53-1003269-01
16 April 2015

Brocade NetIron

FIPS and Common Criteria Guide

Platform Support: Multi-Service IronWare R05.8.00a

BROCADE

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. assume no liability or responsibility to any person or entity with respect to the accuracy of this document or any loss, cost, liability, or damages arising from the information contained herein or the computer programs that accompany it.

The product described by this document may contain open source software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit *http://www.brocade.com/support/oscd*.

Document History

Title	Publication number	Summary of changes	Date
Brocade NetIron FIPS and Common Criteria Guide	*53-1003269-01*	New Document	16 April 2015

Contents

Chapter 3 **Common Criteria Certification**

Appendix A **Acknowledgments and Encrypted Syslog Server**

Appendix B **Configuring an Encrypted Syslog Server**

Appendix C **MITM Test for Common Criteria**

Appendix D **Syslog Messages**

Appendix E **Running Tasks for different NetIron devices in FIPS mode**

About This Document

In this chapter

Audience

This document is designed for system administrators with a working knowledge of Layer 2 and Layer 3 switching and routing to configure and operate the devices in FIPS or Common Criteria mode.

Supported hardware and software

The following hardware platforms are supported in this guide:

- Brocade NetIron CES 2000-4X Series and Brocade NetIron CER 2000-4X-RT Series
- Brocade MLXe series (MLXe-4, MLXe-8, and MLXe-16) with management module (BR-MLX-MR2-M or BR-MLX-MR2-X) and optional interface modules (BR-MLX-10GX20-M, BR-MLX-10GX20-X2, and BR-MLX-10GX4-IPSEC-M).

Supported software

For the complete list of supported software features and the summary of enhancements and configuration notes for this release, refer to the *Brocade NetIron Unified R05.8.00 Release Notes*.

Related publications

The *Brocade MLX Series and Brocade NetIron Family Configuration Guide* supplements the information in this guide.

Visit the Brocade website to locate related documentation for your product and additional Brocade resources.

You can download additional publications supporting your product at www.brocade.com. Select the Brocade Products tab to locate your product, then click the Brocade product name or image to open the individual product page. The user manuals are available in the resources module at the bottom of the page under the Documentation category.

To get up-to-the-minute information on Brocade products and resources, go to MyBrocade. You can register at no cost to obtain a user ID and password.

Release notes are available on MyBrocade under Product Downloads.

White papers, online demonstrations, and data sheets are available through the Brocade website.

Document conventions

This section describes text formatting conventions and important notice formats used in this document.

Text formatting

The narrative-text formatting conventions that are used are as follows:

bold text	Identifies command names
	Identifies the names of user-manipulated GUI elements
	Identifies keywords
	Identifies text to enter at the GUI or CLI
italic text	Provides emphasis
	Identifies variables
	Identifies document titles
`code` text	Identifies CLI output

For readability, command names in the narrative portions of this guide are presented in bold; for example, **show version**.

Notes, cautions, and danger notices

The following notices and statements are used in this manual. They are listed below in order of increasing severity of potential hazards.

NOTE

A note provides a tip, guidance or advice, emphasizes important information, or provides a reference to related information.

CAUTION

A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER

A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Getting technical help

To contact Technical Support, go to http://www.brocade.com/services-support/index.html page for the latest e-mail and telephone contact information.

Document feedback

To send feedback and report errors in the documentation you can use the feedback form posted with the document or you can e-mail the documentation team.

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. You can provide feedback in two ways:

- Through the online feedback form in the HTML documents posted on www.brocade.com.
- By sending your feedback to documentation@brocade.com.

Chapter

Federal Information Processing Standards 1

Table 1 lists the individual Brocade NetIron platforms that support Federal Information Processing Standard (FIPS) ready mode as detailed in FIPS Publication 140-2.

TABLE 1 Devices that support FIPS*

Brocade NetIron XMR	Brocade MLX Series	Brocade NetIron CES 2000 Series BASE package	Brocade NetIron CES 2000 Series ME_PREM package	Brocade NetIron CES 2000 Series L3_PREM package	Brocade NetIron CER 2000 Series Base package	Brocade NetIron CER 2000 Series Advanced Services package	Brocade BR-CER -2024-4 X-RT	Brocade BR-CES-202 4-4X
No	MLXe: Yes** MLX: No	No	No	No	No	No	Yes	Yes

* FIPS CLI supported on all platforms

** Except MLXe-32

This chapter contains steps for configuring FIPS ready mode on the Brocade device in compliance with standards established by the United States government and the National Institute of Standards and Technology (NIST). The sections in this chapter describe FIPS mode, how to enable and disable FIPS mode on the device, and the behavior of the device in FIPS mode.

NOTE: Starting Brocade NetIron 05.8.00 release, the NetIron devices do not support MR management module.

FIPS-supported Devices

FIPS is vendor ready for the following devices:

- Brocade MLXe-4, MLXe-8, and MLXe-16 with MR2 management modules (BR-MLX-MR2-M and BR-MLX-MR2-X)
- Brocade MLXe with management module MR2: 1666 MHz Power PC processor 7448 (version 8004/0202) 166 MHZ bus
- Brocade CER/CES (including 4X-RT models): 800 MHz Power PC processor 8544E (version 8021/0022) 400 MHz bus

NOTE
Refer to the release notes for the software version running on the device to verify that software is FIPS and Common Criteria certified.

FIPS-supported Line Cards

FIPS is vendor-ready for the following line cards:

- BR-MLX-10GX20-M
- BR-MLX-10Gx20-X2
- BR-MLX-10GX4-IPSEC-M

The following SKUs are supported for certification:

Brocade CER platform

- BR-CER-2024F-4X-RT-DC
- BR-CER-2024C-4X-RT-DC
- BR-CER-2024F-4X-RT-AC
- BR-CER-2024C-4X-RT-AC

Brocade CES platform

- BR-CES-2024C-4X-AC
- BR-CES-2024C-4X-DC
- BR-CES-2024F-4X-AC
- BR-CES-2024F-4X-DC

FIPS Security Seal Procedures

Tamper-evident security seals must be applied to the product, based on instructions in the FIPS Security Seal Procedure document for your device. The document is available at my.brocade.com.

Determine if the Brocade NetIron device is FIPS certified

Check the http://csrc.nist.gov/groups/STM/cmvp/validation.html site to determine if your Brocade NetIron device and the software it is running is FIPS certified.

Overview

FIPS are security standards developed by the United States of America government and NIST for use by all non-military government agencies and government contractors. Due to their importance within the security industry, these standards form a baseline for many security requirements.

The FIPS Publication 140-2 is a technical standard and worldwide de facto standard for the implementation of cryptographic modules.

You can configure the Brocade device to run in FIPS mode to ensure that the device is operating according to the standards stated in FIPS Publication 140-2.

NOTE
Once FIPS mode is enabled on the system, even if the mode is disabled at a later point of time, firmware integrity test will always be carried on the device at image copy time.

A Brocade device is FIPS 140-2-compliant when the following requirements have been met:

- Tamper-resistant labels are applied to the device according to the instructions included in the tamper-resistant accessory kit. The accessory kit is purchased separately.
- The device software is placed in FIPS mode with or without any FIPS security policy applied.

You place a device in FIPS mode by entering the **fips enable** command on the management station while the station is connected to the device console port with a serial cable.

In addition, configure an optional set of FIPS policy commands, and then use the **fips zeroize all** command to zero out the shared secrets used by various networking protocols, including the host access passwords, SSH and HTTPS host and client keys based on the configured FIPS Security Policy. After you issue this command, use the **write memory** command, and then place the device in FIPS Administrative mode by reloading the device. After reloading, the device goes into the “FIPS operational mode”. Before reload, the device is in the “FIPS administrative mode”.

The default FIPS policy is for the system to run in a strict mode that fully supports FIPS 140-2 specifications. However, the device allows you the flexibility to configure a modified FIPS policy according to your network requirements. A FIPS policy that varies from the default policy weakens the intent of the FIPS 140-2 specifications; when implemented, the device is not operating in full compliance with these specifications. Refer to “Modifying the FIPS policy” on page 33.

The FIPS approved mode disables the following:

- Telnet access including the **telnet server** command
- AAA authentication for the console including the **enable aaa console** command
- The **ip ssh scp disable** command
- TFTP access
- SNMP access to CSP MIB objects
- Access to SNMPv1, SNMPv2, and SNMPv3 *noAuthNoPriv* security mode
- Access to all commands that allow debugging memory content within the monitor mode
- HTTP access including the **web-management http** command (applies to Brocade MLXe series only)
- HTTPS SSL 3.0 access (applies to Brocade MLXe series only)
- The **web-management allow-no-password** command (applies to Brocade MLXe series only)

The FIPS approved mode clears the following:

- Protocol shared secret and host passwords
- SSH DSA and RSA host keys
- HTTPS RSA host keys and certificate (applies to Brocade MLXe series only)

The FIPS approved mode enables the following:

- SCP
- HTTPS TLS version 1.0 and greater (applies to Brocade MLXe series only)

After defining the FIPS policy, save the configuration, and reboot the device. While the device is booting, several tests are run to ensure the device is FIPS compliant.Some of these tests include several FIPS self-tests such as Known Answer Tests (KAT) and conditional tests that are run to ensure that the cryptographic engine is FIPS-compliant. After these tests are completed successfully, the device continues with its initialization and is operationally in FIPS mode. If any of the self-tests fails, the device attempts to reload again per the FIPS specification.

NOTE
Execution of the **self-test** command in FIPS operational or admin modes may result in the device rebooting as per the fips criteria if any of the algorithm self-test fails.

User roles in FIPS mode

A Brocade device in FIPS mode supports three roles:

- Crypto Officer Role: The Crypto Officer Role on the device in FIPS mode is equivalent to the administrator role, or the super-user, in non-FIPS mode.
- Port Configuration Administrator Role: The Port Configuration Administrator on the device in FIPS mode is equivalent to the port configuration user in non-FIPS mode and has write access to the interface configuration mode only.
- User Role: The User Role on the device in FIPS mode has read-only privileges and no configuration mode access.

In addition to the above roles, we also have the following roles that support IPsec and MACsec protocols.

- MACsec Peer Role: This specific role is available on the device. It allows MACsec key agreement protocol (MKA) sessions to be established with a remote peer based on the MACsec configuration on the Brocade NetIron device. Once the session keys (SAK) are obtained, the MACsec peer role will install the keys on the PHY and start MACsec communication with the peer.
- IPsec Peer Role: This specific role is available on the IPsec-supported line card. It allows Internet Key Exchange (IKE) and IPsec sessions to be established with a remote peer based on the IPsec configuration on the Brocade NetIron device.

Commands disabled in FIPS mode

The device in FIPS mode does not support the following commands:

- **web-management allow-no-password**
- **telnet server**
- **ip ssh scp disable**
- **ip ssh key-authentication no**
- **ip ssh permit-empty-password**
- **web-management http**
- **enable password-display**

A device in FIPS mode does not support TFTP commands, unless a FIPS policy is configured to allow TFTP access, including:

- **copy tftp flash** *ip*
- **boot system tftp** *ip file*
- **ip ssh pub-key-file tftp** *ip file* | *pubkey*
- **ip ssl certificate-data-file tftp** *ip file*
- **ip ssl private-key file tftp** *tftp file*

Cryptographic algorithms in FIPS mode

The following hardware modules in FIPS mode support the FIPS 140-2 approved cryptographic algorithms:

- "Cryptographic Algorithms on the Management module"
- "Cryptographic Algorithms on the Brocade NetIron CES and CER devices"
- "Cryptographic Algorithms on the BR-MLX-10GX4-IPSEC-M module"
- "Cryptographic Algorithms on the BR-MLX-10GX20-M and BR-MLX-10GX20-X2 modules"

NOTE
For information about specific certificate numbers assigned for the cryptographic algorithms, refer to the *Brocade NetIron FIPS 140-2 Non-Proprietary Security Policy*.

Cryptographic Algorithms on the Management module

The management module in FIPS mode supports the following FIPS 140-2 approved cryptographic algorithms:

- Advanced Encryption Algorithm (AES) including AES-CTR and AES-128-CFB
- AES Key Wrap (KW) RFC 3394
- Cipher-based MAC (CMAC) with AES 128
- Secure Hash Algorithm (this includes all SHA variants the module supports: SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512)
- Key-Based Key Derivation Functions (KBKDF SP800-108)
- Keyed-Hash Message Authentication code (HMAC-SHA1, HMAC-SHA256)
- Counter-based Deterministic Random Bit Generator (DRBG)
- Reversible Digital Signature Algorithm (RSA) including RSA2, FIPS 186-4 KeyGen, SigGen, SigVer
- Elliptic curve Digital Signature Algorithm (ECDSA) FIPS 186-4 KeyGen, SigGen, SigVer
- TLS 1.0, 1.1, and TLS 1.2 KDF SP800-135
- SSH Key exchange algorithm diffie-hellman-group-exchange-sha256
- SNMPv3 (in *authPriv* security mode) KDF SP800-135

Allowed exceptions include:

- RSA Key Wrapping
- Diffie-Hellman (DH)
- Message Digest 5 (MD5)
- Hashed Message Authentication codes - Message Digest 5 (HMAC-MD5)
- Non-Deterministic Random Number Generator (NDRNG)
- SSHv2Key Derivation Function (KDF)

The device in FIPS mode does not support the following cryptographic algorithms:

- SSH key exchange algorithms diffie-hellman-group1-sha1, diffie-hellman-group14-sha1
- SNMPv1
- SNMPv2c
- SNMPv3 in *noAuthNoPriv*, and *authNoPriv* security mode

Cryptographic Algorithms on the Brocade NetIron CES and CER devices

The Brocade NetIron CES and CER devices in FIPS mode support the following FIPS 140-2 approved cryptographic algorithms:

- SNMPv3 (in *authPriv* security mode) KDF SP800-135
- TLS 1.2 KDF SP800-135
- Advanced Encryption Algorithm (AES) including AES-CTR and AES-128-CFB128
- Secure Hash Algorithm (this includes all SHA variants the module supports: SHA-1, SHA-256, SHA-384, and SHA-512)
- Keyed-Hash Message Authentication code (HMAC-SHA1, HMAC-SHA256)
- Deterministic Random Bit Generator (DRBG) Hash based
- Reversible Digital Signature Algorithm (RSA) including RSA2
- SSH Key exchange algorithm diffie-hellman-group-exchange-sha256

Cryptographic Algorithms on the BR-MLX-10GX4-IPSEC-M module

The Brocade NetIron BR-MLX-10GX4-IPSEC-M module in FIPS mode support the following FIPS 140-2 approved cryptographic algorithms:

- IKEv2 KDF SP800-135
- ECDSA 186-4 KeyGen, SigVer, SigGen
- KAS ECC SP800-56A
- KAS FCC SP800-56A
- Hash-DRBG SP800-90
- AES (AES-256-ECB)
- GCM (SP800-38D)
- Elliptical Curve Diffie-Hellman (ECDH)

Algorithms running on the on-board security engine:

- AES (AES-128-CBC and AES-256-CBC)
- SHA (SHA-256 and SHA-384)

- HMAC

Algorithm running on the PHY crypto engine:

- AES (AES-128-GCM)

Cryptographic Algorithms on the BR-MLX-10GX20-M and BR-MLX-10GX20-X2 modules

The Brocade NetIron BR-MLX-10GX20-M and the BR-MLX-10GX20-X2 modules have an on-board PHY chip that supports the following FIPS 140-2 approved cryptographic algorithms:

- AES (AES-128-GCM)
- ECDSA
- Deterministic Random Bit Generator (DRBG)
- Component Test Key Derivation Function (CVL)
- Key Agreement Schemes (KAS)

Enhancements to public key authentication in FIPS mode

With public key authentication, a collection of clients' public keys are stored on the device. Clients are authenticated using these stored public keys. Only clients that have a private key that corresponds to one of the stored public keys can gain access to the device using SSH server.

Setting up public key authentication consists of the following steps:

1. Importing authorized public keys into the device.
2. Enabling public key authentication
3. Username restrictions

The client public key file format allows for a username to be provided in the "Subject:" field. Additional private headers can be used for specifying the privilege level we will use the x-brocade-privilege-level:" header. The privilege level can take on 3 values (0 READ-WRITE/ADMINISTRATOR, 4 PORT-CONFIG, 5 READ-ONLY)

Following example public key displays the headers that will be used by the device.

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20121206"
Subject: brcd
x-brocade-privilege-level: 0
AAAAB3NzaC1yc2EAAAABJQAAAQEAkwiApY1x4T/DHII5JzR2OgqcF5vjlubNcvSE
UjkGmiRBDSOicjxS0ZLm1b2xFpVzw8XxSSy8cxvntfs5ortOt80QzynqgL+H2zJa
Lb4Qbu6/1vakJbPb/VUJE66Zezh0c8mze6zTbiP4iQ/Wn2lxpSmlS5cdowmFlZ7B
97xcagJIBl+7JKuvj8P+85ESUf2/pcrogqx7gdr1IpP2nev5s4xwCWFGtr2R/yMF
Q9h0xLcc4A7vLTDuY/h1GzLdICgtNYdqpUhpw+w0DkTKbQuDPd0gkwHkoFwg85lE
4VCDevdC/DeOCNJjNp9NbVD+SW6uL4NymmV7/i0YbPyl3gTESQ==
---- END SSH2 PUBLIC KEY ----
```

NOTE
No exec authorization through AAA server is available because the privilege level is obtained from the public key file private header field (x-brocade-privilege-level) as mentioned above.

For additional information on configuring public keys, refer to the *NetIron Family Configuration Guide.*

Protocol changes in FIPS mode

Table 2 lists the protocols that undergo changes while the device is in FIPS mode with the default policy applied.

TABLE 2 Protocol changes

Protocols/ Algorithms	Supported in FIPS mode	Supported in Non-FIPS mode	For more information on individual protocol changes, refer to the following sections:
BGP	Yes	Yes	"BGP" on page 14
HTTP	No	Yes	"HTTP" on page 15
HTTPS	Yes, with limitations	Yes	"HTTPS" on page 15
IPsec	Yes, with limitations	Yes	"IKEv2/ IPsec" on page 16
IS-IS	Yes	Yes	"IS-IS" on page 16
MACsec	Yes	Yes	"MACsec" on page 17
MPLS	Yes	Yes	"MPLS" on page 17
NTP	Yes, with limitations	Yes	"NTP" on page 17
OSPFv2	Yes	Yes	"OSPFv2" on page 18
OSPFv3	Yes	Yes	"OSPFv3" on page 18
Proprietary 2-way encryption algorithms	No	Yes	"Proprietary 2-way encryption algorithms" on page 18
RADIUS	Yes, with limitations	Yes	"RADIUS" on page 18
SCP	Yes	Yes	"SCP" on page 19
SNMP	Yes, with limitations	Yes	"SNMP" on page 20
SSHv2	Yes, with limitations	Yes	"SSHv2" on page 21
TACACS+	Yes, with limitations	Yes	"TACACS+" on page 22
Telnet	No	Yes	"Telnet" on page 22
TFTP	No	Yes	"TFTP" on page 22

BGP

Border Gateway Protocol (BGP) allows peer-to-peer authentication or client-to-server authentication. To authorize an authentication, use a command such as the following to configure shared secret keys for BGP:

```
device(config)# neighbor 202.55.144.7 password jdoepass
```

Syntax: **[no] neighbor** <*ip-addr*> | <*peer-group-name*> **password** <*string*>

For more information on BGP authentication commands, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

HTTP

HTTP is not supported on the device in FIPS mode.

The **web-management http** command is disabled if it is included in the device's configuration. When the HTTP server is enabled because the **web-management http** command has been configured, the system removes the command from the configuration and the device displays the following messages:

```
FIPS Compliance: HTTP service will been disabled
```

HTTPS continues to be enabled in FIPS mode and the configuration changes the **web-management http** command to the **web-management https** command.

HTTPS

The following HTTPS configurations are affected in FIPS mode:

- The **web-management https** command is maintained and offers equivalent functionality to the disabled **web-management http** command.
- The **web-management allow-no-password** command is disabled.
- The **ip ssl certificate-data-file tftp** command is disabled when TFTP operation is disabled in FIPS mode. SCP supports this command's functionality. Refer to "SCP" on page 19.
- The **ip ssl private-key-file tftp** command is disabled when TFTP operation is disabled in FIPS mode. SCP supports the functionality of this command. Refer to "SCP" on page 19.
- The **crypto-ssl certificate zeroize** command zeros out the RSA key pair and removes the digital certificate.
- SSL version 3 and earlier versions are disabled and TLS 1.0 or later versions are enabled.

The FIPS 140-2 cipher suites consist of the following algorithms:

- Triple DES (FIPS 46-3) or AES (FIPS 197) for symmetric key encryption and decryption.
- Secure Hash Standard (SHA-1, SHA-256, SHA-384, and SHA-512) (FIPS 180-2) for hashing.
- HMAC (FIPS 198) for keyed hash.
- Random number generator CTR (AES 256) DRBG (NIST SP800-90).
- Diffie-Hellman, Ephemeral Diffie-Hellman, or Key Wrapping using RSA keys for key establishment.
- RSA (PKCS #1 v2.1), or ECDSA (ANSI X9.62) for signature generation and verification.

The following cipher suites are allowed in FIPS mode:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA

The cipher suite TLS_RSA_WITH_AES_256_CBC_SHA is the default cipher suite.

TLS implementation in NetIron devices

For devices which act as a SSL server or HTTPS server, the default connection is with TLS1.2. For devices which acts as a SSL client or Syslog/OpenFlow/Secure AAA client, during session negotiation, the TLS version is decided based on the server support. TLS version negotiation starts with the highest number and goes down to the optionally configured minimum TLS version.

You can configure the minimum TLS version on NetIron devices using the following command:

ip ssl server min-version {0 | 1 | 2}

IKEv2/ IPsec

The BR-MLX-10Gx4-IPSEC-M interface module supports creation of virtual private network (VPN) using the IPsec protocol. The IKEv2 protocol is used to negotiate the IPsec service parameters for the VPN.

NOTE
The IKE/IPsec protocol supports importing pre-shared keys (PSK) using the **pki import Brocade_CA pem url flash:** *name* command:

IPsec Critical Security Parameters

The IPsec critical security parameters are listed below:

- Password for MM
- ECDSA Private Key
- ECDH Private Key
- ECDH Shared Secret
- IKE Encrypt/Decrypt
- IKE Authentication Key
- ESP Encrypt/Decrypt
- IKE KDF State
- Pre-Shared Key (PSK)
- DRBG State
- Entropy Data

IS-IS

ISIS allows peer-to-peer authentication or client-to-server authentication. To authorize an authentication, use commands such as the following to configure shared secret keys for ISIS:

```
device(config)# auth-mode md5 level-1
```

Syntax: [no] auth-mode md5 [level-1 | level-2]

```
device(config)# auth-key jdoepass level-1
```

Syntax: [no] auth-key <*string*> **[level-1 | level-2]**

For more information on ISIS authentication commands, refer to *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

MACsec

MACsec standard consists of two main components:

- MACsec (MAC security)
- MKA (MACsec key agreement protocol)

MKA protocol defined as part of IEEE 802.1x-2010 standard is responsible for generating the secret keys (SAKs) used by MACsec for symmetric cryptography. This protocol runs on the management card in the control plane.

MACsec protocol is used for securing communication among the trusted components of a 802.1 LAN.

When MACsec is used to secure the communication between endpoints on a LAN, each packet on the wire is encrypted in the PHY in the data plane using symmetric key cryptography so that communication cannot be monitored or altered on the wire.

MACsec Critical Security Parameters

The MACsec critical security parameters are listed below:

- CAK (Connectivity Association Key either configured manually by user or derived from the MSK obtained from authentication server.)
- CKN (CAK name configured manually by user)
- ICK (Integrity check key)
- KEK (Key encryption key)
- SAK (Secure Association Key used for encryption/decryption of the traffic. This key is derived from the CAK.)

MPLS

Multi Protocol Label Switching (MPLS) allows peer to peer authentication or client to server authentication. To authorize an authentication, use commands such as the following to configure shared secret keys for MPLS.

For MPLS RSVP:

```
device(config)# rsvp-authentication jdoepass
```

Syntax: **[no] rsvp-authentication key** <*string*>

For MPLS LDP:

```
device(config)# session 10.10.10.3 key jdoepass
```

Syntax: [no] **session** <*remote-ip-addr*> **key** <*string*>

For more information on MPLS authentication commands, refer the to *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

NTP

Brocade NetIron FIPS devices support Network Time Protocol (NTP) using SHA1.

device (config-ntp)# authentication-key key-id 1 sha1

Syntax: **[no] authentication-key key-id** *decimal* **sha1**

NOTE
FIPS and CC mode does not support MD5 since MD5 is not a supported hash algorithm.

OSPFv2

OSPF allows peer-to-peer authentication or client-to-server authentication. To authorize an authentication, use commands such as the following to configure shared secret keys for OSPFv2:

```
device(config)# ip ospf authentication-key jdoepass
```

Syntax: **ip ospf authentication-key** <*string*>

```
device(config)# ip ospf md5-authentication key-id 5 key jdoepass
```

Syntax: **ip ospf md5-authentication key-id** <num> **key** <*string*>

```
device(config)# area 1 virtual-link 209.157.22.1 md5-authentication key-id 5 key
jdoepass
```

Syntax: **[no] area** <*ip-addr*> | <*num*> **virtual-link** <*router-id*> **[authentication-key** <*string*>|
md5-authentication key-id <*num*> **key** [0|1] <*string*>]

For more information on OSPF authentication commands, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

OSPFv3

The OSPFv3 protocol uses IPsec with IP ESP and HMAC-SHA-196, and is allowed in FIPS mode.

To authorize an authentication, use commands such as the following to configure shared secret keys for OSPFv3:

```
Brocade(config)#ipv6 ospf authentication ipsec spi %u esp sha1 encrypt jdoepass
```

Syntax: **[no] ipv6 ospf authentication ipsec spi** <*spinum*> **esp sha1 [no-encrypt]** <*key*>

For more information on OSPFv3 authentication commands, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

Proprietary 2-way encryption algorithms

The routing protocols OSPFv2, ISIS, BGP, MPLS LDP, RSVP and the management protocols SNMP and SNTP save authentications parameters using one of the following two proprietary algorithms:

- Global encoding scheme
- Base 64 encoding scheme

These proprietary algorithms are not supported in FIPS mode. When the default FIPS policy is applied, the commands are zeroized.

RADIUS

HMAC-MD5 authentication used in RADIUS is allowed in FIPS mode.

RADIUS allows peer-to-peer authentication or client-to-server authentication. To authorize an authentication, use commands such as the following to configure shared secret keys for RADIUS:

```
device(config)# radius-server host 1.2.3.4 auth-port 1812 acct-port 1813 default
key jdoepass
```

Syntax: [no] **radius-server host** <*ip-addr*> | <*server-name*> [**auth-port** <*number*> **acct-port** <*number*> [**authentication-only accounting-only default**] [**key** [**0 1 2**] <*string*> [**dot1x**]]]

For more information on RADIUS authentication commands, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

SCP

Table 3 lists the Secure Copy (SCP) commands that are available to compensate for equivalent existing functionality of TFTP commands disabled in FIPS mode.

TABLE 3 Corresponding TFTP and SCP commands

Command functionality	TFTP commands not allowed in FIPS mode	SCP commands with corresponding functionality in FIPS mode
Import a digital certificate	**ip ssl certificate-data-file tftp** <*ip-address*> <*certificate-filename*>	**scp** <*certificate-filename*> <*user*>@<*ip-address*>:**sslCert**
Import an RSA private key from a client	**ip ssl private-key-file tftp** <*ip-address*> <*key-filename*>	**scp** <*key-filename*> <*user*>@<*ip-address*>: **sslPrivKey**
Load a public key file from a client	**ip ssh pub-key-file tftp** <*ip-address*> <*key-filename*>	**scp** <*key-filename*> <*user*>@<*ip-address*>: **sshPubKey**

Importing a digital certificate

To import a digital certificate using SCP, enter a command such as the following one:

```
C# scp certfile user@192.168.89.210:sslCert
```

Syntax: **scp** <*certificate-filename*> <*user*>@<*ip-address*>:**sslCert**

NOTE
This command is not supported on NetIron CER devices.

The <*ip-address*> variable is the IP address of the server from which the digital certificate file is downloaded.

The <*certificate-filename*> variable is the file name of the digital certificate that you are importing to the device.

The functionality of the **scp** command is equivalent to that of the disabled **ip ssl certificate-data-file tftp** command.

For more information on the **scp** command, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

Importing an RSA private key from a client

To import an RSA private key from a client using SCP, enter a command such as the following one:

```
C# scp keyfile user@192.168.9.210:sslPrivKey
```

Syntax: **scp** <*key-filename*> <*user*>@<*ip-address*>*:* **sslPrivKey**

NOTE
This command is not supported on NetIron CER devices.

The <*ip-address*> variable is the IP address of the server that contains the private key file.

The <*key-filename*> variable is the file name of the private key that you want to import into the device.

The functionality of the **scp** command is equivalent to that of the disabled **ip ssl private-key-file tftp** command.

For more information on the **scp** command, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

Loading a public key file from a client

To load a public key file from a client using SCP, enter a command such as the following one:

```
C# scp pkeys.txt user@192.168.1.234:sshPubKey
```

Syntax: **scp** <*key-filename*> <*user*>@<*ip-address*>:**sshPubKey**

The <*ip-address*> variable is the IP address of the server that contains the public key file.

The <*key-filename*> variable is the name of the public key file that you want to import into the device.

The functionality of the **scp** command is equivalent to the disabled **ip ssh pub-key-file tftp** command.

For more information on the **scp** command, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

SNMP

In the FIPS mode of operation, the device uses the existing SNMPv3 configuration. However, MIB objects related to keys and passwords output NULL or a 0 value. Refer to “SNMP CSP objects” on page 21.

NOTE
SNMPv1 and SNMPv2c versions are not allowed in FIPS mode. Access is allowed only for SNMPv3 configuration with *authPriv* mode. Other security modes such as *noAuthNoPriv* and *authNoPriv* are not allowed.

SNMP Notification

In FIPS or CC mode of operation, the Brocade NetIron device will generate only SNMPv3 notifications, if it has to be configured for SNMPv3 host in *authPriv* security mode. As a result, both authentication and privacy are configured for a given SNMP target.

Syntax: **[no] snmp-server host** <*ip*> **version v3 priv** <*user*>

NOTE
The device does not validate any configuration of **snmp-server host** command to ensure SNMPv3 *authPriv* configuration. During the notification generation instance, the system goes through the configured SNMP host list and sends notification to only those hosts that have SNMPv3 with *authPriv* security mode.

SNMP CSP objects

The following SNMP MIB objects represent the Critical Security Parameter (CSP) entities that are restricted in FIPS mode:

Enterprise MIB objects:

- snRadiusKey
- snRadiusServerRowKey
- snTacacsKey
- snTacacsServerRowKey
- snVrrpIfAuthPassword
- snAgGblPassword
- snAgGblReadOnlyCommunity
- snAgGblReadWriteCommunity
- snAgGblTelnetPassword
- snAgentUserAccntPassword

Standard MIB objects:

- rip2IfConfAuthKey
- vrrpOperAuthKey
- dvmrpInterfaceKey
- ospfIfAuthKey
- ospfVirtIfAuthKey

SSHv2

Secure Shell version 2 (SSHv2) is allowed in FIPS mode.

The following SSH configurations are affected when the Brocade device is in FIPS mode:

- The SSH server is always enabled; however, to start it, use the **crypto key generate** command to create host keys.
- The **ip ssh aes-only** command is enforced. During SSH connection, encryption is done using AES 256 or AES 128, depending on client's capability.
- The **ip ssh key-authentication** command is disabled.
- The **ip ssh permit-empty-password** command is disabled.
- The **ip ssh pub-key-file tftp** command is disabled.
- The **ip ssh scp** command ensures that SCP is enabled to run in FIPS mode. SCP is needed for file communication and the **ip ssh scp disable** command is disabled in FIPS mode and displays the following message:

  ```
  FIPS Compliance: SCP needs to be enabled
  ```

- The **crypto key zeroize** command removes configured SSH keys.

Use the command **show ip ssh config** to display SSH configuration information. For more information on the **show ip ssh config** command, refer to the *Brocade NetIron Security Configuration Guide*.

SSH key generation time is affected by the increased security of authentication and encryption algorithms both in and out of FIPS mode. Refer to the Brocade NetIron configuration guides in my.brocade.com. for SSH key generation time ranges.

The **ip ssh password-authentication [no | yes]** command is used to disable the password authentication for SSH. The **ip ssh interactive-authentication [no | yes]** command is used to disable the interactive authentication for SSH. For more information about these commands, refer to the *Brocade NetIron Security Configuration Guide*.

The SSH cipher list is given below:

Brocade NetIron release	SSH cipher supported
Pre-5.8 FIPS mode	aes256-cbc, aes128-cbc
5.8 FIPS mode	aes256-ctr, aes192-ctr, aes128-ctr, aes256-cbc,aes192-cbc, and aes128-cbc
5.8 JITC mode	aes256-ctr, aes192-ctr, and aes128-ctr
5.8 CC mode	aes256-cbc and aes128-cbc

TACACS+

HMAC-MD5 packet encryption used in TACACS+ is allowed in FIPS mode.

TACACS+ allows peer-to-peer authentication or client-to-server authentication. To authorize an authentication, use commands such as the following to configure shared secret keys for TACACS+:

```
device(config)# tacacs-server host 1.2.3.4 auth-port 49 accounting-only
```

Syntax: [no] tacacs-server host <ip-addr> **|** <server-name> **[auth-port** <*number*> **accounting-only]**

For more information on TACACS+ authentication commands, refer to the *Brocade MLX Series and Brocade NetIron Family Configuration Guide*.

Telnet

Telnet is disabled in FIPS mode as part of the default FIPS policy on the device. Attempts to start the Telnet server fail in FIPS mode.

TFTP

The following TFTP commands are disabled and return an error when TFTP operation is not allowed on the device in FIPS mode:

- All **copy tftp** commands
- **boot system tftp** <*ip-address*> <*filename*>
- **boot system auxiliary flash** <*file*>

The following TFTP commands are disabled. Use SCP commands with equivalent functionality instead. Refer to "SCP" on page 19.

- **ip ssl certificate-data-file tftp** <*ip-address*> <*certificate-filename*>
- **ip ssl private-key-file tftp** <*ip-address*> <*key-filename*>
- **ip ssh pub-key-file tftp** <*ip-address*> <*key-filename*>

System reset and boot in FIPS mode

Firmware digital signature verification and POST testing takes place as the device progresses through the boot sequence.

The following actions and limitations take effect when the device is operationally in FIPS mode according the FIPS default policy:

- Boot from TFTP or Auxiliary flash card is disabled.
- Monitor mode memory access command set is disabled. Configure an alternative FIPS policy to the default policy to access the command set. Refer to "Modifying the FIPS policy" on page 33.
- Boot up interruption is disabled with the exception of the option to access monitor mode during the boot sequence. Refer to "Accessing monitor mode in the event of continuous failure" on page 40.
- Access to memory test mode is disabled.
- Debug commands are disabled from the application prompt in FIPS mode.

Debugging in FIPS mode

The device reloads automatically when it encounters a system reset and enters FIPS failure state. The cause of failure logs on the console and the device performs a self-reboot.

You can conduct debugging in monitor mode when a flexible FIPS policy is applied on the device and in the event of continuous failure. Refer to "Access to monitor mode" on page 39.

Placing the device in FIPS mode

Placing the device in FIPS mode is a multiple step process. The general steps are as follows:

1. Copy the signatures files (Optional step. Use names in Table 4, or Table 5).
2. Enable FIPS Mode. Refer to“Enabling FIPS mode” on page 27
3. Modify the default FIPS Policy. (Optional step. Refer to“Modifying the FIPS policy” on page 33)
4. For strict FIPS mode of operation, zeroize shared secrets and host keys. Refer to“Clearing shared secrets and host keys” on page 34.
5. Save the configuration. Refer to“Saving the configuration” on page 36
6. Reload the device. Refer to “Reloading the device” on page 36

Copying the signature files

Copy the needed signature files. Refer to the “FIPS” chapter in the *Brocade NetIron Software Upgrade Guid*e for the required signature file information.

For the Brocade MLXe devices, the following signature files need loaded to the management module with specific destination file names.

NOTE
Wherever the ".sig" extension appears in the source file name it could mean to use either ".sig", or ".sha256". ".sig" is meant if the device is currently running 5.6a or older code. ".sha256", if 5.6aa or newer.

TABLE 4 Required signature files for the Brocade MLXe devices

Image name on flash	Image type	Signature source file name	Signature destination file name	RSA2048/SHA256 bit signature source file name
primary	Management Application	xmrXXXXX.sig	primary.sig	xmrXXXXX.sha256
secondary			secondary.sig	
Monitor	Management Monitor	xmbXXXXX.sig	monitor.sig	xmbXXXXX.sha256
lp-monitor	Interface Module Monitor	xmlbXXXXX.sig	lp-mon.sig	xmlbXXXXX.sha256
lp-primary-0	Interface Module Application	xmlpXXXXX.sig	lp-pri.sig	xmlpXXXXX.sha256
lp-secondary-0			lp-sec.sig	

The NetIron CER devices requires the following signature files need loaded to the management module with specific destination file names.

TABLE 5 Required signature files for the NetIron CER devices

Image name on flash	Image type	Signature source file name	Signature destination file name	RSA2048/SHA256 bit signatures file name
primary	Management Application	ceXXXXX.sig	primary.sig	ceXXXXX.sha256
secondary			secondary.sig	
Monitor	Management Monitor	cebXXXXX.sig	monitor.sig	cebXXXXX.sha256

NOTE
These signature files are specific to the version of the images in currently in code flash of the device.

NOTE
The **fips policy allow tftp-access** command must be enabled if FIPS is enabled, and using the TFTP commands.

For Brocade NetIron MLXe devices

1. Place the needed signature files on an accessible SCP or TFTP server.
2. Copy the management monitor image signature file by entering one of the following commands:
   - Using SCP on a remote client:

     ```
     C:> scp xmb<xxxxx>.sig <user>@<device-IpAddress>:flash:monitor.sig
     ```

   - Using TFTP at the Privileged EXEC level of the CLI:

     ```
     copy tftp flash <tftp-srvr> xmb<xxxxx>.sig monitor.sig
     ```
3. Copy the interface module monitor image signature file by entering one of the following commands:
   - Using SCP on a remote client:

     ```
     C:> scp xmlb<xxxxx>.sig <user>@<device-IpAddress>:flash:lp-mon.sig
     ```

   - Using TFTP at the Privileged EXEC level of the CLI:

     ```
     copy tftp flash <tftp-srvr> xmlb<xxxxx>.sig lp-mon.sig
     ```
4. Copy the interface module application image signature file by entering one of the following commands:
   - Using SCP on a remote client:

     ```
     C:> scp xmlp<xxxxx>.sig <user>@<device-IpAddress>:flash:[lp-pri.sig |
     lp-sec.sig]
     ```

   - Using TFTP at the Privileged EXEC level of the CLI:

     ```
     copy tftp flash <tftp-srvr> xmlp<xxxxx>.sig [lp-pri.sig | lp-sec.sig]
     ```

5. Copy the management module application image signature file by entering one of the following commands:
   - Using SCP on a remote client:

     ```
     C:> scp xmr<xxxxx>.sig <user>@<device-IpAddress>:flash:[primary.sig |
     secondary.sig]
     ```

   - Using TFTP at the Privileged EXEC level of the CLI:

     ```
     copy tftp flash <tftp-srvr> xmr<xxxxx>.sig [primary.sig | secondary.sig]
     ```

For Brocade CES/CER 2000-4X devices

1. Place the needed signature files on an accessible SCP or TFTP server.
2. Copy the management monitor image signature file by entering one of the following commands:
   - Using SCP on a remote client:

     ```
     C:> scp ceb<xxxxx>.sig <user>@<device-IpAddress>:flash:monitor.sig
     ```

   - Using TFTP at the Privileged EXEC level of the CLI:

     ```
     copy tftp flash <tftp-srvr> ceb<xxxxx>.sig monitor.sig
     ```

3. Copy the management module application image signature file by entering one of the following commands:

- Using SCP on a remote client:

  ```
  C:> scp ce<xxxxx>.sig <user>@<device-IpAddress>:flash:[primary.sig |
  secondary.sig]
  ```

- Using TFTP at the Privileged EXEC level of the CLI:

  ```
  copy tftp flash <tftp-srvr> ce<xxxxx>.sig [primary.sig | secondary]
  ```

4. Copy application image file by entering one of the following commands:

- Using SCP on remote client:

  ```
  scp ce<xxxxx>.bin <user>@<device-IpAddress>:flash:primary
  ```

- Using TFTP at the privileged EXEC Level of the CLI:

  ```
  copy tftp flash <tftp-srvr> ce<xxxxx>.bin [primary | secondary]
  ```

FIPS self test

The FIPS self test will verify that the signatures match the application and monitor images for the management and interface modules that are loaded onto the device.

NOTE
The **fips self-tests** command should only be executed prior to the device being placed into FIPS operational or admin modes. In the FIPS operational or admin modes, if any self test fails, this command results in restarting the device as per FIPS criteria.

1. From Privileged EXEC level of the CLI on the console, execute **fips self-test** to verify that the FIPS Known Answer Test and Conditional Tests pass.

In the following example, the FIPS Known Answer Test and Conditional Tests passed:

```
device# fips self-tests
WARNING: Issuing of this command may result in your device reloading.
WARNING: Please verify firmware images are installed correctly first.
Are you sure? (enter 'y' or 'n'): y
FIPS Power On Self Tests and KAT tests successful.
Running continuous DRBG check.
Running continuous DRBG check successful.
Pairwise consistency check successful.

FIPS KAT and Conditional Tests... PASSED
```

NOTE
This check must pass before saving the configuration and reloading the device.

The following log message is only output to the console terminal and no trap messages are generated as the system is not fully operational when this event happens:

```
“Crypto module initialization and Known Answer Test (KAT) passed”
```

Enabling FIPS mode

1. Attach a management station (PC or terminal) to the management module serial (console) port using a serial cable.

   When the device is not in a console session, FIPS-related commands return errors.

2. Verify that the device is in non-FIPS mode using the following command:

   ```
   device(config)#fips show
   ```
   Syntax: fips show

   The **fips show** command lists the current configuration of the device and can be run in both FIPS and non-FIPS modes to establish whether the device is truly in FIPS mode.

   The output of the **fips show** command confirms that the device is in FIPS mode and identifies the device as either administratively or operationally in FIPS mode.

The following example shows the output of the **fips show** command before the **fips enable** command is entered, and administrative status is off and operational status is off:

```
device# fips show
Not a FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0

FIPS mode   : Administrative status OFF: Operational status OFF
FIPS CC mode: Administrative status OFF: Operational status OFF
```

If the device is already in administrative FIPS mode, you can modify the FIPS policy. Refer to "Modifying the FIPS policy" on page 33.

3. Assume the Crypto-officer role and log into the Brocade device.
4. Copy the signature files of all affected images to the flash memory.
5. Use the following command to place the device administratively in FIPS mode:

```
device(config)#fips enable

WARNING: This will enable FIPS on this device. Please refer

       : to the NetIron Federal Information Processing Standards Guide for

      : more details. Also, be advised that Software/Firmware Integrity checks

       : will always be performed on this device on subsequent reloads, even

       : if FIPS or Common Criteria is disabled in the future.

Are you sure? (enter 'y' or 'n'): y
```

Syntax: [no] fips enable

The following example shows the output of the **fips enable** command on Brocade MLX Series and Brocade NetIron XMR devices:

```
device(config)#fips enable
WARNING: This will enable FIPS on this device. Please refer
       : to the NetIron Federal Information Processing Standards Guide for
       : more details. Also, be advised that Software/Firmware Integrity checks
       : will always be performed on this device on subsequent reloads, even
       : if FIPS or Common Criteria is disabled in the future.
Are you sure? (enter 'y' or 'n'): y
This device is now running in FIPS administrative mode.
At this time you can alter this system's FIPS default security policy
and then enter FIPS operational mode.

Note: Making changes to the default FIPS security policy weakens
the security of the device and makes the device non-compliant with
FIPS 140-2 Level 2, design assurance Level 3
The default security policy defined in the FIPS
Security Policy Document ensures that the device complies with all
FIPS 140-2 specifications. Commands to alter the default security policy
are available to the crypto-officer; however, Brocade does not recommend
making changes to the default security policy at any time.
=====================================

To enter FIPS mode, complete the following steps:
1. Optionally, configure FIPS policy commands that meets your network
   requirements. You must explicitly configure the following services if you want
   to use them when the device is operational in FIPS mode:

      - Allow TFTP access.
          Current status: Enabled
      - Allow SNMP Access to the Critical Security Parameter (CSP) MIB objects.
          Current status: Disabled
      - Allow access to all commands within the monitor mode.
          Current status: Disabled
      - Allow cleartext password display in some commands.
          Current status: Disabled
      - Retention of shared secret keys for all protocols and the host passwords.
          Current status: Clear
      - Retention of HTTPS RSA host keys and certificate.
          Current status: Retain

2. Enter the "fips zeroize all" command, which zeroes out the shared secrets
    used by various networking protocols, including the host access passwords,
    SSH and HTTPS host-keys with the digital signature based on the configured
    FIPS Security Policy.
3. Save the running configuration.
4. Reload the device.
5. Enter the "fips show" command to verify that the device entered
  FIPS or CC operational mode.
=====================================

The system will disable the following services or commands after reload:
1. Telnet server will be disabled. The "telnet server" command will be removed.
2. SSL Client will be enabled.
3. SCP will be enabled. The "ip ssh scp disable" command will be removed.
4. Configuration "boot system {slot1|slot2} <file>" will be removed as FIPS mode
does not allow system to boot from Storage Card.
5. Configuration "lp boot system {slot1|slot2} <file> <slot>" will be removed as
FIPS mode does not allow system to boot from Storage Card.
6. Configuration "boot system tftp <ip> <file>" will be removed as FIPS mode does
not allow system to boot from TFTP.
7. Configuration "enable password-display" will be removed.
8. HTTP server will be disabled. The "web-management http" command will be
removed.
9. HTTPS server will change as follows:
      -SSL 3.0 will be disabled.
      -TLS version 1.0 and greater will be used.
      -RC4 cipher will be disabled.[Latent functionality RC4 is no longer
supported]
      -Passwords will be required; the "web-management allow-no-password"
        command will be removed.
Passwords/Keys which dont comply FIPS standards will be removed on reload.
Please see FIPS config guide for complete details.


=====================================
Additionally, in FIPS  only operational mode, the system will have the following
restrictions
FIPS1. Configuration for CLI logging "logging cli-command" will be removed.

      - Start SSL client task for secure syslog server.
          Current status: Enabled
```

The following example shows the output of the **fips enable** command on CER devices.

```
device(config)#fips enable
WARNING: This will enable FIPS on this device. Please refer
       : to the NetIron Federal Information Processing Standards Guide for
       : more details. Also, be advised that Software/Firmware Integrity checks
       : will always be performed on this device on subsequent reloads, even
       : if FIPS or Common Criteria is disabled in the future.
Are you sure? (enter 'y' or 'n'): y
This device is now running in FIPS administrative mode.
At this time you can alter this system's FIPS default security policy
and then enter FIPS operational mode.

Note: Making changes to the default FIPS security policy weakens
the security of the device and makes the device non-compliant with
FIPS 140-2 Level 2, design assurance Level 3
The default security policy defined in the FIPS
Security Policy Document ensures that the device complies with all
FIPS 140-2 specifications. Commands to alter the default security policy
are available to the crypto-officer; however, Brocade does not recommend
making changes to the default security policy at any time.
=====================================

To enter FIPS mode, complete the following steps:
1. Optionally, configure FIPS policy commands that meets your network
   requirements. You must explicitly configure the following services if you want
   to use them when the device is operational in FIPS mode:

      - Allow TFTP access.
          Current status: Enabled
      - Allow SNMP Access to the Critical Security Parameter (CSP) MIB objects.
          Current status: Disabled
      - Allow access to all commands within the monitor mode.
          Current status: Disabled
      - Allow cleartext password display in some commands.
          Current status: Disabled
      - Retention of shared secret keys for all protocols and the host passwords.
          Current status: Clear
      - Retention of HTTPS RSA host keys and certificate.
          Current status: Retain

2. Enter the "fips zeroize all" command, which zeroes out the shared secrets
    used by various networking protocols, including the host access passwords,
    SSH and HTTPS host-keys with the digital signature based on the configured
    FIPS Security Policy.
3. Save the running configuration.
4. Reload the device.
5. Enter the "fips show" command to verify that the device entered
  FIPS or CC operational mode.
=====================================

The system will disable the following services or commands after reload:
1. Telnet server will be disabled. The "telnet server" command will be removed.
2. SSL Client will be enabled.
3. SCP will be enabled. The "ip ssh scp disable" command will be removed.
4. Configuration "boot system {slot1|slot2} <file>" will be removed as FIPS mode
does not allow system to boot from Storage Card.
5. Configuration "lp boot system {slot1|slot2} <file> <slot>" will be removed as
FIPS mode does not allow system to boot from Storage Card.
```

```
6. Configuration "boot system tftp <ip> <file>" will be removed as FIPS mode does
not allow system to boot from TFTP.
7. Configuration "enable password-display" will be removed.
8. HTTP server will be disabled. The "web-management http" command will be
removed.
9. HTTPS server will change as follows:
      -SSL 3.0 will be disabled.
      -TLS version 1.0 and greater will be used.
      -RC4 cipher will be disabled.[Latent functionality RC4 is no longer
supported]
      -Passwords will be required; the "web-management allow-no-password"
        command will be removed.
Passwords/Keys which dont comply FIPS standards will be removed on reload.
Please see FIPS config guide for complete details.


=====================================
Additionally, in FIPS  only operational mode, the system will have the following
restrictions
FIPS1. Configuration for CLI logging "logging cli-command" will be removed.

      - Start SSL client task for secure syslog server.
          Current status: Enabled
device#
```

6. You can verify the status of the device as administrator in FIPS mode by using the **fips show** command.

   The following example shows the output of the **fips show** command on the Brocade MLX Series devices after the **fips enable** command is entered and administrative status is on and operational status is off:

```
device# fips show
FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0
FIPS mode   : Administrative status ON: Operational status ON
FIPS CC mode: Administrative status OFF: Operational status OFF

System Specific:
OS monitor access status is: Disabled

Management Protocol Specific:
Telnet server                  : Disabled
Telnet client                  : Disabled
TFTP client                    : Disabled
HTTPS SSL 3.0                  : Disabled
SNMP v1, v2, v2c       : Disabled

SNMP Access to security objects: Disabled
Password Display               : Disabled
Critical security Parameter updates across FIPS boundary:
(i.e. during "fips zeroize" ..., or "no fips enable")   :
Protocol Shared secret and host passwords: Clear
SSH RSA Host keys                        : Clear
HTTPS RSA Host Keys and Signature        : Clear
```

The following example shows the output of the **fips show** command on the CER devices after the **fips enable** command is entered and administrative status is on and operational status is off:

```
device# fips show
FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0

FIPS mode   : Administrative status ON: Operational status ON
FIPS CC mode: Administrative status OFF: Operational status OFF

System Specific:
OS monitor access status is: Disabled

Management Protocol Specific:
Telnet server                  : Disabled
Telnet client                  : Disabled
TFTP client                    : Disabled
HTTPS SSL 3.0                  : Disabled
SNMP v1, v2, v2c               : Disabled
SNMP Access to security objects: Disabled
Password Display               : Disabled
Critical security Parameter updates across FIPS boundary:
(i.e. during "fips zeroize" ..., or "no fips enable")   :
Protocol Shared secret and host passwords: Clear
```

Modifying the FIPS policy

You can modify the default FIPS policy after the device is administratively in FIPS mode.

NOTE
Making changes to the default FIPS security policy weakens the security of the device and makes the device non-compliant with FIPS 140-2 Level 2. The default security policy defined in the FIPS Security Policy Document ensures that the device complies with all FIPS 140-2 specifications. Commands to alter the default security policy are available to the crypto-officer; however, Brocade does not recommend making changes to the default security policy at any time.The output of the **fips enable** command displays which protocols that constitute the FIPS policy are set in compliance with FIPS standards by default and can be adjusted to set a more flexible policy. The remaining protocols that constitute the FIPS policy are set to the appropriate status automatically during reload due to the **fips enable** command. The default FIPS policy is detailed in this chapter’s overview. Refer to “Overview” on page 8.

When you make no changes to the FIPS policy, the default FIPS policy is applied on the device and the device operates in strict FIPS mode upon reload, in full compliance with FIPS 140-2 specifications.

To set a more flexible FIPS policy on the Brocade device, use the following commands as desired to modify the default FIPS policy.

- Allow TFTP access:

  ```
  device(config)# fips policy allow tftp-access
  ```

  Syntax: **[no] fips policy allow tftp-access**

- Allow SNMP access to the Critical Security Parameter (CSP) MIB objects:

  ```
  device(config)# fips policy allow snmp-csp-access
  ```

  Syntax: **[no] fips policy allow snmp-csp-access**

- Allow access to monitor mode for debugging both from application and from boot prompts:

  ```
  device(config)# fips policy allow monitor-full-access
  ```

  Syntax: **[no] fips policy allow monitor-full-access**

  NOTE
  During an application reset, monitor access is restored to allow debugging. Refer to “Access to monitor mode” on page 39.

- Allow display of secrets and passwords in encrypted or cleartext format.

  ```
  device(config)# fips policy allow password-display
  ```

  Syntax: **[no] fips policy password-display**

  NOTE
  In the FIPS default mode of operation, **enable password-display** cannot be configured. The various **show** commands will always mask the secret or password with "....."

  To override this behavior, the crypto-officer could configure this policy, use the **fips policy password-display** command which then allows **enable password-display** to be configured. The various show commands will now display the secret or password in either encrypted or cleartext form, depending on their implementation.

- Retain the shared secret keys for all protocols and the host passwords:

  ```
  device(config)# fips policy retain shared-secrets
  ```

  Syntax: **[no] fips policy retain shared-secrets**

- Retain the HTTPS RSA host keys and the HTTPS Server digital certificate:

  ```
  device(config)# no fips policy retain rsa-host-keys
  ```

  Syntax: **[no] fips policy retain rsa-host-keys**

Clearing shared secrets and host keys

After you have reviewed the FIPS policy, use the following command to clear the shared secrets and host keys used by various networking protocols.

```
device(config)# fips zeroize all
```

Syntax: **[no] fips zeroize all | shared-secret | host-keys**

The **all** option zeroizes all shared secrets and host keys. The **shared-secret** option zeroizes shared secret keys only. The **host-keys** option zeroizes host keys only.

For example, entering the **fips zeroize shared-secret** zeroizes only the shared secret keys of various networking protocols and host access passwords.

NOTE
This command may cause operational failure within networking protocols using shared secrets and should be used with careful consideration.

The default FIPS policy calls for the zeroization of all keys using the **fips zeroize all** command option. The **fips zeroize all** option zeroizes all keys irrespective of the configured FIPS policy. When you apply a less strict FIPS policy than the default, zeroize at your discretion. The configured FIPS policy is maintained only when zeroization occurs due to the **no fips enable** command and the device subsequently enters non-FIPS mode. Refer to "Disabling FIPS mode" on page 38.

NOTE
Run the **clear ikev2 sa** command to manually reset the IPsec tunnel once the FIPS mode is disabled.

Table 6 lists the various keys used in the system that are zeroized on the management module in compliance with FIPS.

TABLE 6 Key zeroization on the management module

Keys used	Command option handling
DH Private Keys	Host-keys
FCSP Challenge Handshake Authentication Protocol (CHAP) Secret	Host-keys
SSH Session Key	Host-keys
SSH RSA Private Key	Host-keys
RNG Seed key	N/A
Passwords	Shared-secret

TABLE 6 Key zeroization on the management module

Keys used	Command option handling
TLS Private Key	Host-keys
TLS pre-master secret	Host-keys
TLS session key	Host-keys
TLS authentication key	Host-keys
RADIUS, TACACS+ secret	Shared-secret
Authentication passwords for various networking protocols	Shared-secret

Table 7 lists the various keys used in the system that are zeroized for MACsec on the interface modules in compliance with FIPS.

TABLE 7 Key zeroization for MACsec on interface modules

Keys used	Command option handling
CAK	Connectivity Association Key either configured manually by user or derived from the MSK obtained from authentication server.
CKN	CAK name either configured manually by user or derived from EAP session-id obtained from authentication server.
SAK	Secure Association Key used for encryption/decryption of the traffic. This key is derived from the CAK.
ICK	Integrity check.
KEK	Key encryption key

Table 8 lists the various keys used in the system that are zeroized for IKEv2/ IPsec on the interface modules in compliance with FIPS.

TABLE 8 Key zeroization for IKEv2/IPsec on interface modules

Keys used	Command option handling
Password for MM	
ECDH Shared Secret	
ECDSA Private Key	SP800-90
ECDH Private	SP800-90
IKE Encrypt/Decrypt	IKE v2 KDF
IKE Authentication Key	IKE v2 KDF
ESP Encrypt/Decrypt	IKE v2 KDF
ESP Authentication Key	IKE v2 KDF

TABLE 8 Key zeroization for IKEv2/IPsec on interface modules

Keys used	Command option handling
IKE KDF State	IKE v2 PRF
Pre-Shared Key (PSK)	RFC 5996
DRBG State	SP800-90
Entropy Data	NDRNG

Saving the configuration

After zeroizing, use the **write memory** command to save the configuration.

```
device (config)# write memory
```

Syntax: **write memory**

NOTE
Keep a backup copy of the startup configuration in the event of system reset.

Reloading the device

After you have saved the configuration, reload the device using the **reload** command:

```
device# reload
```

Syntax: **reload**

Various tests, including Power-On Self Tests (POSTs) and Known Answer Tests (KATs), are run by the Brocade device during reload, during the transition between non-FIPS and FIPS mode.

POSTs check for the consistency of the FIPS approved algorithms implemented on the device.

KATs are used to exercise various features of FIPS-approved algorithms.

All interfaces on the device are down until the tests are completed successfully.

Possible POST failure messages indicating that the tests did not pass successfully include:

```
Crypto module initialization and KNown Answer Test (KAT) failed with reason:(Error
Code 0x80000000)'CKR_VENDOR_DEFINED'

FIPS: MP Primary image verification failed

FIPS: MP Secondary image verification failed
FIPS: MP Monitor image verification failed
```

If there is a failure while the POSTs are being run, the device reboots or switches over to a standby management module. For information on access to monitor mode to perform debugging, refer to "Access to monitor mode" on page 39.

Use the **fips self-test** command to run tests on demand, in both FIPS and non-FIPS mode. Refer to "FIPS self test" on page 27.

After these tests are completed successfully, the device reloads in FIPS mode and FIPS mode is successfully enabled and operational on the Brocade device.

You can verify the status of the device as operationally in FIPS mode by using the **fips show** command.

```
device(config)# fips show
```

Syntax: **fips show**

The following example shows the output of the **fips show** command on Brocade MLX Series devices, after the device reloads successfully in the default strict FIPS mode, and administrative status is on and operational status is on:

```
device(config)# fips show
FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0

FIPS mode   : Administrative status ON: Operational status ON
FIPS CC mode: Administrative status OFF: Operational status OFF

System Specific:
OS monitor access status is: Disabled

Management Protocol Specific:
Telnet server                  : Disabled
Telnet client                  : Disabled
TFTP client                    : Disabled
HTTPS SSL 3.0                  : Disabled
SNMP v1, v2, v2c               : Disabled
SNMP Access to security objects: Disabled
Password Display               : Disabled
Critical security Parameter updates across FIPS boundary:
(i.e. during "fips zeroize" ..., or "no fips enable")   :
Protocol Shared secret and host passwords: Clear
```

The following example shows the output of the **fips show** command on CER devices, after the device reloads successfully in the default strict FIPS mode, and administrative status is on and operational status is on:

```
device(config)# fips show
FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0
FIPS mode   : Administrative status ON: Operational status ON
FIPS CC mode: Administrative status OFF: Operational status OFF

System Specific:
OS monitor access status is: Disabled

Management Protocol Specific:
Telnet server                  : Disabled
Telnet client                  : Disabled
TFTP client                    : Disabled
HTTPS SSL 3.0                  : Disabled
SNMP v1, v2, v2c                   : Disabled
SNMP Access to security objects: Disabled
Password Display               : Disabled
Critical security Parameter updates across FIPS boundary:
(i.e. during "fips zeroize" ..., or "no fips enable")   :
Protocol Shared secret and host passwords: Clear
SSH RSA Host keys                        : Clear
HTTPS RSA Host Keys and Signature        : Clear
```

MACsec and software release upgrade

If the device has MACsec configuration (such as **dot1x-mka-enable**) and you are upgrading the software, the bypass test (also known as the configuration integrity test) is automatically executed when the device is in FIPS mode. This test compares the HMAC values based on the available MACsec configurations. The output of the **show running** command displays the **fips bypass-test macsec config-integrity** command along with a value generated at runtime. If the HMAC value does not match, it means bypass test has failed. This will be treated as a failure in the conditional test and the device will reset.

The output of the **show dot1x-mka config** command provides information about the bypass status for every port.

Disabling FIPS mode

Use the following command to disable FIPS mode on the Brocade device.

```
Brocade(config)# no fips enable
```

After you enter the command, a warning displays that FIPS mode will be disabled.

This command performs the following policy-related operations:

- Enables TFTP access.
- Re-enables SNMP access to Critical Security Parameter (CSP) MIB objects.
- Re-enables SNMPv1, v2c, and all of SNMPv3 modes that were disabled in FIPS mode for future SNMPv3 user configuration.

- Re-enables access to monitor mode.
- Zeroizes shared secrets, SSH and HTTPS host keys, and the HTTPS certificate based on the configured FIPS policy.
- Zeroizes the MACsec and the IKEv2/IPsec CSP items on the management module as well as on the affected interface modules.

Changes to the running configuration are not saved to the startup configuration; therefore, when the device reloads it returns to FIPS mode.

Use the **write memory** command to save the running configuration.

NOTE
Run the **clear ikev2 sa** command to manually remove the IPsec tunnel once the FIPS mode is disabled. You can run the **clear ikev2 sa** command after executing the **fips zeroize all** command as well.

Access to monitor mode

The device in strict FIPS mode with the default policy applied does not allow access to monitor mode commands that perform memory access on both the management module and all the interface modules.

When the device is operating in FIPS mode, you can access all monitor mode commands, including memory debug commands, in the following instances:

- A flexible FIPS policy with the command **fips policy allow monitor-full-access** configured allows access memory debug commands.
- A strict FIPS policy does not allow access to memory debug commands. To apply a more flexible policy and allow access to all monitor commands, either configure a more flexible FIPS policy or disable FIPS mode to enter monitor mode. Refer to “Accessing monitor mode from FIPS mode” on page 40.

  NOTE
  Making changes to the default FIPS policy on the device is not recommended and weakens the security of the device.

- In the event of continuous failure on the Brocade device you can access all monitor mode commands. Refer to “Accessing monitor mode in the event of continuous failure” on page 40.

Perform the necessary operations after allowing the device access to the memory debug commands. Refer to “Debugging in monitor mode” on page 40.

To enable FIPS mode on the device after you have completed your use of monitor mode, refer to “Returning to FIPS mode from monitor mode” on page 41.

Accessing monitor mode from FIPS mode

A flexible FIPS policy with the command **fips policy allow monitor-full-access** configured allows access to monitor mode memory debug commands.

When the default FIPS policy is applied and the device is in strict FIPS mode, take the following steps to set a more flexible FIPS policy and allow access to memory debug commands:

NOTE
Making changes to the default FIPS policy on the device is not recommended and weakens the security of the device.

1. Clear the Critical Security Parameters (CSP). The device zeroizes the CSP based on the configured FIPS zeroization policy. Use the following command.

   ```
   device(config)# fips zeroize all
   ```

   Syntax: [no] fips zeroize all | shared-secret | host-keys

2. Allow access to the restricted memory commands within monitor mode by using the following FIPS policy command.

   ```
   device(config)# fips policy allow monitor-full-access
   ```

   Syntax: fips policy allow monitor-full-access

All commands in monitor mode, specifically the previously restricted memory access commands, are available for use. Refer to "Debugging in FIPS mode" on page 23.

If you do not want to apply any FIPS policy but the default and still need to enter monitor mode, disable FIPS mode on the device using the **no fips enable** command. Refer to "Disabling FIPS mode".

Once FIPS is disabled, all monitor mode commands are available.

Accessing monitor mode in the event of continuous failure

In the event of continuous failure, enter monitor mode by pressing **b** during a boot cycle.

When in monitor mode, only a restricted CLI is available if the device was previously running in FIPS mode. This restricted CLI does not allow the use of commands that refer to reading or writing memory location. To access the memory debug commands, erase the startup configuration file using the **erase startup-config** command. After the startup configuration is erased, the device lifts restrictions and starts with a blank configuration and FIPS mode is disabled. and use the **reload** command to reload the device. Refer to "Reloading the device" on page 36.

Debugging in monitor mode

After allowing access to monitor mode, the memory debug commands disabled in strict FIPS mode are available for use.

The monitor mode command set allows you to perform the following actions:

- debug the system reset
- erase the configuration (reset CSPs)
- set an IP address
- boot from TFTP

Returning to FIPS mode from monitor mode

After the necessary actions are performed in monitor mode, take the following steps to return the device to FIPS mode:

1. Use **Ctrl + Z** during reboot to exit monitor mode and return to the application prompt.
2. Re-create the CSP values.

Use the **fips enable** command to re-enable FIPS mode on the device. Refer to “Enabling FIPS mode” on page 27.

Chapter

Upgrading and Downgrading Software on FIPS-enabled Devices 2

For a complete information about upgrading NetIron software from existing versions to latest versions, please refer to the *Brocade NetIron Software Upgrade Guide* for instructions.

Upgrading FIPS-enabled devices

FIPS 140-2 compliance is a combination of implemented hardware procedures and the activation of a software-based security policy.

FIPS 140-2 certification is achieved when the device meets certain physical security and software conditions:

- FIPS physical security requirements: Tamper-evident security seals (available in Brocade FIPS kit purchased separately) must be applied to the product, based on FIPS Security Seal Procedure document (available on my.brocade.com).
- FIPS software compliance: The devices are configured to run in FIPS operational mode with the default FIPS security policy.

NOTE
After enabling the FIPS mode on your device, you cannot disable it without losing the device configuration. For disabling the FIPS mode, it is recommended that you contact Brocade Technical Support and perform the procedure under qualified guidance.

NOTE
Refer to the release notes for the software version running on the device to verify that software is FIPS and Common Criteria certified.

Image verification in FIPS or CC mode

NOTE
Refer to the latest version of the Brocade NetIron Unified R05.8.00 Release Notes for all R05.8.00 images.

Upgrade from non-sha256 signatures to sha256 signature packages requires two upgrade cycles to update the signature files to sha256 signatures for lp-autoupgrade to use the sha256 signatures for manifest file signature check.

When upgrading from a release that does not support SHA256 signatures to a release that does, please upgrade twice to the same release as follows. First upgrade to the release that supports SHA256 signatures. Reload the device. Then upgrade again to the same release that supports SHA256 signatures, and reload the device again. This ensures that the device will have the SHA256 signatures on the device.

NOTE
LP auto-upgrade is not supported in FIPS mode.

FIPS R05.8.00a images for Brocade MLXe devices

Table 9 lists the minimum required images to upgrade to R05.8.00a.

NOTE
The software described in this section applies only to the Brocade MLXe devices. You cannot use this software on other Brocade devices.

NOTE
Once a device has been FIPS cryptographically validated (as indicated in the **fips show** command output), signature verification of images is always done at the time of uploading the images to the device. To invalidate a FIPS cryptographically module, please contact Brocade technical support.

TABLE 9 Required images for a basic upgrade to R05.8.00a

Image description	Image name	Signature name for upgrade from devices running (legacy) 5.7.00a and older code using DSA1024/SHA1 signatures	RSA2048/SHA256 bit signatures file name
Combined application image for management modules	xm05800.bin	xm05800.sig	xm05800.sha256
Monitor image for management modules	xmb05800.bin	xmb05800.sig	xmb05800.sha256
Monitor image for interface modules	xmlb05800.bin	xmlb05800.sig	xmlb05800.sha256
Boot image for management modules	xmprm05800.bin	xmprm05800.sig	xmprm05800.sha256
Boot image for interface modules	xmlprm05800.bin	xmlprm05800.sig	xmlprm05800.sha256
Combined FPGA image for interface modules	lpfpga05800.bin	lpfpga05800.sig	lpfpga05800.sha256

Performing a basic upgrade

The overall procedure for a basic upgrade involves copying only the new application, boot, monitor, and combined FPGA image. How to upgrade additional images, refer to the *Brocade NetIron Software Upgrade Guide* for instructions.

There are two ways to perform an upgrade to FIPS-enabled devices:

- Using Secure Copy (SCP). For more information about SCP, refer to the Brocade NetIron configuration guides.
- Using a TFTP server. To upgrade Using TFTP at the Privileged EXEC level of the CLI (fips policy allow tftp-access is enabled):, you must first enter the following command in config mode:

Brocade (config)# **fips policy allow tftp-access**

NOTE
If the device is in FIPS mode, the **fips policy allow tftp-access** command is required. If the device is not in FIPS mode, TFTP is allowed.

NOTE
Once FIPS mode is enabled on the system, even if the mode is disabled at a later point of time, firmware integrity test will always be carried on the device at image copy time.

Downgrading from a FIPS environment to a non-FIPS environment

While a FIPS-supported image is running on the device, at any given time the image can be running in FIPS or non-FIPS mode. In either mode, SSH host-keys are lost because the FIPS supported image saves the host-keys as a file in flash memory, but the downgraded non-FIPS image stores host keys in the backplane EEPROM.

NOTE
Once FIPS mode is enabled on the system, even if the mode is disabled at a later point of time, firmware integrity test will always be carried out on the device at image copy time.

NOTE
In FIPS mode, do not attempt to downgrade to a release that does not support SHA256 signatures. Generally, NI releases prior to 5.6 patch C (but not 5.6 patch AA) do not support SHA256 signatures. In FIPS mode, downgrading to release that does not support SHA256 signatures is not supported.

NOTE
All shared-secret passwords (including any MD5 passwords) are lost when downgrading from a FIPS environment to a non-FIPS environment.

After the device is placed in non-FIPS mode, you can use SCP or TFTP to download and initialize an older image. Use the following steps to revert to a non-FIPS compliant image:

1. Log on to the device by entering your user name and password.
2. Disable FIPS by entering the **no fips enable** command at the prompt.
3. Regenerate ssh host keys or other shared secrets as needed for access after reload.
4. To replace the startup configuration with the **no fips enable** configuration, enter the **write memory** command.

   ```
   Brocade# write memory
   ```

5. Reload the configuration by entering the **reload** command.

Chapter

Common Criteria Certification 3

Table 10 lists the individual Brocade NetIron platforms that support Common Criteria certification requirements.

TABLE 10 Devices that support Common Criteria*

Features supported	Brocade NetIron XMR	Brocade MLX Series	Brocade NetIron CES 2000 Series BASE package	Brocade NetIron CES 2000 Series ME_PREM package	Brocade NetIron CES 2000 Series L3_PREM package	Brocade NetIron CER 2000 Series Base package	Brocade NetIron CER 2000 Series Advanced Services package
FIPS CC mode	No	MLXe: Yes** MLX: No	No	No	No	No	Yes**

* FIPS Common Criteria CLI supported on all platforms

** Only MLXe-4, MLXe-8, and MLXe-16 and CER 2000 Advanced Services Package are FIPS Common Criteria Certified.

This chapter contains steps for configuring Common Criteria mode on Brocade devices in compliance with the Common Criteria standards. Because Common Criteria mode enforces security restrictions additional to the FIPS mode, procedures and information are provided in relation to those for the FIPS mode. For information about enabling FIPS mode on the device, see Chapter 1, "Federal Information Processing Standards".

Determine if a device is Common Criteria certified

Check the https://www.niap-ccevs.org/CCEVS_Products/pcl.cfm site to determine if your device and the software it is running has achieved Common Criteria certification.

Overview

NOTE
Refer to the release notes for the software version running on the device to verify that software is FIPS and Common Criteria certified.

Common Criteria certification for a device enforces a set of security standards and feature limitations on a device to be compliant with the Common Criteria standards, similar to placing the device in FIPS mode. These restrictions are in addition to the requirements of the FIPS mode. When the device is placed in the Common Criteria mode, several security features that are available in the FIPS mode are unavailable on the device.

You can enable the Common Criteria mode on a device directly from non-FIPS mode, or on a device already in FIPS mode. The following table summarizes the transitions:

TABLE 11 Transition to Common Criteria mode

From	To non-FIPS mode	To FIPS mode	To Common Criteria mode
Non-FIPS mode	Not Applicable	Use the **fips enable** command	Use the **fips enable common-criteria** command
FIPS mode	Use the **no fips enable** command	Not Applicable	Use the **fips enable common-criteria** command
Common Criteria mode	Use the **no fips enable** command	Use the following commands and actions in a sequence: **no fips enable** or **no fips enable common-criteria** save the configuration and reload the device **fips enable**	Not Applicable

Notice the following:

- Disabling FIPS mode from the Common Criteria mode using the **no fips enable** command downgrades the device directly into non-FIPS mode.
- You cannot directly transition from Common Criteria mode to FIPS mode. To transition to FIPS mode, you should disable FIPS mode, save the configuration, reload the device, and then enable FIPS mode.

Features unavailable in Common Criteria mode

Some of the security features that are allowed in FIPS mode are disabled in Common Criteria mode:

1. **SSHv2:** Host and client key generation methods using DSA and RSA-1024 key size are not supported (Only RSA 2048 and higher key sizes are supported). Therefore, the following commands are not supported:
   - **crypto key generation dsa**
   - **crypto key client generation dsa**
   - **crypto key zero dsa**
   - **crypto key client zero dsa**
   - **crypto key gen rsa modulus 1024**
   - **crypto key zero rsa modulus 1024**
2. **TLS/HTTPS:** RSA 1024 key size for SSL or TLS private key generation is not supported (NetIron devices support only 2048 and above key sizes).
3. **SSH key exchange:** SSH key exchange method `DiffieHellmanGroup1Sha1` is not supported. Only `diffie-hellman-group14-sha1` is supported.
4. **Syslog:** Logging to host that uses UDP for transport is not supported. Only TLS host is supported. Therefore, the **logging host [ipv6] <ip-address> | <ipv6-address> ssl-port** *<port>* command is supported, only the UDP port option is not allowed. See "Configuring encrypted Syslog servers in Common Criteria mode" on page 53 for more information.

NOTE

TLS 1.0, 1.1, and 1.2 are supported. Secure syslog feature uses TLS to securely send the log messages to the log server. In order to restrict the device to use TLS 1.2 for maximum security, configure the **ip ssl server min-version 2** command.

5. AAA servers: Only local and TLS-encrypted TACACS+ servers are supported.

Enabling Common Criteria mode

When you enable Common Criteria mode on the device, it enters the Common Criteria Administrative mode. Similar to FIPS, Common Criteria also has administrative and operational modes:

- **Common Criteria Administrative mode:** Log in to the device console and enable the Common Criteria mode. You can optionally modify the default Common Criteria security policy in this mode.

  NOTE

  You must validate the software image with the corresponding signature file using the FIPS configuration guidelines. Failure in signature verification results in the device continuously rebooting after device reload.

- **Common Criteria Operational mode:** Transition to Common Criteria Operational mode from Common Criteria Administrative mode. After you transition the device to the Administrative mode, you must zeroize the FIPS keys, save the configuration, and reload the device to transition to the Operational mode.

Entering the Common Criteria Administrative mode

You can enable Common Criteria mode on a device with the following command:

```
(Brocade)# fips enable common-criteria
```

Syntax: [no] fips enable common-criteria

The device prompt displays the detailed banner information as follows.

```
Brocade(config)#fips enable common-criteria
WARNING: This will enable FIPS and Common Criteria on this device. Please refer
       : to the NetIron Federal Information Processing Standards Guide for
       : more details. Also, be advised that Software/Firmware Integrity checks
       : will always be performed on this device on subsequent reloads, even
       : if FIPS or Common Criteria is disabled in the future.
Are you sure? (enter 'y' or 'n'): y
This device is now running in CC administrative mode.
At this time you can alter this system's CC default security policy
and then enter CC operational mode.

Note: Making changes to the default CC security policy weakens
the security of the device and makes the device non-compliant
with CC and FIPS 140-2 Level 2, design assurance Level 3.
The default security policy defined in the FIPS
Security Policy Document ensures that the device complies with all
```

```
FIPS 140-2 specifications. Commands to alter the default security policy
are available to the crypto-officer; however, Brocade does not recommend
making changes to the default security policy at any time.
=====================================

To enter CC mode, complete the following steps:
1. Optionally, configure FIPS policy commands that meets your network
   requirements. You must explicitly configure the following services if you
   want to use them when the device is operational in CC mode:

      - Allow TFTP access.
          Current status: Enabled
      - Allow SNMP Access to the Critical Security Parameter (CSP) MIB objects.
          Current status: Disabled
      - Allow access to all commands within the monitor mode.
          Current status: Disabled
      - Allow cleartext password display in some commands.
          Current status: Disabled
      - Retention of shared secret keys for all protocols and the host passwords.
          Current status: Clear
      - Retention of HTTPS RSA host keys and certificate.
          Current status: Retain

2. Enter the "fips zeroize all" command, which zeroes out the shared secrets
    used by various networking protocols, including the host access passwords,
    SSH and HTTPS host-keys with the digital signature based on the configured
    FIPS Security Policy.
3. Save the running configuration.
4. Reload the device.
5. Enter the "fips show" command to verify that the device entered
  FIPS or CC operational mode.
=====================================

The system will disable the following services or commands after reload:
1. Telnet server will be disabled. The "telnet server" command will be removed.
2. SSL Client will be enabled.
3. SCP will be enabled. The "ip ssh scp disable" command will be removed.
4. Configuration "boot system {slot1|slot2} <file>" will be removed as FIPS mode
does not allow system to boot from Storage Card.
5. Configuration "lp boot system {slot1|slot2} <file> <slot>" will be removed as
FIPS mode does not allow system to boot from Storage Card.
6. Configuration "boot system tftp <ip> <file>" will be removed as FIPS mode does
not allow system to boot from TFTP.
7. Configuration "enable password-display" will be removed.
8. HTTP server will be disabled. The "web-management http" command will be
removed.
9. HTTPS server will change as follows:
      -SSL 3.0 will be disabled.
      -TLS version 1.0 and greater will be used.
      -RC4 cipher will be disabled.[Latent functionality RC4 is no longer
supported]
      -Passwords will be required; the "web-management allow-no-password"
        command will be removed.
Passwords/Keys which dont comply FIPS standards will be removed on reload.
Please see FIPS config guide for complete details.

=====================================
Additionally, in CC operational mode, the system will disable the following
services or commands after reload:
CC1. UDP Syslog servers will be disabled from configuration.
```

```
CC2. DSA keys will be deleted from configuration, and will be disabled .
CC3. RSA key sizes will be restricted to 2048 and above in the configuration.
CC4. Non-TLS TACACS+ servers will be disabled from configuration.
CC5. RADIUS servers will be disabled from the configuration.
CC6. For SSH Key Exchange, only diffie-hellman-group14-sha1 algorithm is allowed.
      - Start SSL client task for secure syslog server.
          Current status: Enabled
```

Displaying Common Criteria information

After you have enabled Common Criteria administrative mode on the device, you can display the relevant information with the **fips show** command.

```
Brocade# fips show
[Not a] FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0

FIPS mode   : Administrative status ON: Operational status OFF
FIPS CC mode: Administrative status ON: Operational status OFF
Some shared secrets inherited from non-fips mode may not be fips compliant and has
to be zeroized
The system need to be reloaded to operationally enter FIPS mode.

System Specific:
OS monitor access status is: Disabled

Management Protocol Specific:

Telnet server                  : Disabled
Telnet client                  : Disabled
TFTP client                    : Disabled
HTTPS SSL 3.0                  : Disabled
SNMP v1, v2, v2c               : Disabled
SNMP Access to security objects: Disabled
Password Display               : Disabled
Any AAA server (including      :
RADIUS, non TLS-TACACS+, None) : Disabled

Protocol Shared secret and host passwords: Clear
SSH RSA Host keys                        : Clear
HTTPS RSA Host Keys and Signature        : Clear*
```

NOTE
* For MLXe chassis only. Not for CER.

After you have enabled Common Criteria operational mode by zeroizing the FIPS keys, saving the configuration, and reloading the device, enter the **fips show** command to verify the operational mode status:

```
Brocade# fips show
FIPS Validated Cryptographic Module
MP FIPS Version: BRCD-IP-CRYPTO-VER-3.0
LP FIPS Version: BRCD-LP-CRYPTO-VER-1.0

FIPS mode   : Administrative status ON: Operational status ON
FIPS CC mode: Administrative status ON: Operational status ON
System Specific:
```

```
OS monitor access status is: Disabled

Management Protocol Specific:

Telnet server                  : Disabled
Telnet client                  : Disabled
TFTP client                    : Disabled
HTTPS SSL 3.0                  : Disabled
SNMP v1, v2, v2c               : Disabled
SNMP Access to security objects: Disabled
Password Display               : Disabled
Any AAA server (including      :
RADIUS, non TLS-TACACS+, None) : Disabled

Critical security Parameter updates across FIPS boundary:
(i.e. during "fips zeroize" ..., or "no fips enable")   :
Protocol Shared secret and host passwords: Clear
SSH RSA Host keys                        : Clear
HTTPS RSA Host Keys and Signature        : Clear
```

TABLE 12 fips show command output description

Field...	Description...
OS monitor access status	The following policy allows full access to the OS monitor mode. This includes read, write access for debug purpose. **fips policy allow monitor-full-access.**
Telnet server	Telnet client and server are always disabled in FIPS CC Operational mode.
Telnet client	
TFTP client	To allow TFTP access in FIPS mode use **fips policy allow tftp-access.**
HTTPS SSL 3.0	Always disabled in FIPS mode.
SNMP v1, v2, v2c	Always disabled in the FIPS CC mode. SNMPv3 in *noAuthNoPriv*, and *authNoPriv* security mode is not supported. Only SNMPv3 in *authPriv* security mode is supported.
SNMP Access to Security objects	To allow snmp read access to the critical security parameters and mib objects, use **fips allow snmp-csp-access.**
Password display	To allow password display, use **fips allow password-display.**
AAA server	To allow any AAA server (including RADIUS and non TLS-Encrypted TACACS+ servers) to be used in FIPS common-criteria mode, use **fips policy allow common-criteria aaa-server-any.**
Protocol shared password	To retain the protocol shared secrets and host access passwords between FIPS and non-FIPS mode, use **fips policy retain shared-secrets.**
HTTPS DSA Host keys	To retain the SSH RSA host keys between FIPS and non-FIPS mode, use **fips policy retain rsa-host-keys** (for MLX platform ONLY).

NOTE

Making changes to the default FIPS security policy weakens the security of the device and makes the device non-compliant with FIPS 140 Level 2. The default security policy defined in the FIPS Security Policy Document ensures that the device complies with all FIPS 140-2 specifications. Commands to alter the default security policy are available to the crypto-officer; however, Brocade does not recommend making changes to the default security policy at any time.

Configuring encrypted Syslog servers in Common Criteria mode

Brocade devices, when enabled for Common Criteria mode, do not support syslog servers that use UDP transport. However, other parameters that are defined for syslog server connections, such as specifying the hold time for queued messages and traps when the device reloads or switches over are applicable for encrypted syslog connections as well.

When you enable Common Criteria mode on a device, the device is in the Common Criteria Administrative mode, where syslog server configuration that uses UDP transport is retained. You can configure encrypted syslog server connections in this mode. Syslog messages that are generated when the device is in the Administrative mode, are sent to the UDP syslog servers, not encrypted syslog server that you have configured. When the device is put in the Common Criteria Operational mode, existing syslog servers that use UDP transport are removed, and only encrypted syslog server connections are accepted. Conversely, when a device is downgraded from Common Criteria mode, the encrypted syslog server connections that were configured are removed, and the device supports only un-encrypted UDP syslog servers.

The following table summarizes these transitions.

TABLE 13 Syslog server connections during transition to and from Common Criteria mode

From	To non-FIPS mode	To FIPS mode	To Common Criteria mode
Non-FIPS mode	Not Applicable.	No change. FIPS mode does not support encrypted syslog servers.	All the UDP servers are removed when the device is put in CC Operational mode. Only encrypted syslog server connections are allowed in the CC Operational mode.
FIPS mode	No Change	Not Applicable.	All the UDP servers are removed when the device is put in CC Operational mode. Only encrypted syslog server connections are allowed in the CC Operational mode.
Common Criteria mode	All the SSL servers are removed. Non-FIPS mode does not support encrypted syslog server connections.	Not Allowed. You must disable Common Criteria mode to revert to Non-FIPS mode, and then re-enable FIPS mode. FIPS mode does not support encrypted syslog server connections.	Not Applicable.

Configuring an encrypted syslog server

You can configure up to six encrypted syslog servers, but only one is active at any time, with the other servers acting as standby. When you add an encrypted syslog server, if there is no active syslog server, a session is established with the configured server. If a new connection is added when an active session exists, a new session with another encrypted syslog server is not attempted.

A new syslog server session is attempted in the following scenarios:

- Current active encrypted syslog server configuration is removed or the SSL connection to the active syslog server is closed
- During a device reload
- During switch over of the management module
- No active syslog server is found when the device sends syslog messages

Attempts to connect to a new syslog server starts with the first configured syslog server. The device attempts to establish an SSL connection with a server until a successful SSL connection is established. During this interval, the trap hold down timer is started and all the syslog messages are queued. When the timer expires, the device sends queued log messages to the connected syslog server.

Configuring encrypted syslog servers requires two steps:

- Installing the SSL Client certificate from a remote machine
- Adding encrypted syslog servers

Installing the SSL client certificate

Before you can configure an encrypted syslog server for the device, you must install the SSL client certificate. Do one of the following to install the SSL client certificate.

Using TFTP:

1. Use TFTP to copy the SSL Client Certificate and private key from the remote machine if TFTP is enabled on the device. Enter the following commands in sequence in any order:

```
Brocade# copy tftp flash 10.25.101.121 cert.p12 client-certificate
Brocade# copy tftp flash 10.25.101.121 privkeyfile client-private-key
```

Syntax: **copy tftp flash** *<remote_ip>* *<cert_file>* **client-certificate**

and

Syntax: **copy tftp flash** *<remote_ip>* *<priv_key_file>* **client-private-key**

The **remote_ip** keyword specifies the IP address of the remote host where the SSL Client certificate and private key are present. The **cert_file** keyword specifies the filename of the SSL Client Certificate, and the **priv_key_file** keyword specifies the filename of the private key.

Using SCP

1. Use SCP to copy the SSL Client Certificate and private key from the remote machine. Enter the following commands in sequence in any order at the remote host where the SSL Client Certificate and private key are present:

```
Host# scp cert.p12 user@10.25.105.121:sslclientcert
Host# scp privkeyfile user@10.25.105.121:sslclientprivkey
```

Syntax: **scp <cert_file> user@<remote_ip>:sslclientcert**

and

Syntax: **scp <priv_key_file> user@<remote_ip>:sslclientprivkey**

The **remote_ip** keyword specifies the IP address of the device. The **cert_file** keyword specifies the filename of the SSL Client Certificate, and the **priv_key_file** keyword specifies the filename of the private key.

Adding an encrypted syslog server

To configure an encrypted server connection, enter the following command:

```
Brocade (config)# logging host 10.25.105.201 ssl-port 60514
```

Syntax: **logging host [ipv6] <ip-address> | <ipv6-address> ssl-port** *<port>*

The **ip-address** keyword specifies the syslog server. The **ssl-port** keyword specifies the SSL port that will be used to connect to the specified syslog server.

NOTE
You can configure an encrypted syslog server connection only after the device has been placed in the Common Criteria mode. While you can configure these when the device is in the Administrative mode, the configuration takes effect only after the device is put in the Common Criteria Operational mode.

Displaying the configured server connections

You can display the active encrypted syslog server connection with the **show ip ssl** command:

```
Brocade# show ip ssl
Session  Source IP      Source Port   Remote IP        Remote Port
0        10.25.105.80   633           10.25.105.201    60514
```

In addition, you can use the show logging command to display the active SSL-encrypted syslog server along with the logging level information.

```
Brocade# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 27 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning
Current active SSL syslog server: 10.25.105.201:60514
```

AAA servers in Common Criteria mode

Common Criteria mode requires that devices support NDPP version 1.1. This requires that the communication of the device with the AAA servers takes place over a TLS-encrypted session.

Even though you can configure multiple TLS-encrypted TACACS+ servers, but only one connection can be active at any time. If another TLS-encrypted TACACS+ session is attempted at the same time as the first TACACS+ session, the connection attempt is rejected. Additionally, since the TACACS+ server may accept only a single TACACS+ session over the TCP or the TLS-encrypted connection, you are recommended to use this only for authentication.

When you enable Common Criteria mode on a device, the device is in the Common Criteria Administrative mode, where non TLS-encrypted TACACS+ configuration is still used. You can configure TLS-encrypted TACACS+ servers in this mode. In Administrative mode, the device can still use the non TLS-encrypted TACACS+ servers. When the device is put in the Common Criteria Operational mode, existing non-TLS-encrypted TACACS+ servers are disabled and only TLS-encrypted TACACS+ servers are enabled. Conversely, when a device is downgraded from Common Criteria mode, the TLS-encrypted TACACS+ servers are disabled, and the device supports only non-TLS encrypted TACACS+ servers.

NOTE
You can modify the default Common Criteria policy to allow non-TLS-encrypted TACACS+ server, but this will make the device non-compliant to Common Criteria requirements.

Configuring a TLS-encrypted TACACS+ connection

Configuring a TLS-encrypted TACACS+ server requires two steps:

1. Installing the SSL Client certificate from a remote machine. (Refer to "Installing the SSL client certificate," in this chapter)
2. Adding TLS-encrypted TACACS+ servers using the tacacs-server CLI (see below).

To configure a TACACS+ server connection that uses TLS encryption, enter the following command:

```
(Brocade)# tacacs-server host 10.25.105.201 ssl-auth-port 2323 default
```

Syntax: **[no] tacacs-server host <ip-addr> | <server-name> [{ssl-auth-port} <number>
[authentication-only | authorization-only | accounting-only | default] [key <string>]]**

The **ssl-auth-port** keyword specifies that the TACACS+ server uses a TLS-encrypted TCP connection.

Even though the command supports adding a non TLS-encrypted TACACS+ server, for the device to use the TACACS+ server in Common Criteria mode, you must configure a TLS-encrypted TACACS+ server. If you do not specify the port number, the default option of **auth-port** (port 49 with no TLS encryption) is used.

Modifying the Common Criteria policies to use non-encrypted AAA servers

If required, you can modify the Common Criteria policies to allow AAA servers that do not use TLS encryption to be enabled, such as RADIUS and non TLS-encrypted TACACS+. When non TLS-encrypted AAA servers are allowed, you cannot use the TLS-encrypted TACACS+ servers on the device.

NOTE
Modifying the default Common Criteria policy will make the device non-compliant to Common Criteria standards.

To allow any AAA server to work with the device in Common Criteria mode, enter the following command:

```
(Brocade)# fips policy allow common-criteria aaa-server-any
```

Syntax: **[no] fips policy allow common-criteria aaa-server-any**

Use the **[no]** version of this command to disable the non-TLS encrypted AAA servers. If any non-encrypted AAA servers were available on the device, they are disabled when the device is in the common criteria operational mode.

Use the **show aaa** command which also shows the **SSL-auth-port** for TLS-encrypted TACACS+ servers.

```
Brocade # show aaa
TACACS default key: ...
TACACS retries: 1
TACACS timeout: 5 seconds
TACACS+ Server: IP=10.25.105.201 SSL-Auth-Port=60520 Usage=any Key=...
                opens=0 closes=0 timeouts=0 errors=0
                packets in=0 packets out=0
no connection
***** Radius server not configured
```

Downgrading from Common Criteria mode to non-FIPS mode

You cannot directly downgrade to FIPS mode. You first downgrade to non-FIPS mode, then re-enable FIPS mode using the procedures detailed in the earlier chapter.

After the device is placed in non-FIPS mode, you can use SCP or TFTP to download and initialize an older image. Use the following steps to revert to a non-FIPS compliant image:

1. Log on to the device by entering your user name and password.
2. Disable Common Criteria mode by entering the **no fips enable** or **no fips enable common-criteria** command at the prompt.
3. Regenerate ssh host keys or other shared secrets as needed for access after reload.
4. To replace the startup configuration with the **no fips enable** configuration, enter the **write memory** command.

   ```
   Brocade# write memory
   ```

5. Reload the configuration by entering the **reload** command.

Appendix

Acknowledgments and Encrypted Syslog Server A

This appendix presents the acknowledgments for portions of code from various vendors that are included in the Brocade devices covered in this manual.

OpenSSL Licensing

NOTE
This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)

NOTE
OpenSSL has been compiled without the Heartbeat extension.

License

This is a copy of the current LICENSE file inside the CVS repository.

```
  LICENSE ISSUES
  ==============

  The OpenSSL toolkit stays under a dual license, i.e. both the conditions of
  the OpenSSL License and the original SSLeay license apply to the toolkit.
  See below for the actual license texts. Actually both licenses are BSD-style
  Open Source licenses. In case of any license issues related to OpenSSL
  please contact openssl-core@openssl.org.

  OpenSSL License
  ---------------

/* ====================================================================
 * Copyright (c) 1998-2011 The OpenSSL Project.  All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 *    software must display the following acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
 *
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 *    endorse or promote products derived from this software without
 *    prior written permission. For written permission, please contact
 *    openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 *    nor may "OpenSSL" appear in their names without prior written
 *    permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 *    acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * ====================================================================
 *
 * This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com).  This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

 Original SSLeay License
 -----------------------

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to.  The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code.  The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *    "This product includes cryptographic software written by
 *     Eric Young (eay@cryptsoft.com)"
 *    The word 'cryptographic' can be left out if the rouines from the library
 *    being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 *    the apps directory (application code) you must include an acknowledgement:
 *    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
 * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * The licence and distribution terms for any publically available version or
 * derivative of this code cannot be changed.  i.e. this code cannot simply be
 * copied and put under another distribution licence
 * [including the GNU Public Licence.]
 */
```

Appendix B
Configuring an Encrypted Syslog Server

The information available in this appendix is a representative configuration example of the many types of Syslog servers available. It describes how to setup an encrypted syslog server running on Ubuntu 10.4. The setup procedure for encrypted syslog server on other Linux operating systems such as Red Hat or Centos is the same as mentioned in this document except for difference in commands.

You will need to set up stunnel as a server and client in your server. As a server, stunnel listens on port 60516 to connections from its client peers, and all connections are forwarded to the locally-running rsyslog listening at port 61514. As a client, rsyslog forwards message to stunnel local portal at port 61514, and local stunnel forwards data via the network to port 60514 to its remote peer.

Set up stunnel

1. Install the stunnel utility with the following command:

   ```
   $ sudo apt-get install stunnel4
   ```

2. Edit the file with path `/etc/default/stunnel4` to start the service on system startup. Use an editor such as vi.

   ```
   $ sudo vi /etc/default/stunnel4
   ```

3. Change the line `Enabled=0` to `Enabled=1`.

Create a certificate with the OpenSSL tool

1. Enter the following command:

   ```
   cd /etc/stunnel
   ```

2. Enter the following command to create the `/etc/stunnel/stunnel.pem` file with certificate and key for SSL:

   ```
   $openssl req -new -x509 -days 365 -nodes -out stunnel.pem -keyout
    /etc/stunnel/stunnel.pem
   ```

3. Enter the following command to change the permissions for the certificate that you generated.

   ```
   $ sudo chmod 600 /etc/stunnel/stunnel.pem
   ```

Create a configuration file

1. Enter the following command to open the stunnel.conf file:

   ```
   $sudo vi /etc/stunnel/stunnel.conf
   ```

2. Comment out the features that you don't require, such as [pop3s], [ssmtp], and [imaps] sections.

3. Change the line `cert=/etc/stunnel/mail.pem` to `cert=/etc/stunnel/stunnel.pem`.
4. Add the following lines and save the file.

   ```
   ; Certificate/key is needed in server mode
   cert = /etc/stunnel/stunnel.pem
   key = /etc/stunnel/stunnel.pem

   ; Some debugging stuff useful for troubleshooting
   debug = 7
   foreground=yes

   [ssyslog]
   accept  = 60514
   connect = 61514
   ```

Change the stunnel4 startup file

1. Enter the command `cd /etc/init.d/stunnel4` and change `ENABLED=0` to `ENABLED=1`.

Restart the stunnel service

1. Enter the following command:

   ```
   $sudo  /etc/init.d/stunnel4 restart
   ```

Configure rsyslog

Ubuntu 10.04.3 comes with Rsyslog 4.2.0 as its default logger. You can add MySQL output support and the Reliable Event Logging Protocol (RELP). Enter the following command:

```
root@linux:~$sudo apt-get install rsyslog-mysql rsyslog-relp
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  dbconfig-common librelp0
The following NEW packages will be installed:
  dbconfig-common librelp0 rsyslog-mysql rsyslog-relp
0 upgraded, 4 newly installed, 0 to remove and 0 not upgraded.
Need to get 677kB of archives.
After this operation, 2,335kB of additional disk space will be used.
Do you want to continue [Y/n]? y
```

During the installation process, do the following:

1. Create the tables that are needed in MySQL when prompted.
2. Set the MySQL root password
3. Create a password that the rsyslog processes will use in its configuration files.

Enable accepting remote logs

To turn on accepting remote logs, edit the `/etc/rsyslog.conf` file by commenting out the following lines:

```
# provides TCP syslog reception
$ModLoad imtcp
$InputTCPServerRun 61514
```

Restart rsyslog service

Enter the following command:

```
root@linux:~$sudo service rsyslog restart
```

NOTE
It is recommended to reboot the Linux server after the setup.

Print log messages

Enter the following command to update the log-watcher window with logged messages as they arrive.

```
root@linux:~$tail  -f  /var/log/messages
```

You can also configure a web UI to display the syslog messages using the Reliable Event Logging Protocol (RELP). See http://www.linuxjournal.com/content/centralized-logging-web-interface for more information.

TLS encrypted syslog server configuration and validation

Certificates (both Server and Trusted) will need to meet the following criteria.

- Only RSA certificates is accepted.
- Public Key should be greater than or equal to 2048 bits.
- The device should have server certificate installed.
- Expired certificate is not accepted.
- A certificate with an empty Subject Alternative Name (SAN) field is rejected.
- When the server’s certificate signature is invalid, the client rejects a certificate based on the public key provided in the issuer's self-signed certificate.
- A certificate with a mismatching Subject Alternative Name (SAN) IP address field is rejected.
- Correct cipher suites.

In common criteria mode, when device acts as a TLS client while connecting to a remote server, the client needs to perform validation of the server certificate.

1. Create the TLS encrypted syslog server's private key using the **openssl genrsa** command.

```
openssl genrsa -out rsakey2048.pem 2048
Generating RSA private key, 2048 bit long modulus
..+++
.............................................................+++
e is 65537 (0x10001)
```

2. Create the TLS encrypted syslog server's self-signed certificate, also including the IP address of the server in the Subject Alternative Name (SAN) extension of the certificate.

    a. Create a configuration file which looks like the following.

```
cat req_san.config.txt

[ req ]
default_bits                     = 2048                 # Size of keys
default_keyfile                 = key.pem             # name of generated keys
default_md                     = sha256                             # message 
digest algorithm
string_mask                      = nombstr              # permitted characters
distinguished_name               = req_distinguished_name
req_extensions                   = v3_req

[ req_distinguished_name ]
# Variable name                  Prompt string
#-------------------------        ----------------------------------
0.organizationName               = Organization Name (company)
organizationalUnitName           = Organizational Unit Name (department, 
division)
emailAddress                     = Email Address
emailAddress_max                 = 40
localityName                     = Locality Name (city, district)
stateOrProvinceName              = State or Province Name (full name)
countryName                      = Country Name (2 letter code)
countryName_min                  = 2
countryName_max                  = 2
commonName                       = Common Name (hostname, IP, or your name)
commonName_max                   = 64

[ v3_req ]
basicConstraints                 = CA:FALSE
subjectKeyIdentifier             = hash

[ extensions_section ]
subjectAltName=IP:192.168.10.201
```

    b. Create the certificate giving the configuration file created in the previous step as parameter.

```
openssl req -new -x509 -key rsakey2048.pem -out 
rsacert2048_days1095_sha256_SAN.pem -days 1095 -sha256 -config 
./req_san.config.txt -extensions extensions_section

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Organization Name (company) []:Brocade
Organizational Unit Name (department, division) []:Engineering
Email Address []:
Locality Name (city, district) []:San Jose
State or Province Name (full name) []:California
Country Name (2 letter code) []:US
```

```
Common Name (hostname, IP, or your name) []:SP_EMIS TLS Encrypted SYSLOG
server
```

c. To view the TLS encrypted syslog server's self-signed certificate that is created:

```
openssl x509 -inform PEM -noout -text -in rsacert2048_days1095_sha256_SAN.pem

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            f9:aa:bd:da:1b:5a:3e:51
        Signature Algorithm: sha256WithRSAEncryption
       Issuer: O=XYZ, OU=ABC, L=San Jose, ST=California, C=US, CN=SP_EMIS TLS
Encrypted SYSLOG server
        Validity
            Not Before: Mar 31 22:20:47 2014 GMT
            Not After : Mar 30 22:20:47 2017 GMT
       Subject: O=XYZ, OU=ABC, L=San Jose, ST=California, C=US, CN=SP_EMIS TLS
Encrypted SYSLOG server
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:ca:5f:78:de:07:b7:15:21:b4:9d:e9:66:b7:5e:
                    48:8b:96:ed:4b:f3:5d:dc:d7:95:27:ed:ca:1d:00:
                    9d:d6:06:5b:f5:df:d2:0c:54:69:53:4a:38:d1:52:
                    2d:bf:6c:a4:2b:7d:dd:ad:e7:2c:5a:4f:1c:0e:8b:
                    59:7a:04:f1:54:b8:00:99:51:21:f7:42:81:17:4c:
                    cc:94:86:00:8b:c6:c0:0d:3b:7a:19:66:3c:e5:33:
                    be:5f:b5:2c:d9:df:74:1c:07:f5:41:82:c0:b2:48:
                    9e:c3:7b:cc:2e:07:4e:d8:2a:17:69:48:ae:f2:97:
                    4a:fd:7e:4b:34:2d:36:49:bb:3a:79:c6:c4:9c:1e:
                    5f:1b:d7:59:a0:3e:27:02:2f:2b:eb:60:26:95:20:
                    bb:2a:e8:5b:9b:56:b6:2e:62:eb:a1:21:f4:95:1c:
                    e1:d6:ca:4e:74:0a:a1:6a:f6:b0:27:7f:f4:e2:d2:
                    92:f9:db:25:49:9f:c1:87:d3:ed:1f:d1:98:6c:da:
                    15:04:c1:bb:16:66:78:02:ab:81:a0:98:c2:62:75:
                    b1:4e:96:0a:fd:25:84:64:f3:e6:35:5e:06:05:79:
                    c6:83:73:d6:33:6b:57:64:ad:4d:b5:f4:3d:f6:e7:
                    e5:a3:71:d0:c9:e5:77:7a:4a:11:c0:89:ca:1a:35:
                    72:df
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Alternative Name:
                IP Address:192.168.10.201
    Signature Algorithm: sha256WithRSAEncryption
        88:a5:6c:d3:15:c2:10:20:c9:36:73:ba:c5:72:4c:e4:26:78:
        dc:ec:21:a2:2b:ec:4b:5a:42:85:be:fe:c4:1f:01:97:f0:5c:
        e2:51:1a:3b:84:15:c9:cb:63:35:b1:e6:b8:3e:2a:76:47:5f:
        ce:1b:59:80:43:81:95:b8:aa:1b:11:7f:80:6f:3f:97:d9:0c:
        43:7e:53:b0:04:80:be:52:da:4b:0b:b4:70:07:a8:b6:d8:09:
        55:9f:4e:08:7a:c1:df:7e:da:dd:c0:59:f3:9d:c6:f5:2b:ec:
        66:89:9e:c9:5f:6c:d1:e7:fe:1e:d3:b1:6e:9f:84:3c:fb:ed:
        e5:c9:2c:7f:8c:85:f4:97:bb:99:3c:cd:1e:3e:d2:a1:6e:09:
        3a:05:b6:c1:76:b9:54:ec:34:a8:a9:6f:ca:30:34:cb:ec:05:
        5d:17:a3:cb:21:3a:69:e3:7d:28:d2:15:c0:19:0e:34:00:8d:
        68:ce:cd:0a:65:db:e4:88:b6:d1:67:40:3a:3d:22:bf:dc:22:
        16:ec:4f:08:a7:54:7f:42:73:9b:f7:88:1a:70:73:8c:81:a8:
        5c:b4:55:5f:7e:94:75:ec:93:f0:48:08:18:4a:3c:ea:c6:6b:
```

```
        48:d6:b2:f4:ff:de:23:df:d5:fd:a0:bd:8a:cb:c7:69:f9:3a:
        d9:5f:c5:0f
```

3. Upload the self-signed certificate to the device. Verify the certificate after upload, using either the signature or the fingerprint of the certificate.

```
scp rsacert2048_256_days1095_SAN.pem lab1@192.168.105.82:ssltrustedcert

lab1@192.168.105.82's password:
rsacert2048_256_days1095_SAN.pem              100% 1448     1.4KB/s   00:00
Connection to 192.168.105.82 closed by remote host.
```

   The SCP command can be executed from a remote system (like Linux or Windows). Use the **ssltrustedcert** option to upload a trusted certificate (certificate.pem) to the device specified by the IP Address as user.

   The device can have up to three dynamic trusted certificates. Once three dynamic trusted certificates are uploaded, running the command again will return an error.

   To display the list of dynamic trusted certificates on the device, use the **show ip ssl certificate** CLI described in the previous section.

   To delete the dynamic trusted certificate list on the device, use an empty certificate file. Following is an example of deleting the dynamic trusted certificate list:

```
> ls -la empty.file
ls: empty.file: No such file or directory
> empty.file
> ls -la empty.file
-rw-r--r-- 1 lab engr 0 Mar 31 12:47 empty.file
> scp empty.file lab@192.168.10.82:ssltrustedcert
```

4. Verify the certificate after upload to the device, using either the signature or the fingerprint of the certificate.

   The **show ip ssl certificate** command displays the dynamic trusted certificate list. The dynamic trusted certificate list can be modified by the **scp ssltrustedcert** command.

```
Device# show ip ssl certificate

No SSL sessions in use.
Trusted Certificates:
 Dynamic:
  Signature Algorithm: sha256WithRSAEncryption
  Validity:
   Not Before: Mar 31 2014 13:22:47
   Not After : Mar 30 2017 13:22:47
  X509v3 extensions:
   X509v3 Subject Alternative Name:
    IP Address:192.168.10.201
  Signature:
    88:a5:6c:d3:15:c2:10:20:c9:36:73:ba:c5:72:4c:e4:26:78:
    dc:ec:21:a2:2b:ec:4b:5a:42:85:be:fe:c4:1f:01:97:f0:5c:
    e2:51:1a:3b:84:15:c9:cb:63:35:b1:e6:b8:3e:2a:76:47:5f:
    ce:1b:59:80:43:81:95:b8:aa:1b:11:7f:80:6f:3f:97:d9:0c:
    43:7e:53:b0:04:80:be:52:da:4b:0b:b4:70:07:a8:b6:d8:09:
    55:9f:4e:08:7a:c1:df:7e:da:dd:c0:59:f3:9d:c6:f5:2b:ec:
    66:89:9e:c9:5f:6c:d1:e7:fe:1e:d3:b1:6e:9f:84:3c:fb:ed:
    e5:c9:2c:7f:8c:85:f4:97:bb:99:3c:cd:1e:3e:d2:a1:6e:09:
    3a:05:b6:c1:76:b9:54:ec:34:a8:a9:6f:ca:30:34:cb:ec:05:
    5d:17:a3:cb:21:3a:69:e3:7d:28:d2:15:c0:19:0e:34:00:8d:
    68:ce:cd:0a:65:db:e4:88:b6:d1:67:40:3a:3d:22:bf:dc:22:
```

```
    16:ec:4f:08:a7:54:7f:42:73:9b:f7:88:1a:70:73:8c:81:a8:
    5c:b4:55:5f:7e:94:75:ec:93:f0:48:08:18:4a:3c:ea:c6:6b:
    48:d6:b2:f4:ff:de:23:df:d5:fd:a0:bd:8a:cb:c7:69:f9:3a:
    d9:5f:c5:0f
```

5. Configure the TLS encrypted syslog server with the server private key, and certificate. The following example shows the successful configuration of a Linux syslog server with the DHE-RSA-AES256-SHA cipher.

```
openssl s_server -accept 60892 -cert rsacert2048_256_days1095_SAN.pem -key
rsakey2048.pem -cipher DHE-RSA-AES256-SHA

Using default temp DH parameters
Using default temp ECDH parameters
ACCEPT
-----BEGIN SSL SESSION PARAMETERS-----
MHUCAQECAgMBBAIAOQQgw8qyfvnc6W0z65juN+RuUeurjFO3qVuNXTMDQPdAGdwE
MI6hWek1E/a69dWIJ6VImumyQTTuv90P+8AzwIpb2JHc3MWliE0qZJ6wsFg4jvDQ
Y6EGAgRSXsY3ogQCAgEspAYEBAEAAAA=
-----END SSL SESSION PARAMETERS-----
Shared ciphers:AES128-SHA:AES256-SHA:DHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA
CIPHER is DHE-RSA-AES256-SHA
Secure Renegotiation IS NOT supported
<14>Mar 31 2014 14:58:57 XM82 FIPS mode enabled by operator from console
<14>Mar 31 2014 14:58:57 XM82 CLI CMD: "fips enable common-criteria" from
console
```

6. Configure the device with the IP address of the TLS encrypted syslog server.

```
logging host 192.158.105.82 ssl-port 60892
```

NOTE
The port number should be the same as used in the **openssl s_server** command in step 5.

Testing TLS cipher suites on a Linux Syslog Server

To test a TLS cipher suite, perform the following steps on the Linux syslog server:

1. Generate openssl certificate and key.
2. Configure device with the **ssl-port** *port-number* command.

```
MLXe(config)# logging host 1.1.1.1 ssl-port 60892
```

3. Start Wireshark capture.
4. Run one of the following four commands to test one of the four different cipher suites:

```
openssl s_server -accept 60892 -cert /usr/local/src/rsyslog-7.4.1/contrib/
gnutls/cert.pem -key /usr/local/src/rsyslog-7.4.1/contrib/gnutls/key.pem
-cipher DHE-RSA-AES128-SHA
openssl s_server -accept 60892 -cert /usr/local/src/rsyslog-7.4.1/contrib/
gnutls/cert.pem -key /usr/local/src/rsyslog-7.4.1/contrib/gnutls/key.pem
-cipher DHE-RSA-AES256-SHA
openssl s_server -accept 60892 -cert /usr/local/src/rsyslog-7.4.1/contrib/
gnutls/cert.pem -key /usr/local/src/rsyslog-7.4.1/contrib/gnutls/key.pem
-cipher AES128-SHA
openssl s_server -accept 60892 -cert /usr/local/src/rsyslog-7.4.1/contrib/
gnutls/cert.pem -key /usr/local/src/rsyslog-7.4.1/contrib/gnutls/key.pem
-cipher AES256-SHA
```

5. Perform some action that generates syslog messages on the Brocade device.

The following example shows the successful configuration of a Linux syslog server with the DHE-RSA-AES256-SHA cipher suite on a MLXe that just completed reboot.

```
openssl s_server -accept 60892 -cert /root/server2048.crt -key
/root/server2048.key -cipher DHE-RSA-AES256-SHA
Using default temp DH parameters
Using default temp ECDH parameters
ACCEPT
bad gethostbyaddr
-----BEGIN SSL SESSION PARAMETERS-----
MHUCAQECAgMBBAIAOQQgw8qyfvnc6W0z65juN+RuUeurjFO3qVuNXTMDQPdAGdwE
MI6hWek1E/a69dWIJ6VImumyQTTuv90P+8AzwIpb2JHc3MWliE0qZJ6wsFg4jvDQ
Y6EGAgRSXsY3ogQCAgEspAYEBAEAAAA=
-----END SSL SESSION PARAMETERS-----
Shared ciphers:AES128-SHA:AES256-SHA:DHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA
CIPHER is DHE-RSA-AES256-SHA
Secure Renegotiation IS NOT supported
<14>Oct 16 2013 10:01:01 MLXe8-R1 FIPS mode enabled by operator from console
<14>Oct 16 2013 10:01:01 MLXe8-R1 CLI CMD: "fips enable common-criteria" from
console
<14>Oct 16 2013 10:01:01 MLXe8-R1 CLI CMD: "interface management 1" from
console
<14>Oct 16 2013 10:01:01 MLXe8-R1 CLI CMD: "ip address 1.1.177.31
255.255.240.0" from console
<14>Oct 16 2013 10:01:01 MLXe8-R1 CLI CMD: "enable" from console
<14>Oct 16 2013 10:01:01 MLXe8-R1 CLI CMD: "end" from console
<14>Oct 16 2013 10:01:02 MLXe8-R1 System: Warm start
<14>Oct 16 2013 10:02:08 MLXe8-R1 System: SSL Syslog server 1.1.1.1:60892 is
now active syslog server
```

Appendix

C

MITM Test for Common Criteria

Common criteria provides for the man-in-the-middle (MITM) test using the TLS.

The NetIron Target of Evaluation (TOE) behaves as a TLS server and client and provides for four test cases to test MITM and identify invalid certificates.

Test1. Server Test – Admin Interface

Modify at least one byte in the server's nonce in the server Hello handshake message and verify that the server denies the client's Finished handshake message.

Test2. Client test – Syslog client

Modify the server's selected ciphersuite in the server Hello handshake message to be a ciphersuite not available in the client Hello handshake message. The evaluator verifies that the client rejects the connection after receiving the server Hello message.

Test3. Client test – Syslog client

If a DHE or ECDHE ciphersuite is supported, modify the signature block in the server's KeyExchange handshake message, and verify that the client rejects the connection after receiving the server KeyExchange.

Test4. Client test – Syslog client

Modify a byte in the server Finished handshake message and verify that the client sends a fatal alert upon receipt and does not send any application data.

Test1. Server Test example:

[Linux: unmodified SSL client – openssl s_client] < - > [Linux: MITM tcptunnel] < - > [TOE: unmodified DUT with https]

The Linux box is the same for the openssl s_client, and MITM tcptunnel.

Test 1: The tcptunnel forwards the messages from server unmodified.

```
------------------------------------------------------------------------------------------

[root@centos64-81-160 ~]# openssl s_client -connect 10.20.81.160:61100 -cipher AES256-SHA
-msg

...

>>> TLS 1.2 ChangeCipherSpec [length 0001]
```

```
   01

>>> TLS 1.2 Handshake [length 0010], Finished

   14 00 00 0c 94 56 31 19 b5 ff 24 14 f7 bb ba 3f

<<< TLS 1.2 ChangeCipherSpec [length 0001]

   01

<<< TLS 1.2 Handshake [length 0010], Finished

   14 00 00 0c bc dc c6 8a ec 45 7f 53 fd 14 3b 2d
```

Test 2: The tcptunnel modifies (Exclusive OR byte in the DUT response at offset provided on command line) before forwarding to client

```
-------------------------------------------------------------------------------------------------------------------------------

[root@centos64-81-160 src]# ./tcptunnel --local-port 61100 –remote-port 443
–remote-host=10.20.81.102 –mitm-test=15

MITM: **** mitm-test offset 15

MITM: Changing -70 to -69

[root@centos64-81-160 src]# [root@centos64-81-160 ~]# openssl s_client -connect
10.20.81.160:61100 -cipher AES256-SHA -msg

...

>>> TLS 1.2 ChangeCipherSpec [length 0001]

   01

>>> TLS 1.2 Handshake [length 0010], Finished

   14 00 00 0c 79 a1 9c 9f bd 3f 6c 7c ba 4d 69 e5

<<< TLS 1.2 Alert [length 0002], fatal bad_record_mac

   02 14

140348367210152:error:140943FC:SSL routines:SSL3_READ_BYTES:sslv3 alert bad record
mac:s3_pkt.c:1275:SSL alert number 20

140348367210152:error:140790E5:SSL routines:SSL23_WRITE:ssl handshake
failure:s23_lib.c:177:
```

Tests2, 3, and 4 Client Test example:

The NetIron client needs to install a specific, self-signed server certificate for this test. The NetIron client sends a SSL Fatal Alert message upon detecting an invalid certificate and sends this alert before ending the TLS connection. This is confirmed by viewing the show log command output and conclude that NetIron was correctly detecting the invalid certificates.

Appendix

Syslog Messages

D

The following table lists some of the syslog messages in FIPS mode.

TABLE 14 FIPS syslog messages

Message level	Message	Explanation
Alert	Time is updated by NTP server *ip-address* from NO_CLOCK to <*new time*> GMT+00 <*new date*>	Indicates time is updated by an NTP server.
Alert	Clock Changed from old time <*old time*> GMT+00 <*old date*> to new time <*new time*> GMT+00 <*new date*>	Indicates time is updated using the **clock set** command.
Informational	SSH login by *user* from src IP *ip-address*, src MAC *mac-address* to USER EXEC mode using RSA as Server Host Key.	Indicates entry into the "user exec" mode for all sessions for the mentioned user. Similar message is logged for “privileged exec” mode.
Informational	SSH logout by *user* from src IP *ip-address*, src MAC *mac-address* from USER EXEC mode using RSA as Server Host Key.	Indicates exit from "user exec" mode for all sessions for the mentioned user. Similar message is logged for “privileged exec” mode.
Informational	SSH session for *user* from src IP *ip-address*, MAC *mac-address* in PRIVILEGED EXEC mode has timed out.	Indicates SSH logout has occurred due to timeout. Similar message is logged for “user exec” mode.
Informational	SSH session closed by *user* from src IP *ip-address*, MAC *mac-address* in PRIVILEGED EXEC mode.	Indicates SSH logout has occurred due to termination. Similar message is logged for “user exec” mode.
Informational	SSH session killed for user src IP *ip-address*, MAC *mac-address* in PRIVILEGED EXEC mode.	Indicates SSH logout has occurred because the session was killed.
Informational	Super user login success in console session.	Indicates user has logged in with super user password.
Informational	Logging CLI_CMD operation enabled by *user* from console session. "logging cli-command" by *user* from console.	Indicates audit log command "logging cli-command" is enabled.
Informational	Logging CLI_CMD operation disabled by *user* from console session.	Indicates audit log command "logging cli-command" is disabled.
Informational	"reload" by un-authenticated user from console	Indicates initiation of device reload through console.
Informational	SSL Syslog server *ip-address:portnum* is now connected.	Indicates encrypted syslog server is connected in the server end.
Informational	SSL Syslog server *ip-address:portnum* is now disconnected.	Indicates encrypted syslog server is disconnected in the server end.

TABLE 14 FIPS syslog messages (Continued)

Message level	Message	Explanation
Informational	SSH login by *user* from src IP *ip-address* from src MAC *mac-address* to USER EXEC mode using RSA as Server Host Key. Brocade scp -t file: secondary.sig Brocade transfer to device completed SSH logout by *user* from src IP *ip-address* from src MAC *mac-address* from USER EXEC mode using RSA as Server Host Key.	Indicates SCP transfer.
Informational	*yyyy month dd hh:mm:ss*	The year is always displayed in the four digit yyyy format.
Informational	Error - Incorrect username or password in console session.	Indicates incorrect username or password.
Informational	Brocade(config)#wr m Message: "write memory" by *user* from console.	Audit log will display the commands in expanded form.
Informational	console login by *user* to USER EXEC mode.	Displays all "login" events including the user and session details. Similar message is logged for “logout” events and “privileged exec” mode.
Informational	Module in slot 2 is rebooted due to FIPS HW sec engine KAT failure	The message is displayed on the MP indicating that the module in slot 2 has restarted due to KAT failure on the MP.
Informational	Module in slot 5 is rebooted due to FIPS HW macsec engine KAT failure	The message is displayed on the MP indicating that the module in slot 5 has restarted due to MACsec engine KAT failure on the MP.
Informational	Entropy generated using the HW crypto engine on slot 2 is same as the previous... regenerating	The message indicates that entropy generated is similar to the previously generated entropy. This message is observed after the first failure.
Informational	Module 5 is reset by mgmt (reason: FIPS KAT failure)	This message is displayed on the MP for KAT failure on LP due to the LP reset issue.

Appendix

Running Tasks for different NetIron devices in FIPS mode E

The information available in this appendix is a list of running tasks in both the Brocade NetIron CER and Brocade MLXe series.

TABLE 15 List of MLXe tasks

Task Name	Priority	Description
idle	0	idle collector task to find idle cpu usage
con	27	OS console task
mon	31	OS monitor task
flash	20	flash access task
dbg	30	debug task
boot	29	boot task
main	3	main parent
itc	6	InterTask Communication task
tmr	5	Timer task
ip_rx	5	IP Receive path task
sfm_mgr	9	Switch fabric manager task
scp	5	System control task, interacts with modules, ports
lpagent	5	communicates with various interface modules
console	5	console task
vlan	5	VLAN handler task
mac_mgr	5	MAC manager task
mrp	5	Metro Ring Protocol handler task
vsrp	5	VSRP protocol handler task
erp	5	ERP protocol handler task
mxrp	5	MSRP protocol handler task
snms	5	Aggregator task for AAA, Syslog, Trap, LLDP, FDP, CDP
rtm	5	Route Table Manager task
rtm6	5	IPv6 Route Table Manager task
ip_tx	5	IP Transmit path task
rip	5	RIP protocol handler task
l2vpn	5	L2VPN (VLL, VLL-Local, VPLS) handler task
mpls	5	MPLS protocol manager

TABLE 15 List of MLXe tasks

Task Name	Priority	Description
nht	5	Next Hop table manager task
mpls_glue	5	MPLS task to communicate with internal engine
bgp	5	BGP protocol handler task
bgp_io	5	BGP I/O controller task
ospf	5	OSPF protocol handler task
ospf_r_calc	5	OSPF Route calculation task
isis	5	ISIS protocol task
isis_spf	5	ISIS Shortest Path Forward (SPF) calculation task
mcast	5	Multicast protocol task
msdp	5	MSDP protocol manager
vrrp	5	VRRP protocol manager
ripng	5	RIP for IPv6 manager task
ospf6	5	OSPF for IPv6 manager task
ospf6_rt	5	OSPF for IPv6 route calculation task
mcast6	5	Multicast for IPv6 manager task
vrrp6	5	VRRP for IPv6 manager task
bfd	5	Bidirection Fault Detection (BFD) protocol manager task
ipsec	5	IPSec protocol manager task
l4	5	L4 (ACL, Rate Limit) manager task
stp	5	Spanning Tree Protocol (STP) manager task
gvrp_mgr	5	GVRP protocol manager task
snmp	5	SNMP protocol manager task
rmon	5	RMON SNMP table task
web	5	HTTP and HTTP-S (SSL/TLS) server and client task
lacp	5	LACP protocol manager task
dot1x	5	802.1X protocol manager task
dot1ag	5	802.1ag protocol manager task
loop_detect	5	L2 loop detection task
ccp	5	MCT Cluster Communication Protocol (CCP): P2P sync between peers
cluster_mgr	5	MCT FSM manager for client and peers
hqos	5	Hierarchical QoS manager task
statistcs	5	Statistics collector manager
hw_access	5	Task doing periodic polling of the temp sensors
sfm_mon	8	Soft error related monitoring on SFMs (only applicable to certain part # of hSFM)

TABLE 15 List of MLXe tasks

Task Name	Priority	Description
ntp	5	Network Time Protocol (NTP) manager task
openflow_ofm	5	OpenFlow flow manager task
openflow_opm	5	Openflow protocol manager task
dhcp6	5	DHCP for IPv6 manager task
fid_mgr	5	FID Manager
sysmon	5	System monitor task
ospf_msg_task	6	OSPF message handler task
ssl	5	SSL client task
ssh_0	5	SSH client #1, max 16 clients are allowed

TABLE 16 List of CER, CER-4X, CER-RT-4X tasks

Task Name	Priority	Description
idle	0	idle collector task to find idle CPU usage
con	27	OS console task
mon	31	OS monitor task
flash	20	flash access task
dbg	30	debug task
boot	29	boot task
main	3	main parent
itc	6	InterTask Communication task
tmr	5	Timer task
ip_rx	5	IP Receive path task
scp	5	System control task, interacts with modules, ports
lpagent	5	communicates with various interface modules
console	5	console task
vlan	5	VLAN handler task
mac_mgr	5	MAC manager task
mrp	5	Metro Ring Protocol handler task
vsrp	5	VSRP protocol handler task
erp	5	ERP protocol handler task
mxrp	5	MSRP protocol handler task
snms	5	Aggregator task for AAA, Syslog, Trap, LLDP, FDP, CDP
rtm	5	Route Table Manager task
rtm6	5	IPv6 Route Table Manager task

TABLE 16 List of CER, CER-4X, CER-RT-4X tasks

Task Name	Priority	Description
ip_tx	5	IP Transmit path task
rip	5	RIP protocol handler task
l2vpn	5	L2VPN (VLL, VLL-Local, VPLS) handler task
mpls	5	MPLS protocol manager
nht	5	Next Hop table manager task
mpls_glue	5	MPLS task to communicate with internal engine
bgp	5	BGP protocol handler task
bgp_io	5	BGP I/O controller task
ospf	5	OSPF protocol handler task
ospf_r_calc	5	OSPF Route calculation task
isis	5	ISIS protocol task
isis_spf	5	ISIS Shortest Path Forward (SPF) calculation task
mcast	5	Multicast protocol task
msdp	5	MSDP protocol manager
vrrp	5	VRRP protocol manager
ripng	5	RIP for IPv6 manager task
ospf6	5	OSPF for IPv6 manager task
ospf6_rt	5	OSPF for IPv6 route calculation task
mcast6	5	Multicast for IPv6 manager task
vrrp6	5	VRRP for IPv6 manager task
bfd	5	Bidirection Fault Detection (BFD) protocol manager task
ipsec	5	IPSec protocol manager task
l4	5	L4 (ACL, Rate Limit) manager task
stp	5	Spanning Tree Protocol (STP) manager task
gvrp_mgr	5	GVRP protocol manager task
snmp	5	SNMP protocol manager task
rmon	5	RMON SNMP table task
web	5	SSL/TLS client task
lacp	5	LACP protocol manager task
dot1x	5	802.1X protocol manager task
dot1ag	5	802.1ag protocol manager task
loop_detect	5	L2 loop detection task
ccp	5	MCT Cluster Communication Protocol (CCP): P2P sync between peers
cluster_mgr	5	MCT FSM manager for client and peers
hw_access	5	HW access manager task
ntp	5	Network Time Protocol (NTP) manager task

TABLE 16 List of CER, CER-4X, CER-RT-4X tasks

Task Name	Priority	Description
openflow_ofm	5	OpenFlow flow manager task
openflow_opm	5	Openflow protocol manager task
dhcp6	5	DHCP for IPv6 manager task
sysmon	5	System monitor task
ospf_msg_task	6	OSPF message handler task
ssl	5	SSL client task
lp	3	Virtual LP (VLP) task
LP-I2C	3	Virtual LP (VLP) I2C bus controller access task
ssh_0	5	SSH client #1, max 16 clients are allowed