DELL Technologies

Configuration and Deployment

Dell EMC ECS with Kemp ECS Connection Manager

IP load balancer deployment reference guide

Abstract

This document describes how to configure the Kemp ECS Connection Manager® with Dell EMC ECS™

December 2019

H18066

Revisions

Date	Description
December 2019	Initial release

Acknowledgments

This paper was produced by the Unstructured Technical Marketing Engineering and Solution Architects team.

Author: Rich Paulson

Table of contents

Executive summary

The explosive growth of unstructured data and cloud-native applications has created demand for scalable cloud storage infrastructure in the modern datacenter. ECS is the third-generation object store by Dell EMC designed from the ground up to take advantage of modern cloud storage APIs and distributed data protection, providing active/active availability spanning multiple datacenters.

Managing application traffic both locally and globally can provide high availability (HA) and efficient use of ECS storage resources. HA is obtained by directing application traffic to known-to-be-available local or global storage resources. Optimal efficiency can be gained by balancing application load across local storage resources.

The ECS HDFS client, CAS SDK and ECS S3 API extensions are outside of the scope of this paper. The ECS HDFS client, which is required for Hadoop connectivity to ECS, handles load balancing natively. Similarly, the Centera Software Development Kit for CAS access to ECS has a built-in load balancer. The ECS S3 API also has extensions leveraged by certain ECS S3 client SDKs which allow for balancing load to ECS at the application level.

Dell EMC takes no responsibility for customer load balancing configurations. All customer networks are unique, with their own requirements. It's extremely important for customers to configure their load balancers according to their own circumstance. We only provide this paper as a guide. Kemp or a qualified network administrator should be consulted before making any changes to your current load balancer configuration.

Objectives

This document is targeted for customers interested in a deploying ECS with the Kemp ECS Connection Manager family of ADCs/load balancers.

External load balancers (traffic managers) are highly recommended with ECS for applications that do not proactively monitor ECS node availability or natively manage traffic load to ECS nodes. Directing application traffic to ECS nodes using local DNS queries, as opposed to a traffic manager, can lead to failed connection attempts to unavailable nodes and unevenly distributed application load on ECS.

Audience

This document is intended for administrators who deploy and configure Dell EMC ECS with Load Balancers. This guide assumes a high level of technical knowledge for the devices and technologies described.

1 Solution overview

1.1 ECS Overview

ECS provides a complete software-defined strongly-consistent, indexed, cloud storage platform that supports the storage, manipulation, and analysis of unstructured data on a massive scale. Client access protocols include S3, with additional Dell EMC extensions to the S3 protocol, Dell EMC Atmos, Swift, Dell EMC CAS (Centera), NFS, and HDFS.

Object access for S3, Atmos, and Swift is achieved via REST APIs. Objects are written, retrieved, updated and deleted via HTTP or HTTPS calls using REST verbs such as GET, POST, PUT, DELETE, and HEAD. For file access, ECS provides NFS version 3 natively and a Hadoop Compatible File System (HCFS).

ECS was built as a completely distributed system following the principle of cloud applications. In this model, all hardware nodes provide the core storage services. Without dedicated index or metadata nodes the system has limitless capacity and scalability.

Service communication ports are integral in the Kemp ECS Connection Manager configuration. See Table 1 below for a complete list of protocols used with ECS and their associated ports. In addition to managing traffic flow, port access is a critical piece to consider when firewalls are in the communication path.

For more information on ECS ports refer to the ECS Security Configuration Guide at https://support.emc.com/docu92972_ECS_3.3_Security_Configuration_Guide.pdf

For a more thorough ECS overview, please review ECS Overview and Architecture whitepaper at https://www.dellemc.com/en-af/collaterals/unauth/white-papers/products/storage-1/h14071-ecs-architectural-guide-wp.pdf

Table 1 ECS protocols and associated ports

Protocol	Transfer Protocol or Daemon Service	Port
S3	HTTP	9020
	HTTPS	9021
Atmos	HTTP	9022
	HTTPS	9023
Swift	HTTP	9024
	HTTPS	9025
NFS	portmap	111
	mountd, nfsd	2049
	lockd	10000

1.2 ECS Constructs

Understanding the main ECS constructs is necessary in managing application workflow and load balancing. This section details each of the upper-level ECS constructs.

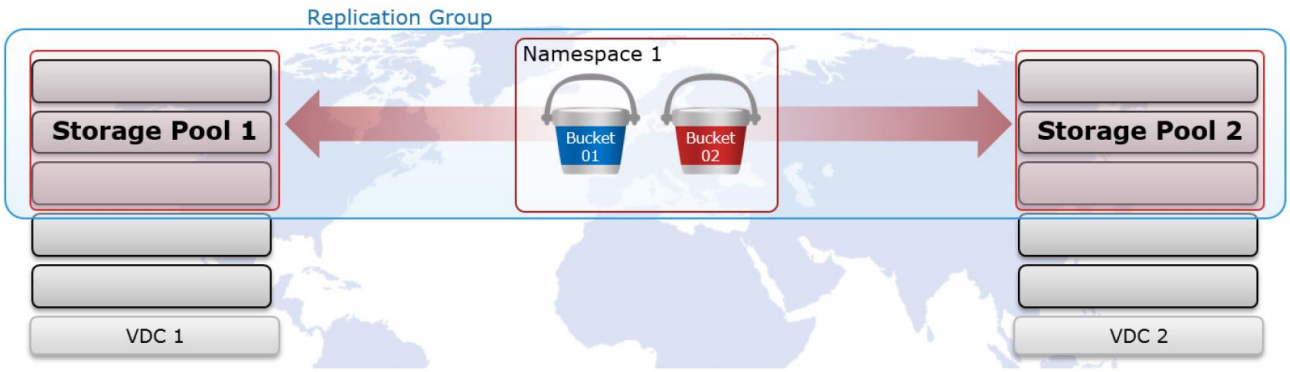


Figure 1. ECS upper-level constructs

Storage Pool - The first step in provisioning a site is creating a storage pool. Storage pools form the basic building blocks of an ECS cluster. They are logical containers for some or all nodes at a site.

ECS storage pools identify which nodes will be used when storing object fragments for data protection at a site. Data protection at the storage pool level is rack, node, and drive aware. System metadata, user data and user metadata all coexist on the same disk infrastructure.

Storage pools provide a means to separate data on a cluster, if required. By using storage pools, organizations can organize storage resources based on business requirements. For example, if separation of data is required, storage can be partitioned into multiple different storage pools. Erasure coding (EC) is configured at the storage pool level. The two EC options on ECS are 12+4 or 10+2 (aka cold storage). EC configuration cannot be changed after storage pool creation.

Only one storage pool is required in a VDC. Generally, at most two storage pools should be created, one for each EC configuration, and only when necessary. Additional storage pools should only be implemented when there is a use case to do so, for example, to accommodate physical data separation requirements. This is because each storage pool has unique indexing requirements. As such, each storage pool adds overhead to the core ECS index structure.

A storage pool should have a minimum of five nodes and must have at least three or more nodes with more than 10% free space in order to allow writes.

Virtual Data Center (VDC) - VDCs are the top-level ECS resources and are also generally referred to as a site or zone. They are logical constructs that represent the collection of ECS infrastructure you want to manage as a cohesive unit. A VDC is made up of one or more storage pools.

Between two and eight VDCs can be federated. Federation of VDCs centralizes and thereby simplifies many management tasks associated with administering ECS storage. In addition, federation of sites allows for expanded data protection domains that include separate locations.

Replication Group - Replication groups are logical constructs that define where data is protected and accessed. Replication groups can be local or global. Local replication groups protect objects within the same VDC against disk or node failures. Global replication groups span two or more federated VDCs and protect objects against disk, node, and site failures.

The strategy for defining replication groups depends on multiple factors including requirements for data resiliency, the cost of storage, and physical versus logical separation of data. As with storage pools, the minimum number of replication groups required should be implemented. At the core ECS indexing level, each storage pool and replication group pairing is tracked and adds significant overhead. It is best practice to create the absolute minimum number of replication groups required. Generally, there is one replication group for each local VDC, if necessary, and one replication group that contains all sites. Deployments with more than two sites may consider additional replication groups, for example, in scenarios where only a subset of VDCs should participate in data replication, but, this decision should not be made lightly.

Namespace - Namespaces enable ECS to handle multi-tenant operations. Each tenant is defined by a namespace and a set of users who can store and access objects within that namespace. Namespaces can represent a department within an enterprise, can be created for each unique enterprise or business unit, or can be created for each user. There is no limit to the number of namespaces that can be created from a performance perspective. Time to manage an ECS deployment, on the other hand, or, management overhead, may be a concern in creating and managing many namespaces.

Bucket - Buckets are containers for object data. Each bucket is assigned to one replication group. Namespace users with the appropriate privileges can create buckets and objects within buckets for each object protocol using its API. Buckets can be configured to support NFS and HDFS. Within a namespace, it is possible to use buckets as a way of creating subtenants. It is not recommended to have more than 1000 buckets per namespace. Generally, a bucket is created per application, workflow, or user

1.3 Kemp ECS Connection Manager Overview

ECS Connection Manager enables scalable and highly available application deployments with a variety of scheduling methods, application level health checking, intelligent content switching and SSL/ TLS acceleration.

An intuitive and easy-to-use web user interface helps to simplify the management of application delivery services for complex environments whether these be in private, public or hybrid clouds. API access allows for seamless integration with modern orchestration and automation frameworks. Comprehensive Layer 7 ADC functionality in a virtual package provides customers with the needed flexibility for today's demanding and dynamic Enterprise application environments.

- Layer 4 and Layer 7 Load Balancing and Cookie Persistence
- SSL Offload/SSL Acceleration
- Application Acceleration: HTTP Caching, Compression & IPS Security
- Full HTTP/2 support
- WAF - Web Application Firewall
- Global Server Load Balancing (GEO)
- Edge Security Pack (Microsoft TMG Replacement)
- Application Health Checking
- Adaptive (Server Resource) Load Balancing
- Content Switching

For more information on the Kemp ECS Connection Manager, visit the Kemp Technologies website

DNS is recommended if you want to either globally distribute application load, or to ensure seamless failover during site outage conditions where static failover processes are not feasible or desirable. Kemp ECS Connection Manager can manage client traffic based on results of monitoring both network and application layers and is largely mandatory where performance and client connectivity is required.

With ECS, monitoring application availability to the data services across all ECS nodes is necessary. This is done using application level queries to the actual data services that handle transactions as opposed to relying only on lower network or transport queries which only report IP and port availability.

1.4 Kemp ECS Connection Manager Constructs

A general understanding of the Kemp ECS Connection Manager constructs is critical to a successful architecture and implementation. The below is a list of the most common ECS Connection Manager constructs:

Virtual Services - The virtual service advertises an IP address and port to the external world and listens for client traffic

Virtual Service Templates - Adding Virtual Services can be both repetitive and prone to error when being performed over multiple ECS Connection Managers. Kemp has developed a general template mechanism that will allow consistency and ease of use when creating Virtual Services.

Using templates to set up and configure a Virtual Service is a two-stage process. Initially the templates must be imported into the ECS Connection Manager. When imported, the templates can then be used when adding a new Virtual Service.

Real Servers - A real server configuration includes the IP address of the individual ECS nodes and port numbers that the real server receives sessions on.

Real Server Check Method - The Kemp ECS Connection Manager utilizes health checks to monitor the availability of the Real Servers. If one of the servers does not respond to a health check within a defined time interval for a defined number of times, the weighting of this server is reduced to zero. This zero weighting has the effect of removing the Real Server from the available Real Servers in the Virtual Service until it can be determined that this Real Server is back online.

Global Balancing - Directs web facing traffic to the closest and fastest performing data center through intelligent DNS responses and provides failover support from a data center suffering from an outage to another data center that has capacity available.

One-Arm Deployment

- The load balancer has one physical network card connected to one subnet
- A Single Ethernet port (eth0) is used for both inbound and outbound traffic
- Real Servers and Virtual Services will be part of the same logical network - sometimes called flat-based - this implies that both have public IP addresses if used for services within the Internet
- Server NAT does not make sense for one-armed configurations
- Does not automatically imply the use of Direct Server Return (DSR) methods on the Real Servers
- IP address transparency will function properly if clients are located on the same logical network as the ECS Connection Manager in a DSR configuration. IP address transparency is not supported when clients are located on the same logical network as the ECS Connection Manager in a NAT configuration.

Two-Arm Deployment

- The load balancer has two network interfaces connected to two subnets - this may be achieved by using two physical network cards or by creating VLANs on a single network interface
- Virtual Services and Real Servers are on different subnets

1.5 Solution architecture

Figure 2 below shows the relationships between applications, virtual services and real servers (ECS nodes).

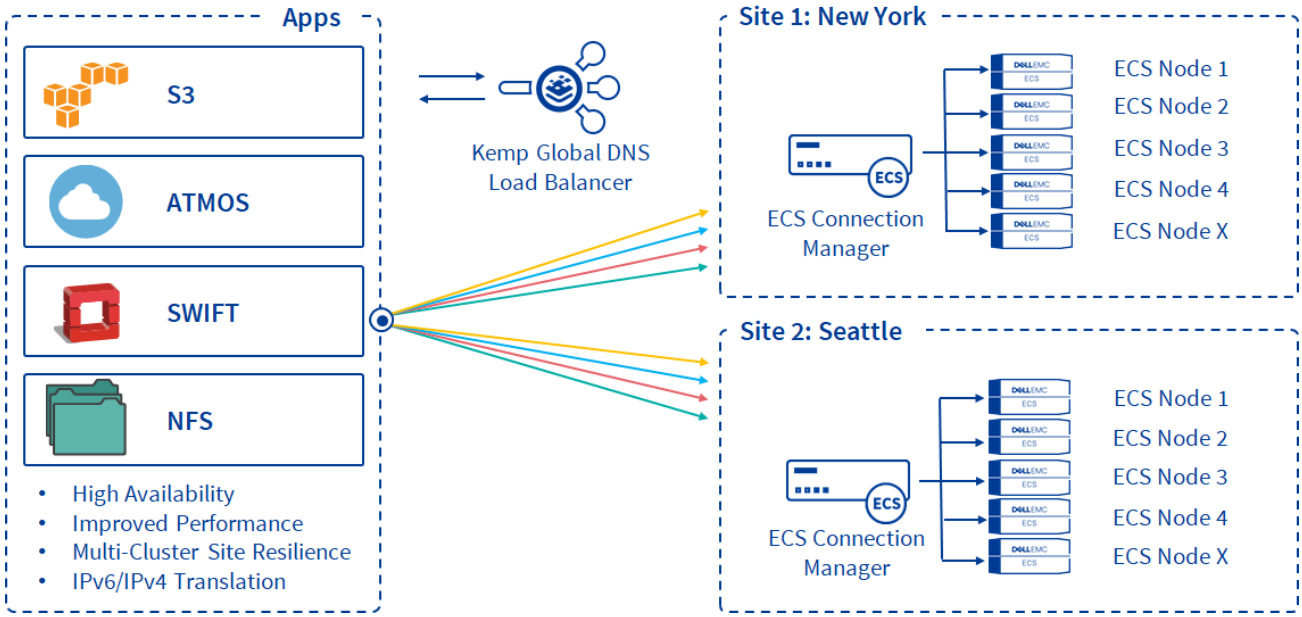


Figure 2. Kemp and ECS architectural overview

1.6 Key components

The following components were used for the examples described below.

Table 2 Components and versions

Component	Version
Dell EMC ECS EX300 Appliance	3.4
ECS Connection Manager H1 Appliance	7.2.49.0.18124

2 Solution implementation

This section describes several deployment options and examples which can be used when deploying a Kemp ECS Connection Manager with Dell EMC ECS.

2.1 Kemp ECS Connection Manager Deployment Options

There are many deployment options based on the environment and customer requirements. This section will cover some of the more common deployments and configuration options.

2.1.1 Single ECS Connection Manager (Virtual or Physical)

Deployments using a single ECS Connection Manager delivers the necessary functionality to load balance Dell EMC ECS but does introduce a single point of failure into the environment. This configuration although supported is not recommended due to the possibility of a system outage should this single unit going offline for either planned or unplanned maintenance.

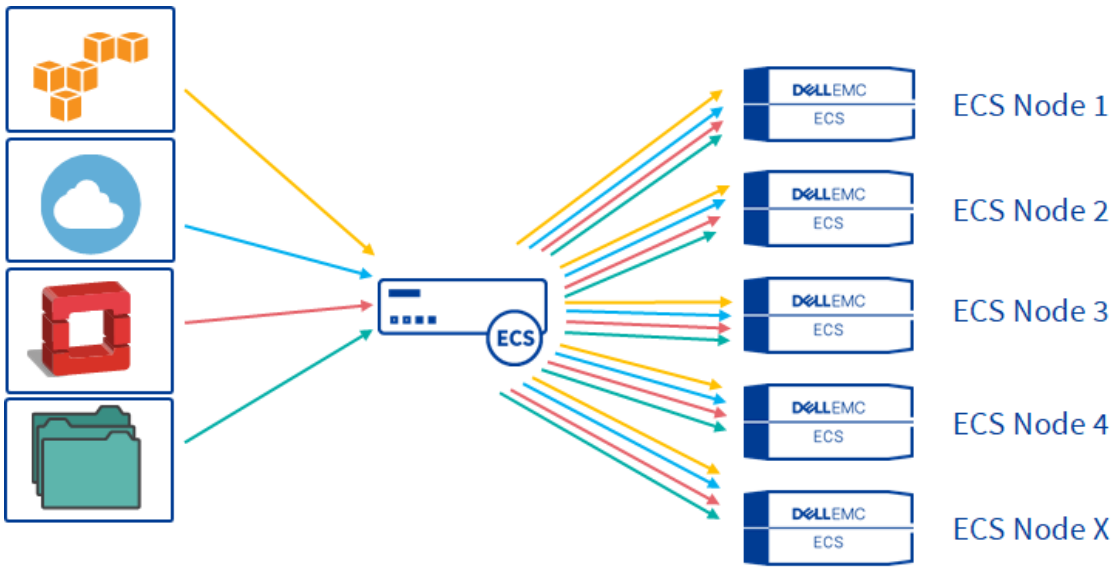


Figure 3. Single Kemp ECS Connection Manager servicing the ECS cluster

2.1.2 ECS Connection Manager in HA Pair (Virtual or Physical)

The recommended deployment for a single site is running the ECS Connection Managers in an active/passive HA configuration. HA enables two physical or virtual machines to become one logical device. Only one of these units is active and handling traffic at any one time while the other unit is a hot standby (passive). This provides redundancy and resiliency, meaning if one ECS Connection Manager goes down for any reason, the hot standby becomes active, therefore avoiding any downtime.

There are some prerequisites to be aware of before setting up HA:

- Two ECS Connection Managers must:
  - Be located on the same subnet.
  - Be in the same physical location.
  - Not be located further than 100 meters from each other.
  - Use the same default gateway.
- A layer 2 connection (Ethernet/VLAN) is required.

- Ensure that any switches do not prevent MAC spoofing. For example, on Hyper-V, go to the network adapter settings in the Virtual Machine settings and select the Enable MAC address spoofing check box.
- Latency on the link between the two ECS Connection Managers must be below 100 milliseconds.
- Multicast traffic flow is required in both directions between the devices. This includes disabling Internet Group Management Protocol (IGMP) snooping on the various switches between the ECS Connection Managers.
- Three IP addresses are required for each subnet in which the ECS Connection Manager is configured.
    - Active unit
    - Standby unit
    - Shared interface
- Use Network Time Protocol (NTP) to keep times on the ECS Connection Managers up-to-date. This ensures that the times are correct on any logs and that Common Address Redundancy Protocol (CARP) message timestamps are in sync.
- Ensure that you have more than one interconnection between the two ECS Connection Managers to avoid data loss or lack of availability.

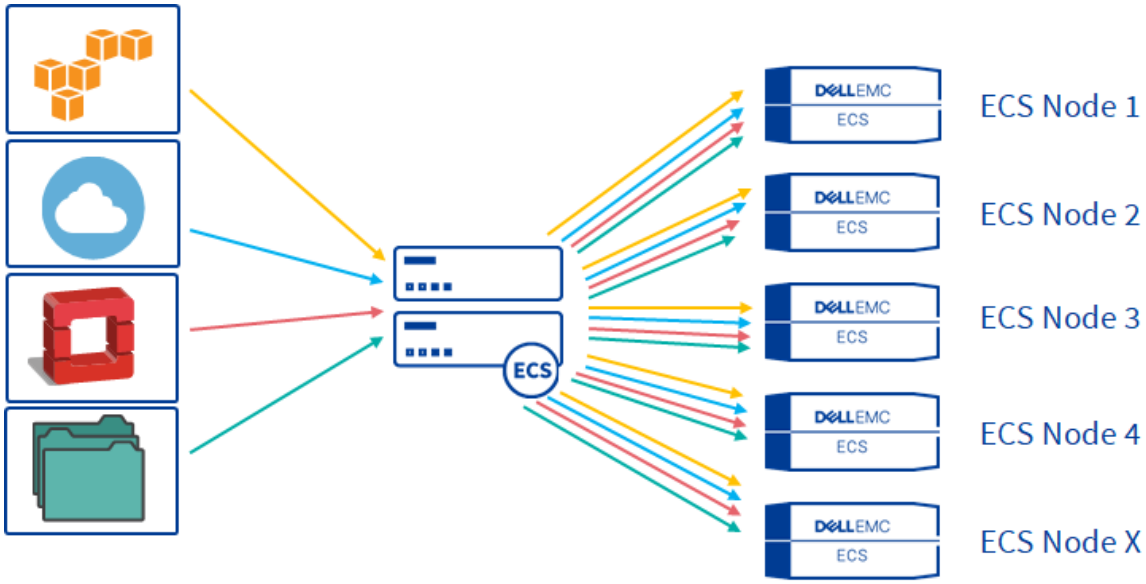


Figure 4. Active/Passive HA Kemp ECS Connection Manager servicing the ECS cluster

2.1.3 Global Server Load Balancing / GEO

When Dell EMC ECS is deployed across multiple locations and there is a requirement for site reliance, Kemp GEO can be leveraged to provide this availability. GEO offers the ability to move past the single data center, allowing for multi data center High Availability (HA). Even when a primary site is down, traffic is diverted to the disaster recovery site. Also included in GEO is the ability to ensure clients connect to their fastest performing and geographically closest data center.

GEO can be deployed in a distributed (active/active) high availability configuration, with multiple GEO ECS Connection Managers securely synchronizing information. Introducing GEO into existing Authoritative Domain Name Services (DNS) requires minimal integration work and risk, allowing you to fully leverage the existing DNS investment.

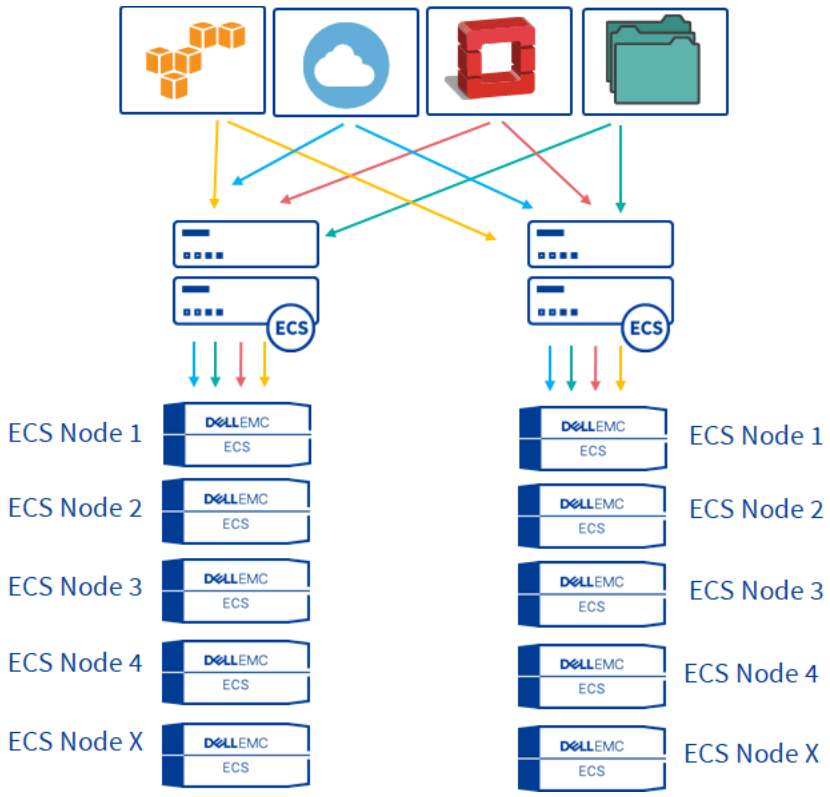


Figure 5. Kemp ECS Connection Manager deployed in a distributed GSLB configuration

2.1.4 ECS Configuration

There is generally no special configuration required to support load balancing strategies within ECS. ECS is not aware of any Kemp ECS Connection Manager systems and is strictly concerned, and configured with, ECS node IP addresses, not, virtual addresses of any kind.

Regardless of whether the data flow includes a traffic manager, each application that utilizes ECS will generally have access to one or more buckets within a namespace. Each bucket belongs to a replication group and it is the replication group which determines both the local and potentially global protection domain of its data as well as its accessibility. Local protection involves mirroring and erasure coding data inside disks,

nodes, and racks that are contained in an ECS storage pool. Geo-protection is available in replication groups that are configured within two or more federated VDCs. They extend protection domains to include redundancy at the site level.

Buckets are generally configured for a single object API. A bucket can be an S3 bucket, an Atmos bucket, or a Swift bucket, and each bucket is accessed using the appropriate object API. As of ECS version 3.2 objects can be accessed using S3 and/or Swift in the same bucket. Buckets can also be file enabled. Enabling a bucket for file access provides additional bucket configuration and allows application access to objects using NFS and/or HDFS.

Application workflow planning with ECS is generally broken down to the bucket level. The ports associated with each object access method, along with the node IP addresses for each member of the bucket's local and remote ECS storage pools, are the target for client application traffic. This information is what is required during Kemp ECS Connection Manager configuration. In ECS, data access is available via any node in any site that serves the bucket. In directing the application traffic to a Kemp ECS Connection Manager virtual service, instead of directly to an ECS node, load balancing decisions can be made which support HA and provide the potential for improved utility and performance of the ECS cluster.

2.2 Implementation workflow

The below example configuration options describe various methods for directing client traffic to Dell EMC ECS using a Kemp ECS Connection Manager.

Example 1: ECS Connection Manager terminated SSL Communication
SSL termination at the load balancer

Example 2: Global Service Load Balancing with Fixed Weighting
Disaster Recovery and HA

Example 3: NFS access via the LoadMaster
Persistent NFS traffic

Example 4: IPv6 to IPv4 Translation
Leverage IPv6 traffic

Example 5: ECS S3 Scheduling for XOR
Leverage storage efficiency with ECS Connection Manager Scheduling

Example 6: Health Monitoring
Monitor health of the ECS nodes

It is recommended, when appropriate, to terminate SSL on the Kemp ECS Connection Manager and offload encryption processing overhead off of the ECS storage. Each workflow should be assessed to determine if traffic requires encryption at any point in the communication path.

Generally, storage administrators use SSL certificates signed by a trusted Certificate Authority (CA). A CA-signed or trusted certificate is highly recommended for production environments. For one, they can generally be validated by clients without any extra steps. Also, some applications may generate an error message when encountering a self-signed certificate. In our example we generate and use a self-signed certificate.

Both the Kemp ECS Connection Manager and ECS software have mechanisms to produce the required SSL keys and certificates. Private keys remain on the Kemp ECS Connection Manager and/or ECS. Clients must have a means to trust a device's certificate. This is one disadvantage to using self-signed certificates. A self-signed certificate is its own root certificate and as such client systems will not have it in their cache of known

(and trusted) root certificates. Self-signed certificates must be installed in the certificate store of any machines that will access ECS.

Note: Local applications may use the S3-specific application ports, 9020 and 9021. For workflows over the Internet it is recommended to use ports 80 and 443 on the front end and ports 9020 and 9021 on the backend. This is because the Internet can handle these ports without problem. Using 9020 or 9021 may pose issues when used across the Internet.

2.3 Installation and configuration steps

2.3.1 ECS Connection Manager-terminated SSL Communication

The simplest and most common use is for the client to the Kemp ECS Connection Manager traffic to be encrypted and ECS Connection Manager to ECS is not. In this scenario the ECS Connection Manager offloads the CPU-intensive SSL processing from ECS.

Here is a general overview of the steps we'll walk through in this example:

Step 1: Create an SSL key and self-signed certificate using OpenSSL.
Step 2: Import the certificate to the ECS Connection Manager.
Step 3: Create a virtual server for ECS Connection Manager terminated SSL connectivity to a single ECS cluster
Step 4: Test connectivity to ECS

Step 1: Create the SSL key and self-signed certificate

OpenSSL is used in this example to generate the certificates. Note that certificate generation can be accomplished on any system with suitable tools like OpenSSL. By default, OpenSSL is installed on most Linux releases.

Note: A Client Signed Request (CSR) can generated on the ECS Connection Manager and provided to a Certificate Authority to obtain the valid certificate. This is the recommended way to generate CSRs. Reference the Appendix for the steps to generate a CSR.

Note: When an SSL-enabled Virtual Service is configured on the ECS Connection Manager and no certificate is specified, a self-signed certificate is installed automatically.

For the purposes of our example we'll be generating a self-signed certificate using OpenSSL. The general steps to create a certificate using OpenSSL are as follows:

a. Generate a private key.
b. Modify the configuration file to add Subject Alternative Names (SANs).
c. Generate a self-signed certificate.

Create an SSL key and self-signed certificate using OpenSSL

A private key is required for self-signed and CA requests certificates. An example of how to generate the private key is shown in the below figure. Permissions are modified to safeguard from accidental modification or deletion.

```
# openssl genrsa -des3 -out server.key 2048
Generating RSA private key, 2048 bit long modulus
.....................+++
.....................................................+++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:

# chmod 0400 server.key
```

OpenSSL does not allow passing of SANs through the command line so a configuration file is created to define them. A sample configuration file for openssl can be used as a template.

Copy the openssl.cnf file to a temporary directory where the certificate will be generated.

```
# cp /etc/pki/tls/openssl.cnf request.cnf
```

Edit the request.cnf file to include the SANs by adding the IP addresses or DNS entries.

```
[ alternate_names ]
DNS.1 = s3.ecstme.org
DNS.2 = atmos.ecstme.org
DNS.3 = swift.ecstme.org
IP.1 = 10.246.156.204
```

In the [req] section, add the following lines if not present in the configuration file as shown in the below figure

```
X509_extensions     = V3_ca   # the extensions to add to the self-signed cert
req_extensions      = V3_ca     # for cert signing request
```

In the [v3_ca] section, add the following lines as shown in the below figure. This indicates that there are alternate names provided.

```
[ V3_ca ]
subjectAltName      = @alternate_names
basicConstraints    = CA:FALSE
keyUsage            = nonrepudiation, digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth
```

Finally in section [CA_default], uncomment or add the copy_extension line as pictured in the below figure.

```
copy_extension = copy
```

Create the Self-Signed Certificate

The command to create the self-signed certificate is shown in the below figure. Note that we set the Common Name to "*.s3.ecstme.org" to support an S3 wildcard DNS entry. The validity of this certificate by default is one month, if more days are desired, specify the command with "-days <# of days> (i.e." –days 365").

```
# openssl req -x509 -new -key server.key -config request.cnf -out server.crt
Enter pass phrase for server.key:
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.
-----
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:MA
Locality Name (eg, city) [Default City]:Hopkinton
Organization Name (eg, company) [Default Company Ltd]:Dell EMC
Organizational Unit Name (eg, section) []:UDS
Common Name (eg, your name or your server's hostname) []:*.s3.ecstme.org
Email Address []:admin@ecstme.org
```

Verify that the SANs and CN are correct by using the below command

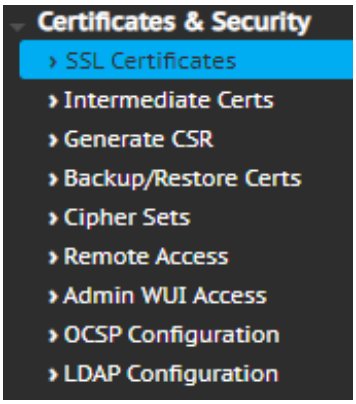
```
# openssl x509 -in server.crt -noout -text
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            ca:cf:6b:7a:32:8d:54:14
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, ST=MA, L=Hopkinton, O=DellEMC, OU=UDS,
CN=*.s3.ecstme.org/emailAddress=admin@ecstme.org
        Validity
            Not Before: Sep 13 01:59:03 2018 GMT
            Not After : Oct 13 01:59:03 2018 GMT
        Subject: C=US, ST=MA, L=Hopkinton, O=DellEMC, OU=UDS,
CN=*.s3.ecstme.org/emailAddress=admin@ecstme.org
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:ea:21:39:3e:76:b1:07:50:47:5d:8c:2a:22:08:
                    1b:c8:84:f2:1b:78:5a:4c:88:13:ca:a0:2a:3d:20:
                    12:be:38:57:5a:fa:04:a8:c2:f6:fd:42:a3:56:de:
                    27:ea:6d:9f:23:53:81:da:82:44:5f:8b:c8:09:c6:
                    d4:1b:03:24:99:d7:c8:18:40:5f:3d:a1:b6:39:6e:
                    ac:0a:40:e3:7e:d5:1a:89:af:c7:40:4f:5e:df:6c:
                    10:c6:27:a4:3d:3b:65:66:60:84:8d:5a:0e:c5:d0:
                    37:5f:46:8e:c9:97:28:74:6d:b7:2c:61:a7:d8:4d:
                    8d:20:3e:35:43:42:ea:90:69:f2:6e:e4:fb:62:d6:
                    4d:4a:72:ce:d0:dc:b3:83:8f:b8:36:1a:31:ef:ce:
                    a0:a4:9b:d4:2d:d7:d5:31:f3:f5:42:86:4a:20:15:
                    d8:a8:74:4f:7b:97:57:9e:bd:31:69:34:04:ba:51:
                    0e:19:4c:02:76:55:be:fc:b0:9a:26:e1:f4:9e:18:
```

```
                d6:9d:82:b1:29:87:17:6e:32:af:39:8a:14:64:1f:
                e5:a1:ec:99:df:69:c3:96:ef:43:f9:26:9a:fa:54:
                2a:67:74:1a:86:b8:6a:17:3e:e6:6e:34:06:0b:df:
                0d:99:72:41:ec:57:45:f8:f7:34:ba:93:4e:c4:79:
                d2:91
            Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Extended Key Usage:
                TLS Web Server Authentication
            X509v3 Subject Key Identifier:
                FB:64:99:11:25:46:E3:9B:B6:A6:FA:C7:DF:D2:8C:3A:27:E6:CD:1F
            X509v3 Key Usage:
                Digital Signature, Non Repudiation, Key Encipherment
            X509v3 Subject Alternative Name:
                DNS:s3.ecstme.org, DNS:atmos.ecstme.org, DNS:swift.ecstme.org, IP Address:10.246.156.204
            X509v3 Basic Constraints:
                CA:FALSE
    Signature Algorithm: sha256WithRSAEncryption
        24:d1:e7:4b:7f:1c:da:39:14:06:8d:87:60:f5:c3:c8:d3:08:
        08:ed:3b:f7:44:56:ff:93:50:d0:74:a9:56:69:2e:e5:09:0d:
        1e:2a:58:e3:51:78:e6:11:1b:c1:e5:ca:f5:ef:73:d8:e0:13:
        76:cb:e0:4c:e4:7c:98:4e:8d:03:31:6e:bb:c1:a2:62:40:6c:
        35:55:af:22:99:e2:9a:eb:62:c4:2e:9a:50:03:81:ea:a9:9e:
        d7:f0:73:e8:00:0f:93:af:d6:d6:6c:28:76:f7:b1:62:00:a5:
        4c:df:6f:cc:df:38:b6:28:3d:67:c3:c8:3e:c1:9b:a8:5d:0a:
        8d:61:3e:71:bc:ec:18:8b:78:9e:ca:74:6a:ce:a3:b5:6f:3a:
        56:2e:36:3e:22:1c:cb:92:87:ce:bc:0a:9d:e1:24:83:7d:6f:
        ec:4a:66:2e:ad:09:48:1d:0e:21:e5:6d:12:c2:39:24:f9:1c:
        35:dc:a0:d2:27:ce:21:e4:8c:dd:37:5f:3c:bc:0f:8b:08:c3:
        c0:bf:83:5e:67:b4:66:77:74:1e:d3:7e:76:e7:a5:a6:46:71:
        94:0f:40:48:6a:cc:15:e6:09:83:09:a5:89:91:af:2b:18:f3:
        98:da:22:73:84:dd:98:97:86:03:b6:21:a0:07:8c:45:6a:39:
        59:bf:a3:44
```

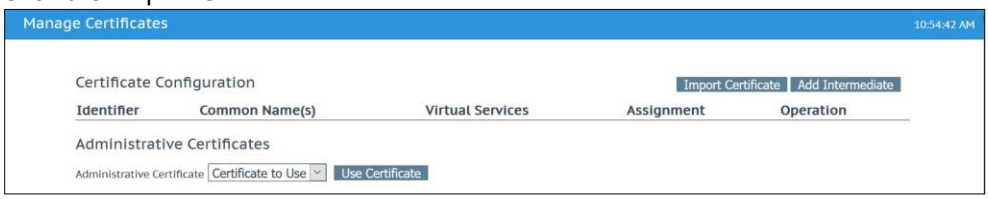
Step 2: Import the certificate into the ECS Connection Manager

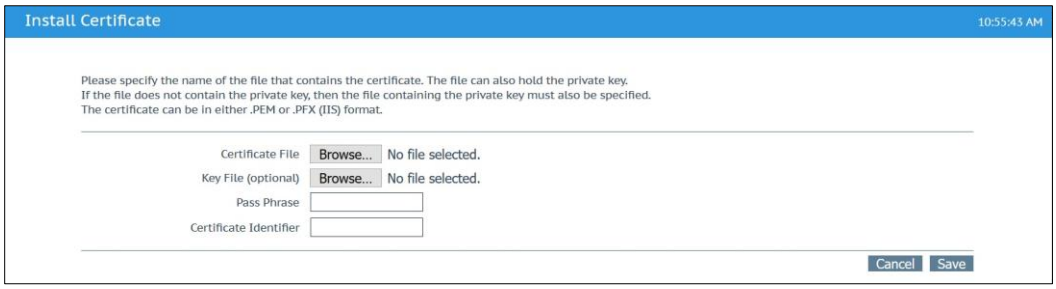
Import the Self-signed certificate to the ECS Connection Manager. The certificate and private key are required for import.

On the ECS Connection Manager, navigate to **Certificates & Security/ SSL Certificates** and select **SSL Certificates**



Click the 'Import Certificate' button

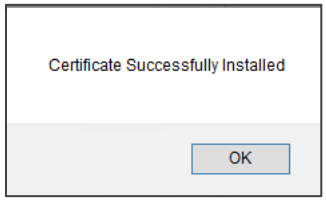




Provide the following:

a. Select **Browse** for Certificate File and select the trusted CA or self-signed certificate
b. Select **Browse** for **Key File** and select the .key file
c. Under **Certificate Identifier**, provide a name for the certificate

Click 'Save'



Click 'OK'

Step 3: Create a virtual server for ECS Connection Manager terminated SSL connectivity to a single ECS cluster

Create a new Virtual Server to publish ECS on port 443. This configuration will provide encryption to the ECS Connection Manager and the ECS Connection Manager will forward the traffic to ECS un-encrypted using the S3 protocol and port 9020.

Within the ECS Connection Manager Web User Interface (WUI), navigate to **Virtual Services/ Add New**

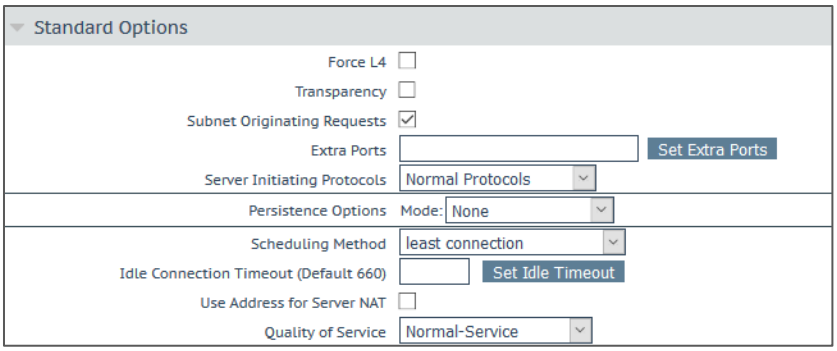
. Enter the required fields:

a. IP address for the Virtual Service
b. Virtual Service Port number (443)
c. Friendly name for the Virtual Service (i.e. S3-HTTPS-Offload)
d. Select TCP as the protocol

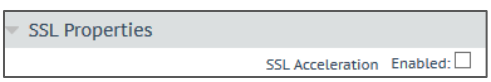
Click **Add this Virtual Service**
Under **Basic Properties** ensure **Service Type** is **HTTP-HTTP/2-HTTPS**

Expand **Standard Options**

a. If ECS Connection Manager is configured with multiple interfaces, be sure to enable **Subnet Originating Requests**
b. Select the appropriate Scheduling Method.

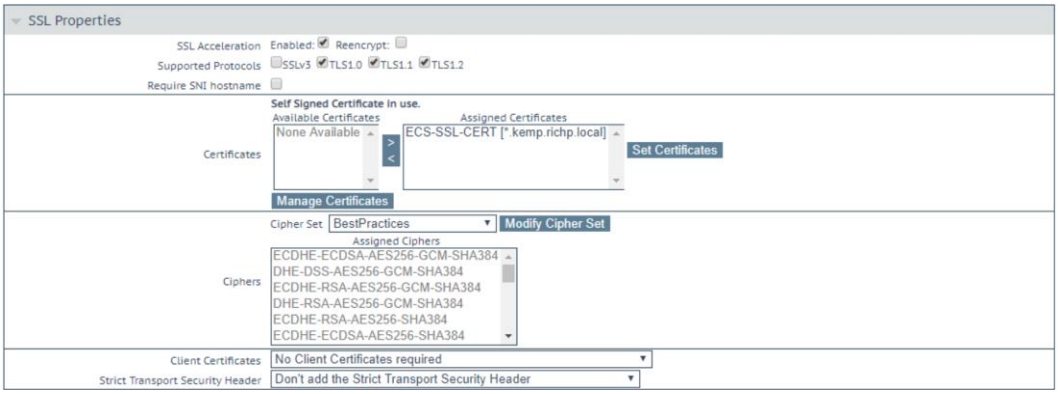


Expand **SSL Properties** and check the box **Enabled** for **SSL Acceleration**



Under the SSL Properties configure the following:

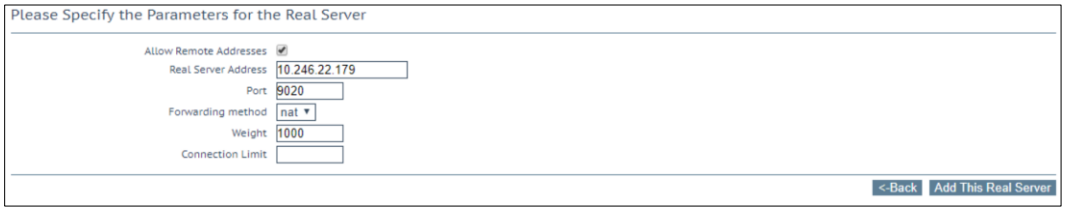
a. Clear the box for **Reencrypt**
b. Check the box for **TLS1.0, TLS 1.1, and** TLS 1.2
c. Select the Certificate that was imported earlier and click the arrow to move it to **Assigned Certificates.**
d. Click **Set Certificates**
e. Select **BestPractices** as the **Cipher Set**



Expand **Real Servers**

Use of the S3 Ping operation is recommended in monitoring ECS S3 service port availability on ECS software running on dedicated ECS hardware. Reference the section of this document labeled 'Health Monitoring' to configure S3 Ping as the **Real Server Check Method** on the ECS Connection Manager

To add the ECS nodes as Real Servers click the **Add New** button



Enter the required fields

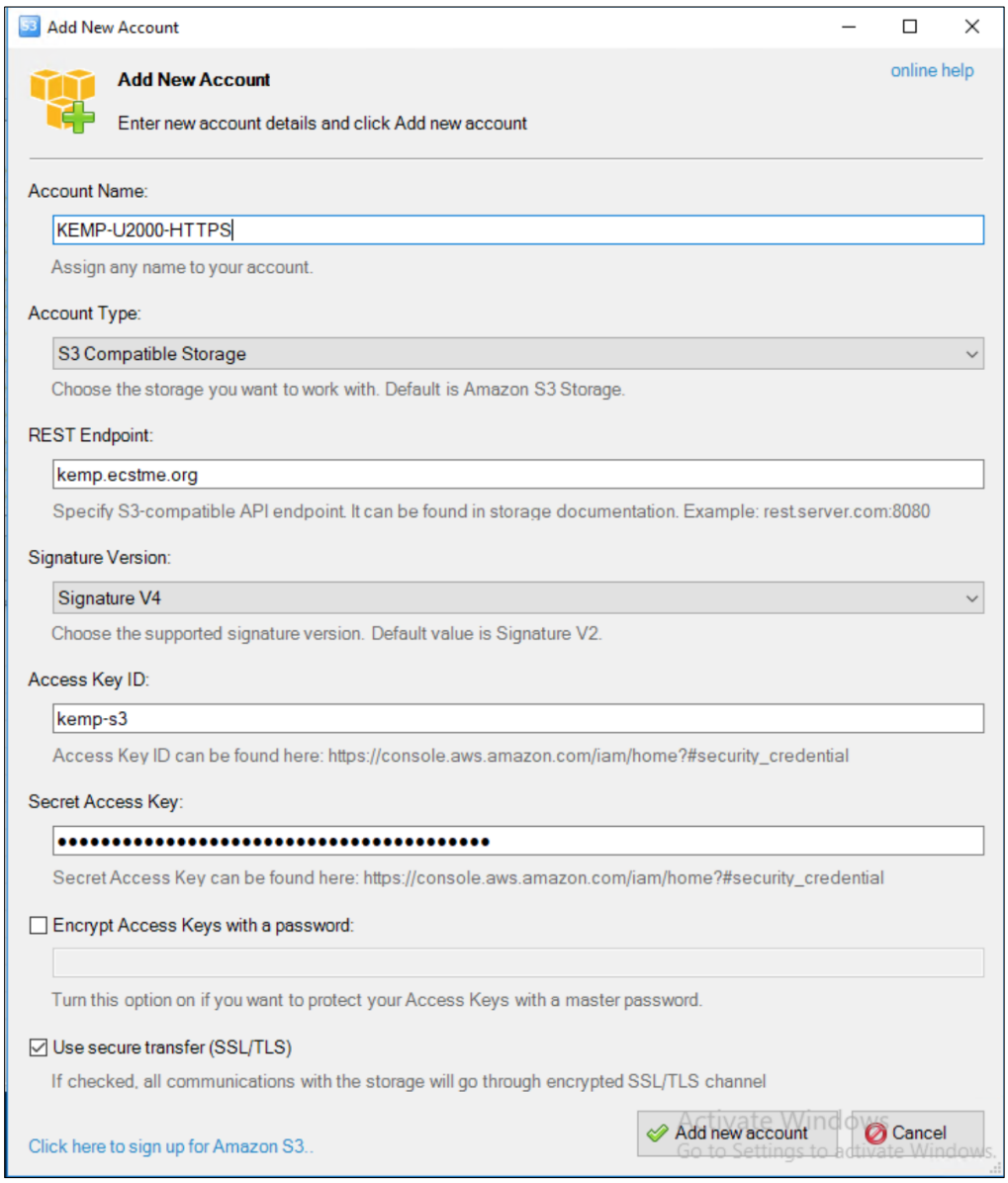
a. For Real Server Address, enter the IP Address of the ECS Node
b. Enter Port 9020
c. Forwarding method NAT
d. Leave the default Weight of 1000
e. Click Add this Real Server

Continue to add each additional ECS node in the cluster to **Real Servers**
Navigate to **Virtual Services/ View Modify Services** to confirm the health of the Virtual Service.

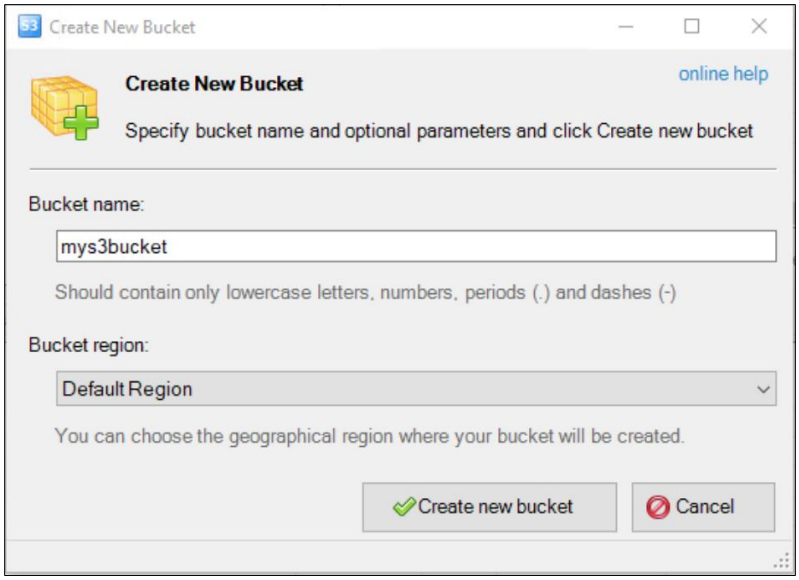
Step 4: Test connectivity to ECS

S3 Browser is a freeware Windows client for Amazon S3 which can be used to connect to ECS.

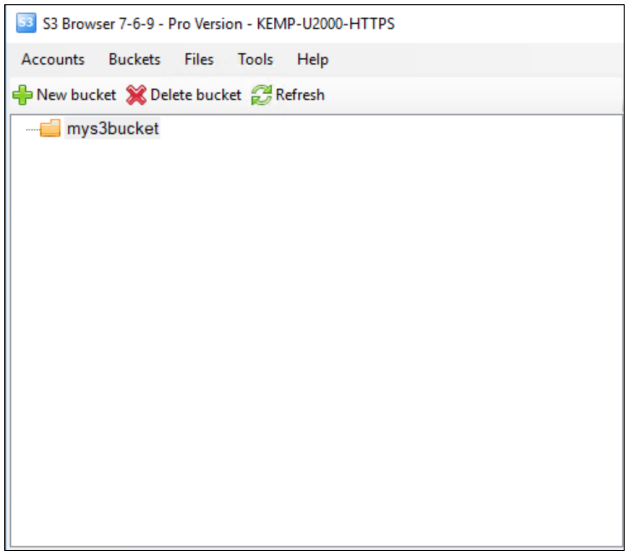
To add a new HTTPS account, select ‘Add new account’ from the Accounts menu item. Provide an account name to identify the connection, select ‘S3 Compatible Storage’, enter the DNS A record that points to the ECS Connection Manager. ECS supports both V2 and V4 Signatures. Lastly, enter you’re ECS Object User ID and S3 password, ensure that ‘Use secure transfer (SSL/TLS) is checked.



Create a new bucket. Specify the bucket name and click **Create new bucket**



Verify that the bucket has been successfully created



As with the virtual servers created for S3 application traffic, virtual servers can be created for Atmos and Swift protocols using the appropriate ports.

2.3.2 Global Service Load Balancing with Fixed Weighting

Fixed weighted scheduling is usually used in Disaster Recovery (DR) sites. The highest weight Real Server is only used when other Real Server(s) are given lower weight values. However, if the highest weighted server fails, the Real Server with the next highest priority number is available to serve clients. The weight for each Real Server should be assigned based on the priority among the remaining Real Servers. When the failed Real Server becomes available, it automatically starts receiving requests.

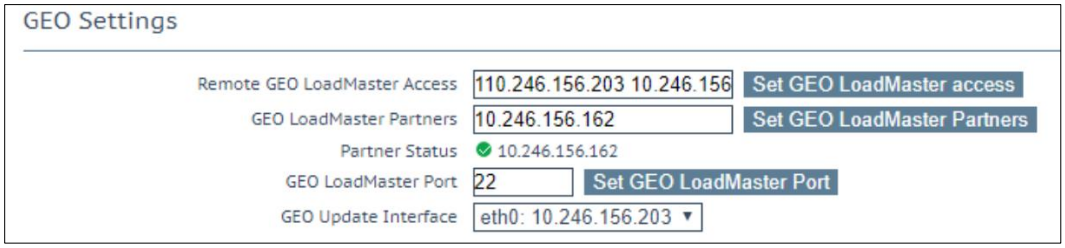
This section will walk through the configuration of Global Server Load Balancing for the ECS Virtual Services on the ECS Connection Manager. Here is a general overview of the steps we'll walk through in this example:

Step 1: Configure Remote GEO ECS Connection Manager Access
Step 2: Set up GEO Clusters
Step 3: Create a Global Balancing Fully Qualified Domain Name
Step 4: Configure Source of Authority (SOA) and Resource Check Parameters
Step 5: Configure DNS Delegation

Step 1: Configure Remote GEO ECS Connection Manager Access

GEO settings can be synchronized between ECS Connection Managers to simplify the configuration. Follow the steps below on each of the ECS Connection Managers in the GSLB configuration. Open the ECS Connection Manager WUI:

1. In the main menu, select **Certificates & Security** and **Remote Access**.



2. Under **GEO Settings** complete the following fields:

   a. For **Remote GEO ECS Connection Manager Access**, type the IP address of all of remote ECS Connection Managers and the local ECS Connection Manager that are included in this GLSB configuration and click **Set GEO ECS Connection Manager Access**.
   b. For **GEO ECS Connection Manager Partners**, type the IP address of just the remote ECS Connection Managers in the GSLB configuration and click **Set GEO ECS Connection Manager Partners**.
   c. For **GEO ECS Connection Manager Port**, keep the default port (22).
   d. Ensure the correct **GEO Update Interface** is selected.

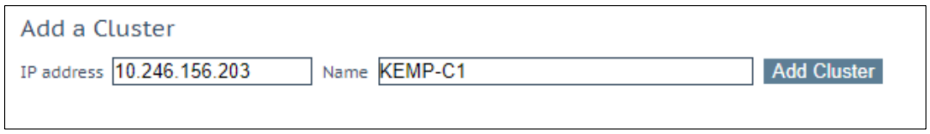
Repeat these steps on the additional ECS Connection Managers. When all ECS Connection Managers are configured with the remote IP address set, the Partner Status should show Green.

Step 2: Set up GEO Clusters

GEO Clustered can be configured to allow for the monitoring of Virtual Services as layer 7. Each ECS Connection Manager in the GSLB configuration must be added to the Cluster. These steps can be done on one of the ECS Connection Managers in the GSLB configuration and will be replicated to the others.

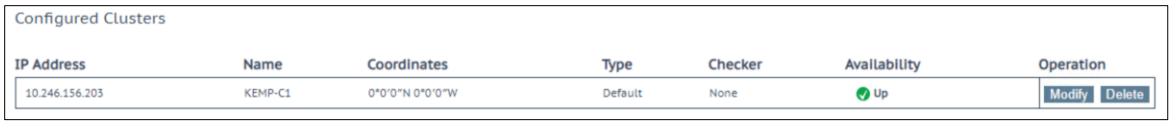
1. In the main menu, select **Global Balancing**.

2. Select **Manage Clusters**.



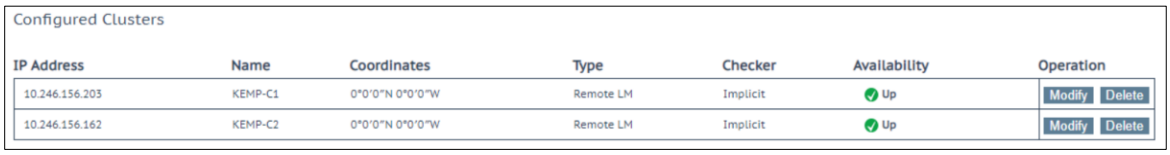
3. Type the **IP address** of one of the ECS Connection Managers in the GLSB configuration and give it a friendly name.

4. Click **Add Cluster**



5. Click **Modify**

6. Change **Type** to **Remote LM**



Repeat these steps on the additional ECS Connection Managers. Once complete all ECS Connection Managers will show Availability as Up.

Step 3: Create a Global Balancing Fully Qualified Domain Name

To configure Global Balancing on the Kemp ECS Connection Manager, follow the steps below in the ECS Connection Manager WUI. These steps can be done on one of the ECS Connection Managers in the GSLB configuration and will be replicated to the others.

1. In the main menu, select **Global Balancing**.

2. Select **Manage FQDN**.

3. Type the Fully Qualified Domain Name (FQDN) of the workload and click **Add FQDN**.

4. Select the preferred **Selection Criteria**. This is going to be set based on requirements, some configurations may require **Location Based/ Proximity** or Round Robin while others can use **Fixed Weighting** for active/passive configurations.

5. Enter the Virtual Service IP address for the one of the ECS Connection Manager and Select the cluster this IP address is associated with.

6. Click **Add Address**.

7. Set **Checker** to **Clustered Checks** and select the configured Virtual Service to monitor under Mapping Menu.

8. Enter the IP address for the Virtual Service on the other ECS Connection Manager and select the associated Cluster.

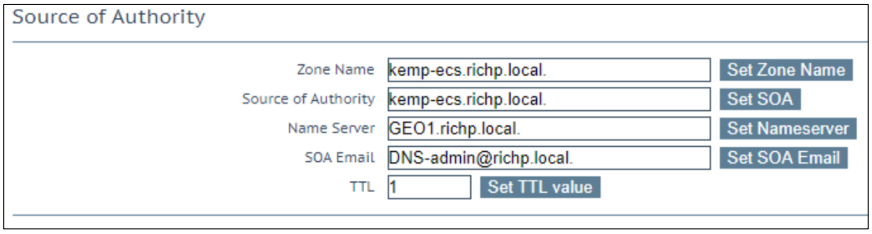
9. Enter the Virtual Service IP address for the one of the ECS Connection Manager and Select the cluster this IP address is associated with.

Repeat these steps for any addition ECS Connection Managers in the GSLB configuration.

Step 4: Configure Source of Authority (SOA) and Resource Check Parameters

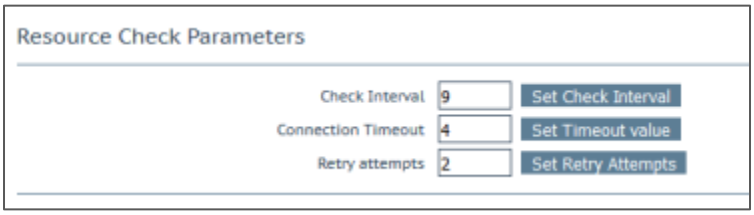
The ECS Connection Managers in this configuration are responsible for resolving the FQDN therefore the SOA parameters should be complete:

1. In the main menu, select **Global Balancing**.
2. Select **Miscellaneous Params**.



3. Configure the following settings:

Field	Example	Description
Zone Name	example.com	The name of the zone when using DNSSEC. This should be left blank if not using DNSSEC.
Source of Authority	kemp-ecs.example.com	The name of the domain owner.
Name Server	GEO1.example.com	The name of the DNS server.
SOA Email	DNS-Admin@example.com	Email address of the DNS administrator (who to contact if there is any issue with the DNS authoritative record).
TTL	1	(Time to Live), which is measured in seconds, defines how long a DNS answer is valid for.



4. Enter the following settings for **Resource Check Parameters**:

Check Interval = 9
Connection Timeout = 4
Retry attempts = 2

5. Enter the following setting for **Stickiness:**

Stickiness = 0

Step 5: Configure DNS Delegation

After the ECS Connection Manager configuration is complete, delegation of DNS must be performed. This differs depending on the DNS provider.

1. A new entry (sometimes called a node) must be created under the parent zone.
2. The node must have both ECS Connection Managers configured as name servers.

2.3.3 NFS via the ECS Connection Manager

It is not recommended to balance NFS traffic load across ECS nodes without using persistence. This is because ECS nodes locally cache specific metadata attributes that NFS clients often query and read ahead (prefetch) NFS file data. These caching mechanisms allow fewer trips to disk which reduces system response time and generally improve sequential read throughput. Load balancing each client's NFS traffic severely reduces the benefits of these ECS cache mechanisms.

A client application should be tied to a single ECS node for the duration of the session. Only during a failure should the connection between client and ECS be moved to another ECS node.

When using ECS Connection Manager to publish NFS traffic, four (4) Virtual Services are required. ECS Connection Manager has a feature known as Port Following which does support the NFS persistence requirements. The following TCP and UDP ports must be configured for NFS traffic:

portmap	**111**
mountd, nfsd	2049
lockd	10000

This section will walk through the configuration of NFS Virtual Services on the ECS Connection Manager. Here is a general overview of the steps we'll walk through in this example:

Step 1: Create and configure Virtual Service for TCP (111,2049,10000)
Step 2: Create and configure Virtual Service for UDP 111
Step 3: Create and configure Virtual Service for UDP 2049

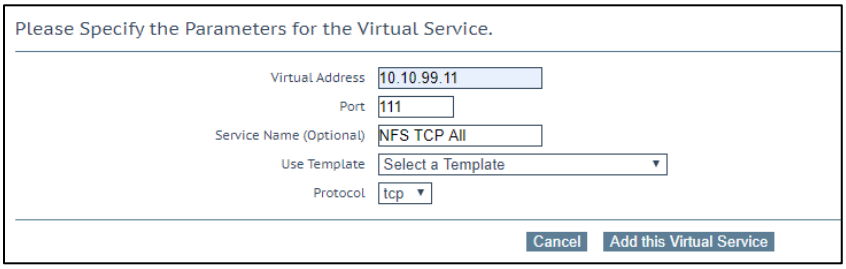
Step 4: Create and configure Virtual Service for UDP 10000

Step 1: Create and configure Virtual Service for TCP (111,2049,10000)

TCP Virtual Services allow for “Extra Ports” to be defined therefore all NFS TCP ports can be configured on one Virtual Service.

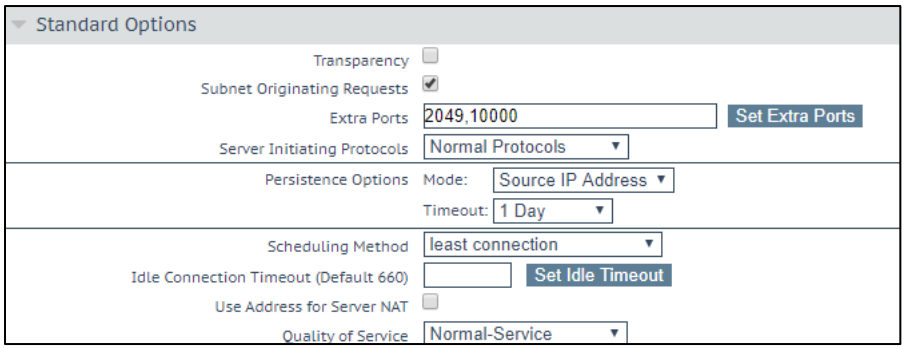
1. In the main menu, select **Virtual Services/ Add New**

   a. Enter the **Virtual IP Address** (VIP)
   b. Enter port **111**
   c. Provide a name for the Virtual Service
   d. Ensure **TCP** is selected for protocol.
   e. Click **Add this Virtual Service**



2. Expand **Standard Options** and enter the following:

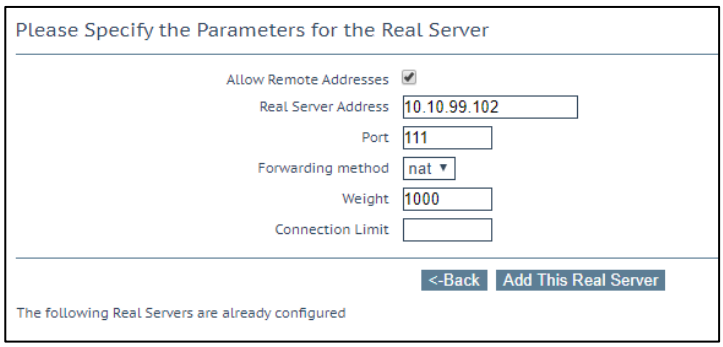
   a. Under **Extra Ports** enter **2049,10000** and click **Set Extra Ports**
   b. Under **Persistence Options** enter **Source IP Address**
   c. Change the **Persistence Timeout** to **1 Day**
   d. Change **Scheduling Method** to **least connection**



3. Expand **Real Servers** and enter the following:

   a. Set **Checked Port** to **111** and click **Set Check Port**
   b. Click **Add New …**
   c. Enter the IP address of an ECS Node
   d. Ensure **Port 111** is entered and click **Add This Real Server**

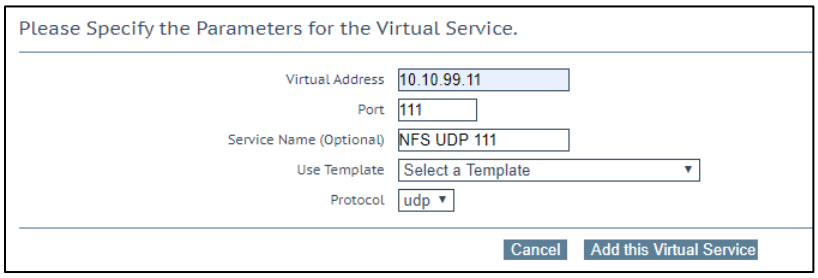
Repeat these steps for all additional ECS Node IP Addresses



Step 2: Create and configure Virtual Service for UDP 111

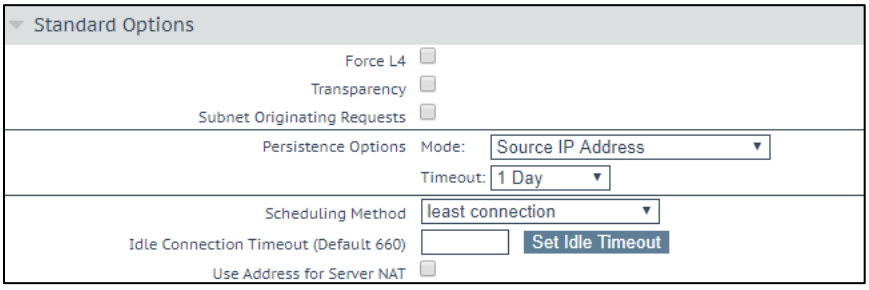
1. In the main menu, select **Virtual Services/ Add New**

   a. Enter the **Virtual IP Address** (VIP)
   b. Enter port **111**
   c. Provide a name for the Virtual Service
   d. Ensure **UDP** is selected for protocol.
   e. Click **Add this Virtual Service**



2. Expand **Standard Options** and enter the following:

   a. Uncheck **Force L4**
   b. Under **Persistence Options** enter **Source IP Address**
   c. Change the **Persistence Timeout** to **1 Day**
   d. Change **Scheduling Method** to **least connection**



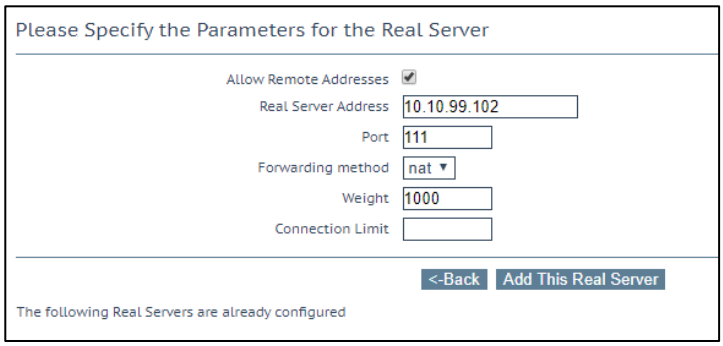
3. Expand **Advanced Options** and enter the following:

   a. Under **Port Following** select the TCP Virtual Service created in Step 1

4. Expand **Real Servers** and enter the following:

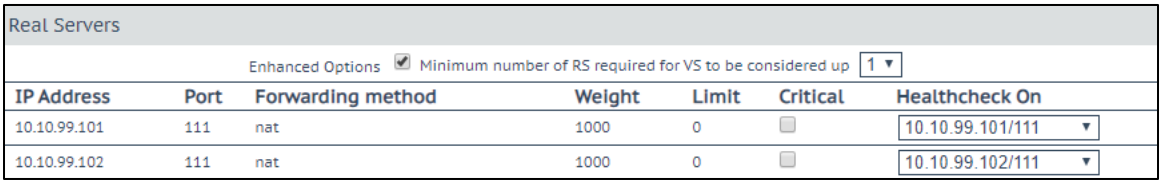
   a. Leave **Real Server Check Method** as **ICMP Ping**
   b. Click **Enhanced Options** check box
   c. Click **Add New …**
   d. Enter the IP address of an ECS Node
   e. Ensure **Port 111** is entered and click **Add This Real Server**

Repeat these steps for all additional ECS Node IP Addresses



5. Once all ECS Nodes have been added, click the back button to return to the **Real Servers** section

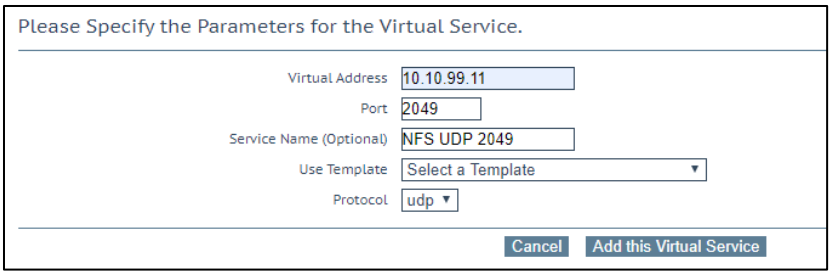
   a. For each Real Server, Under **Healthcheck On**, select the Real Server for the TCP Virtual Service created in Step 1.



Step 3: Create and configure Virtual Service for UDP 2049

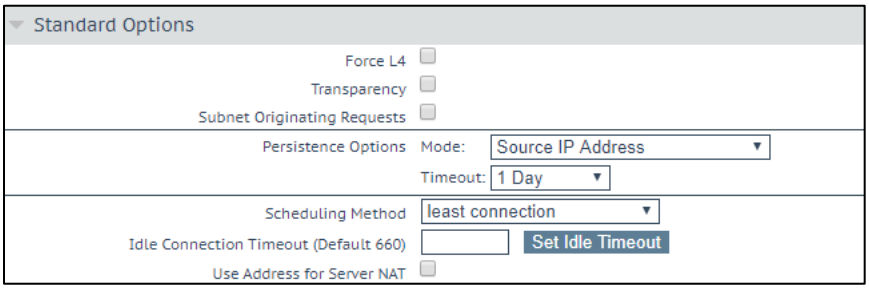
1. In the main menu, select **Virtual Services/ Add New**

   a. Enter the **Virtual IP Address** (VIP)
   b. Enter port **2049**
   c. Provide a name for the Virtual Service
   d. Ensure **UDP** is selected for protocol.
   e. Click **Add this Virtual Service**



2. Expand **Standard Options** and enter the following:

   a. Uncheck **Force L4**
   b. Under **Persistence Options** enter **Source IP Address**
   c. Change the **Persistence Timeout** to **1 Day**
   d. Change **Scheduling Method** to **least connection**



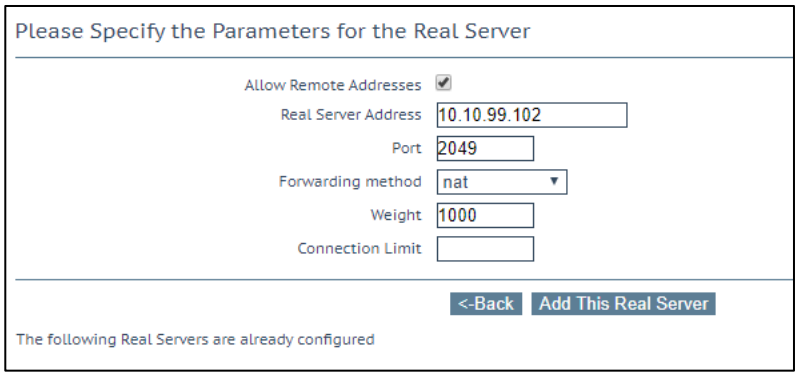
3. Expand **Advanced Options** and enter the following:

   a. Under **Port Following** select the TCP Virtual Service created in Step 1

4. Expand **Real Servers** and enter the following:

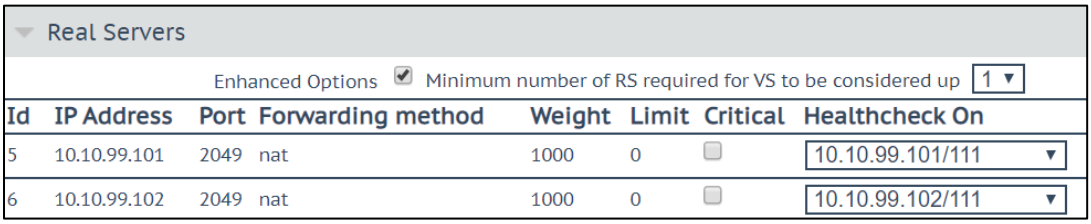
   a. Leave **Real Server Check Method** as **ICMP Ping**
   b. Click **Enhanced Options** check box
   c. Click **Add New …**
   d. Enter the IP address of an ECS Node
   e. Ensure **Port 2049** is entered and click **Add This Real Server**

   Repeat these steps for all additional ECS Node IP Addresses



5. Once all ECS Nodes have been added, click the back button to return to the **Real Servers** section

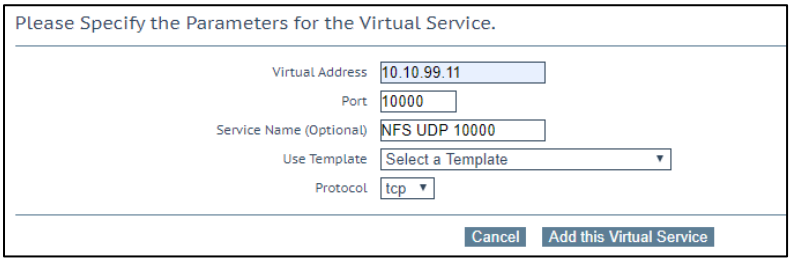
   a. For each Real Server, Under **Healthcheck On**, select the Real Server for the TCP Virtual Service created in Step 1.



Step 4: Create and configure Virtual Service for UDP 10000

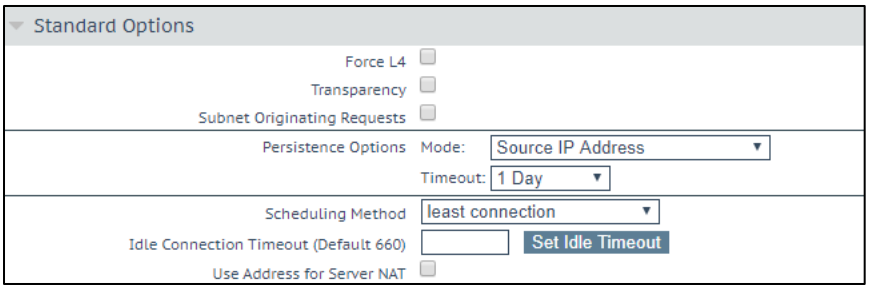
1. In the main menu, select **Virtual Services/ Add New**

   a. Enter the **Virtual IP Address** (VIP)
   b. Enter port **10000**
   c. Provide a name for the Virtual Service
   d. Ensure **UDP** is selected for protocol.
   e. Click **Add this Virtual Service**



2. Expand **Standard Options** and enter the following:

   a. Uncheck **Force L4**
   b. Under **Persistence Options** enter **Source IP Address**
   c. Change the **Persistence Timeout** to **1 Day**
   d. Change **Scheduling Method** to **least connection**



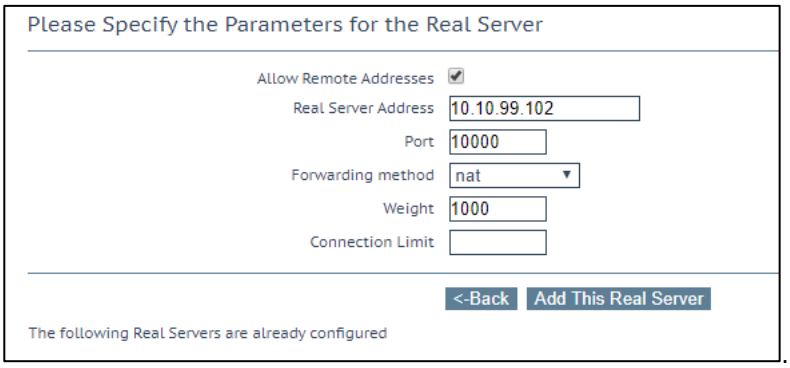
3. Expand **Advanced Options** and enter the following:

   a. Under **Port Following** select the TCP Virtual Service created in Step 1

4. Expand **Real Servers** and enter the following:

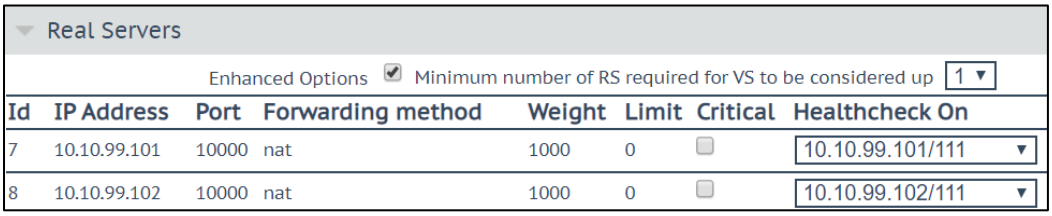
   a. Leave **Real Server Check Method** as **ICMP Ping**
   b. Click **Enhanced Options** check box
   c. Click **Add New …**
   d. Enter the IP address of an ECS Node
   e. Ensure **Port 10000** is entered and click **Add This Real Server**

Repeat these steps for all additional ECS Node IP Addresses



5. Once all ECS Nodes have been added, click the back button to return to the **Real Servers** section

   a. For each Real Server, Under **Healthcheck On**, select the Real Server for the TCP Virtual Service created in Step 1.



2.3.4 IPv6 to IPv4 Translation

IPv6 is the latest version of the Internet Protocol, which identifies devices across the internet, so they can be located. The previous version, IPv4, uses a 32-bit addressing scheme to support 4.3 billion devices however with the explosive growth of the internet, personal computers, and smartphones and now Internet of Things means that the world needs more addresses.

IPv6 was created, which instead uses 128-bit addressing to support 7.9×1028 times as many addresses as IPv4.

Kemp ECS Connection Manager Load Balancers are IPv6 Ready

The ECS Connection Manager can support and is able to translate between IPv4 and IPv6. This does not require the IT department to understand new skills and tricks when handling IPv6 traffic mixed with IPv4 traffic, they do what they should do, deal with the different payloads seamlessly and effectively.

One possible workflow as illustrated below is where the virtual service uses an IPv6 address name/IP to access the ECS appliance nodes which are configured on an IPv4 network.

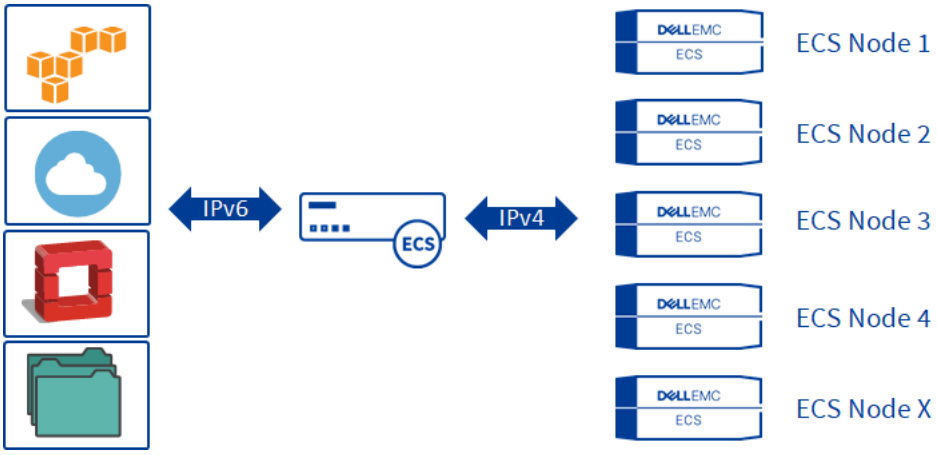


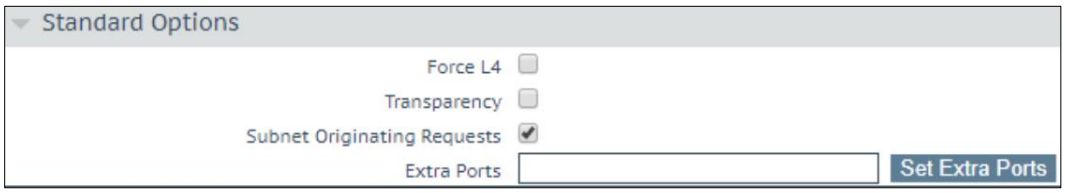
Figure 6. Virtual Service using an IPv6 address with the ECS nodes on IPv4

Virtual IP Address	Prot	Name	Layer	Certificate Installed	Status	Real Servers	Operation
[2018::2]:80	tcp	kemp	L7		Up	10.246.22.163:9020 10.246.22.164:9020 10.246.22.165:9020 10.246.22.166:9020	Modify Delete

Figure 7. Virtual Service configured to use an IPv6 address with the ECS nodes on IPv4

It's recommended to enable 'Subnet Originating Requests' if the ECS Connection Manager is configured with multiple interfaces.

This setting can be configured globally from System Configuration > Miscellaneous Options > Network Options in the main menu or on a per-Virtual Service basis. The below figure shows it as enabled from the virtual service screen.



2.3.5 GEO affinity

Geo-affinity is a feature that focuses on maximizing XOR storage efficiency, while minimizing read-latency impact.

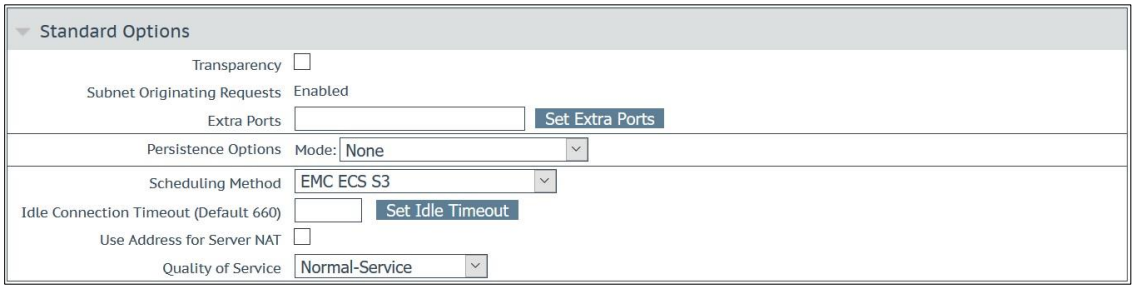
Note: For geo-pinning to work well, you must use a replication group with 3 or more sites, with low latency between sites. If application performance is more important than maximizing storage efficiency, you should **not** use geo-affinity, but instead, make sure your application and the ECS bucket are located in the same site.

To take advantage of the storage efficiencies gained on ECS by XOR, data must be written evenly across 3 or more sites. While writing data evenly across multiple sites leads to increased storage efficiency, reading data in a similar fashion may lead to increased WAN overhead and storage inefficiencies due to caching of remote data. This is because for ECS to provide data that is spread out across multiple sites in a strongly consistent manner, it maintains a record of each object's owner. The object owner is the VDC in which the object was written and serves as the definitive source and ultimate authority for changes to that object. When an object is read from a non-owner site, ECS must communicate with the owner site across the WAN to determine the latest version of the object.

If you can direct applications to the site where an object was originally written, WAN traffic can be minimized and caching of ECS objects at non-owning sites eliminated or dramatically minimized. This results in higher performance for application workflow and minimal caching of remote data

This is where a geo-affinity (or "geo-pinning") algorithm is beneficial. Geo-pinning ensures that all requests for a particular object (be they read or write), are sent to the same site. This feature is built directly in to the Kemp ECS Connection Manager specifically for ECS.

To configure ECS Connection Manager Geo-pinning with ECS, select the Scheduling Method “EMC ECS S3 under the Standard Options section for your Virtual Service.



When an object is written to the ECS VDC, ECS Connection Manager performs a hash of the object’s path, so it knows where the object was written.

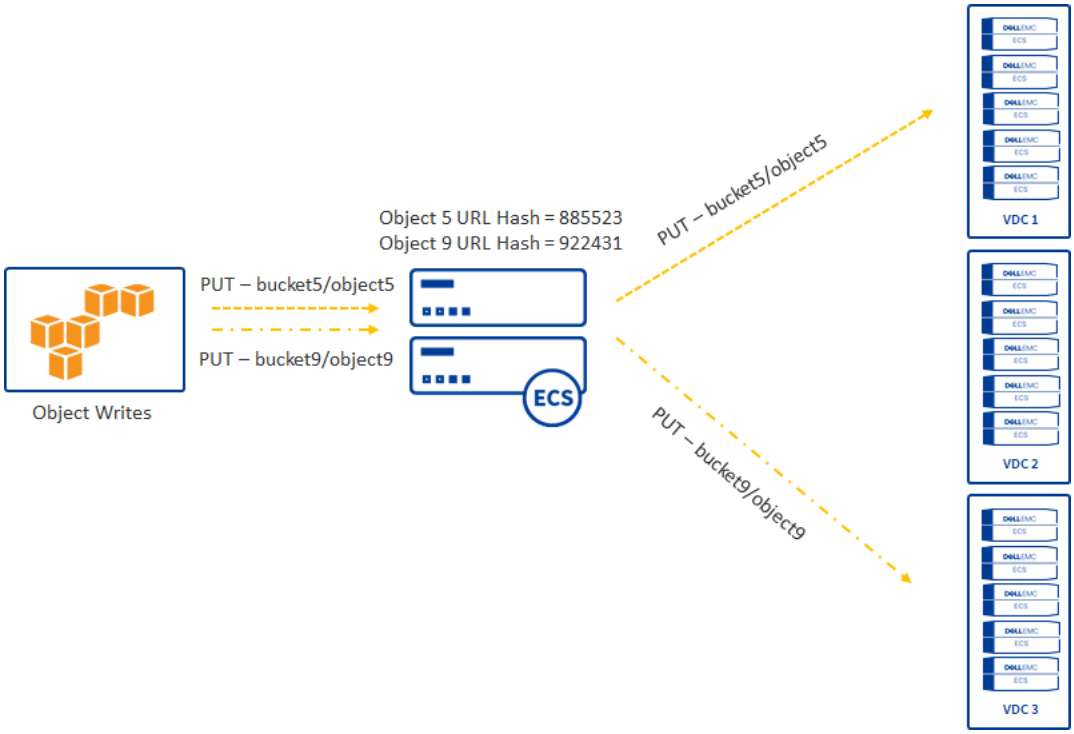


Figure 8. Writing an object using EMC ECS S3

Objects are always read from site where it was originally written. This leads to lower WAN traffic, higher performance, and no caching needed for remote data.

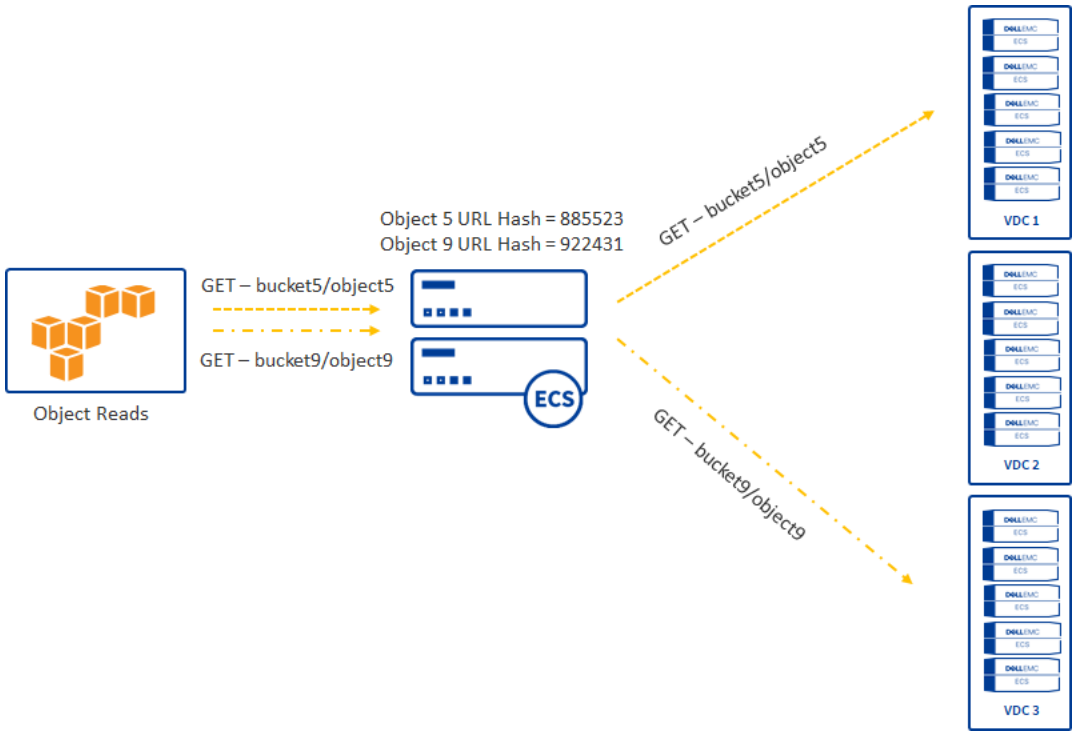


Figure 10. Reading an object using EMC ECS S3

2.3.6 Health Monitoring

The Kemp ECS Connection Manager utilizes health checks to monitor the availability of the Real Servers. If one of the servers does not respond to a health check within a defined time interval for a defined number of times, the weighting of this server is reduced to zero. This zero weighting has the effect of removing the Real Server from the available Real Servers in the Virtual Service until it can be determined that this Real Server is back online.

Use of the S3 Ping operation is recommended in monitoring ECS S3 service port availability on ECS software running on dedicated ECS hardware. This operation is documented inside the Dell EMC ECS REST API Reference Guide which can be found at https://community.emc.com/docs/DOC-73931

The S3 Ping operation is dependent upon the fabric layer inside the ECS software. The fabric layer of the ECS software stack provides clustering and system health among other things. It is responsible for keeping required services up and running and managing resources such as disks, containers, and the network. It tracks and reacts to environmental changes such as failure detection and provides alerts related to system health. The S3 Ping operation uses the fabric layer to determine the state of the node's maintenance mode.

Several different health check types are available however the two types used for the ECS S3 Ping check method are the HTTP and HTTPS protocols.

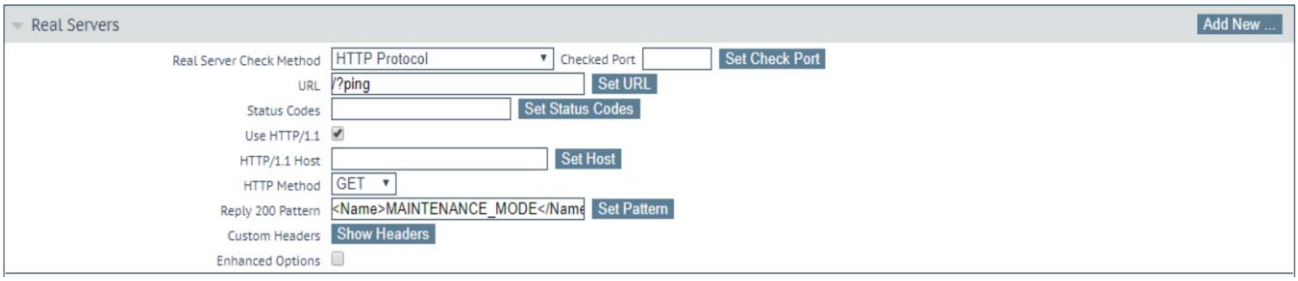


Figure 11. Configuration of the custom ECS health monitor using the HTTP protocol

In the above figure the check method is configured using the HTTP Protocol. The S3 Ping operation is an unauthenticated request, so no username or password is required. The details to configure are as follows:

Real Server Check Method: HTTP or HTTPS Protocol
Use HTTP/1.1 Host: Select the checkbox to enable
HTTP Method: The S3 Ping operation is a GET command
Reply 200 Pattern: Set the pattern to '<Name>MAINTENANCE_MODE</Name><Status>OFF</Status>'.

If the ECS Connection Manager receives a reply pattern from any of the ECS nodes that do not match the Reply 200 Pattern then the node will marked as disabled and automatically re-enabled once the pattern matches.

Note: The S3 Ping operation works with both ports 9020 and 9021. It also works for Atmos services.

Figure 12. Virtual Service displaying an ECS node that is in Maintenance Mode

Enhanced Options can be configured to determine what the minimum number of ECS nodes that are required to be up for the virtual service to be considered up.

If the Enhanced Options check box is disabled (the default), the Virtual Service is considered available if at least one Real Server is available. If the Enhanced Options check box is enabled, you can specify the minimum number of Real Servers that must be available to consider the Virtual Service to be available.

Organizations may choose to consider a VDC unavailable if less than a required minimum number of nodes are up. A three-node minimum may be chosen as it is the minimum number of nodes required for writes to complete. ECS customers decide on their own logic and configuration on how best to accomplish this.

3 Best practices

Utilizing a load balancer with ECS is highly recommended. Highlights of some of the best practices to use when deploying the ECS Connection Manager include:

Table 3 Best practices and recommendations

Recommendation
Do not use the Kemp ECS Connection Manager for CAS traffic. The Centera SDK has a built-in load balancer and cannot function without direct access to all nodes.
Traffic management is best utilized for data traffic
Terminate SSL connections on the Kemp ECS Connection Manager when possible to reduce load on the ECS nodes.
For NFS workloads configure the ECS Connection Manager to keep client sessions terminated on a single ECS node. Only during node failure should an NFS session be torn down and established on another ECS node. Balancing NFS traffic across ECS nodes is inefficient because it doesn't take advantage of ECS caching.
Utilize the ECS Connection Manager “url hash” feature to improve efficiency when using three or more ECS sites

A Technical support and resources

Dell.com/support is focused on meeting customer needs with proven services and support.

Storage technical documents and videos provide expertise that helps to ensure customer success on Dell EMC storage platforms.

A.1 Related resources

Document	Location
Dell EMC ECS product documentation	https://www.dell.com/support/article/us/en/19/sln319451/dell-emc-ecs-3-4-x-product-documentation-index?lang=en
Dell EMC ECS Architecture and Overview	https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage-1/h14071-ecs-architectural-guide-wp.pdf
Dell EMC ECS Networking and Best Practices	https://dellemc.com/resources/en-us/asset/white-papers/products/software/h15718-ecs-networking-bp-wp.pdf
Dell EMC ECS General Best Practices	https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage/h16016-ecs-best-practices-guide-wp.pdf
Dell EMC ECS High Availability Design	https://www.dellemc.com/resources/en-us/asset/white-papers/products/storage-2/h16344-elastic-cloud-storage-ha-design.pdf
Kemp ECS Connection Manager Product Overview Documents	https://support.kemptechnologies.com/hc/en-us/articles/204373265-KEMP-LoadMaster
Kemp ECS Connection Manager GEO Feature Description	https://support.kemptechnologies.com/hc/en-us/articles/203125189-GEO-Feature-Description
Kemp ECS Connection Manager IPv6 Load Balancing	https://kemptechnologies.com/solutions/ipv6-address-load-balancing/
Kemp ECS Connection Manager Dell EMC ECS Template	https://kemptechnologies.com/loadmaster-documentation/