packet from being fragmented, or enter `no` to allow the ICMP packet to be fragmented.
`pattern`	Hexadecimal pattern, such as `00ffaabb`, to fill the optional data buffer at the end of the ICMP packet. The size of the buffer is determined by data-size <bytes_int>.
`repeat-count`	Number of times to repeat the ping. The default is 5.
`source`	Network interface from which the ping is sent. Enter either `auto` or a FortiADC network interface IP address. The default is auto.
`timeout`	Response timeout in seconds. The default is 2.
`tos`	Type-of-service option value, either: `default` — Do not indicate. (That is, set the TOS byte to 0.) `lowcost` — Minimize cost. `lowdelay` — Minimize delay. `reliability` — Maximize reliability. `throughput` — Maximize throughput.

ttl	Time-to-live (TTL) value. The default is 64.
validate-reply	Whether or not to validate ping replies.
view-settings	Display the current ping option settings.

Example

```
FortiADC-VM # execute ping-option view-settings
Ping Options:
 Repeat Count: 5
 Data Size: 56
 Timeout: 2
 Interval: 1
 TTL: 64
 TOS: 0
 DF bit: unset
 Source Address: auto
 Pattern:
 Pattern Size in Bytes: 0
 Validate Reply: no

FortiADC-VM # execute ping-option ?
 data-size ping option settings
 df-bit set DF bit in IP header <yes | no>
 pattern hex format of pattern, e.g. 00ffaabb
 repeat-count integer value to specify how many times to repeat ping
 source auto | <source interface ip>
 timeout integer value to specify timeout in seconds
 tos IP type-of-service option
 ttl integer value to specify time-to-live
 validate-reply validate reply data <yes | no>
 view-settings view the current settings for ping option

FortiADC-VM # execute ping-option repeat-count 3

FortiADC-VM # execute ping-option view-settings
Ping Options:
 Repeat Count: 3
 Data Size: 56
 Timeout: 2
 Interval: 1
 TTL: 64
 TOS: 0
 DF bit: unset
 Source Address: auto
 Pattern:
 Pattern Size in Bytes: 0
 Validate Reply: no
```

execute ping/ping6

Use these commands to perform an ICMP `ECHO` request (also called a ping) to a host by specifying its fully qualified domain name (FQDN) or IPv4 address, using the options configured by execute ping-option/ping6-option.

Pings are often used to test IP-layer connectivity during troubleshooting.

Syntax

```
execute {ping|ping6} {<hostname> | <ipaddress>}
```

`<hostname>`	Fully qualified domain name (FQDN) of the host to ping.
`<ipaddress>`	IP address to ping.

Example

This example pings a host with the IP address 172.16.1.10.

```
execute ping 172.16.1.10
```

The CLI displays the following:

```
PING 172.16.1.10 (172.16.1.10): 56 data bytes
     64 bytes from 172.16.1.10: icmp_seq=0 ttl=128 time=0.5 ms
     64 bytes from 172.16.1.10: icmp_seq=1 ttl=128 time=0.2 ms
     64 bytes from 172.16.1.10: icmp_seq=2 ttl=128 time=0.2 ms
     64 bytes from 172.16.1.10: icmp_seq=3 ttl=128 time=0.2 ms
     64 bytes from 172.16.1.10: icmp_seq=4 ttl=128 time=0.2 ms
--- 172.16.1.10 ping statistics ---
     5 packets transmitted, 5 packets received, 0% packet loss
     round-trip min/avg/max = 0.2/0.2/0.5 ms
```

The results indicate that a route exists between the FortiADC appliance and 172.16.1.10. It also indicates that during the sample period, there was no packet loss, and the average response time was 0.2 milliseconds.

Example

This example pings a host with the IP address 10.0.0.1.

```
execute ping 10.0.0.1
```

The CLI displays the following:

```
PING 10.0.0.1 (10.0.0.1): 56 data bytes
```

After several seconds, no output appears. The administrator halts the ping by pressing Ctrl+C. The CLI displays the following:

```
--- 10.0.0.1 ping statistics ---
5 packets transmitted, 0 packets received, 100% packet loss
```

The results indicate the host may be down, or there is no route between the FortiADC appliance and 10.0.0.1. To determine the point of failure along the route, further diagnostic tests are required, such as execute traceroute.

Example

This example pings a host with the IP address 2001:0db8:85a3:::8a2e:0370:7334.

```
execute ping6 2607:f0b0:f:420::
```

The CLI displays the following:

```
PING 2607:f0b0:f:420:: (2607:f0b0:f:420::): 56 data bytes
```

After several seconds, no output appears. The administrator halts the ping by pressing Ctrl+C. The CLI displays the following:

```
--- 2607:f0b0:f:420:: ping statistics ---
5 packets transmitted, 0 packets received, 100% packet loss
```

The results indicate the host may be down, or there is no route between the FortiADC appliance and 2607:f0b0:f:420::. To determine the point of failure along the route, further diagnostic tests are required, such as execute traceroute.

execute reboot

Use this command to restart the FortiADC appliance.

Syntax

```
execute reboot
```

Example

This example shows the reboot command in action.

```
execute reboot
```

The CLI displays the following:

```
This operation will reboot the system !
Do you want to continue? (y/n)
```

After you enter `y` (yes), the CLI displays the following:

```
System is rebooting...
```

If you are connected to the CLI through a local console, the CLI displays messages while the reboot is occurring.

If you are connected to the CLI through the network, the CLI will not display any notification while the reboot is occurring, as this occurs after the network interfaces have been shut down. Instead, you may notice that the connection is terminated. Time required by the reboot varies by many factors, such as whether or not hard disk verification is required, but may be several minutes.

execute reload

Use this command to reload the system.

Syntax

```
execute reload
```

Example

```
FortiADC-VM # execute reload
This operation will reload the system!
Do you want to continue? (y/n)y

System is checking ...
```

execute restore

Use the following commands to manually import system files from an FTP/TFTP server or a disk as indicated:

- `execute restore configdisk` — Use this command to restore the configuration from the backup file on the disk.
- `execute restore config-file` — Imports a zip file that includes the configuration text file, error page files, script files, ISP address book files, and certificate files. It is imported from a TFTP server.
- `execute restore image` — Imports a firmware image. It is imported from an FTP or TFTP server.
- `execute restore image alternative` — Boots alternate firmware. If partition1 is active and then system boots partition2 after executing the command and vice versa.
- `execute restore isp-address` — Imports an ISP address book text file. When you perform the restore operation, the imported address book takes priority over entries from the predefined address book (default for the firmware image). In systems with multiple VDOMs, the command applies to the current VDOM only. It is imported from a TFTP server.
- `execute restore waf-signature` — Imports a WAF signature database update. It is imported from an FTP or TFTP server.
- `execute restore geoip-db ftp` — Imports a GEO IP database. It is imported from a FTP server.
- `execute restore geoip-db tftp` — Imports a GEO IP database. It is imported from a TFTP server.
- `execute restore geoip-db factory-default` — Set the GEO IP database to default.
- `execute restore reputation-block-list` — Imports a block list IP reputation database. It is imported from a TFTP server.

Back up a configuration before restoring a different version. This command restores configuration changes only, and does not affect settings that remain at their default values. Default values might vary by firmware version.

Syntax

```
execute restore config disk <filename>
execute restore config-file tftp <filename> <ip> <password>
execute restore image <ftp|tftp|tftp-ha-sync|ftp-ha-sync> <filename> <ip>
execute restore image alternative
execute restore isp-address tftp <filename> <ip>
execute restore waf-signature <ftp|tftp> <filename> <ip>
execute restore reputation-block-list tftp <filename> <ip>
execute restore geoip-db ftp <geoip database name> <ip>
execute restore geoip-db tftp <geoip database name> <ip>
execute restore geoip-db factory-default
```

`<filename>`	Name of the file.
`<ip>`	IP address of the FTP/TFTP server.
`<password>`	Optional. The password is used to decrypt the backup ZIP file.
execute restore image	
`<ftp|tftp|tftp-ha-sync|ftp-ha-sync>`	Select one of the following options: • ftp — Import the firmware image from an FTP server. • tftp — Import the firmware image from a TFTP server. • tftp-ha-sync — Import the firmware image from a TFTP server and transfer the firmware to the peer node to upgrade an HA cluster. For details on the HA cluster upgrade process, refer to the FortiADC Handbook. • ftp-ha-sync — Import the firmware image from a FTP server and transfer the firmware to the peer node to upgrade an HA cluster. For details on the HA cluster upgrade process, refer to the FortiADC Handbook.
execute restore waf-signature	
`<ftp|tftp>`	Select one of the following options: • ftp — Import the WAF Signature from an FTP server. • tftp — Import the WAF Signature from a TFTP server.
execute restore geoip-db ftp/ execute restore geoip-db tftp	
<geoip database name>	Name of the GEO IP database.

Example

```
FortiADC-VM # execute restore config-file tftp backup.zip 192.168.1.23
This operation will overwrite the current settings!
Do you want to continue? (y/n)
```

The FortiADC appliance then applies the configuration backup and reloads.

```
FortiADC-VM # execute restore config disk FortiADC-backup
```

This command downloads a configuration file named FortiADC-backup from the disk to the FortiADC appliance.

execute scan-report export

Use this command to export the WVS (Web Vulnerability Scan) reports to an FTP or TFTP server.

Syntax

```
execute scan-report export [ftp|tftp] <id> <server> <username> <password> [type {xml|html}]
```

`[ftp|tftp]`	Select whether to export the report to an FTP or TFTP server.
`<id>`	Enter an index ID for the exported report.
`<server>`	Enter the server address.
`<username> <password>`	Enter the username and password to log in to the server.
`[type {xml|html}]`	Specify the type of the exported report. The HTML files are packaged into a tar.gz file.

execute scan-report import

Use this command to import a scan report . FortiADC supports scan reports from the following products:

- Acunetix
- IBM AppScan Standard
- WhiteHat
- HP WebInspect
- Qualys
- Telefonica FAAST
- ImmuniWeb
- FortiWeb
- FortiADC

By analyzing the scan results in the imported report, FortiADC automatically generates a WAF profile to prevent the reported attacks. In this command, you will required to specify the name of the generated WAF profile and the actions to be taken upon the attacks.

Syntax

```
execute scan-report import {ftp/tftp <filename> <ip> | restapi <key> <app>} vendor <vendor>
     profile <profile-name> high <action> medium <action> low <action>
```

`ftp/tftp <filename> <ip>`	Import a scan report file by FTP or TFTP.
`restapi <key> <app>`	REST API key. Specify it only when vendor is WhiteHat and Telefonica FAAST. It

	is used for retrieving a report from the WhiteHat and Telefonica FAAST portal using the REST API.
`vendor <vendor>`	Vendor report type, including: • fortiadc • fortiweb • acunetix • appscan • whitehat • webinspect • qualys • faast • immuniweb Some types of reports have specific requirements. For details, see WhiteHat Sentinel scanner report requirements, Telefónica FAAST scanner report requirements, and HP WebInspect scanner report requirements.
`profile <profile-name>`	• Enter a new name for the WAF profile generated to prevent the reported attacks, or • Enter the name of an existing profile. The WAF settings based on the scan report will be merged to an existing WAF profile. If there are conflict settings, the new ones will overwrite the existing ones.
`high <action>`	Specify the action that FortiADC will take if High severity attacks are detected.
`medium <action>`	Specify the action that FortiADC will take if Medium severity attacks are detected.
`low <action>`	Specify the action that FortiADC will take if Low severity attacks are detected.

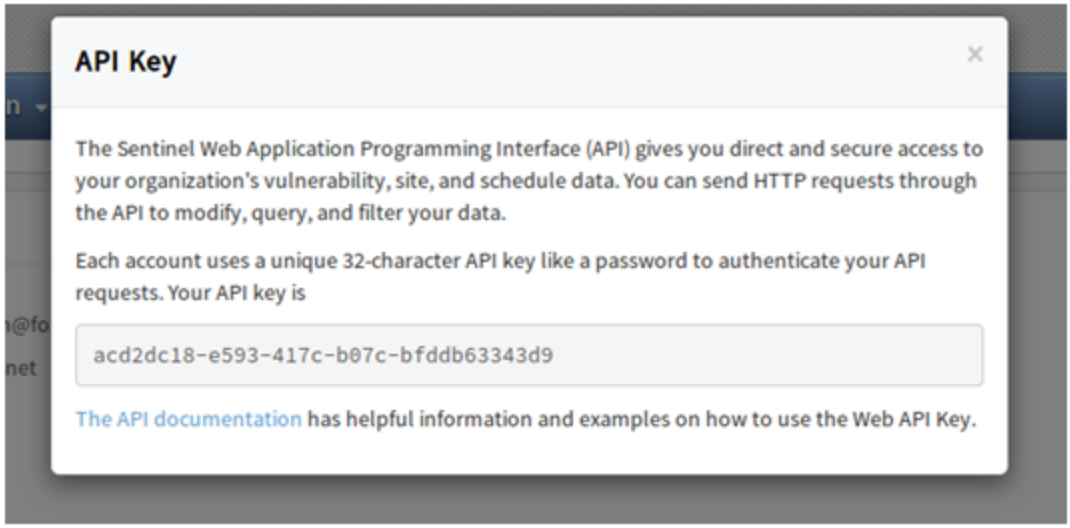
WhiteHat Sentinel scanner report requirements

To allow [[[Undefined variable FortiWebVariables.FortiWeb]]] to generate rules using a WhiteHat Sentinel scanner report, ensure that the parameters “display_vulnerabilities” and “display_description” are enabled when you run the scan.

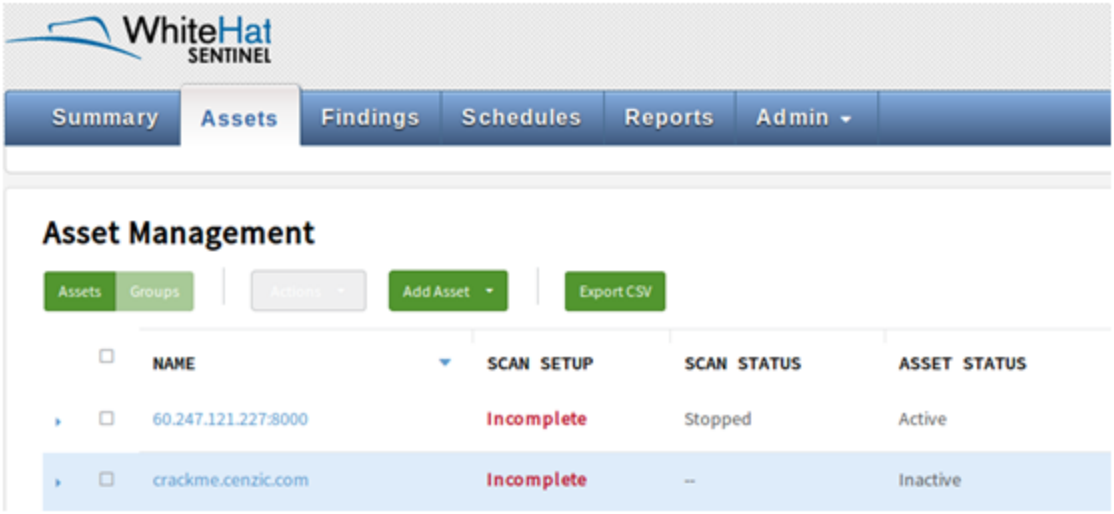
You can upload a WhiteHat Sentinel scanner report using either a report file you have downloaded manually or directly import the file from the WhiteHat portal using the RESTful API. Importing a scanner file from the WhiteHat portal requires the API key and application name that WhiteHat provides.

To retrieve the WhiteHat API key and application name

1. Go to the following location and log in:
   https://source.whitehatsec.com/summary.html#dashboard
2. In the top right corner, click **My Profile**.
3. Click View My API Key and enter your password.
   Your API key is displayed. For example:



4. To view the application name, navigate to the Assets tab. The application name is the NAME value. For example:

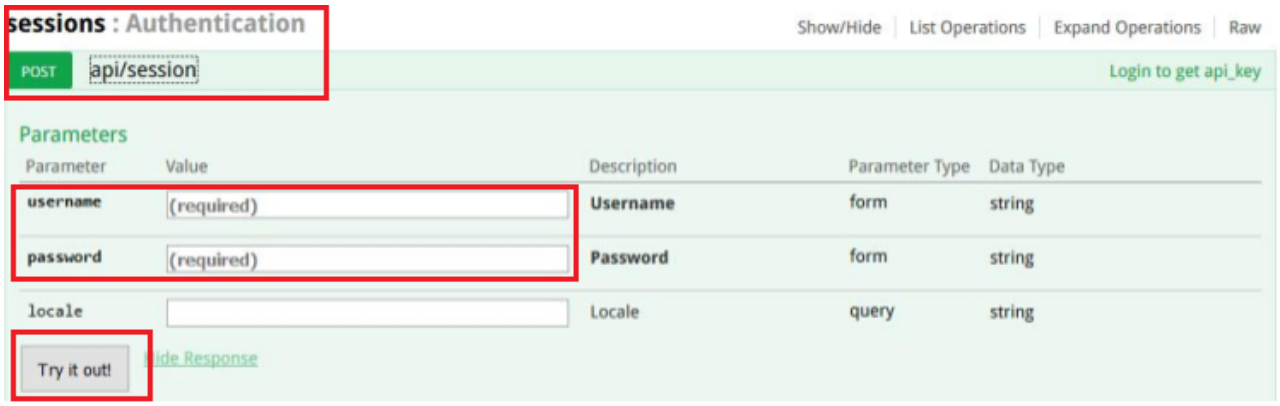


Telefónica FAAST scanner report requirements

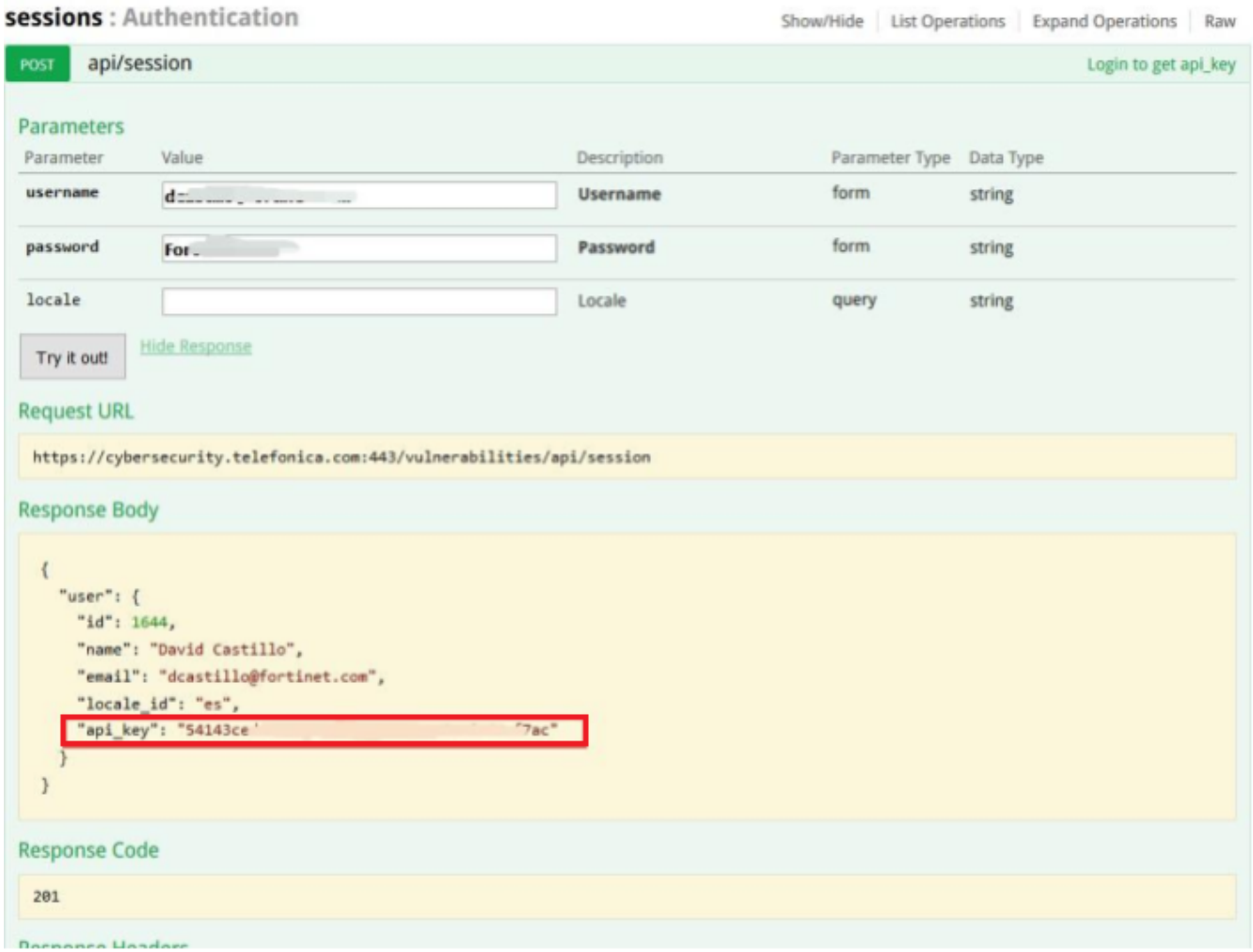
You can upload a Telefónica FAAST scanner report using either a report file you have downloaded manually or directly import the file from the Telefónica FAAST portal using the RESTful API. Importing a scanner file from the Telefónica FAAST portal requires the API key that Telefónica FAAST provides. One Telefónica FAAST scanner account can apply for an API key.

To apply for a Telefónica FAAST API key

1. Go to the following location and log in:
   https://cybersecurity.telefonica.com/vulnerabilities/es/api_docs
2. In the **session : Authentication** page, please select **POST > api/session** for the method, and fill in the blanks for **username** and **password**. Then click **Try it out**.



3. The API key will be gave in the **Response Body** if the username and password are authorized.



HP WebInspect scanner report requirements

To generate rules from HP WebInspect, when you export the report, for the **Details** option, select either **Full** or **Vulnerabilities**.

execute scripting-shared-table

Use this command to list, show, or delete Lua shared tables.

Syntax

```
execute scripting-shared-table list {<VS-name>|any}
execute scripting-shared-table show {<table-name>|all} {<VS-name>|any}
execute scripting-shared-table delete {<table-name>|all} {<VS-name>|any}
```

`list`	Lists Lua shared tables based on the applicable virtual server. This will list all table information in the format of `(tableName):(tableInfo)`. Select either of the following options: • <VS-name> — Enter a virtual server name. All Lua shared tables from this specified virtual server name will be listed. • any — Instead of a specified virtual server name, the command will loop over all virtual servers to list Lua shared tables from any virtual server.
`show`	Shows Lua shared tables based on the applicable "table name" and "virtual server name". This will dump one shared table in the format of `(key):(value)`. To minimize lock errors and process delays, only one table will be shown at a time. Select either of the following options: • <table-name> or all — A table name is required to precede the virtual server name. You can either specify a table name or select "all" to apply the command to all tables. • <VS-name> or any — A virtual server name is required. You can either specify a virtual server name or select "any" to apply the command to any virtual server.
`delete`	Deletes Lua shared tables from all the processes of the applicable virtual server. Select either of the following options: • <table-name> or all — A table name is required to precede the virtual server name. You can either specify a table name or select "all" to apply the command to all tables. • <VS-name> or any — A virtual server name is required. You can either specify a virtual server name or select "any" to apply the command to any virtual server.

Examples

```
FortiADC-VM # exec scripting-shared-table list VS1
Printed Table 'all', current entry count: 2.
(VS1TableDemo1): (shmkey=1321348 table_size=128 entry_size=2048 mem_limit=20971520)
(VS1TableDemo2): (shmkey=1321349 table_size=128 entry_size=2048 mem_limit=20971520)


FortiADC-VM # exec scripting-shared-table list any
List of applicable VS names:
```

```
VS1
VS2

VS1:

Printed Table 'all', current entry count: 2.
(VS1TableDemo1): (shmkey=1321348 table_size=128 entry_size=2048 mem_limit=20971520)
(VS1TableDemo2): (shmkey=1321349 table_size=128 entry_size=2048 mem_limit=20971520)

VS2:


FortiADC-VM # exec scripting-shared-table show VS1TableDemo1 VS1
Printed Table 'VS1TableDemo1', current entry count: 3.
(1002): (VS1Value10002)
(1003): (VS1Value10003)
(1001): (VS1Value10001)


FortiADC-VM # exec scripting-shared-table show all VS1
Printed Table 'all', current entry count: 2.
(VS1TableDemo1): (shmkey=1321348 table_size=128 entry_size=2048 mem_limit=20971520)
(VS1TableDemo2): (shmkey=1321349 table_size=128 entry_size=2048 mem_limit=20971520)


FortiADC-VM # exec scripting-shared-table delete VS1TableDemo1 VS1
Shared table 'VS1TableDemo1' is destroyed from VS 'VS1'.


FortiADC-VM # exec scripting-shared-table delete all VS1
All shared tables with VS 'VS1' are destroyed.


FortiADC-VM # exec scripting-shared-table list any
List of applicable VS names:
VS1
VS2

VS1:

VS2:
```

execute shutdown

Use this command to prepare the FortiADC appliance to be powered down by halting the software, clearing all buffers, and writing all cached data to disk.

Power off the FortiADC appliance only after issuing this command. Unplugging or switching off the FortiADC appliance without issuing this command could result in data loss.

Syntax

```
execute shutdown
```

Example

```
FortiADC-VM # execute shutdown
This operation will halt the system!
Do you want to continue? (y/n) y

System is shutting down...(power-cycle needed to restart)
```

If you are connected to the CLI through a local console, the CLI displays a message when the shutdown is complete.

If you are connected to the CLI through the network, the CLI will not display any notification when the shutdown is complete, as this occurs after the network interfaces have been shut down. Instead, you may notice that the connection times out.

execute ssh

Use this command to open an SSH connection to a remote host using the specified username.

Syntax

```
execute ssh <user@host> [port]
```

`<user@host>`	Username@host or IP address. username@ is optional. If not specified, the user named **admin** is inferred by default.
`[port]`	Specify a port if not the commonly used port 22.

Example

```
FortiADC-docs $ execute ssh admin2@192.168.0.1
FortiADC-QA #
```

execute statistics-db

Use this command to reset statistics or restore traffic statistics.

Syntax

```
execute statistics-db {reset|restore}
```

reset	Reset traffic statistics.
restore	Restore traffic statistics from its backup.

Example

```
FortiADC-VM # execute statistics-db restore
You need to wait 2 minutes at least until statistics db restore completes
```

execute ssli mode

Use this command to enable ssli mode. For information on how to deploy SSLi, see SSL Deployment Guide.

Syntax

```
execute ssli mode {enable|disable}
```

execute telnet

Use this command to open an Telnet connection to a remote host.

Syntax

execute telnet <ip> [port]

<ip>	IP address of the remote host.
[port]	Specify a port if not the commonly used port 23.

Example

```
FortiADC-VM # execute telnet 192.168.0.1
Entering character mode
Escape character is '^]'.
Remote Host login: admin
Password:
Welcome!
Remote Host #
```

execute traceroute

Use this command to use ICMP to test the connection between the FortiADC appliance and another network device, and display information about the time required for network hops between the device and the FortiADC appliance.

Syntax

```
execute traceroute {<hostname> | <ipaddress>}
```

<hostname>	Fully qualified domain name (FQDN) of the other network device.
<ipaddress>	IP address of the other network device.

Example

This example tests connectivity between the FortiADC appliance and docs.fortinet.com. In this example, the trace times out after the first hop, indicating a possible connectivity problem at that point in the network.

```
FortiADC# execute traceroute docs.fortinet.com
traceroute to docs.fortinet.com (65.39.139.196), 30 hops max, 38 byte packets
1  172.16.1.200 (172.16.1.200) 0.324 ms 0.427 ms 0.360 ms
 2  * * *
```

execute vm license

Use this command to upload license files for a virtual appliance deployment.

Syntax

```
execute vm license tftp <filename> <ip> [<password>]
```

<filename>	Name of the license file.
<ip>	IP address of the TFTP server.
<password>	Password if the license file is encrypted.

Example

```
FortiADC-VM # execute vm license tftp license.lic 192.168.1.23
This operation will replace the current vm license and reload the system!
Do you want to continue? (y/n)
```

execute web-category-test

Use this command to see the FortiGuard web category that a specified URL has been mapped to. You can also find a lookup tool on http://fortiguard.com/webfilter.

Syntax

```
execute web-category-test <url>
```

Example

```
FortiADC-VM # execute web-category-test docs.fortinet.com
```

execute SSL client-side session statistics

Use this command to see the SSL client-side session reuse statistics. You can see the statistics of session ID reuse and session ticket reuse.

Syntax

```
execute ssl-client-side-session-statistics show/clean <datasource>
```

Example

```
FortiADC-VM # execute ssl-client-side-session-statistics show VS
FortiADC-VM # execute ssl-client-side-session-statistics clean VS
```

execute SSL handshake record statistics

Use this command to see the SSL handshake record statistics. You can see the statistics of successful and failed handshakes.

Syntax

```
execute ssl-handshake-n-record-statistics show/clean <datasource>
```

Example

```
FortiADC-VM # execute ssl-handshake-n-record-statistics show VS
FortiADC-VM # execute ssl-handshake-n-record-statistics clean VS
```

execute waf block-ip

Use the following commands to view, filter, or release any client IP address that is currently blocked by WAF modules prior to the block expiry period.

- execute waf block-ip list on page 595 — Generates a list of WAF blocked IP addresses.
- execute waf block-ip filter on page 595— Allows you to filter through the list of WAF blocked IP addresses.
- execute waf block-ip release on page 596— Allows you to release one or all WAF blocked IP addresses.

execute waf block-ip list

Use this command to generate a list of all the IP addresses that are currently blocked by WAF modules through the **Block** or **Period Block** actions.

Syntax

```
execute waf block-ip list
```

execute waf block-ip filter

Use this command to filter through the WAF blocked IPs via the IP address or the name of the virtual server that has blocked the IP address.

Syntax

```
execute waf block-ip filter {clear|ip|show|vs-name} {<ip>|<vs-name>}
```

clear	Clears the filters.
ip	Filter by a single IP or an IP range.
show	Show the filters.
vs-name	Filter by a virtual server name.
<ip>	If **ip**, specify the IP or an IP range to filter by. For example, `1.1.1.1-2.2.2.2`
<vs-name>	If **vs-name**, specify the name of the virtual server that has blocked the IP address.

Example

```
FortiADC-VM # execute waf block-ip filter ip 50.1.0.1

FortiADC-VM # execute waf block-ip filter show
ip range: 50.1.0.1 - 50.1.0.1
virtual server: any

FortiADC-VM # execute waf block-ip filter vs-name VS1

FortiADC-VM # execute waf block-ip filter show
ip range: any
virtual server: VS1
```

execute waf block-ip release

Use this command to release one or all WAF blocked IP addresses.

Syntax

```
execute waf block-ip release {all|ip|vs-name} {<ip>|<vs-name>}
```

all	Release all the IP addresses currently blocked by the WAF.
ip	Release a single IP or an IP range.
vs-name	Release an IP by their virtual server name.
<ip>	If **ip**, specify the IP or an IP range to release. For example, `'1.1.1.1-2.2.2.2'`
<vs-name>	If **vs-name**, specify the name of the virtual server that has blocked the IP address.

Example

```
FortiADC-VM # execute waf block-ip release ip 50.1.0.1

FortiADC-VM # execute waf block-ip release vs-name VS1
```

```
FortiADC-VM # execute waf block-ip release vs-name VS1 ip '1.1.1.1'
```

execute web-vulnerability-scan

Use this command to execute the web vulnerability scan.

Syntax

execute web-vulnerability-scan <start/stop>

<starat/stop>>	Start initiates the scan. Stop terminates the scan.

See the "config security waf scanner" on page 1.

execute web-vulnerability-scan mitigate

By analyzing the scan results in the WVS (Web Vulnerability Scan) report, FortiADC automatically generates a WAF profile to prevent the reported attacks.

In this command, you can specify the WVS reports to be analyzed, the name of the generated WAF profile, and the actions to be taken upon the attacks.

Syntax

```
execute web-vulnerability-scan mitigate id <id1,id2…> profile <profile-name> [high <action>
     medium <action> low <action>]
```

id <id1,id2…>	Specify the ID of the WVS report. You can separate multiple reports by ",".
profile <profile-name>	• Enter a new name for the WAF profile generated to prevent the reported attacks, or • Enter the name of an existing profile. The WAF settings based on the scan report will be merged to an existing WAF profile. If there are conflict settings, the new ones will overwrite the existing ones.
high <action>	Specify the action that FortiADC will take if High severity attacks are detected.
medium <action>	Specify the action that FortiADC will take if Medium severity attacks are detected.
low <action>	Specify the action that FortiADC will take if Low severity attacks are detected.

execute forticloud create-account

Use this command to create a FortiCloud account.

Syntax

```
execute forticloud create-account <account-id>
```

Example

```
(M) ADC1 # execute forticloud create-account sam
execute forticloud create-account <account-id> <password>

(M) ADC1 # execute forticloud create-account sam
<password> Password

(M) ADC1 # execute forticloud create-account sam pwd
Create account success.
```

execute forticloud login

Use this command to log into a FortiCloud account.

Syntax

```
execute forticloud login <account-id>
```

Example

```
ADC-6 # execute forticloud login xxxxx@fortinet.com xxxxx
Login success.
```

execute forticloud try

Use this command to test a connection to a FortiCloud account.

Syntax

```
execute forticloud try <account-id> <password>
```

Example

```
ADC-6 # execute forticloud try nazhao@xxxxx xxxxx
This account is valid.
```

execute fctems

Use the following commands to check or configure the EMS server certificate for your FortiClient EMS connector.

- execute fctems is-verified on page 599 — to check if the configured EMS server has a verified certificate.
- execute fctems test-connectivity on page 599 — to test the connectivity of the FortiClient EMS server.
- execute fctems unverify on page 600 — to unverify the FortiClient EMS server.
- execute fctems verify on page 600 — to verify the FortiClient EMS server.

execute fctems is-verified

Use this command to check if the configured EMS server has a verified certificate.

Syntax

```
execute fctems is-verified <datasource>
```

Example

```
ADC# execute fctems is-verified ems-226
Configured FortiClient EMS has been verified.
```

execute fctems test-connectivity

Use this command to test the connectivity of the FortiClient EMS server.

This provides the connection status of your FortiClient EMS connector configuration. If the connection is not successful, further detail is provided for the status condition.

Syntax

```
execute fctems test-connectivity <datasource>
```

Example

```
ADC# execute fctems test-connectivity ems-226
Connection test had an error -2: EMS server connection failed. Authentication denied.

ADC# execute fctems test-connectivity ems-223
Connection test was successful.
```

execute fctems unverify

Use this command to unverify the FortiClient EMS server.

Syntax

```
execute fctems unverify <datasource>
```

Example

```
ADC# execute fctems test-connectivity ems-226
FortiClient EMS certificate successfully unverified.
```

execute fctems verify

Use this command to verify the FortiClient EMS server. In order for the FortiClient EMS and FortiADC to communicate, FortiClient EMS provides an EMS server certificate. This certificate must be reviewed for correctness, and accepted if deemed valid. You will be prompted to trust the remote certificate to complete the verification.

After you have verified the EMS server certificate is verified, you need to authorize the FortiADC as a Fabric Device in FortiClient EMS.

Syntax

```
execute fctems verify <datasource>
...
Do you wish to add the above certificate to trusted remote certificates? Do you want to
     continue? (y/n)
```

Example

```
ADC# execute fctems test-connectivity ems-226
Subject: C = US, ST = California, L = Sunnyvale, O = Fortinet, OU = FortiClient, CN =
     FCTEMS8822003243, emailAddress = support@fortinet.com
...
Do you wish to add the above certificate to trusted remote certificates? Do you want to
     continue? y
Certificate successfully configured and verified.
```

execute update-now

Use this command to trigger package updates for all FortiGuard services in FortiADC.

If `update-dldb` is **disabled** in `config system fortiguard`, then this command will trigger all packages to update except for the DLP database.

Syntax

```
execute update-now
```

execute update-dldb

Use this command to trigger package updates for the FortiGuard DLP database.

This command will not work if `update-dldb` is **disabled** in `config system fortiguard`.

Syntax

```
execute update-dldb
```

get

Use `get` commands to display configuration settings and values. You must have read permission for the configuration object you want to display.

`show` commands display user-configured setings but not default settings; `get` commands display all settings, including both user-configured settings and defaults.

For example, you might get the current DNS settings:

```
FortiADC-VM # get system dns
primary : 8.8.8.8
secondary : 0.0.0.0

FortiADC-VM #
```

Notice that the command displays the setting for the secondary DNS server, even though it has not been configured, or has reverted to its default value.

Also unlike `show`, unless used from within an object or table, `get` requires that you specify the object or table whose settings you want to display.

For example, at the root prompt, the following command is valid:

```
FortiADC-VM # get system dns
primary : 8.8.8.8
secondary : 0.0.0.0
```

The following command displays no output:

```
FortiADC-VM # get
```

Like `show`, depending on whether or not you have specified an object, `get` displays one of two different outputs:

- The configuration you have just entered but not yet saved
- The configuration as it currently exists on the flash disk

For example, immediately after configuring the secondary DNS server setting but before saving it, `get` displays two different outputs. In the following example, the first output from `get` indicates the value that you have configured but not yet saved; the second output from `get` indicates the value that was last saved to disk.

```
FortiADC-VM # config system dns

FortiADC-VM (dns) # set secondary 192.168.1.10

FortiADC-VM (dns) # get
primary : 8.8.8.8
secondary : 192.168.1.10

FortiADC-VM (dns) # get system dns
primary : 8.8.8.8
secondary : 0.0.0.0
```

If you were to now enter `end`, saving your setting to disk, `get` output for both syntactical forms would again match. However, if you were to enter `abort` at this point and discard your recently entered secondary DNS setting instead of saving it to disk, the configuration would therefore match the second output, not the first.

If you have entered settings but cannot remember how they differ from the existing configuration, the two different forms of `get`, with and without the object name, can be a useful way to remind yourself.

Most `get` commands, such as `getsystem dns`, are used to display configured settings. You can find information the configuration details in the corresponding config command reference.

Other `get` commands, such as get router info ospf, get router info routing-table, get security waf-signature-status, get security scan-report, "get security scan-task" on page 1get system performance, and get system status, are used to display status, not configuration.

get firewall global

Use this command to display the current timeout settings for the firewall.

Syntax

```
get firewall global
```

Example

```
FortiADC-VM # get firewall global
generic-timeout               : 600
tcp-established-timeout       : 3600
tcp-syn-recv-timeout          : 60
tcp-syn-sent-timeout          : 120
tcp-close-timeout             : 3
tcp-fin-wait-timeout          : 120
tcp-last-ack-timeout          : 30
udp-timeout                   : 30
udp-stream-timeout            : 180
```

get router info ospf

Use this command to display status for OSPF.

Syntax

```
FortiADC-VM # get router info ospf ?
database database
interface show ospf interfaces
neighbor show ospf neighbors
route show ospf routing table
```

```
status show ospf status

FortiADC-VM # get router info ospf database ?
asbr-summary show ospf database ASBR summary link states
brief show ospf LSA list
external show ospf database external link states
max-age LSAs in MaxAge list
network show ospf database network link states
nssa-external show ospf database NSSA external link states
router show ospf database router link states
self-originate show ospf database self-originated link states
summary show ospf database network summary link states
```

Example

```
FortiADC-VM # get router info ospf status
OSPF Routing Process, Router ID: 1.1.1.2
Supports only single TOS (TOS0) routes
This implementation conforms to RFC2328
RFC1583Compatibility flag is disabled
OpaqueCapability flag is disabled
Initial SPF scheduling delay 200 millisec(s)
Minimum hold time between consecutive SPFs 1000 millisec(s)
Maximum hold time between consecutive SPFs 10000 millisec(s)
Hold time multiplier is currently 1
SPF algorithm has not been run
SPF timer is inactive
Refresh timer 10 secs
Number of external LSA 0. Checksum Sum 0x00000000
Number of opaque AS LSA 0. Checksum Sum 0x00000000
Number of areas attached to this router: 0

FortiADC-VM # get router info ospf database summary

OSPF Router with ID (1.1.1.2)
```

get router info routing-table

Use this command to display the routing table.

Syntax

```
FortiADC-VM # get router info routing-table ?
all show all routing table entries
kernel-all show all routing table entries
kernel-connected show connected routing table entries
kernel-llb show llb routing table entries
kernel-static show static routing table entries
```

Example

```
FortiADC-VM # get router info routing-table all
Codes: K - kernel route, C - connected, S - static, O - OSPF, P - PPPoE
> - selected route, * - FIB route

S>* 0.0.0.0/0 [10/0] via 172.30.147.254, port1
C>* 169.254.0.0/16 is directly connected, haport0
C>* 172.30.144.0/22 is directly connected, port1
```

get security waf-signature-status

Use this command to display version information for the WAF signature updates from FortiGuard.

Syntax

```
get security waf-signature-status
```

Example

```
FortiADC-VM # get security waf-signature-status
Version : 1.1.0
Engine Version : 1.0
Signature Number : 1758
Release Date : 2015-07-06 11:00:00 UTC
```

get security scan-report

Use this command to list all scans.

Syntax

```
get security scan-report
```

Example

```
ID:0 Taskname:1 Created Time:10:08:55,10-26-18
ID:1 Taskname:1 Created Time:15:25:17,10-25-18
```

get security scan-task

Use this command to list all tasks.

Syntax

get security scan-task

Example

ID:0 TaskName:task-2 Status:STOP

ID:1 TaskName:task-1 Status:STOP

ID:2 TaskName:3 Status:STOP

ID:3 TaskName:1 Status:STOP

get system ha-status

Use this command to display ha status information including:

- Mode
- State
- Sync status and sync statistics
- Serial number
- Node ID
- IP address
- Monitor status
- Peer count

Syntax

```
get system ha-status
```

Example

```
(M) FADC-VM (global) # get system ha-status
Mode: active-active
State: primary
```

```
Config-sync: In sync (not sync)
Serial-number: FADV010000039883
Node-id: 1
IP address: 169.254.3.131
Last change time: Tue Mar 15 15:39:42 2016
Last change reason: Device initialization

Monitor status
System Harddisk: pass
Link Up: port1
Down: port2
Remote IP
Up:
Down:

Sync statistics: Sent Received
L4 session and persistence sync pkts: 0 0
L7 persistence sync pkts: 0 0
Device management errors:
Duplicate node id: 0
Version mismatch: 0

Peer count: 1
State: secondary(working)
Serial-number: FADV010000039890
Node-id: 2
IP address: 169.254.122.212
```

get system performance

Use this command to display CPU usage, memory usage, average system load, and up time.

Normal idle load varies by hardware platform, firmware, and configured features. To determine your specific baseline for idle, configure your system completely, reboot, then view the system load. After at least 1 week of uptime with typical traffic volume, view the system load again to determine the normal non-idle baseline.

System load is the average of percentages relative to the maximum possible capability of this hardware/system platform. It includes:

- Average system load
- Number of HTTP daemon/proxy processes or children
- Memory usage
- Disk swap usage

Syntax

```
get system performance
```

Example

```
FortiADC-VM # get system performance
CPU usage: 2% used, 98% idle
Memory usage: 40% used
System Load: 0
Uptime: 12 days 23 hours 32 minutes
```

get system status

Use this command to display system status information including:

- Firmware version, build number and date
- License and registration status
- Serial number
- WAF database version
- IP Reputation database version
- Log disk availability
- Hostname
- Current HA mode
- Uptime
- System time

Firmware versions are tagged to indicate the maturity level of its feature development:

- Feature (GA.F) — New features are included in this version.
- Mature (GA.M) — This version primarily includes bug fixes and/or vulnerability patches, and no new or major features.

Syntax

```
get system status
```

Example

```
FortiADC # get system status
Version:                FortiADC-VM v7.4.1,build0310,231030 (GA.F)
VM Registration:        Valid: License has been successfully authenticated with
registration servers.
VM License File:        License file and resources are valid.
VM Resources:           2 CPU/2 allowed, 7564 MB RAM, 29 GB Disk
Serial-Number:          FADVMSTM22000199
WAF Signature DB:       00001.00043 (Expire: 2023-8-13)
IP Reputation DB:       00004.00822 (Expire: 2023-8-13)
```

```
Geography IP DB:          00002.00189
Geography Regions:        00002.00024 (CN)
Credential Stuffing DB:   00001.00438 (Expire: 2023-8-13)
Certificate DB:           00001.00047
Regular Virus DB:         00091.05971 (Expire: 2023-8-13)
Extended Virus DB:        00091.05641 (Expire: 2023-8-13)
Extreme Virus DB:         00091.05670 (Expire: 2023-8-13)
AV Engine:                00006.00285 (Expire: 2023-8-13)
IPS-DB:                   00025.00619 (Expire: 2023-8-13)
IPS-ETDB:                 00025.00619 (Expire: 2023-8-13)
IPS Engine:               00004.00021 (Expire: 2023-8-13)
Bootloader Version:       n/a
Hard Disk:                Capacity 29 GB, Used 1 GB ( 5.73%), Free 27 GB
Log Size:                 34 MB, 0%
Hostname:                 FortiADC-XENAWS
HA Configured Mode:       standalone
HA Effective Mode:        Standalone
Distribution:             International
CM Agent status:          (Disabled)
Uptime:                   12 days  7 hours  54 minutes
Last Reboot:              Wed Nov 01 13:28:27 PDT 2023
System Time:              Mon Nov 13 20:23:22 PST 2023
```

get system traffic-group

Use this command to display a traffic group.

Syntax

```
get system traffic-group <traffic-group name>
```

Example

```
down2000D (global) # get system traffic-group default
failover-order : 0 1 2 3 4 5 6 7
preempt : disable
network-failover : disable
```

get system traffic-group status

Use this command to display traffic-group status.

Syntax

```
get system traffic-group-status detail/brief
```

Example

Use the following command to get detailed status information about the traffic group.

```
down2000D (global) # get system traffic-group-status detail
Traffic group: default
Current device node: 0
Next device node: 1
Preempt: no
Floating IP addresses: vlan1101InDris 10.76.12.110


Traffic group: trafficGrp1
Current device node: 0
Next device node: 1
Preempt: yes
Floating IP addresses: port5 10.76.76.76
```

Use the following command to get brief status information about the traffic group.

```
down2000D (global) # get system traffic-group-status brief
Traffic group: default
Current device node: 0
Next device node: 1
Traffic group: trafficGrp1
Current device node: 0
Next device node: 1
```

get router info bgp all

Use this command to display all BGP information.

Syntax

```
get router info bgp all
```

Example

```
FortiADC-VM # get router info bgp all
BGP table version is 0, local router ID is 10.0.6.217
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete
```

```
Network Next Hop Metric LocPrf Weight Path
*> 2.1.0.0/16 10.0.0.1 0 32768 ?
*>i38.0.0.0/24 172.15.2.29 0 100 0 102 i
* i172.15.1.0/24 172.15.1.218 0 100 0 i
*> 0.0.0.0 0 32768 i
*>i172.15.2.0/24 172.15.1.218 0 100 0 i
*> 192.168.11.0 0.0.0.0 0 32768 i
Total number of prefixes 5
```

get router info bgp ip

Use this command to display BGP information related to a specified IPv4 address.

Syntax

```
get router info bgp ip <ipv4 address>
```

Example

```
FortiADC-VM # get router info bgp ip 38.0.0.10
BGP routing table entry for 38.0.0.0/24
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Not advertised to any peer
102
172.15.22.29 from 172.15.1.218 (10.0.6.238)
, metric 0, localpref 100, valid, internal, best
Last update: Mon Jan 2 22:50:53 2017
```

get router info bgp neighbors

Use this command to display BGP neighbor information.

Syntax

```
get router info bgp neighbors
```

Example

```
FortiADC-VM (root) # get router info bgp neighbors
BGP neighbor is 172.15.1.218, remote AS 101, local AS 101, internal link
BGP version 4, remote router ID 10.0.6.238
BGP state = Established, up for 03:34:16
```

```
Last read 00:00:15, hold time is 180, keepalive interval is 60 seconds
Configured hold time is 180, keepalive interval is 60 seconds
Neighbor capabilities:
4 Byte AS: advertised and received
Route refresh: advertised and received(old & new)
Address family IPv4 Unicast: advertised and received
Message statistics:
Inq depth is 0
Outq depth is 0
Sent Rcvd
Opens: 2 0
Notifications: 0 0
Updates: 3 4
Keepalives: 216 215
Route Refresh: 0 0
Capability: 0 0
Total: 221 219
Minimum time between advertisement runs is 5 seconds
For address family: IPv4 Unicast
Community attribute sent to this neighbor(both)
3 accepted prefixes
Connections established 1; dropped 0
Last reset never
Local host: 172.15.1.217, Local port: 179
Foreign host: 172.15.1.218, Foreign port: 27671
Nexthop: 172.15.1.217
Nexthop global: ::
Nexthop local: ::
BGP connection: non shared network
Read thread: on Write thread: off
```

get router info bgp regexp

Use this command to display BGP information by a regular expression.

Syntax

```
get router info bgp regexp <name line>
```

Example

```
FortiADC-VM (root) # get router info bgp regexp .*
BGP table version is 0, local router ID is 10.0.6.217
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete
Network Next Hop Metric LocPrf Weight Path
*> 2.1.0.0/16 10.0.0.1 0 32768 ?
*>i38.0.0.0/24 172.15.2.29 0 100 0 102 i
```

```
* i172.15.1.0/24 172.15.1.218 0 100 0 i
*> 0.0.0.0 0 32768 i
*>i172.15.2.0/24 172.15.1.218 0 100 0 i
*> 192.168.11.0 0.0.0.0 0 32768 i
Total number of prefixes 5
```

get router info bgp summary

Use this command to display BGP summary information.

Syntax

```
get router info bgp summary
```

Example

```
FortiADC-VM (root) # get router info bgp summary
BGP router identifier 10.0.6.217, local AS number 101
RIB entries 9, using 1008 bytes of memory
Peers 1, using 4560 bytes of memory
Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd
172.15.1.218 4 101 222 224 0 0 0 03:37:33 3
Total number of neighbors 1
```

get router info6 bgp all

Use this command to display all IPv6 BGP information.

Syntax

```
get router info6 bgp all
```

Example

```
FortiADC-VM (bgp) # get router info6 bgp all
BGP table version is 0, local router ID is 10.0.6.217
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete
Network Next Hop Metric LocPrf Weight Path
*> 2015::/64 :: 0 32768 i
* i2016::/64 2016::2 0 100 0 i
```

```
*> :: 0 32768 i
*>i2017::/64 2016::2 0 100 0 i
* i2020::/64 2017::2 0 100 0 102 i
Total number of prefixes 4
```

get router info6 bgp ip

Use this command to display BGP information related to a specified IPv6 address.

Syntax

```
get router info6 bgp ip <ipv6 address>
```

Example

```
FortiADC-VM (bgp) # get router info6 bgp ip6 2017::0103
BGP routing table entry for 2017::/64
Paths: (1 available, best #1, table Default-IP-Routing-Table)
Not advertised to any peer
Local
2016::2 (metric 1) from 2016::2 (10.0.6.238)
, metric 0, localpref 100, valid, internal, best
Last update: Tue Jan 3 02:45:25 2017
```

get router info6 bgp neighbors

Using this command to display BGP IPv6 neighbor information.

Syntax

```
get router info6 bgp neighbors
```

Example

```
FortiADC-VM (bgp) # get router info6 bgp neighbors
BGP neighbor is 2016::2, remote AS 101, local AS 101, internal link
BGP version 4, remote router ID 10.0.6.238
BGP state = Established, up for 00:14:57
Last read 00:00:57, hold time is 180, keepalive interval is 60 seconds
Configured hold time is 180, keepalive interval is 60 seconds
Neighbor capabilities:
4 Byte AS: advertised and received
```

```
Route refresh: advertised and received(old & new)
Address family IPv4 Unicast: advertised and received
Address family IPv6 Unicast: advertised and received
Message statistics:
Inq depth is 0
Outq depth is 0
Sent Rcvd
Opens: 14 1
Notifications: 0 2
Updates: 12 9
Keepalives: 20 17
Route Refresh: 0 0
Capability: 0 0
Total: 46 29
Minimum time between advertisement runs is 5 seconds
For address family: IPv4 Unicast
Community attribute sent to this neighbor(both)
3 accepted prefixes
For address family: IPv6 Unicast
Community attribute sent to this neighbor(both)
2 accepted prefixes
Connections established 3; dropped 2
Last reset 00:15:08, due to BGP Notification received
Local host: 2016::1, Local port: 179
Foreign host: 2016::2, Foreign port: 57424
Nexthop: 10.0.6.217
Nexthop global: 2016::1
Nexthop local: ::
BGP connection: shared network
Read thread: on Write thread: off
```

get router info6 bgp regexp

Use this command to display IPv6 BGP information by a regular expression.

Syntax

```
get router info6 bgp regexp <name line>
```

Example

```
FortiADC-VM (bgp) # get router info6 bgp regexp .*
BGP table version is 0, local router ID is 10.0.6.217
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete
Network Next Hop Metric LocPrf Weight Path
*> 2015::/64 :: 0 32768 i
* i2016::/64 2016::2 0 100 0 i
```

```
*> :: 0 32768 i
*>i2017::/64 2016::2 0 100 0 i
* i2020::/64 2017::2 0 100 0 102 i
Total number of prefixes 4
```

get router info6 bgp summary

Use this command to display ipv6 BGP summary information.

Syntax

```
get router info6 bgp summary
```

Example

```
FortiADC-VM (bgp) # get router info6 bgp summary
BGP router identifier 10.0.6.217, local AS number 101
RIB entries 7, using 784 bytes of memory
Peers 2, using 9120 bytes of memory
Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd
2016::2 4 101 23 40 0 0 0 00:08:07 2
Total number of neighbors 1
```

show

Use `show` commands to display configuration settings and values. You must have read permission for the configuration object you want to display.

`show` commands display user-configured setings but not default settings; `get` commands display all settings, including both user-configured settings and defaults.

For example, you might show the current DNS settings:

```
FortiADC-VM # show system dns
config system dns
   set primary 8.8.8.8
end
```

Notice that the command does not display the setting for the secondary DNS server. This indicates that it has not been configured, or has reverted to its default value.

Like `get`, depending on whether or not you have specified an object, `show` displays one of two different outputs:

- The configuration you have just entered but not yet saved
- The configuration as it currently exists on the flash disk

For example, immediately after configuring the secondary DNS server setting but before saving it, `show` displays two different outputs. In the following example, the first output from `show` indicates the value that you have configured but not yet saved; the second output from `show` indicates the value that was last saved to disk.

```
FortiADC-VM # config system dns

FortiADC-VM (dns) # set secondary 192.168.1.10

FortiADC-VM (dns) # show
config system dns
   set primary 8.8.8.8
   set secondary 192.168.1.10
end

FortiADC-VM (dns) # show system dns
config system dns
   set primary 8.8.8.8
end
```

The `-f` option is supported in the `show` command to grep the context of the search results, for example, running `show full-configuration | grep -f port2` displays the configuration context containing port2.

If you have entered settings but cannot remember how they differ from the existing configuration, the two different forms of `show`, with and without the object name, can be a useful way to remind yourself.

If you were to now enter `end`, saving your setting to disk, `show` output for both syntactical forms would again match. However, if you were to enter `abort` at this point and discard your recently entered secondary DNS setting instead of saving it to disk, the FortiADC appliance's configuration would therefore match the second output, not the first.

When VDOMs are enabled, and if you log in as `admin`, the top level of the shell changes: the two top level items are `show global` and `show vdom`.

`show global` displays settings that only `admin` or other accounts with the **super_admin_prof** access profile can change.

`show vdom` displays each VDOM and its respective settings.

This menu and CLI structure change is not visible to non-global accounts; VDOM administrators' navigation menus continue to appear similar to when VDOMs are disabled, except that global settings such as network interfaces, HA, and other global settings do not appear.

Appendix A: Virtual domains

This appendix describes CLI commands when you use the virtual domains feature. It includes the following topics:

Overview

A Virtual Domain (VDOM) is a complete FortiADC instance that runs on the FortiADC platform. VDOM configuration objects contain all of the system and feature configuration options of a full FortiADC instance and can be used to divide a FortiADC into two or more virtual units that function independently, allowing it to support multi-tennant deployments.

The VDOM feature supports two Virtual Domain Modes that allow the VDOMs to function independently with its own networking or as administrative domains (ADOMs) with shared networking between all ADOMs. When the VDOM is in the Independent Network mode, you can provision an administrator account with privileges to access and manage only their assigned VDOM. The VDOM user can then configure their VDOM as desired untethered to other VDOMs. Alternatively, when the VDOM is in Share Network mode, it functions as an ADOM that shares the same networking interfaces and routing between all the ADOMs. The ADOM functionality enables the administrator to constrain access privileges to a subset of server load-balancing servers by defaulting all interface settings to the root ADOM.

The Virtual Domains feature is not enabled by default and requires an administrator with "super admin" or "global admin" access to enable. The **admin** account holder (also known as the "super admin") can enable and configure all VDOMs and provision accounts with "global admin" access that grants administrators permissions to enable and configure VDOMs as well. The super admin and global admin have unrestricted access to all virtual domains that have been created on the system and can provision administrator accounts to access their assigned domains.

After the Virtual Domain feature is enabled, virtual domain administrators can enter their assigned VDOM/ADOM and see a subset of the typical menus or CLI commands appear, allowing access to only the feature configurations, logs and reports specific to their VDOM/ADOM. Unlike super admin and global admin users, VDOM/ADOM administrators do not have access to global settings.

Differences between super admin/global admin, and VDOM/ADOM administrators when virtual domains are enabled:

	Super admin or global admin user	VDOM/ADOM administrators
Access to global settings (`config global`)	Yes	No
Can create administrator accounts	Yes — administrator accounts can be assigned to access other virtual domains on the system.	Yes — administrator accounts can only be assigned access to the VDOM/ADOM administrator's own virtual domain.
Can create and access all VDOMs/ADOMs	Yes	No

GUI and CLI functional availability for administrators of VDOM, root ADOM, and non-root ADOM

For administrators provisioned to access only their assigned virtual domains, the GUI and CLI functions available to them depend on their Virtual Domain Mode and whether their virtual domain is root or non-root. VDOMs configured in the Independent Network mode function independently within its own network, allowing the VDOM administrator to have full unrestricted access to all configurations within their own VDOM. Administrators of VDOMs in the Independent Network mode have full unrestricted access to all configurations within their own VDOM; as these VDOMs function independently within their own network, modifications can be made without affecting other VDOMs on the system. In contrast, administrators of ADOMs (VDOMs in Share Network mode) do not have full access to all configurations due to all ADOMs sharing the same network interfaces and routing as the root ADOM. As a result, administrators of non-root ADOMs have restricted access, partial access, or completely no access to GUI and CLI functions relating to networking.

The following table lists the difference in CLI function availability between root and non-root ADOM administrators.

Configuration		Root ADOM	Non-root ADOM
config system	interface	`set vdom` is not available since interface settings are automatically defaulted to the root ADOM.	Read-only access for interface settings. Data pulled from root ADOM.
config link-load-balance	flow-policy	Read-write access.	Read-only access. Data pulled from root ADOM.
	gateway	Read-write access.	Read-only access. Data pulled from root ADOM.
	link-group	Read-write access.	Read-only access. Data pulled from root ADOM.
	persistence	Read-write access.	Read-only access. Data pulled from root ADOM.

Configuration		Root ADOM	Non-root ADOM
	proximity-route	Read-write access.	Read-only access. Data pulled from root ADOM.
	virtual-tunnel	Read-write access.	Read-only access. Data pulled from root ADOM.
config router	access-list	Read-write access.	Read-only access. Data pulled from root ADOM.
	access-list6	Read-write access.	Read-only access. Data pulled from root ADOM.
	bgp	Read-write access.	Read-only access. Data pulled from root ADOM.
	isp	Read-write access.	Read-only access. Data pulled from root ADOM.
	md5-ospf	Read-write access.	Read-only access. Data pulled from root ADOM.
	ospf	Read-write access.	Read-only access. Data pulled from root ADOM.
	policy	Read-write access.	Read-only access. Data pulled from root ADOM.
	prefix-list	Read-write access.	Read-only access. Data pulled from root ADOM.
	prefix-list6	Read-write access.	Read-only access. Data pulled from root ADOM.
	setting	Read-write access.	Read-only access. Data pulled from root ADOM.
	static	Read-write access.	Read-only access. Data pulled from root ADOM.
config firewall	connlimit	Read-write access.	Not available.
	connlimit6	Read-write access.	Not available.
	nat-snat	Read-write access.	Not available.
	policy	Read-write access.	Not available.
	policy6	Read-write access.	Not available.
	qos-filter	Read-write access.	Not available.
	qos-filter6	Read-write access.	Not available.
	qos-queue	Read-write access.	Not available.
	vip	Read-write access.	Not available.

Configuration		Root ADOM	Non-root ADOM
config security dos	dos-protection-profile	Read-write access.	Read-write access.
	http-access-limit	Read-write access.	Read-write access.
	http-connection-flood-protection	Read-write access.	Read-write access.
	http-request-flood-protection	Read-write access.	Read-write access.
	ip-fragmentation-protection	Read-write access.	Not available.
	tcp-access-flood-protection	Read-write access.	Read-write access.
	tcp-slowdata-attack-protection	Read-write access.	Read-write access.
	tcp-synflood-protection	Read-write access.	Not available.
config global-dns-server	address-group	Read-write access.	Not available.
	dns64	Read-write access.	Not available.
	dsset-info-list	Read-write access.	Not available.
	general	Read-write access.	Not available.
	policy	Read-write access.	Not available.
	remote-dns-server	Read-write access.	Not available.
	response-rate-limit	Read-write access.	Not available.
	trust-anchor-key	Read-write access.	Not available.
	zone	Read-write access.	Not available.
config global-load-balance	analytic	Read-write access.	Not available.
	data-center	Read-write access.	Not available.
	host	Read-write access.	Not available.
	link	Read-write access.	Not available.
	servers	Read-write access.	Not available.
	setting	Read-write access.	Not available.
	topology	Read-write access.	Not available.
	virtual-server-pool	Read-write access.	Not available.

Enabling the Virtual Domain feature and selecting the Virtual Domain Mode

Before you begin:

- Save a backup of the configuration. Enabling VDOMs changes the structure of your configuration, so you want to be able to easily revert to the system state before VDOMs were enabled.

To enable the Virtual Domain and select the Virtual Domain Mode:

1. Log in with as the `admin` administrator or global administrator.
   Other administrators do not have permissions to configure VDOMs.
2. Use the following command:
```
config system global
   set vdom-admin {enable|disable}
   set vdom-mode {independent-network|share-network}
end
```

`vdom-admin`	Enable the Virtual Domain feature.
`vdom-mode`	Select either of the following virtual domain modes: • independent-network — each VDOM functions independently within its own network, unaffected by activity from other VDOMs on the system. • share-network — VDOMs function as administrative domains (ADOMs), sharing the same network interface and routing between all ADOMs.

   FortiADC terminates your administrative session.
3. Log in again.
   When VDOMs are enabled, and if you log in as `admin` or global admin, the top level of the shell changes: the two top level items are `config global` and `config vdom`.
   - `config global` contains settings that only `admin` or other accounts with the **prof_admin** access profile can change.
   - `config vdom` contains each VDOM and its respective settings.

   This menu and CLI structure change is not visible to non-global accounts; VDOM administrators' navigation menus continue to appear similarly to when VDOMs are disabled, except that global settings such as network interfaces, HA, and other global settings do not appear.
4. Continue by defining VDOMs.

Creating virtual domains

Some settings can only be configured by the `admin` administrator or global administrator — they are *global*. Global settings apply to the appliance overall regardless of VDOM, such as:

- network interfaces
- system time
- backups
- administrator accounts

- access profiles
- FortiGuard connectivity settings
- HA and configuration sync
- SNMP
- X.509 certificates
- TCP `SYN` flood anti-DoS setting
- `exec ping` and other global operations that exist only in the CLI

Only the `admin` administrator or global administrator can configure global settings.

Other settings can be configured separately for each VDOM. They essentially define each VDOM. For example, the policies of VDOM-A are separate from VDOM-B.

Initially, only the `root` VDOM exists, and it contains settings such as policies that were global before VDOMs were enabled. Typically, you will create additional VDOMs, and few if any administrators will be assigned to the `root` VDOM. After VDOMs are created, the `admin` account or global admin usually assigns other administrator accounts to configure their VDOM-specific settings. However, as the root account, the `admin` administrator does have permission to configure all settings, including those within VDOMs.

To create a VDOM:

1. Log in with the `admin` account.
   Other administrators do not have permissions to configure VDOMs.
2. Enter the following commands:

```
config vdom
edit <VDOM_name>
```

where `<VDOM_name>` is the name of your new VDOM. (Alternatively, to configure the default `root` VDOM, type `root`.

The new VDOM exists, but its settings are not yet configured.

Editing a virtual domain

For virtual domains in Independent Network mode, FortiADC allows you to create and impose custom policies or restrictions on each virtual domain you have added. You can modify the dynamic and static parameters of each VDOM by following the instructions below. Dynamic parameters determine how much of a dynamic resource, such as connections per second, a VDOM can use. Static parameters determine how much of a static resource, such as real servers, a VDOM can use.

To edit a virtual domain:

1. Enable vdom.

2. Execute the following commands. A value of 0 means the parameter has no limit.

```
config global
   config system vdom
      edit <VDOM_name>
         L4CPS : 0
         L7CPS : 0
         L7RPS : 0
         SSLCPS : 0
```

```
SSLTHROUGHPUT : 0
CONCURRENTSESSION : 0
virtualserver : 0
realserver : 0
healthcheck : 0
sourcepool : 0
errorpage : 0
localuser : 0
usergroup : 0
INBOUND : 0
OUTBOUND : 0
```

Dynamic parameters

L4CPS	The number of layer 4 connections created per second. When the creation speed exceeds this value, only this number of connections will be created per second. The rest will be dropped.
L7CPS	The number of layer 7 TCP connections created by the httproxy frontend per second. When the creation speed exceeds this value, only this number of connections will be created per second. Additional TCP syn requests will be dropped on the client side.
L7RPS	The number of HTTP GET requests handled by the httproxy from the client side per second. When the number of requests per second exceeds this value, only this number of requests will be handled. Additional HTTP GET requests will be dropped.
SSLCPS	The number of SSL connections created by the httproxy frontend per second. When the creation speed of new SSL connections exceeds this value, only this number of connections will be created per second. Additional connections will not be allowed and additional syn packets will be dropped during that second.
SSLTHROUGHPUT	The volume of SSL encrypted TCP traffic from both the incoming and outgoing side. When the traffic throughput exceeds this value, additional packets from the client will be dropped and new connections will not be allowed.
CONCURRENTSESSION	The total number of living connections for ADC traffic. Living connections include L4, L7, and L7 SSL. When the number of living connections exceeds this number, additional connections will not be allowed.
INBOUND	The maximum volume of inbound traffic allowed. Only L4 and L7 SLB TCP traffic will be counted.
OUTBOUND	The maximum volume of outbound traffic allowed. Only L4 and L7 SLB TCP traffic will be counted.

Static parameters

virtualserver	The maximum number of virtual servers that can be configured using "config load-balance virtual-server" in the chosen VDOM.
realserver	The maximum number of real servers that can be configured using "config load-balance real-server" in the chosen VDOM.

healthcheck	The maximum number of healthcheck members that can be configured using "config system health-check" in the chosen VDOM.
sourcepool	The maximum number of IP pools that can be configured using "config load-balance ippool" in the chosen VDOM.
errorpage	The maximum number of error page files that can be configured using "config load-balance error-page" in the chosen VDOM.
localuser	The maximum number of local users that can be configured using "config user local" in the chosen VDOM.
usergroup	The maximum number of user groups that can be configured using "config user user-group" in the chosen VDOM.

Assigning interfaces to a virtual domain

For virtual domains in Independent Network mode, you need to assign network interfaces to the virtual domain. If the Virtual Domain Mode is Share Network (ADOM mode), all network interface settings are defaulted to the root settings, so assigning network interfaces is unnecessary.

The following commands assign a network interface to a VDOM:

```
FortiADC-VM # config global

FortiADC-VM (global) # config system interface

FortiADC-VM (interface) # edit port10

FortiADC-VM (port10) # set vdom docs-vdom
FortiADC-VM (port10) # end
Changing interface(port10) vdom from root(1) to docs-vdom(233):
change vdom success.
```

Assigning administrators to a virtual domain

The following commands create an administrator account and assign the administrator to a VDOM or ADOM:

```
FortiADC-VM # config global

FortiADC-VM (global) # config system admin

FortiADC-VM (admin) # edit docs-vdom-admin
Add new entry 'docs-vdom-admin' for node 78

FortiADC-VM (docs-vdom-admin) # set access-profile admin_prof
FortiADC-VM (docs-vdom-admin) # set vdom docs-vdom
FortiADC-VM (docs-vdom-admin) # end
```

Disabling virtual domains

You may need to disable virtual domains in certain scenarios, such as switching to a different Virtual Domain Mode.

Before you begin:

- Save a backup of the configuration. Disabling virtual domains changes the structure of your configuration, and deletes most virtual domain related settings. It keeps settings from the `root` VDOM or ADOM only.

To disable virtual domains:

1. Assign interfaces to the root VDOM. For example:
```
FortiADC-VM # config global
FortiADC-VM (global) # config system interface
FortiADC-VM (interface) # edit port10
FortiADC-VM (port10) # set vdom root
FortiADC-VM (port10) # end
Changing interface(port10) vdom from docs-vdom(233) to root(1):
change vdom success.
```

2. Assign admin accounts to the root VDOM or delete them. For example:
```
FortiADC-VM (global) # config system admin
FortiADC-VM (admin) # delete docs-vdom-admin
FortiADC-VM (admin) # end
```

3. Delete non-root VDOMs:
```
FortiADC-VM # config vdom
FortiADC-VM (vdom) # delete docs-vdom
FortiADC-VM (vdom) # end
```

4. Disable VDOMs:

```
FortiADC-VM # config global
FortiADC-VM (global) # config system global
FortiADC-VM (global) # set vdom-admin disable
FortiADC-VM (global) # end
```

The system disables VDOMs and terminates your administrative session.

Viewing virtual domains

Use the following command to show the usage and settings for all VDOMs or ADOMs on the system:

```
get system vdom-status
```

The following example shows the system with two VDOMs set.

```
FortiADC-300D # get system vdom-status
root:
   l4cps: 4.87/-
   l7cps: 90.2/-
   l7rps: 0.0/-
```

```
SSLcps: 3.7/-
SSLThroughput(KB/S): 1550.0/-
ConcurrentSession: 47.0/-
Inbound(KB/S): 255.6/-
Outbound(KB/S): 104669.0/-
VirtualServer: 21/-
RealServer: 33/33
Health Check: 5/-
Source Pool: 0/-
Error-Page: 1/-
LocalUser: 0/-
UserGroup: 2/-
vdom1:
l4cps: 0.0/-
l7cps: 0.0/-
l7rps: 0.0/-
SSLcps: 0.0/-
SSLThroughput(KB/S): 0.0/-
ConcurrentSession: 0.0/-
Inbound(KB/S): 0.0/-
Outbound(KB/S): 0.0/-
VirtualServer: 0/-
RealServer: 0/-
Health Check: 4/-
Source Pool: 0/-
Error-Page: 0/-
LocalUser: 0/-
UserGroup: 0/-
```

The first number represents the current usage. The second number represents the limit set. A dashed line means no limit has been set.

www.fortinet.com