



Amazon Web Services KMS External Key Store (XKS)

KeyControl Cloud Key Management Vault (HYOK)
Integration Guide

2024-02-26

Table of Contents

1. Introduction	1
1.1. Product configuration	1
1.2. Requirements	1
1.3. Overview	2
2. Procedures	3
2.1. Prerequisites	3
2.2. Adding an Elastic Load Balancer	3
2.3. Configure certificates and DNS	10
2.4. Key Administrators - AWS IAM user	12
2.5. Create a Cloud Key Management Vault	13
2.6. Create a CSP Account in the Cloud Key Management Vault	17
2.7. Create the Key Set	18
2.8. Create an External Key Store in AWS	21
2.9. Test the integration	25
3. Additional resources and related products	31
3.1. nShield Connect	31
3.2. nShield as a Service	31
3.3. Entrust digital security solutions	31
3.4. nShield product documentation	31

Chapter 1. Introduction

This guide describes the integration of the Entrust KeyControl Cloud Key Management Vault with Amazon Web Services KMS External Key Store (XKS).

Entrust KeyControl Cloud Key Management Vault provides an External Key Store Proxy inside KeyControl Vault. This feature allows the KeyControl Vault administrator to protect their data within Amazon Web Services (AWS) with 256-bit AES keys residing in KeyControl Vault. KeyControl Vault generates the keys and the keys are stored in KeyControl Vault only.

1.1. Product configuration

Entrust has successfully tested the following software version:

Product	Version	Certification
KeyControl Vault	10.1.1	FIPS 140-2 Level 1
	10.2	

1.2. Requirements

To integrate Entrust KeyControl Cloud Key Management Vault and Amazon Web Services KMS External Key Store (XKS), the server must be set up as follows.

- You must have an AWS account with KMS access allowed.
- There is a minimum of 2 KeyControl instances within a cluster. These instances must be accessible through a load balancer, typically realized using Elastic Load Balancer in the AWS environment.

This integration uses a public endpoint connectivity for AWS XKS. The following are required:

- Your external key store proxy must be reachable at a publicly routable endpoint.
- You must obtain a TLS certificate issued by a public certificate authority supported for external key stores. For a list, see <https://github.com/aws/aws-kms-xksproxy-api-spec/blob/main/TrustedCertificateAuthorities>.
- The subject common name (CN) on the TLS certificate must match the

domain name in the proxy URI endpoint for the external key store proxy. For example, if the public endpoint is <https://myproxy.xks.example.com>, the TLS, the CN on the TLS certificate must be `myproxy.xks.example.com` or `*.xks.example.com`.

- Ensure that any firewalls between AWS KMS and the external key store proxy allow traffic to and from port 443 on the proxy. AWS KMS communicates on port 443 and this value is not configurable.

Familiarize yourself with:

- [Entrust KeyControl and AWS External Key Store \(XKS\) Overview](#)
- [The Amazon Web Services KMS External Key Store \(XKS\) Documentation](#)

1.3. Overview

Entrust KeyControl Cloud Key Management Vault provides an External Key Store Proxy within KeyControl. This feature allows KeyControl administrators to safeguard their data within Amazon Web Services (AWS) using 256-bit AES keys housed in the KeyControl Vault. KeyControl generates the keys, which are exclusively stored in KeyControl.

In this guide:

- **BYOK (Bring Your Own Key):** This approach involves generating and managing encryption keys within an external key management system, such as the Amazon Web Services Key Management Service (AWS KMS). BYOK allows you to maintain control over your encryption keys while utilizing the services provided by AWS.
- **HYOK (Hold Your Own Key):** This method takes data security a step further by enabling you to retain absolute control over encryption keys, even when data is processed in cloud environments. With HYOK, the encryption keys are stored outside the cloud provider's infrastructure.

Entrust supports both BYOK and HYOK approaches to data security. This integration is HYOK implementation, ensuring that encryption keys are held within our control while still harnessing the benefits of cloud services.

For more information about the BYOK approach, refer to *Bring Your Own Key for AWS Key Management Service and Entrust KeyControl Integration Guide*.

Chapter 2. Procedures

Integration steps:

1. [Prerequisites](#)
2. [Adding an Elastic Load Balancer](#)
3. [Configure certificates and DNS](#)
4. [Key Administrators - AWS IAM user](#)
5. [Create a Cloud Key Management Vault](#)
6. [Create a CSP Account in the Cloud Key Management Vault](#)
7. [Create the Key Set](#)
8. [Create an External Key Store in AWS](#)
9. [Test the integration](#)

2.1. Prerequisites

Before integrating Entrust KeyControl Vault server and AWS External Key Store (XKS), ensure the following:

- Entrust KeyControl Vault server is deployed and configured. For details, see [KeyControl Installation](#).
- Entrust KeyControl Compliance Manager is deployed and configured.

For this integration, the KeyControl Vault servers were deployed using AWS EC2 instances. To learn more about deploying KeyControl Vault in Amazon Web Services, refer to [Creating KC Cluster AWS](#).

However, KeyControl Vault servers can also be deployed outside of AWS EC2, provided they fulfill the requirements outlined in [\[procedures:::requirements\]](#).

2.2. Adding an Elastic Load Balancer

After cluster set-up is complete, you must use AWS elastic load balancing for the KeyControl load balancing.

For more information on AWS ELB, refer to [AWS ELB Documentation](#).

2.2.1. Configure target group

To configure the target group:

1. Sign in the Amazon EC2 console.
2. In the navigation pane, under Load Balancing, select **Target Groups**.
3. Select **Create target group**.
4. Under **Basic configuration**:
 - a. Select **Instances** as target type.
 - b. For **Target group name**, enter a name for the new target group.
 - c. For **Protocol**, select **HTTPS**.
 - d. For **Port**, select **443**.
 - e. Select the VPC containing your instances.
 - f. For **Protocol version**, retain the default.

Basic configuration

Settings in this section can't be changed after the target group is created.

Choose a target type

Instances

- Supports load balancing to instances within a specific VPC.
- Facilitates the use of [Amazon EC2 Auto Scaling](#) to manage and scale your EC2 capacity.

IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol: Port:

1-65535

VPC

Select the VPC with the instances that you want to include in the target group.

Protocol version

HTTP1

Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

HTTP2

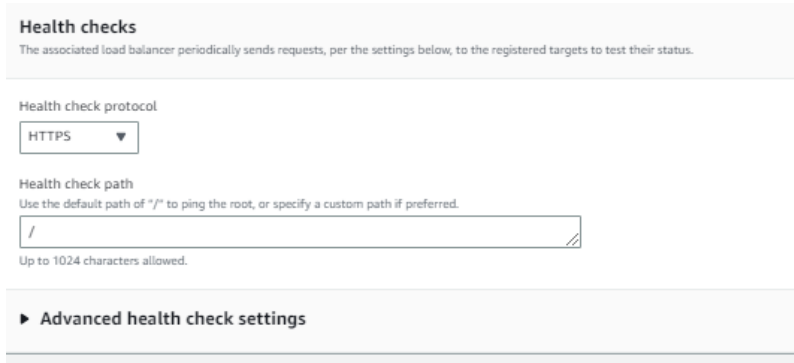
Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

gRPC

Send requests to targets using gRPC. Supported when the request protocol is gRPC.

5. Under **Health checks**:

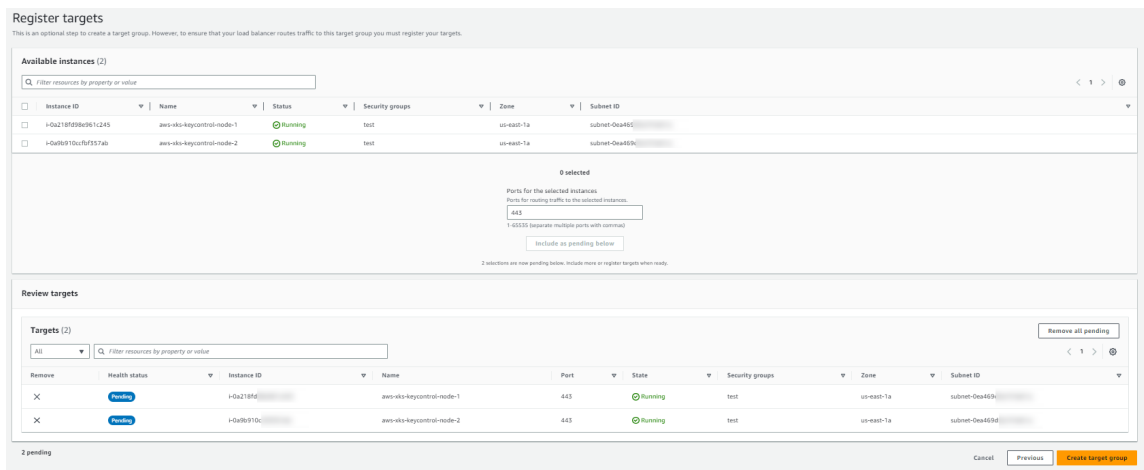
- a. For **Health check protocol**, select **HTTPS**.
- b. Retain the default settings for other properties.



6. Select **Next**.

7. On the **Register Targets** page, complete the following steps. This is an optional step for creating the load balancer. However, you must register this target if you want to test your load balancer and ensure that it is routing traffic to this target.

- a. For **Available instances**, select the two KeyControl instances.
- b. For **Port for the selected instances**, enter **443**, and select **Include as pending** below.
- c. Select **Create target group**.

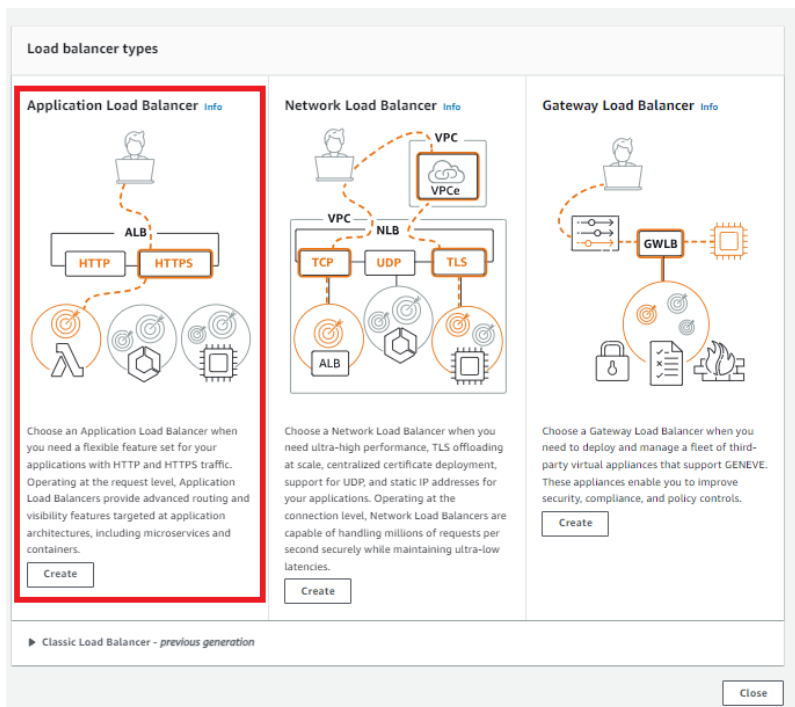


2.2.2. Create an Elastic Load Balancer

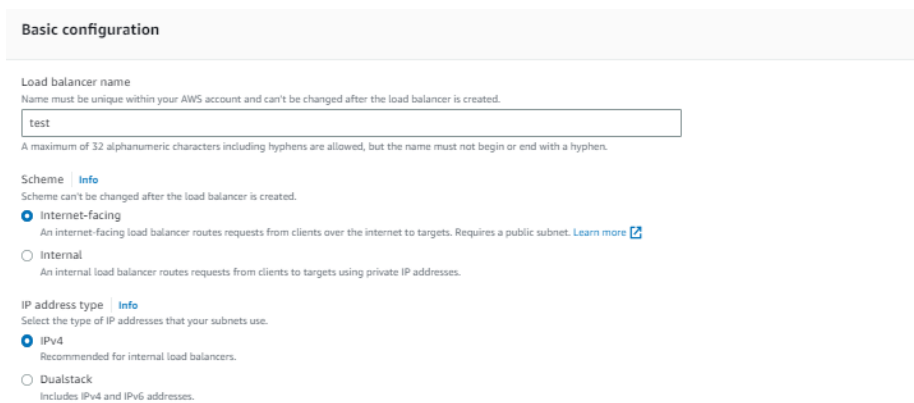
To create an Elastic Load Balancer:

- 1. Sign in to the Amazon EC2 console.

2. On the navigation bar, select a region for your load balancer. You must select the same region that you used for your EC2 instances.
3. In the navigation pane, under **Load Balancing**, select **Load Balancers**.
4. Select **Create Load Balancer**.
5. Select **Application Load Balancer**, select **Create**.



6. Under **Basic configuration**:
 - a. For **Load balancer name**, enter a name for your load balancer.
 - b. For **Scheme**, select **Internet-facing**.
 - c. Retain the **IP address** type default.



7. Under **Network mapping**:
 - a. For **VPC**, select the VPC that you used for your EC2 instances.

- b. For **Mappings**, select at least two Availability Zones and one subnet per zone.
- c. For each Availability Zone that you used to launch your EC2 instances, select the Availability Zone and then select one public subnet for that Availability Zone.
- d. You must select at least one Availability Zone that was used when launching your instances.

Network mapping [Info](#)
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC | [Info](#)
Select the virtual private cloud (VPC) for your targets or you can [create a new VPC](#). Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).

keycontroltest-vpc vpc-036ec5b2 IPv4: 10.0.0.0/16 ↕

Mappings | [Info](#)
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

us-east-1a (use1-az2)

Subnet
subnet-0ea469d keycontroltest-subnet-public1-us-east-1a

IPv4 address
Assigned by AWS

us-east-1b (use1-az4)

Subnet
subnet-0358a1d1 keycontroltest-subnet-public2-us-east-1b

IPv4 address
Assigned by AWS

8. Under **Security groups**:

- a. For **Security group**, select the default security group for the VPC that you selected in the previous step. Alternatively, you can select a different security group.
- b. Ensure that the security group includes rules that allow the load balancer to communicate with registered targets on both the listener port and the health check port.
- c. You must include the VPC source in the inbound rule to allow access to all ports or the port you are using as a listener.

Security groups [Info](#)
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

Security groups
Select up to 5 security groups ↕

test sg-0346003 VPC: vpc-036ec5b24 ✕

9. Under **Listeners and routing**:

- a. For **Protocol**, retain the default setting.
- b. For **Port**, retain the default setting.
- c. For **Default action**, select the **Forward to** action and select the target group that you created and registered.
- d. Keep the **Add-on services** and **Load balancer** tags unchecked and left as default.

This configures a listener that accepts HTTP traffic on port 80 and forwards traffic to the selected target group by default.

The screenshot shows the 'Listeners and routing' configuration page in the AWS Management Console. The page title is 'Listeners and routing' with an 'Info' link. Below the title is a descriptive paragraph: 'A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.'

The main configuration area is titled 'Listener HTTP:80' and includes a 'Remove' button. The configuration fields are:

- Protocol:** HTTP (dropdown menu)
- Port:** 80 (input field, with '1-65535' below it)
- Default action:** Forward to (dropdown menu, currently showing 'Select a target group') with an 'Info' link and a refresh icon.

Below the 'Default action' dropdown is a search bar with a magnifying glass icon and a 'Create target' link. A dropdown menu is open, showing a search bar with 'Q |' and a table of target groups:

In use	
test	HTTPS
Target type: Instance, IPv4	

Below the configuration fields is a section for 'Listener tags - optional' with a note: 'Consider adding tags to your listener. Tags enable you to categorize your AWS resources. You can add up to 50 more tags.' There is an 'Add listener tag' button.

At the bottom of the configuration area is an 'Add listener' button.

10. Review your configuration and select **Create load balancer**. A few default attributes are applied to your load balancer during creation. You can view and edit them after creating the load balancer.
11. Select **Create load balancer**.

Summary

Review and confirm your configurations. [Estimate cost](#)

Basic configuration Edit

test

- Internet-facing
- IPv4

Security groups Edit

- test
sg-0346003

Network mapping Edit

VPC vpc-036ec5b2
keycontroltest-vpc

- us-east-1a
subnet-0ea469d5
keycontroltest-subnet-public1-us-east-1a
- us-east-1b
subnet-0358a1d1
keycontroltest-subnet-public2-us-east-1b

Listeners and routing Edit

- HTTP:80 defaults to
Target group not defined

Add-on services Edit

None

Tags Edit

None

Attributes

ⓘ Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

Cancel Create load balancer

After you receive the notification confirming the successful creation of your load balancer, follow the steps below to verify the status of your instances and test the load balancer.

EC2 > Load balancers

Load balancers (1/1)

Basic: Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Find resources by attribute or tag

Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
test	test-13276	Active	vpc-036ec5b2	2 Availability Zones	application	June 13, 2023, 14:26 (UTC-04:00)

Load balancer: test

- Details
- Listeners
- Network mapping
- Security
- Monitoring
- Integrations
- Attributes
- Tags

Details

Load balancer type Application	Status Active	VPC vpc-036ec5b2	IP address type IPv4
Scheme Internet-facing	Hosted zone Z35SXDOO	Availability Zones subnet-0ea469d5 us-east-1a (use1-az2) subnet-0358a1c1 us-east-1b (use1-az4)	Date created June 13, 2023, 14:26 (UTC-04:00)
Load balancer ARN arn:aws:elasticloadbalancing:us-east-1:594691249913:loadbalancer/app/test/09aa3c2	DNS name test-13276-east-1.elb.amazonaws.com (A Record)		

1. After you are notified that your load balancer was created successfully, select **Close**.
2. In the navigation pane, under **Load Balancing**, select **Target Groups**.
3. Select the newly created target group.
4. Select **Targets** and verify that your instances are ready.

If the status of an instance is **Initial**, the instance is either in the process of being registered or has not passed the minimum number of health checks to be considered healthy. Wait until the status of at least one instance is **Healthy**. For example:

Instance ID	Name	Port	Zone	Health status	Health status details
i-0a218f6c	aws-xks-keycontrol-node-1	443	us-east-1a	healthy	
i-0a9b910c	aws-xks-keycontrol-node-2	443	us-east-1a	healthy	

2.3. Configure certificates and DNS

For the successful set-up of AWS External Key Store (XKS), note that the DNS record and TLS certificate relate to the Fully Qualified Domain Name (FQDN) of the load-balanced endpoint utilized for accessing the service. This is distinct from the KeyControl instances.

Ensure the KeyControl Vault server possesses a publicly accessible IP address and that a DNS record is in place for the designated common name within the public DNS server.



Amazon recommends a round-trip time latency of under 35 milliseconds between the AWS region and the KeyControl.

You must obtain a TLS certificate issued by a public certificate authority supported for external key stores. For a list, see <https://github.com/aws/aws-kms-xksproxy-api-spec/blob/main/TrustedCertificateAuthorities>

To ensure seamless access across the cluster nodes, you must install the TLS certificate on all nodes of the cluster, especially if users plan to access the cluster through other nodes. If an Elastic Load Balancer (ELB) is part of the set-up, the certificate handling process will differ. When using an ELB, the TLS certificate must be managed according to ELB requirements.

1. In the KeyControl Appliance Management:
 - a. Navigate to **Cluster > Servers**.
 - b. Select the server to install the certificate.
 - c. Select **Actions > Install Certificate**. The **Install Custom SSL Certificate** dialog appears.
 - d. Locate and select the **SSL Certificate** file.
 - e. Locate and select the **CA certificate** chain file.
 - f. Select **External** for the Web server.
 - g. Select **Install Certificate**.

Install Custom SSL Certificate

Certificate Private Key

SSL Certificate:
Certificate needs to be in base64 encoded pem format.

CA Certificate:
Certificate needs to be in base64 encoded pem format.

Web server External Internal
Choose which web server to install the custom certificate.

2. After installation, restart the Web service

The screenshot shows the ENTRUST KeyControl Appliance Management interface. At the top, there are navigation tabs for CLUSTER, AUDIT LOG, ALERTS, and SETTINGS. Below this is a table of appliances with columns for Actions, Status, Name, and IP Address. The table shows two appliances, both Online, with names like 'ip-10-0-...-internal' and IP addresses '10.0.9.88' and '10.0.2.14'. Below the table, a detailed view of an appliance is shown, including fields for Name, Status, Authenticated, Domain, IP Address, and Certificate. The Certificate field shows 'Internal Web server: Default' and 'External Web server: Default'. A red box highlights a 'Restart Web Service' button next to the External Web server field.

3. Confirm the installation. The External Web server will now show as **Custom** for the certificate.

This screenshot shows a detailed view of an appliance in the ENTRUST KeyControl Appliance Management interface. The fields shown are Name, Status, Authenticated, Domain, IP Address, Certificate, and Encryption Status. The Certificate field is expanded to show 'Internal Web server: Default' and 'External Web server: Custom', with the 'Custom' text highlighted by a red box. The Encryption Status field shows 'No rekey is in progress'.

4. You can validate the certificate using <https://entrust.sslabs.com/> or a similar tool. For example:

The screenshot shows an SSL Report from Entrust SSL Labs. The report is for the domain 'https://entrust.sslabs.com/'. It was assessed on Monday, August 28, 2023, at 16:58:41 UTC. The overall rating is A+. The report includes a bar chart showing the following scores: Certificate (100), Protocol Support (100), Key Exchange (100), and Cipher Strength (100). Below the chart, there are three green bars indicating security features: 'HTTP Strict Transport Security (HSTS) with long duration deployed on this server.', 'DNS Certification Authority Authorization (CAA) Policy found for this domain.', and a link to 'Visit our documentation page for more information, configuration guides, and books. Known issues are documented here.'



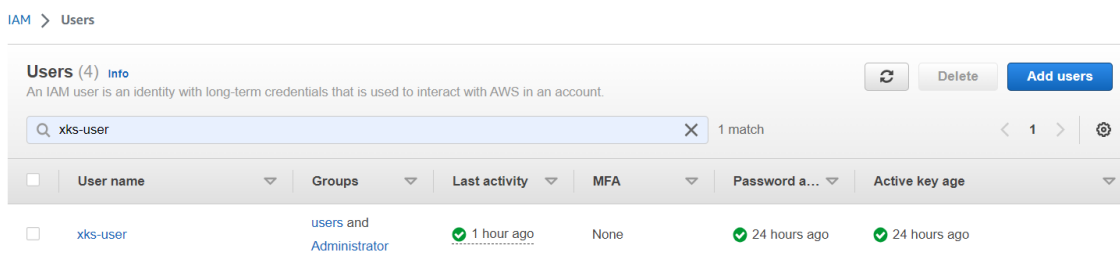
If you are not able to verify the server hostname, ensure that any firewalls between AWS KMS and the external key store proxy allow traffic to and from port 443 on the proxy.

2.4. Key Administrators - AWS IAM user

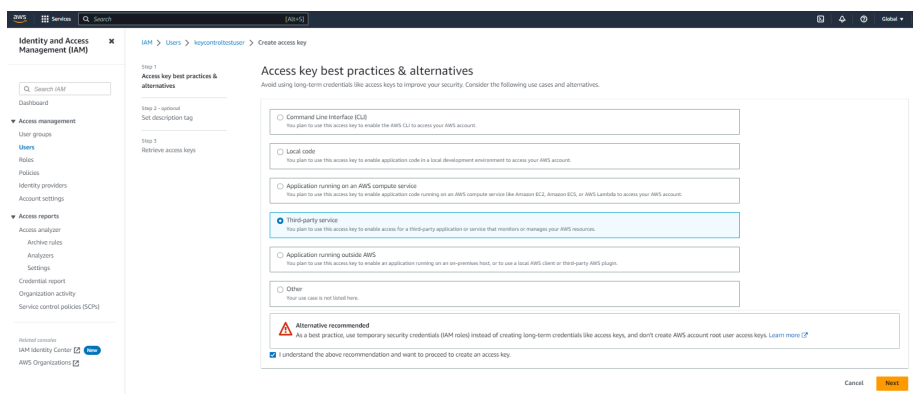
To enable the integration, you must designate an IAM user as a Key Administrator. This user is required to generate an access key that will be used in a later step.

This user must have permissions to manage and use the KMS key for cryptographic operations.

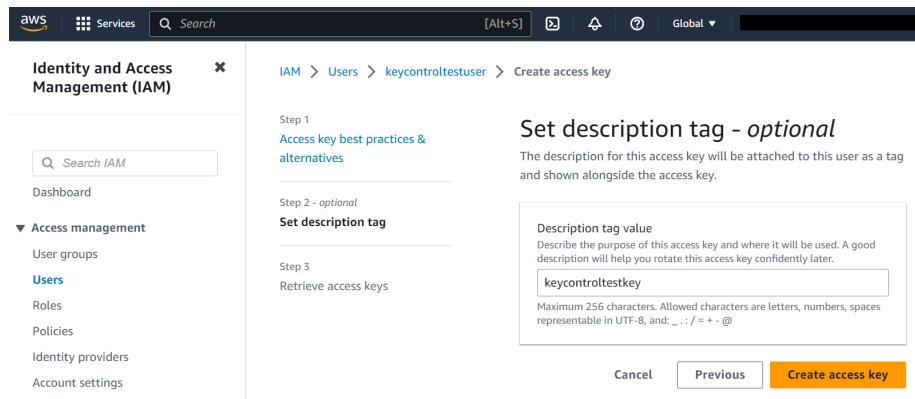
1. Sign in to the AWS Management Console.
2. Search for the **Identity and Access Management (IAM)** service and select it.
3. In the IAM console, select **Access Management** in the left tab and then select **Users**.
4. Create a new user or use an existing user to generate an access key. In this example integration, a new user named **xks-user** is created as the Key Administrator.



5. In the user settings, select **Create access key** and select **third-party service**.



6. Create the access key.



Ensure that you securely store the Access ID and Secret Access Key, as they are required for accessing and managing your AWS resources.

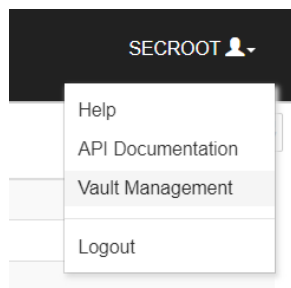
2.5. Create a Cloud Key Management Vault

The KeyControl Vault appliance supports the following types of vaults:

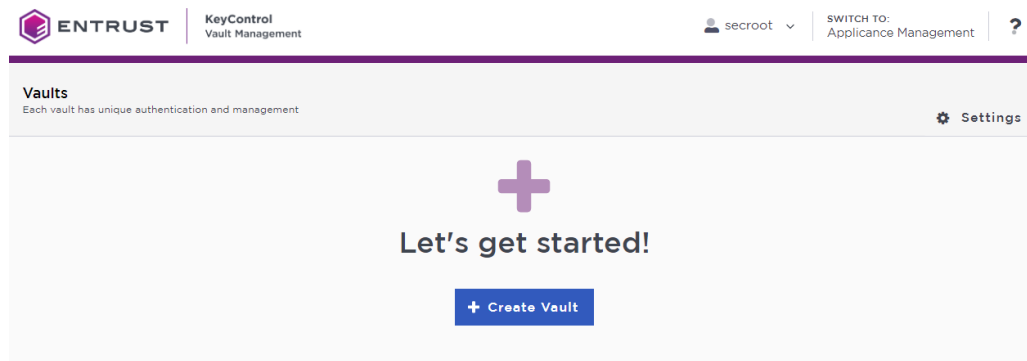
- **Cloud Key Management** - Vault for cloud keys such as BYOK and HYOK.
- **KMIP Vault** - Vault for KMIP Objects.
- **PASM** - Vault for objects such as passwords, files, SSH keys, and so on.
- **Database** - Vault for database keys.
- **Tokenization** - Vault for tokenization policies.
- **VM Encryption** - Vault for encrypting VMs.

To create a Cloud Key Management Vault:

1. Sign in to the KeyControl Vault Server Appliance Manager.
2. Open the drop-down menu and select **Vault Management**.



The KeyControl Vault Management interface appears.



3. Select **Create Vault**.

The **Create Vault** page appears.

4. On the **Create Vault** page:

- For **Type**, select **Cloud Key Management**.
- Enter a **Name** for the vault.
- Provide a **Description** for the vault.

5. Under **Administration**:

- Enter the **Admin Name** who will be responsible for the vault.
- Enter a valid **Admin Email** address.

The screenshot shows the 'Create Vault' form. At the top, it says 'Create Vault' and 'A vault will have unique authentication and management.' Below this, there is a 'Type' section with a dropdown menu set to 'Cloud Key Management'. The 'Name*' field contains 'AWS-XKS'. The 'Description' field contains 'AWS XKS' and has a 'Max. 300 characters' limit. The 'Administration' section includes a note: 'Invite an individual to have complete access and control over this vault. They will be responsible for inviting additional members.' Below this, there are two required fields: 'Admin Name*' and 'Admin Email*'. At the bottom, there are two buttons: 'Create Vault' (in blue) and 'Cancel'.

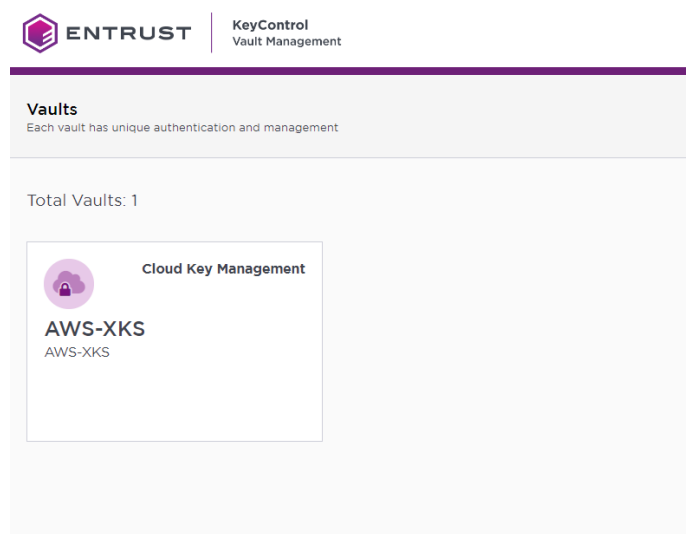
6. Select **Create Vault**.

If you set up an administrator email address when you logged in for the first time, a temporary password is mailed to that address. This is the password you must use when you sign in for the first time to Vaults space in KeyControl.

If you did not set up an email configuration when you logged in for the first time, a password is shown in the Vault Details when you create a Vault for the first time. You must make a note of the password at this time, as it will not be included in the Vault Details afterwards.

7. Select **Close**.

The newly created vault is displayed in the **Vaults** dashboard.



8. To view the details of a vault, hover over the vault and select **View Details**.


Vault Details ✕

AWS-XKS
AWS-XKS


Type
Cloud Key Management

Created
Jun 13, 2023 10:44:24 AM

Vault URL
[Redacted]
[Redacted]

 Copy

API URL
[Redacted]
[Redacted]

 Copy

Administrator
[Redacted]
[Redacted]

[Close](#)

To edit the details of a vault:

1. Hover over the vault and select **Edit**.
2. Make the required changes and select **Apply**.

Vaults

Each vault has unique authentication and management

Edit Vault

Type
Cloud Key Management

Name *

Description

Max. 300 characters

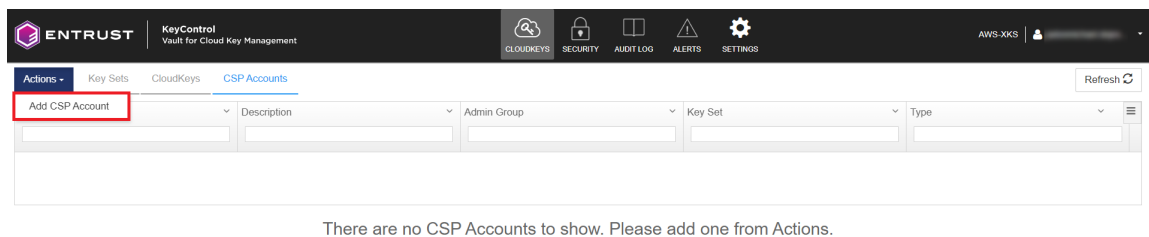
Administrator
[Redacted]
[Redacted]

[Apply](#) [Cancel](#) [Delete Vault](#)

2.6. Create a CSP Account in the Cloud Key Management Vault

To create a CSP Account in the Cloud Key Management Vault:

1. Sign into the newly created vault.
2. Select **Cloud Keys > CSP Accounts > Actions > Add CSP Account**.



The **Add CSP Account** dialog appears.

3. In the **Details** page:
 - a. For **Name**, enter a name for the CSP account.
 - b. Add a **Description**.
 - c. For **Admin Group** select **Cloud Admin Group**.
 - d. For **Type** select **AWS**.
 - e. Enter the **AWS Access Key ID** and **AWS Secret Access Key** from earlier.
 - f. Select the target region as the default region.
 - g. Select **Continue**.

Add CSP Account [Close]

Details | Schedule

Name *
aws_csp

Description
aws_csp

Admin Group *
Cloud Admin Group

Type *
AWS

AWS Access Key ID *
AKIAYU5S4K3

AWS Secret Access Key *
Wef8bWtYphaG2unb623HT

Default Region ⓘ
US East (N. Virginia) us-east-1

Cancel Continue

- 4. In the **Schedule** page:
 - a. Select the required **Rotation Schedule**.
 - b. Select **Apply**.

Add CSP Account [Close]

Details | Schedule

Define a schedule for which access keys are rotated.

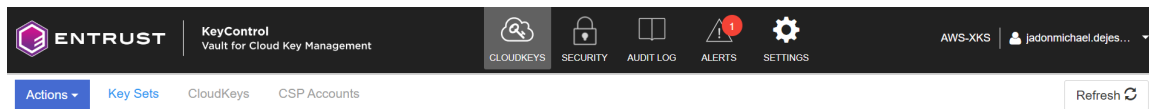
Rotation Schedule *
 Never Define Schedule

Cancel Apply

2.7. Create the Key Set

To create the Key Set:

- 1. Under **CloudKeys**, select **Key Sets > Create a Key Set Now**.

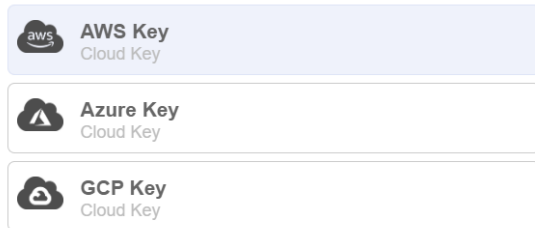


There are currently no Key Sets

[Create a Key Set Now](#)

2. Select **AWS Key** for the type of keys in key set.

Choose the type of keys in this key set: ✕



The **Create Key Set** dialog appears.

3. In the **Details** page:
 - a. Enter a **Name**.
 - b. Enter a **Description**.
 - c. For **Admin Group**, select **Cloud Admin Group**.
 - d. Select **Continue**.

4. In the **CSP Account** page:
 - a. For **CSP Account**, select the `aws_csp` account created earlier.
 - b. Select **Use as External Key Store**.
 - c. Make a note of the XKS credentials, as these are required later.
 - d. Select **Continue**.

Create Key Set ✕

Details CSP Account HSM Schedule

CSP Account *
Choose an existing CSP Account or add a new one to use with this Key Set.

aws_csp ▼

[+ Add CSP Account](#)

External Key Store

Enabling external key store allows KeyControl to encrypt and decrypt KMS keys.

Use as External Key Store

Access Key ID: X5Z6QQ6ZM

Secret Access Key: *****

Copy and add this Access Key ID and Secret Access Key to your AWS KMS Portal

Cancel Continue

- 5. In the **HSM** page:
 - a. Optionally select **Enable HSM**.
 - b. Select **Continue**.

Create Key Set ✕

Details CSP Account HSM Schedule

There is no HSM configured. HSM needs to be configured in Appliance Management, before it can be enabled in the Key Set.

Enable HSM

If checked, the HSM linked to KeyControl will be used for generating cryptographic material for Cloudkeys in this Key Set.

Cancel Verify HSM connection Continue



To set up an HSM linked to KeyControl, follow the installation and set-up instructions in the *Entrust KeyControl nShield HSM Integration Guide*.

- 6. In the **Schedule** page:
 - a. For **Rotation Schedule**, select your required CloudKey rotation.
 - b. Select **Apply**.

Create Key Set ✕

Details CSP Account HSM Schedule

Default CloudKey rotation schedule presented during CloudKey creation.

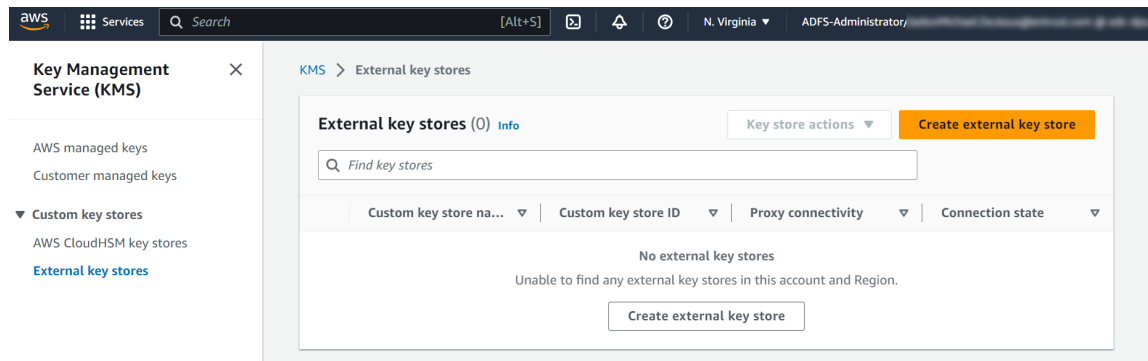
Rotation Schedule *
Never ▼

Cancel Apply

2.8. Create an External Key Store in AWS

To create an External Key Store in AWS:

1. Sign in to the AWS console and navigate to **Key Management Service (KMS)**.
2. In the left panel, select **Custom key stores > External key stores**.
3. For **Key store name**, enter the required name.
4. Select **Create external key store**.

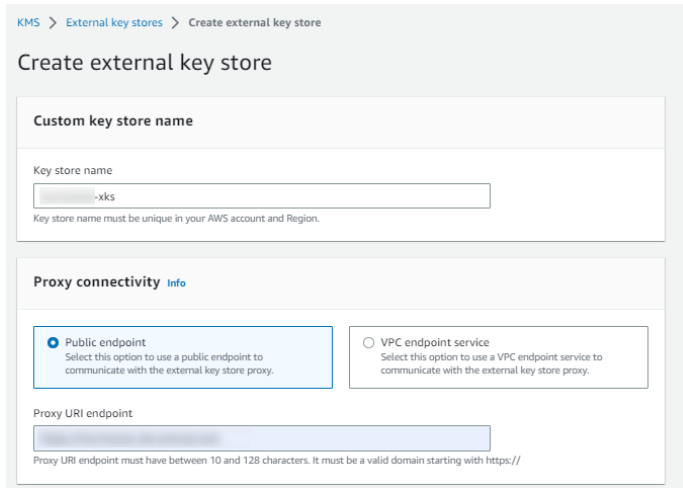


The **Create external key store** page appears.

5. Under **Custom key store name**, provide a descriptive name for the external key store.
6. Under **Proxy connectivity**:
 - a. Select **Public endpoint**.
 - b. For **Proxy URI endpoint**, enter the Proxy URI endpoint in the following format:

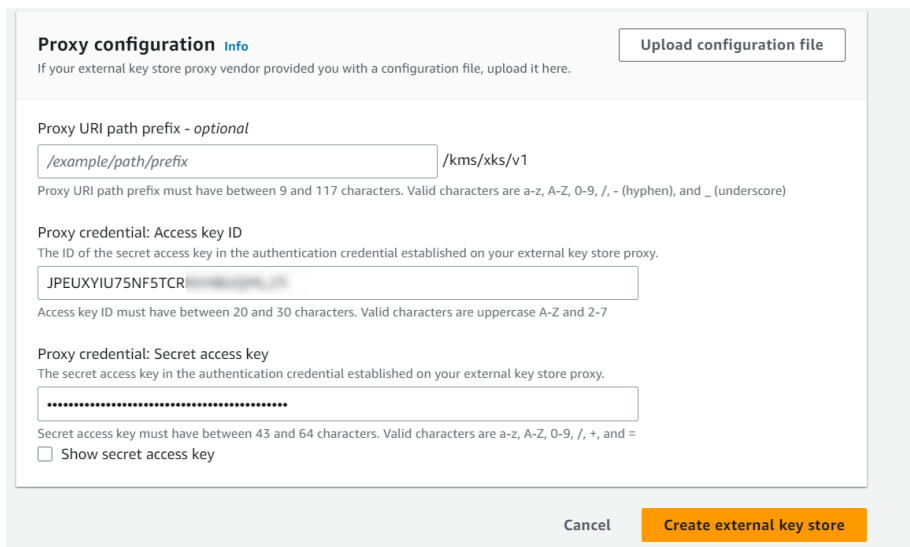
`https://<FQDN of Load Balanced Endpoint>`

Substitute **<FQDN of Load Balanced Endpoint>** with the fully qualified domain name of the load-balanced endpoint utilized for accessing the service, distinct from any of the KeyControl instances.

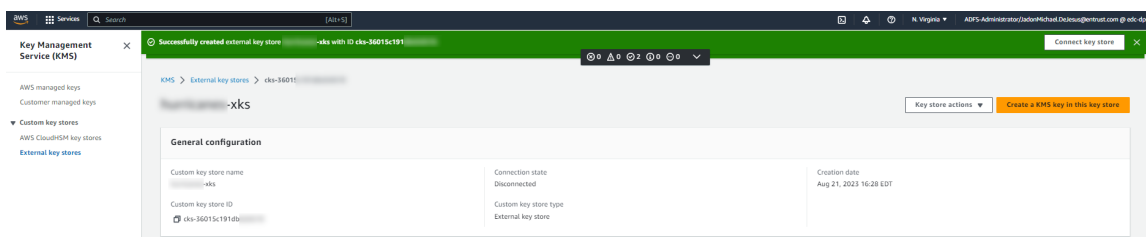


7. Under **Proxy configuration**:

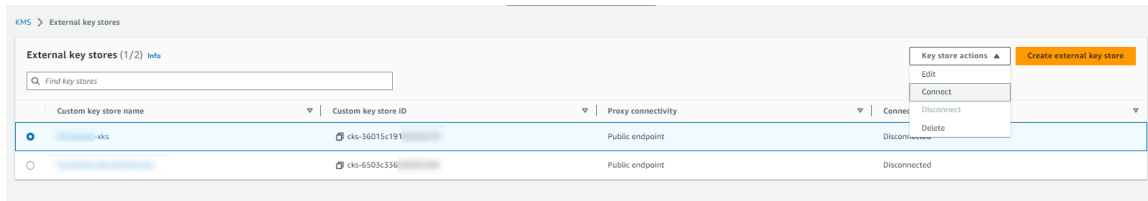
- a. Leave **Proxy URI path prefix** empty.
- b. For **Proxy credential: Access key ID**, enter the previously-saved proxy access key ID.
- c. For **Proxy credential: Secret access key**, enter the previously-saved proxy secret access key.
- d. Select **Create external key store**.



A details page for the new external key store appears.



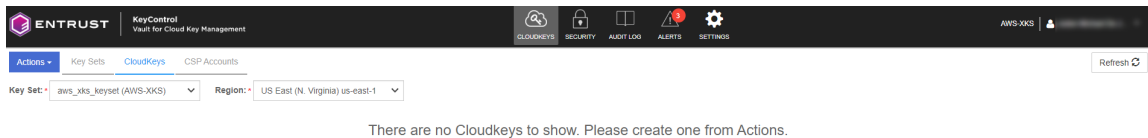
8. Select **External key stores** to view all external key stores.
9. Select **Key store actions** > **Connect** to connect to the external key store.



10. Wait for the Connection state to display as **Connected**.



11. Return to KeyControl Cloud Key Management Vault and select **CLLOUDKEYS** > **CloudKeys**.
12. Select the **Key Set** created earlier along with the **Region**.

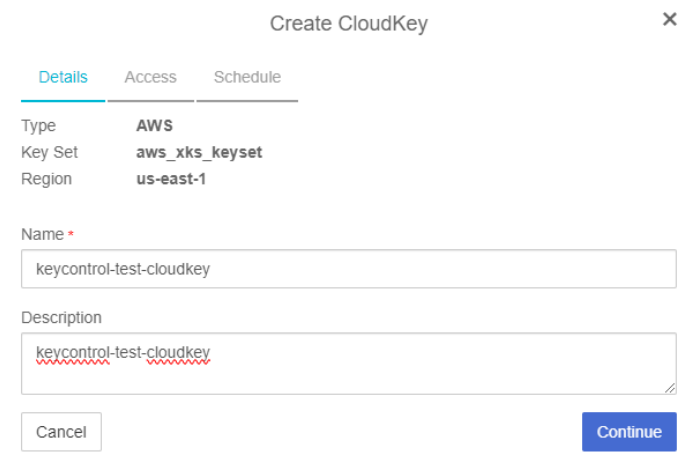


13. Select **Actions** > **Create CloudKey**.

The **Create CloudKey** dialog appears.

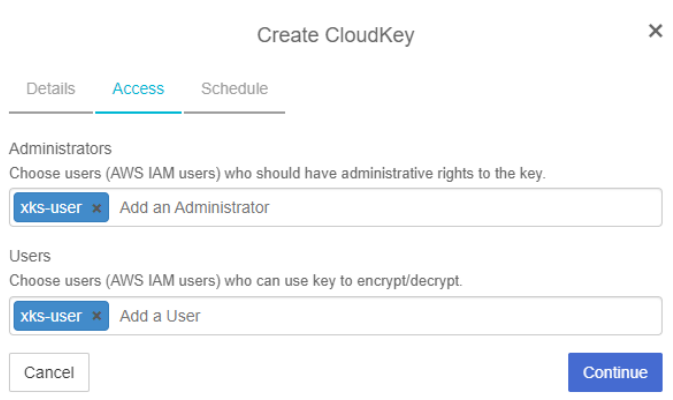
14. In the **Details** page:

- a. For **Name**, enter a name for the CloudKey.
- b. Enter a **Description**.
- c. Select **Continue**



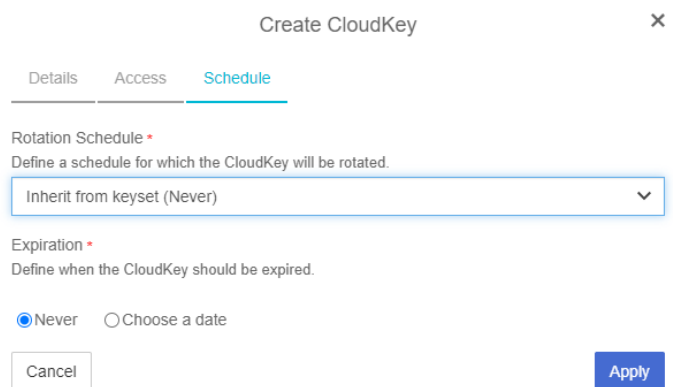
15. In the **Access** page:

- a. For **Administrators**, select AWS IAM users who will have administrative rights.
- b. For **Users**, select AWS IAM users who will be able to use the key to encrypt/decrypt.
- c. Select **Continue**.



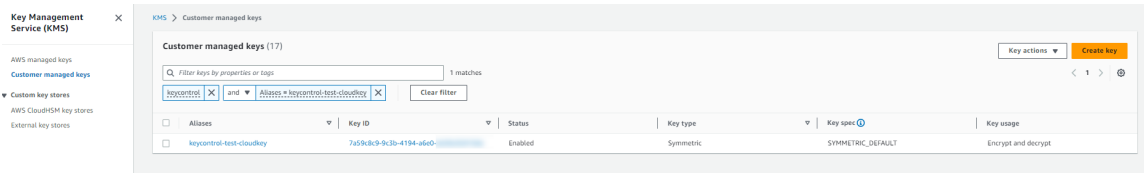
16. In the **Schedule** page:

- a. For **Rotation Schedule**, select a rotation schedule for the CloudKey.
- b. For **Expiration**, select the required condition.
- c. Select **Apply** to finish the process.



After the XKS CloudKey is created in KeyControl, a KMS key pointer is automatically created in AWS KMS with a key alias that matches the KeyControl CloudKey name. This KMS key pointer can be utilized by AWS services to encrypt or decrypt user objects.

17. Return to AWS KMS > **Customer managed keys** to find the created CloudKey.



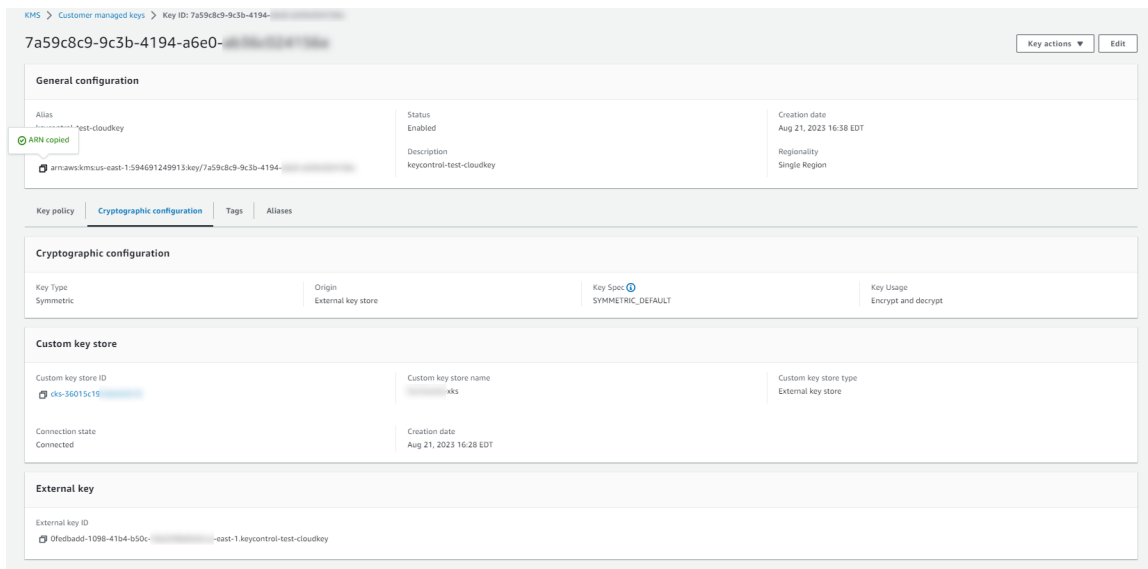
18. Select either the **Aliases** or **Key ID** hyperlink for the CloudKey.

A details page for the CloudKey appears.

19. Select **Cryptographic configuration**.

Note that under **Custom key store**, the **Custom key store name** appears and the **Custom key store type** as listed as **External**.

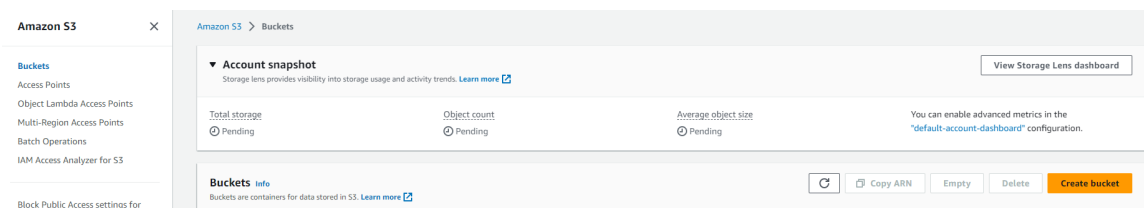
20. Under **General configuration**, copy the AWS KMS ARN for a later step.



2.9. Test the integration

To test the integration:

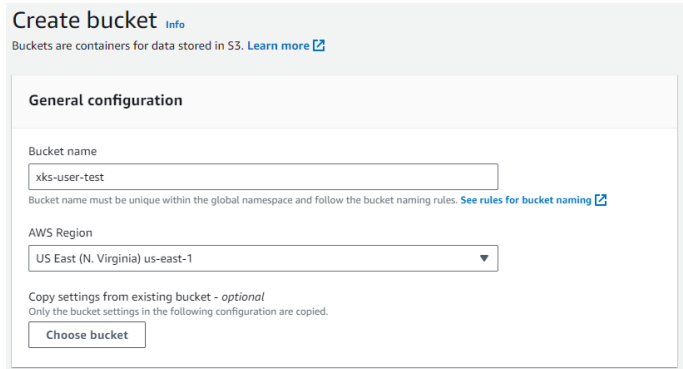
1. Sign in to the AWS Console and access **S3** services.
2. From the left panel, select **Buckets** and then select **Create bucket**.



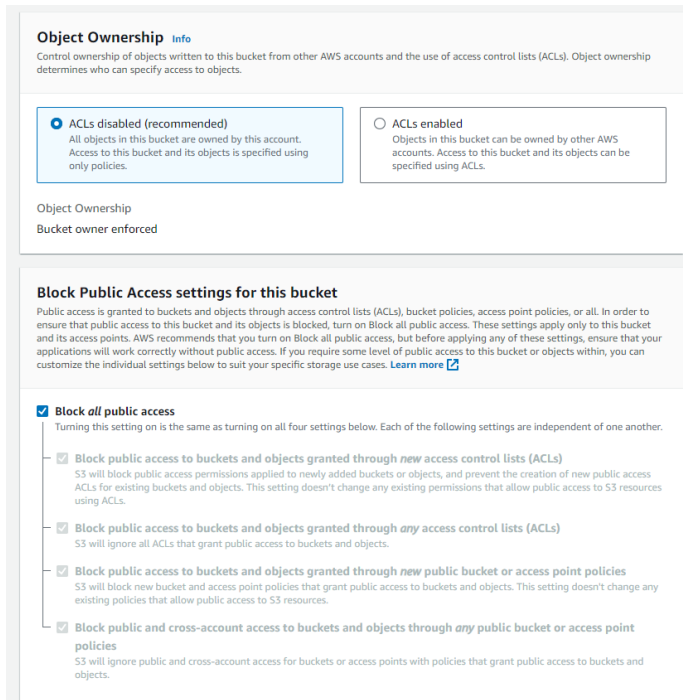
The **Create bucket** page appears.

3. Under **General configuration**:

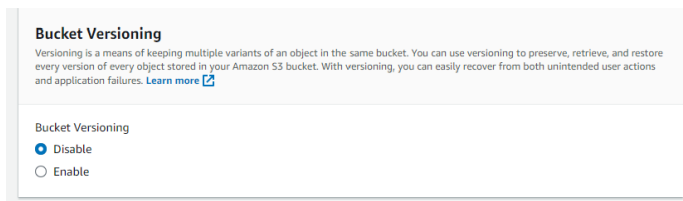
- a. For **Bucket name**, enter the required name for the bucket.
- b. Select an appropriate **AWS Region**.



4. Under **Object Ownership**, select **ACLs disabled**.



5. Under **Bucket Versioning**, set **Bucket Versioning** to **Disable**.



6. Under **Default encryption**:

- a. For **Encryption type**, select **Server-side encryption with AWS Key Management Service keys (SSE-KMS)**.
- b. For **AWS KMS key**:
 - i. Select **Enter AWS KMS key ARN**.
 - ii. Paste the AWS KMS ARN from the previously created CloudKey.
- c. For **Bucket Key**, select **Enable**.
- d. Select **Create bucket** to complete the process.

Default encryption [Info](#)
Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)
 Server-side encryption with AWS Key Management Service keys (SSE-KMS)
 Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing on the Storage tab of the Amazon S3 pricing page](#).

AWS KMS key [Info](#)

Choose from your AWS KMS keys
 Enter AWS KMS key ARN

AWS KMS key ARN
Your KMS key must be in the us-east-1 Region, where this bucket is being created.

Format (using key id): arn:aws:kms:<region>:<account-ID>:key/<key-id>
(using alias): arn:aws:kms:<region>:<account-ID>:alias/<alias-name>

Bucket Key
Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable
 Enable

► **Advanced settings**

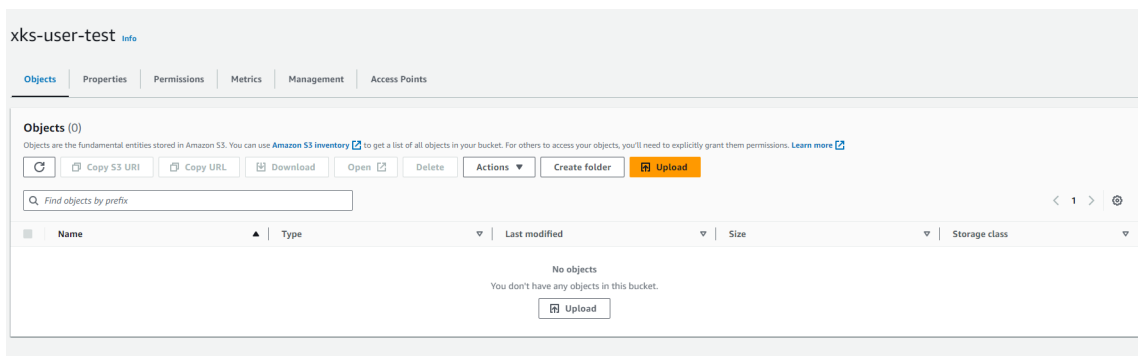
ⓘ After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

The bucket is created.

Buckets (1) [Info](#)
Buckets are containers for data stored in S3. [Learn more](#)

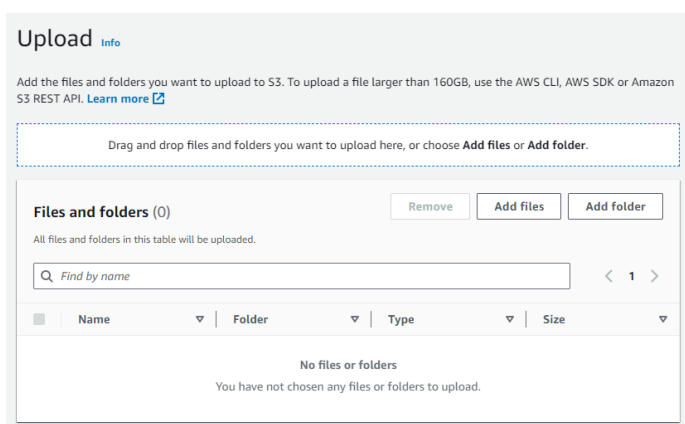
Name	AWS Region	Access	Creation date
<input type="radio"/> xks-user-test	US East (N. Virginia) us-east-1	Bucket and objects not public	August 21, 2023, 16:45:00 (UTC-04:00)

7. Select the hyperlink for the bucket.
- A details page for the bucket appears.
8. Select **Objects**.
9. To test the encryption, select **Upload**.



The **Upload** dialog appears.

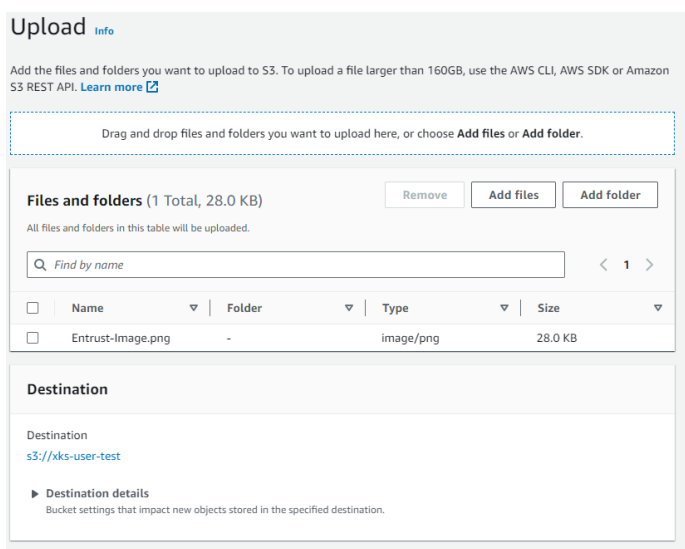
- 10. Select **Add files**.



- 11. Locate and select an image to upload.

The file is added to the list of available images.

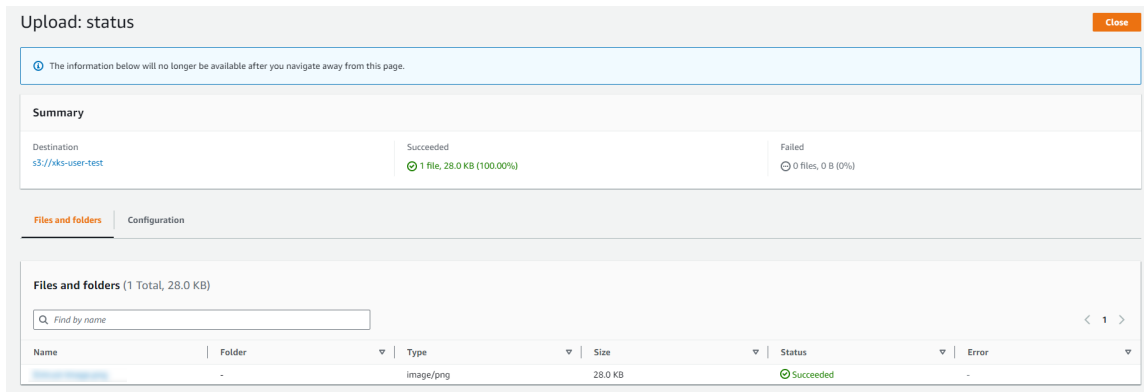
- 12. Select the check box for the image file and select **Upload**.



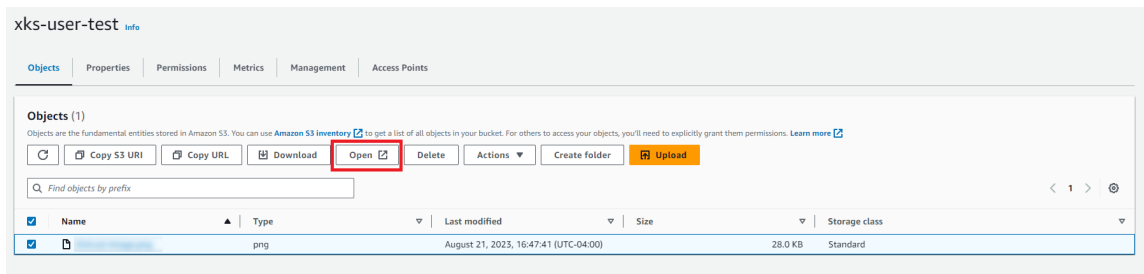
In this example, the **Entrust-Image.png** file was added and can be selected and

uploaded.

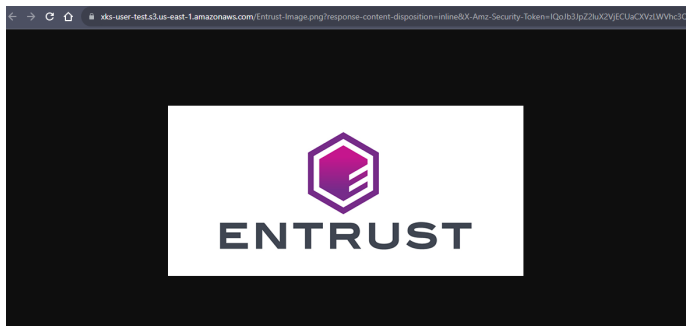
The newly uploaded image is listed within the bucket.



13. Select the new image and select **Open** to view it.



The image starts in a browser window.

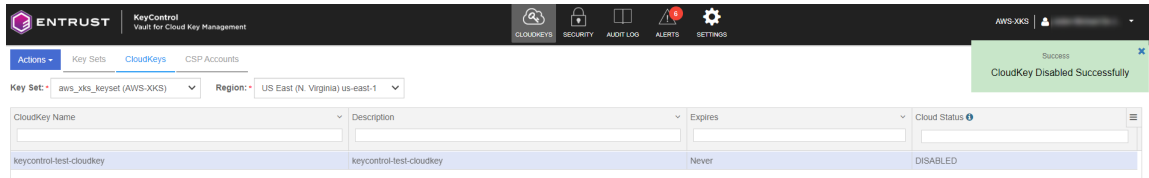


14. Return to KeyControl Cloud Key Management Vault and select **CLOUDKEYS > CloudKeys**.

15. Select the CloudKey and then select **Actions > Disable CloudKey**.

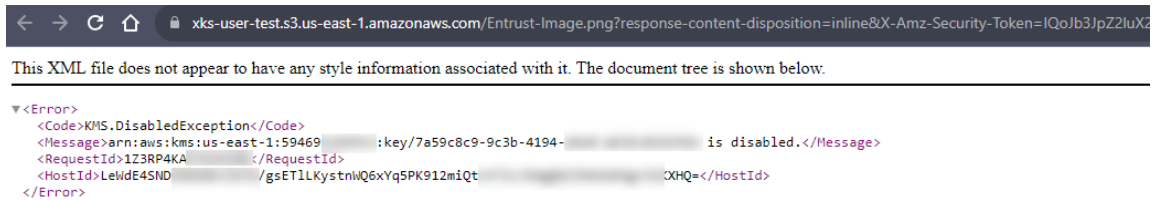


The CloudKey is disabled.

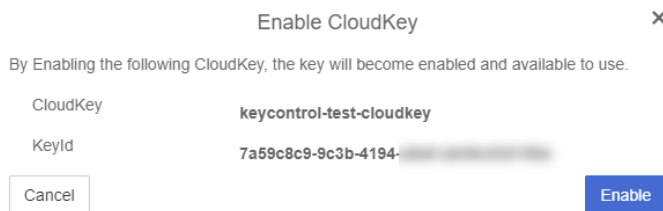


16. Return to the AWS S3 bucket and attempt to open the uploaded image.

The image is not viewable as the CloudKey was disabled.

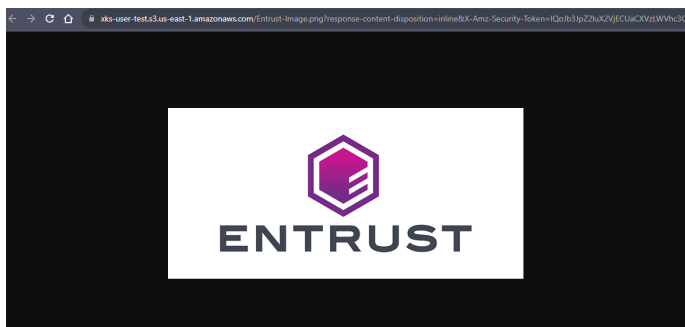


17. Re-enable the **CloudKey** in the KeyControl CloudKey Management Vault.



18. Return to the AWS S3 bucket.

19. Open the uploaded image again. It is now viewable.



This concludes the integration process.

Chapter 3. Additional resources and related products

3.1. nShield Connect

3.2. nShield as a Service

3.3. Entrust digital security solutions

3.4. nShield product documentation