Scality RING and Entrust KeyControl

Integration Guide

2024-11-21

Table of Contents

Chapter 1. Introduction

This document describes the integration of Scality RING Hybrid Cloud Storage Management Solution (referred to as Scality RING in this guide) with the Entrust KeyControl key management solution (KMS). KeyControl serves as a key manager for cloud keys and KMIP objects.

1.1. Product configurations

Entrust has successfully tested the integration of KeyControl with Scality RING in the following configurations:

System	Version
Scality RING	v9.3.0.2
Scaly RING OVA package	scality-OVA-3nodes-9.3.0.2.ova
KeyControl	10.3.1

1.2. Requirements

Before starting the integration process, familiarize yourself with the Scality and KeyControl documentation:

- Scality RING online documentation (authentication required)
- Entrust KeyControl online documentation

Chapter 2. Deploy the Scality RING

For this integration, a three-node Scality RING was deployed on VMware using a pre-packaged environment by Scality, delivered as a single OVA file.

The deployment created five virtual machines:

- Scality Supervisor
- Scality Endpoint
- 3x Scality Storage Node

The following account and user were created for the purpose of this integration at the time of deployment:

Item	Name
Account	`Entrust-KIMP`
User	`Entrust-KIMP-user`

A bucket will be created after the integration is complete, during testing at section Create a bucket with encryption enabled. Note the bucket name.

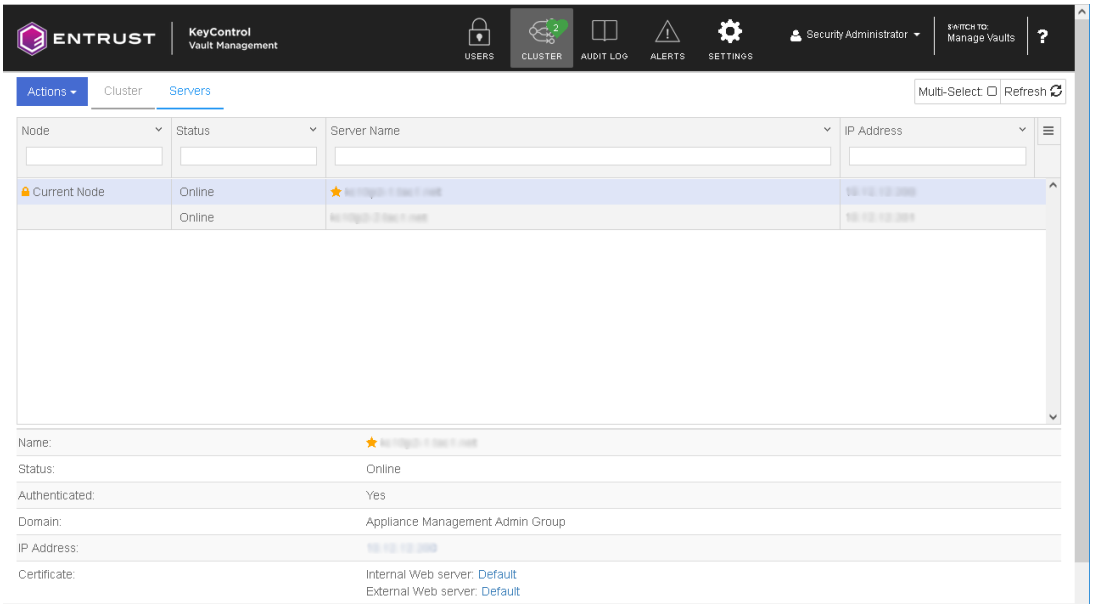
Item	Name
Bucket	`entrust-kmip-bucket`

Chapter 3. Deploy KeyControl

3.1. Deploy a KeyControl cluster

For this integration, a two-node cluster was deployed as follows:

1. Download the KeyControl software from Entrust TrustedCare. This software is available as an OVA or ISO image. This guide deploys an OVA installation.
2. Install KeyControl as described in KeyControl OVA Installation.
3. Configure the first KeyControl node as described in Configuring the First KeyControl Node (OVA Install).
4. Add second KeyControl node to cluster as described in Adding a New KeyControl Node to an Existing Cluster (OVA Install).
5. Both nodes need access to an NTP server, otherwise the above operation will fail.
6. Sign in to the console to change the default NTP server if required.



7. Install the KeyControl license as described in Upgrading Your Trial License.

3.2. Additional KeyControl cluster configuration

After the KeyControl cluster is deployed, additional system configuration can be done as described in KeyControl System Configuration.

3.3. Configure authentication

This guide uses local account authentication.

For AD-managed Security groups, configure the LDAP/AD Authentication Server as described in Specifying an LDAP/AD Authentication Server.

3.4. Create DNS record for the KeyControl cluster

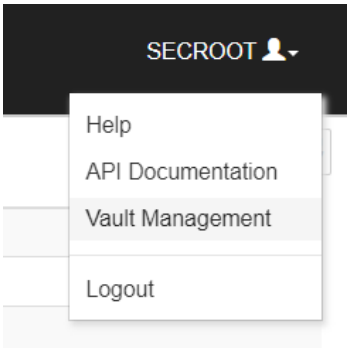
This guide uses the individual IP addresses of the KeyControl nodes.

To use hostnames, configure your DNS server giving each node in the KeyControl a unique name.

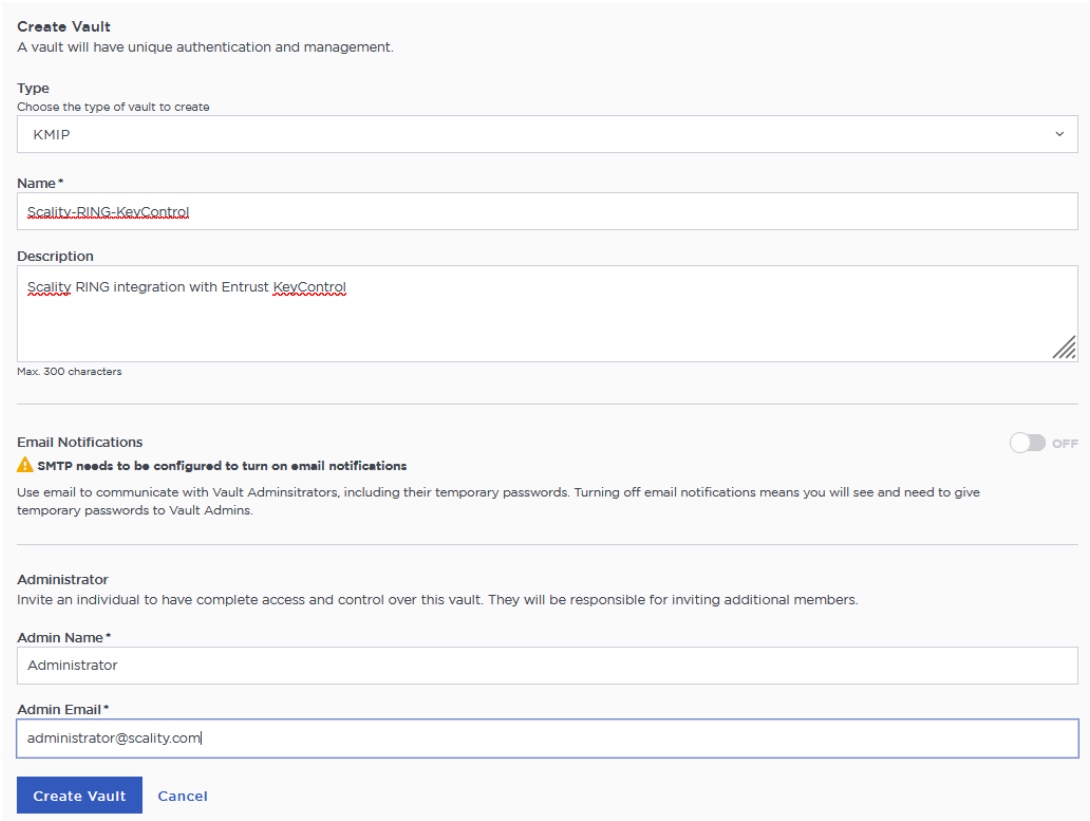
3.5. Create a KMIP vault in KeyControl

The KeyControl Vault appliance supports different types of vaults. This section describes how to create a KMIP vault for this integration.

1. Sign in to the KeyControl Vault Server web user interface using the **secroot** credentials.
2. From the user's dropdown menu, select **Vault Management**.



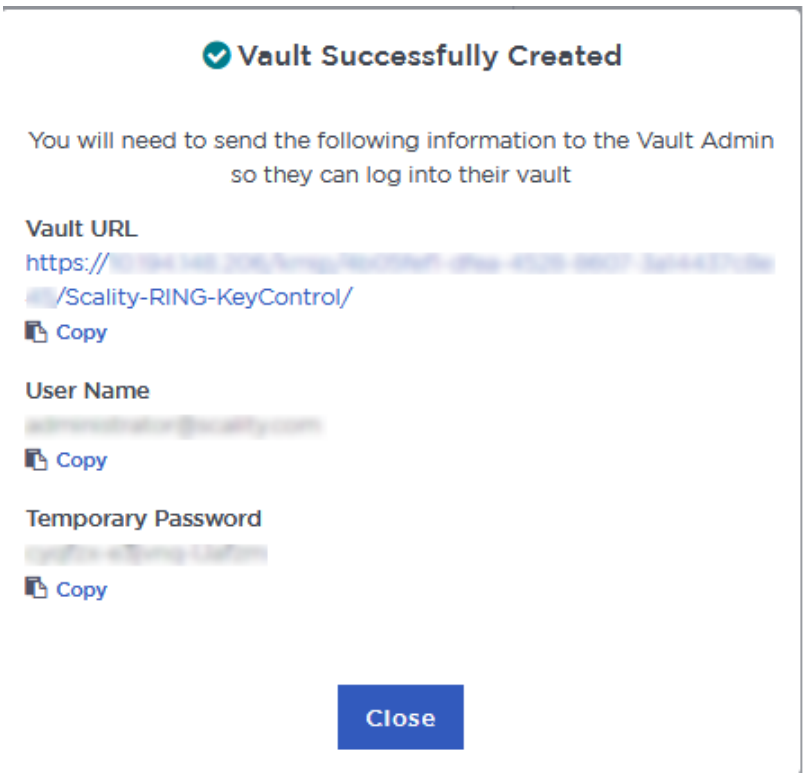
3. Select **Create Vault > KMIP**, then enter your information.



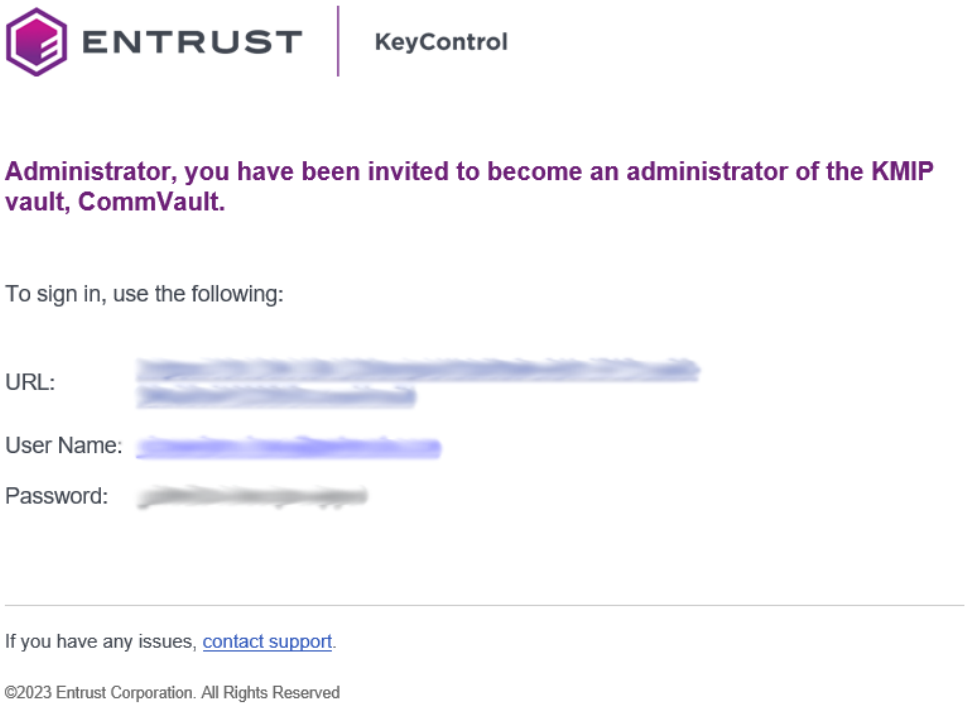
4. Select **Create Vault**, then select **Close**.

   A window with the new vault information appears. In addition, an email with the same vault information is sent to the security administrator **secroot**.

   Example vault information window:

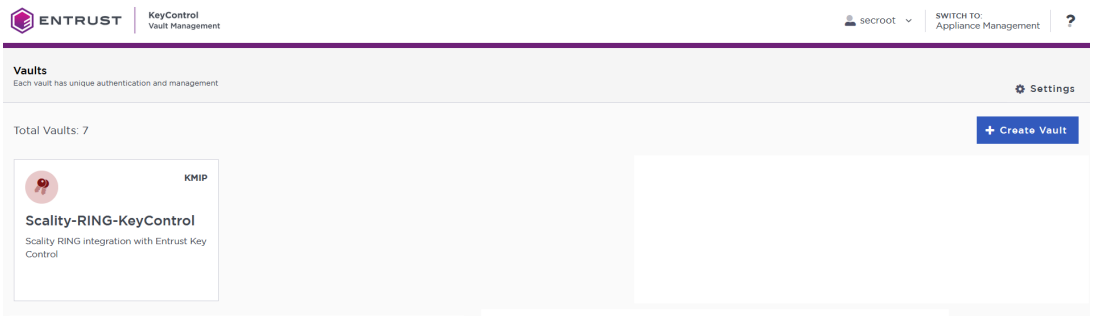


Example email:

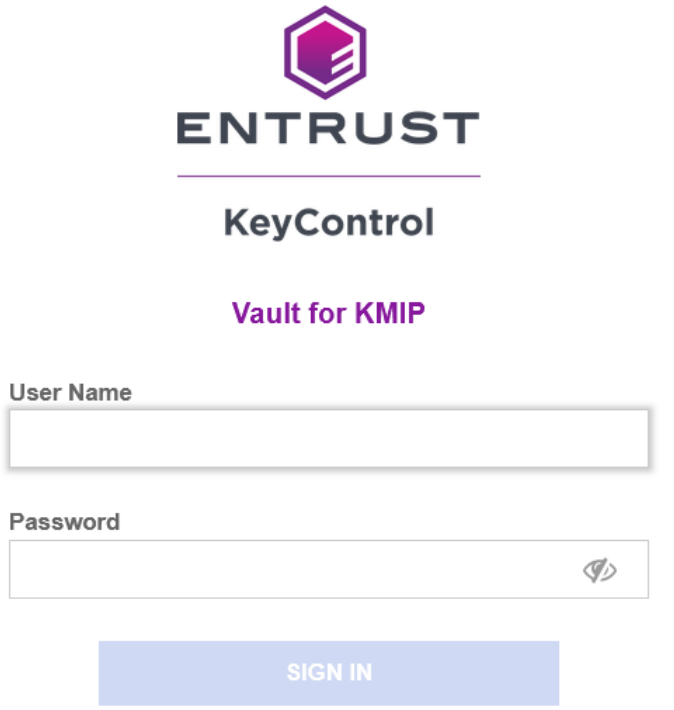


5. Bookmark the **Vault URL** listed above.

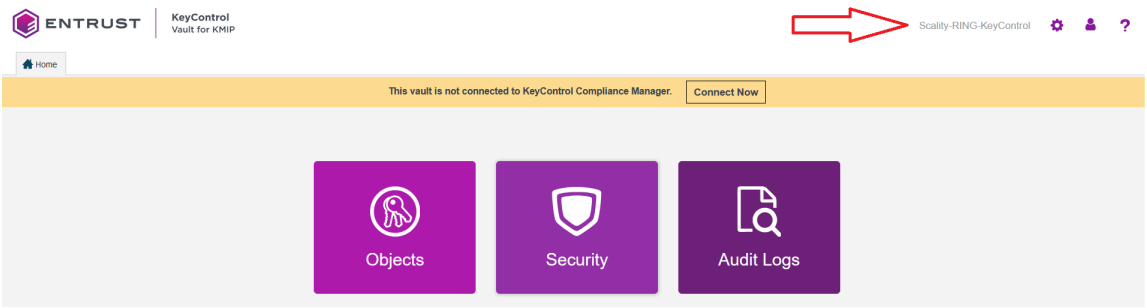
   The new vault is added to the **Vault Management** dashboard.



6. Sign in to the **Vault URL** with the temporary password. Change the temporary password when prompted. Sign in again to verify.



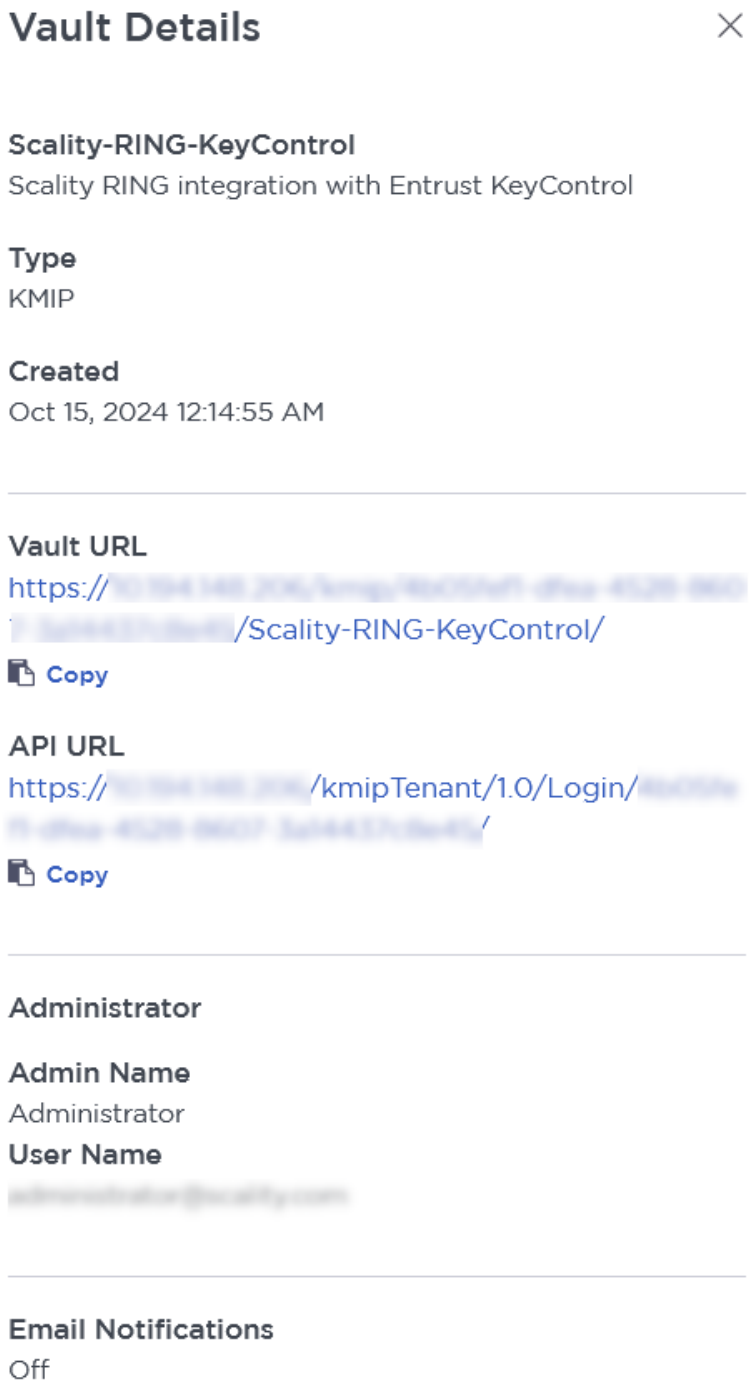
Notice the new vault.



For more information, see Creating a Vault.

3.6. View the KMIP vault details

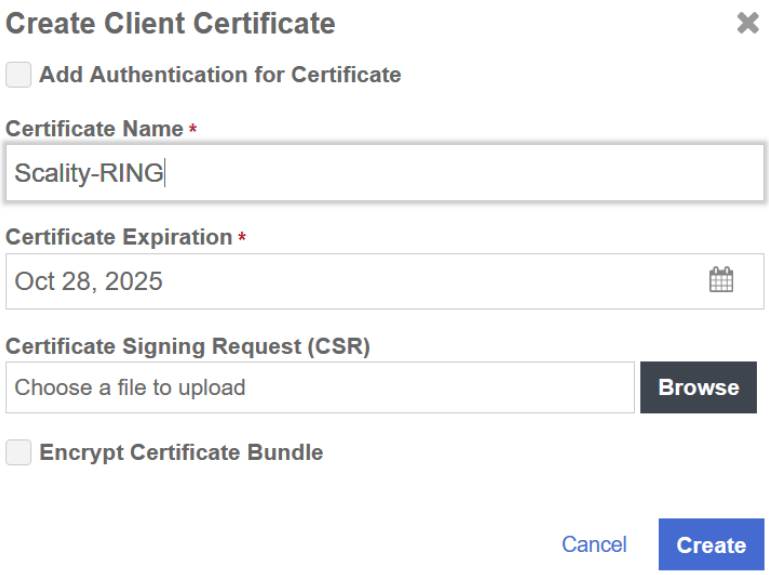
In the **Vault Management** dashboard, hover over the vault and select **View Details**.



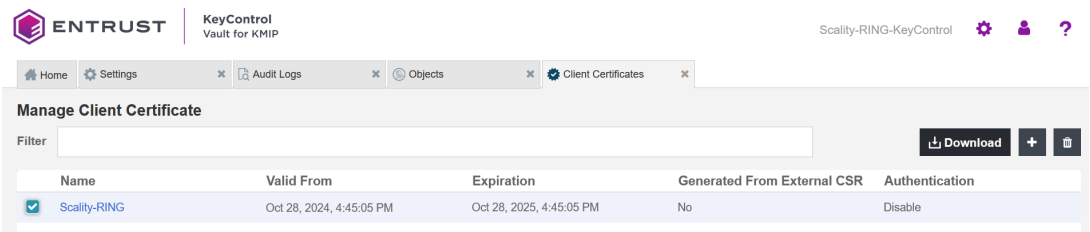
Chapter 4. Integrate Scality RING and KeyControl

4.1. Create the client certificate in KeyControl

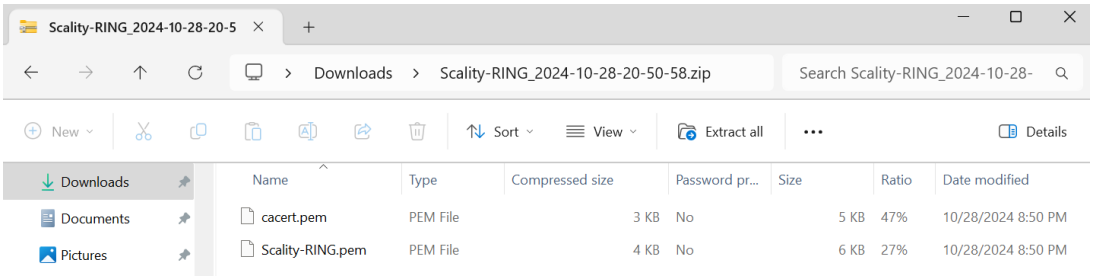
1. Sign in to the KMIP vault URL created in Create a KMIP Vault in the KeyControl.
2. Select the **Client Certificates** tab.
3. Select the **+** icon to create a client certificate. Enter your information. Then select **Create**.



4. Notice the new client certificate.



5. Select the certificate. Then select **Download** and save it for later use.



For more information, see Managing KMIP Client Certificates.

4.2. Configure the Scality RING for KMIP

1. In the supervisor VM, login (SSH) as root. The password is listed in the document titled "Scality OVA 9.3.0.2 Deployment guide - 3-server RING - 4TB.pdf".
2. Go to the federation folder.

```
cd /srv/scality/s3/s3-offline/federation
```

3. Set the `ENV_DIR` environment variable pointing to the folder containing the cluster's configuration, a directory named `env/`.

```
ENV_DIR=s3config
```

4. Create the `kmip/` configuration directory.

```
mkdir env/$\{ENV_DIR}/kmip
```

5. Upload the certificates and key files created in Create the client certificate in KeyControl to `env/${ENV_DIR}/kmip`. We used the WinSCP app to upload the two files.

```
[root@scality-supervisor-01 federation]# ls -al /srv/scality/s3/s3-offline/federation/env/s3config/kmip
total 20
drwxr-xr-x  2 root root   48 Oct 29 08:36 .
drwxr-xr-x 10 root root 4096 Oct 29 08:27 ..
-rw-r--r--  1 root root 4710 Oct 29 08:29 cacert.pem
-rw-r--r--  1 root root 5195 Oct 29 08:29 Scality-RING.pem
```

6. Cat file `/srv/scality/s3/s3-offline/federation/env/s3config/kmip/Scality-RING.pem`. Copy the private key section to the clipboard:

```
-----BEGIN PRIVATE KEY-----
MIIJQgIBAD...
```

```
...
-----END PRIVATE KEY-----
```

7. Create file `/srv/scality/s3/s3-offline/federation/env/s3config/kmip/client.key`. Edit this new file and paste the content of the clipboard.
8. Cat file `/srv/scality/s3/s3-offline/federation/env/s3config/kmip/Scality-RING.pem`. Copy the certificate section to the clipboard:

```
-----BEGIN CERTIFICATE-----
MIIEaTCCA1...
...
-----END CERTIFICATE-----
```

9. Create file `/srv/scality/s3/s3-offline/federation/env/s3config/kmip/client.cert`. Edit this new file and paste the content of the clipboard. Notice the files created.

```
[root@scality-supervisor-01 kmip]# ls -al /srv/scality/s3/s3-offline/federation/env/s3config/kmip
total 28
drwxr-xr-x  2 root root   85 Oct 29 11:39 .
drwxr-xr-x 10 root root 4096 Oct 29 08:27 ..
-rw-r--r--  1 root root 4710 Oct 29 08:29 cacert.pem
-rw-r--r--  1 root root 1590 Oct 29 11:39 client.cert
-rw-r--r--  1 root root 3272 Oct 29 11:39 client.key
-rw-r--r--  1 root root 5195 Oct 29 08:29 Scality-RING.pem
```

10. Edit the `env/${ENV_DIR}/group_vars/all` file. Uncomment the `env_s3.kmip` part of the configuration. Replace the various parameters with yours as described in document titled "Setting Up Server-Side Bucket Encryption — S3 Connector 9.3.0 documentation.pdf".

```
cat /srv/scality/s3/s3-offline/federation/env/s3config/group_vars/all
...
kmip:
  port: 5696
  host: 10.194.148.206
  compoundCreate: true
  bucketAttributeName: x-entrust-kmip-bucket
  pipelineDepth: 8
  key: client.key
  cert: client.cert
  ca:
    - cacert.pem
...
```

Notice the **x-** inserted in front of the bucket name defined in section Deploy the Scality RING.

For further information, refer to document titled "Setting Up Server-Side Bucket Encryption — S3 Connector 9.3.0 documentation.pdf".

4.3. Post installation re-configure procedure

1. In the supervisor VM, login (SSH) as root. The password is listed in the document titled "Scality OVA 9.3.0.2 Deployment guide - 3-server RING - 4TB.pdf".
2. Go to the federation folder.

```
cd /srv/scality/s3/s3-offline/federation
```

3. Set the `ENV_DIR` environment variable pointing to the folder containing the cluster's configuration, a directory named `env/`.

```
ENV_DIR=s3config
```

4. List the stateless hosts.

```
[root@scality-supervisor-01 federation]# ansible -i env/$\{ENV_DIR}/inventory --list-hosts runners_s3
[DEPRECATION WARNING]: DEFAULT_GATHER_SUBSET option, the module_defaults keyword is a more generic version
and can apply to all calls to the
M(ansible.builtin.gather_facts) or M(ansible.builtin.setup) actions, use module_defaults instead. This
feature will be removed from ansible-core in
version 2.18. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
  hosts (3):
    md1-cluster1
    md2-cluster1
    md3-cluster1
```

5. For each listed host, reconfigure s3. If the host is behind a load balancer, deactivate the server before re-configuring it.

```
cd /srv/scality/s3/s3-offline/federation

[root@scality-supervisor-01 federation]# ansible-playbook -i env/$\{ENV_DIR}/inventory run.yml --skip-tags
requirements -t s3 > logfile

cat /srv/scality/s3/s3-offline/federation/logfile

SCALITY S3 CONNECTOR INSTALLER ***********************************************

PLAY [all] *******************************************************************

TASK [Gathering Facts ] ******************************************************
ok: [10.15.20.102(md5-cluster1)]
ok: [10.15.20.101(md1-cluster1)]
ok: [10.15.20.101(md4-cluster1)]
ok: [10.15.20.102(md2-cluster1)]
ok: [10.15.20.103(md3-cluster1)]

...

PLAY RECAP *******************************************************************
md1-cluster1               : ok=172  changed=10   unreachable=0    failed=0
md2-cluster1               : ok=124  changed=8    unreachable=0    failed=0
```

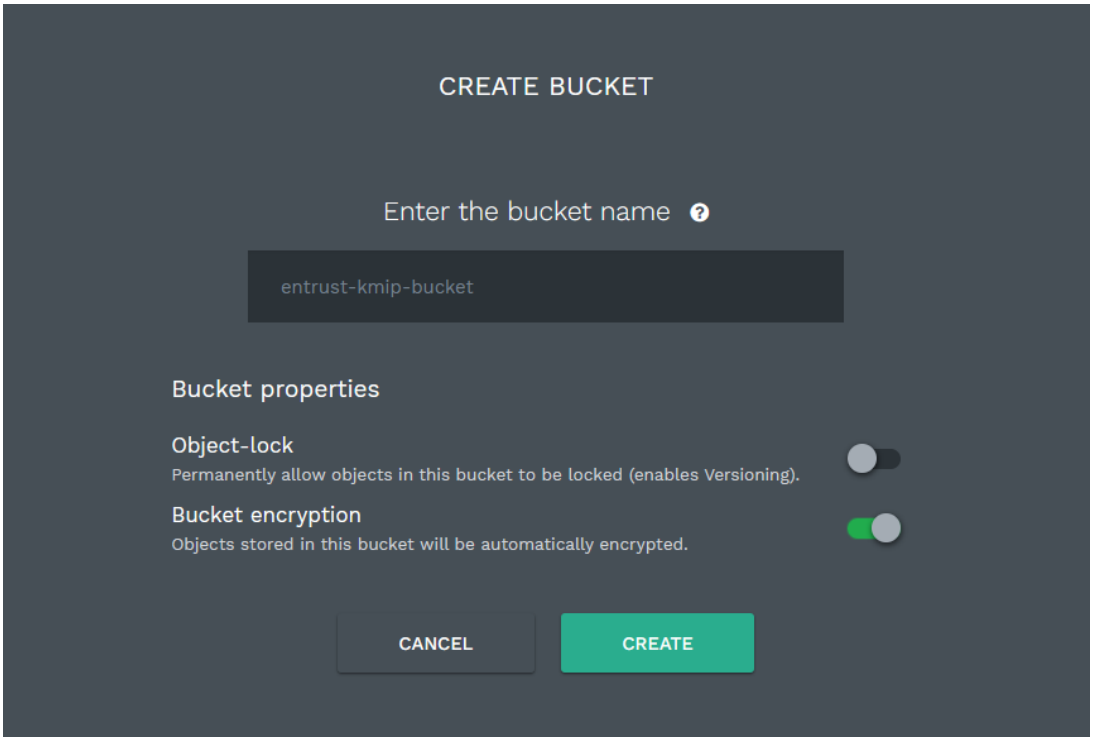
```
md3-cluster1               : ok=124  changed=8    unreachable=0    failed=0
md4-cluster1               : ok=69   changed=0    unreachable=0    failed=0
md5-cluster1               : ok=69   changed=0    unreachable=0    failed=0

If something went wrong (failed tasks), please contact Scality support and attach ./ansible.log
```

Chapter 5. Test the integration

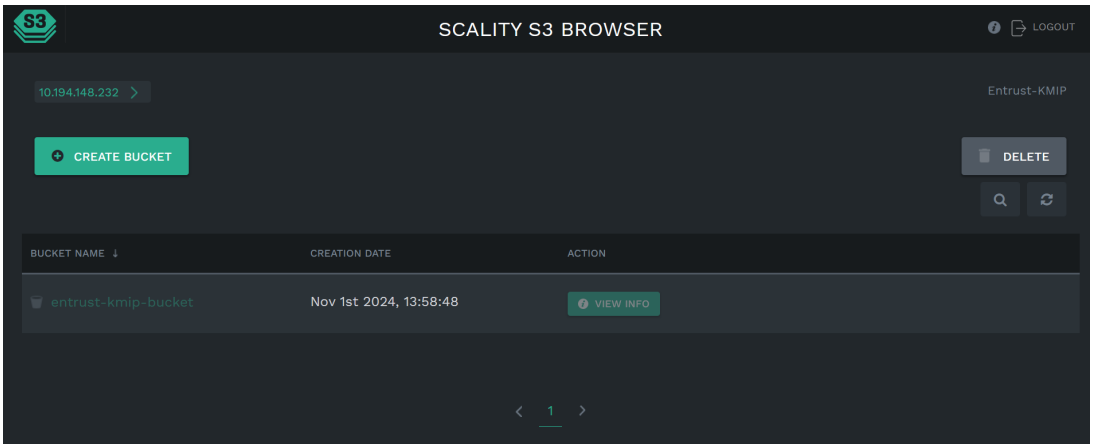
5.1. Create a bucket with encryption enabled

1. On a browser, go to the Scality supervisor URL and login.
2. On the toolbar, select **S3 SERVICE** > **S3 BROWSER**. A new tab should open. Login with the user's **AccessKey** and **SecretAccessKey**.
3. Select **+CREATE BUCKET**. Enter the information as shown. Notice **Bucket encryption** is enabled. Then select **Create**.

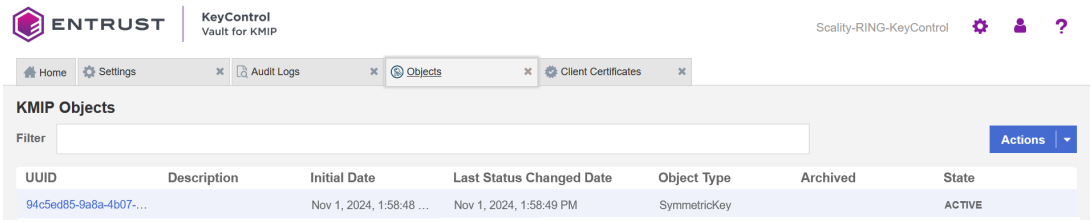
   You cannot use capital letters or spaces in the name.



4. Notice the newly created bucket.



5. Sign in to the vault URL that you created in Create a KMIP Vault in the KeyControl.
6. Select the **Objects** tab. Notice the symmetric key created to protect the bucket created above (creation times match).



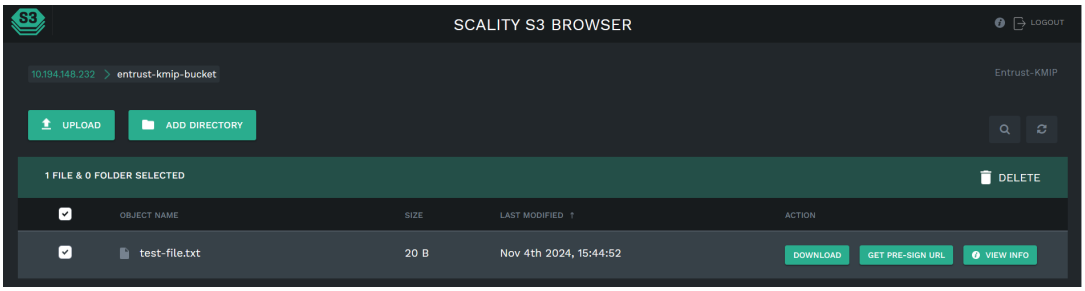
5.2. Upload test object to bucket and verify encryption

An test object named `test-file.txt` will be uploaded to the bucket named **entrust-kmip-bucket** created in Create a bucket with encryption enabled. Once inside the bucket, the object encryption will be verified.

1. Create test object named `test-file.txt` on your PC.

```
C:\Users\xxxxxxxx\Documents>type test-file.txt
This is a test file.
```

2. On a browser, go to the Scality supervisor URL and login.
3. On the toolbar, select **S3 SERVICE** > **S3 BROWSER**. A new tab should open. Login with the user's **AccessKey** and **SecretAccessKey**.
4. Select the bucket named **entrust-kmip-bucket**.
5. Select the **UPLOAD** icon. Then select **UPLOAD OBJECTS** and select the `test-file.txt` file created above.



6. In the supervisor VM, login (SSH) as root. The password is listed in the document titled "Scality OVA 9.3.0.2 Deployment guide - 3-server RING - 4TB.pdf".

7. Go to the federation folder.

```
cd /srv/scality/s3/s3-offline/venv/bin
```

8. Run the following command:

```
[root@scality-supervisor-01 bin]# ./aws --endpoint-url http://10.15.20.120 s3api head-object --bucket
entrust-kmip-bucket --key test-file.txt
{
    "AcceptRanges": "bytes",
    "LastModified": "Mon, 04 Nov 2024 20:44:52 GMT",
    "ContentLength": 20,
    "ETag": "\"3de8f8b0dc94b8c2230fab9ec0ba0506\"",
    "VersionId": "null",
    "ContentType": "text/plain",
    "ServerSideEncryption": "AES256",
    "Metadata": {}
}
```

Notice the **ServerSideEncryption** as **AES256**.

Chapter 6. Integrating with an HSM

For guidance on integrating the Entrust KeyControl with a Hardware Security Module (HSM), consult with your HSM vendor. If you are using an Entrust nShield HSM, refer to the Entrust KeyControl nShield HSM Integration Guide available at Entrust documentation library.

Chapter 7. Additional resources and related products

7.1. nShield Connect

7.2. nShield as a Service

7.3. KeyControl

7.4. KeyControl as a Service

7.5. Entrust products

7.6. nShield product documentation