THALES

Vormetric Data Security Platform

CipherTrust Cloud Key Manager

Installation & Configuration Guide

Release 1.6.2
Document version 1

August 27, 2019

CipherTrust Cloud Key Manager Installation & Configuration Guide

Version 1.6.2

August 27, 2019, Documentation Version 1

Produced in the United States of America

NOTICES, LICENSES, AND USE RESTRICTIONS

Contents

1 Preface

This guide describes how to deploy the CipherTrust Cloud Key Manager (CCKM), and how to configure it to connect with various data sources and cloud services. It also describes how to use the CCKM web interface.

1.1 Intended audience

This guide is for the system administrator who will install and configure the CCKM. This role is the "CCKM Admin." This guide describes how to use the CCKM GUI for managing Azure, Azure China, Azure Germany, Azure Stack, Salesforce, Salesforce Sandbox, and/or AWS Key Management Service (KMS) keys. These users, who must have access credentials to their organizations' Azure, Salesforce, or AWS accounts, are called "Cloud administrators."

1.2 Document version history

Software version Document version	Date	Changes
CCKM version 1.6 Document version 1	11/20/2018	First version of this guide.
CCKM version 1.6 hotfix Document version 2	01/28/2019	This release introduces support in CCKM to allow Azure guest users to take part in B2B collaboration for Azure tenants. In this release, the Microsoft Graph permission in Azure is now an optional permission instead of a required one.
CCKM version 1.6.1 Document version 1	03/26/2019	In this release of CCKM, support for nShield Connect as a Key Provider is now available for all of the Azure cloud services (Azure, Azure China, Azure Germany, and Azure Stack). Support for the Salesforce Sandbox cloud service in CCKM is also now available.
CCKM version 1.6.2 Document version 1	08/27/2019	In this release of CCKM, support for Salesforce cache-only keys, Azure service principal authentication, and certificates as an app credential in Azure is now provided. In addition, a restricted shell within the CCKM is now in place to avoid a security vulnerability that stemmed from the root access.

1.3 Service updates and support information

The license agreement that you have into to acquire the Thales products (“License Agreement”) defines software updates and upgrades, support and services, and governs the term under which they are provided. Any statements made in this guide or collateral documents that conflict with the definitions or terms in the License Agreement shall be superseded by the definitions and terms of the License Agreement. Any references made to “upgrades” in this guide or collateral documentation can either apply to a software update or upgrade.

For support and troubleshooting issues:

- http://help.thalesesecurity.com
- http://support.vormetric.com
- support@thalesesecurity.com
- (877) 267-3247

For Thales Sales:

- https://enterprise-encryption.vormetric.com/contact-sales.html
- sales@thalesesec.net
- (408) 433-6000

2 Overview

The CipherTrust Cloud Key Manager (CCKM) is part of the Vormetric Data Security Platform. It addresses enterprise needs for encrypting data in the cloud while retaining custodianship of encryption keys to comply with data security mandates in cloud storage environments.

The CCKM provides key management and storage and is compatible with the following three service providers:

- Salesforce (Salesforce and Salesforce Sandbox) (BYOK)
- Microsoft Azure (Azure, Azure China, Azure Germany, and Azure Stack) (BYOK)
- Amazon Web Services (AWS) (BYOK)

The CCKM uses the Vormetric Data Security Manager (DSM) as an underlying appliance that generates, stores, and retrieves encryption keys used by the CCKM servers. For the Azure and AWS service providers, CCKM also provides support for the use of the nShield Connect hardware security module (HSM) to generate, store, and retrieve encryption keys used by the CCKM servers. nShield Connect is an HSM that provides secure cryptographic processing within a tamper-resistant casing. With the use of nShield Connect, the keys are stored as blobs within the MongoDB. CCKM uses the MongoDB to hold system and user configuration information.

The CCKM solution is delivered as a virtual appliance that can be installed in one of the following methods:

- On-premises by deploying an `.ova` file
- In Amazon Web Services by deploying an Amazon Machine Image (AMI)
- In Azure and Azure Stack Marketplace by deploying a Virtual Hard Disk (VHD) image

The features and functionality are the same for these deployment scenarios.

2.1. CCKM Components

The components for a complete CCKM solution include:

CCKM	Cloud key management with Web UI.
Vormetric Data Security Manager (DSM)	Not included in the installation file; must already be installed and configured. Supported versions of DSM are **6.1.0.9118** or later. Optional, if you are using CCKM to manage keys in the Azure (Azure, Azure China, Azure Germany, and Azure Stack) or AWS service provider and nShield Connect as your key provider. If you are using CCKM to manage keys in Salesforce, the DSM is required.
nShield Connect hardware security module (HSM) (applicable to Azure and AWS service providers)	Not included in the installation file; must already be installed and configured. Supported versions of nShield Connect are versions **12.40.2** and **12.50.2**. Optional, if you are using CCKM to manage keys in the Azure (Azure, Azure China, Azure Germany, and Azure Stack) or AWS service provider and the nShield Connect as your key provider.
MongoDB	Used to store configuration information for the CCKM. Supported versions of MongoDB are versions **3.4.10**, **4.0.6, and 4.0.10.** If you are using nShield Connect as your key provider, then the source keys you generate from nShield Connect, the backup keys, and the Azure user name and passwords (if used) are stored in MongoDB as blobs. The installation file includes a community (free) edition of MongoDB that can be used for debug purposes. An external MongoDB in replica set configuration is recommended for a production setup.

2.2. How it works: Installation and Configuration Flow

Follow these steps to install and configure the solution:

1 Perform baseline decision making and information gathering.
See Table 1: Setup for CCKM to Connect to DSM, nShield Connect, and MongoDB.

2 Set up the appropriate key management blade to connect CCKM and the key management service. See Table 2 Setup for Key Management Security Blade Options.

3 Installation Prerequisites

Before installing the virtual appliance, it is necessary to gather information and perform some setup steps.

3.1 Virtual Machine Specifications for CCKM Components

3.1.1 With Local MongoDB bundled with CCKM appliance (Recommended for PoCs Only)

POC - Hardware recommendation

- CCKM virtual machine (VM) minimum requirements:
    - 2 CPUs
    - 4 GB Memory
    - 20 GB Hard disk (Thin)

If used in Development/Staging – Hardware recommendation
CCKM VM minimum requirements:

- 2 CPUs
- 4 GB Memory
- 50 GB Hard disk (Thin)

If used in Production – Hardware recommendation (*Not recommended for production as High Availability via CCKM clustering will not be available)
CCKM VM minimum requirements:

- 4 CPUs
- 8 GB Memory (Use 16 GB if API driven high frequency workflows are needed)
- 250 GB Hard disk (Thin)

3.1.2 With External MongoDB (Recommended for Production)

Production – Hardware recommendation
CCKM VM minimum requirements:

- 2 CPUs
- 8 GB Memory (Use 16 GB, if API driven high frequency workflows are needed)
- 10 GB Hard disk (Thin)

External MongoDB VM minimum requirements for each node:

- 4 CPUs
- 8 GB Memory (Use 16 GB, if API driven high frequency workflows are needed)
- 250 GB Hard disk (Thin)

Dev/Staging – Hardware recommendation
CCKM VM minimum requirements:

- 2 CPUs
- 4 GB Memory (Use 8 GB if API driven high frequency workflows are needed to be tested)
- 10 GB Hard disk (Thin)

MongoDB VM minimum requirements:

- 2 CPUs
- 4 GB Memory (Use 8 GB if API driven high frequency workflows are needed to be tested)
- 50 GB Hard disk (Thin)

3.2 At a Glance: Gather Information and Plan Your Installation

Use the tables to organize your plan, gather the configuration details needed, and decide whom else in the organization to include (for example, a DSM administrator, a Salesforce administrator, or an AWS administrator).

Table 1: Setup for CCKM to Connect to DSM, nShield Connect, and MongoDB

	On-Premises	AWS
CCKM Deployment Scenario licensed	Install using .ova file on VMWare ESXi Server Change default password Note the IP address of the VM used Multiple instances can be installed for multiple CCKM servers, if desired	Install using AMI on AWS Change default password Note the IP address of the AMI used Multiple instances can be installed for multiple CCKM servers, if desired
Data Security Manager (DSM)	Optional, if you are using CCKM to manage keys in the Azure (Azure, Azure China, Azure Germany, and Azure Stack) or AWS service provider and nShield Connect as your key provider. If you are using CCKM to manage keys in Salesforce, the DSM is required. The CCKM can use the DSM as the underlying appliance that generates, stores, and retrieves encryption keys used by the CCKM servers. If you plan to use the DSM, then you must install it. Install version 6.1.0.9118 or later. For any DSMs used, the DSM administrator must perform the following in the DSM: 1. **Create a DSM All Admin account** 2. **Provide:** * DSM fully-qualified host name * CCKM DSM domain name * DSM CCKM administrator name and password **Note**: The CCKM DSM domain name must conform to DSM domain name restriction. **Note**: CCKM communicates with the DSM through ports 443 and 8445.	Same as on-premises
nShield Connect	Optional, if you are using CCKM to manage keys in the Azure (Azure, Azure China, Azure Germany, and Azure Stack) or AWS	Same as on-premises

	service provider and the nShield Connect as your key provider. Install version 12.40.2 or later of nShield Connect. For the Azure (Azure, Azure China, Azure Germany, and Azure Stack) or AWS service provider, CCKM also provides support for the use of the nShield Connect HSM to generate source keys. With the use of nShield Connect, the keys are stored as blobs within the MongoDB. **Note**: This documentation assumes that you are familiar with using the nShield Connect HSM. This documentation also assumes that you have installed nShield Connect, and the associated RFS and Security World. For information on how to install an nShield Connect, and the associated RFS and Security World, refer to the *nShield® Connect Installation Guide*. Perform the following steps to use nShield Connect as a key provider: 1. Configure nShield Connect and note its IP address. You will add this IP address when you configure CCKM to connect to the nShield Connect. 2. Set up RFS and note its IP address. You will add this IP address when you configure CCKM to connect with nShield Connect. 3. Set up Security World. 4. Launch CCKM and note its IP address. You will add this IP address when you configure nShield Connect. 5. In the nShield Connect, configure CCKM as an nShield client by specifying the IP address of CCKM. For information on how to configure the CCKM as an nShield client, refer to the *nShield Connect User Guide for Unix*. 6. Within the **Settings** page > **nShield** tab of CCKM, specify the IP addresses of the nShield Connect HSM(s) and the remote file system (RFS) and add the configurations to associated Security World. Refer to section 13.5.3, Configure nShield, for more information. **Note**: CCKM communicates with nShield Connect through port 9004.	
MongoDB configuration database	Choose one of the following and perform the pre-setup steps: • **Embedded (free) MongoDB** No pre-setup required • **External MongoDB (single or replica)** **Supported versions are 3.4.10**, **4.0.6, and 4.0.10** To configure an existing MongoDB, create two new databases and a user document, as described in section 3.4.2, Scenario 2: Configure an existing MongoDB. • **External MongoDB with LDAP (replica)** **Supported versions are 3.4.10**, **4.0.6, and 4.0.10** To configure an existing MongoDB with LDAP, create a new database and a user document, as described in section 3.1.2, Scenario 3: Configure an existing MongoDB with LDAP. **Note**: CCKM communicates with MongoDB through port 27017.	Same as on-premises

Note: For Azure, each of the supported clouds (Azure, Azure China, Azure Germany, and Azure Stack) must be set up independently of each other, as these are separate and distinct key management services within Azure.

Table 2 Setup for Key Management Security Blade Options

	Azure Blade (Azure, Azure China, Azure Germany, and/or Azure Stack)	Salesforce Blade (Salesforce and/or Salesforce Sandbox)	AWS Blade
Create an app as doorway between CCKM and the service	**For Azure, Azure China and Azure Germany:** In the Azure portal, create an Azure Active Directory application and set the required permissions for the application. Then depending on the type of app credential you plan to employ, either create a key (client secret) in the Azure portal or generate a certificate from CCKM, download it, and then upload it to Azure. Note the Application ID and client secret. If CCKM will be used on behalf of users, then in the Azure administrator portal, add each of the users as a Key Vault contributor in Access Control IAM and Access Policies. If CCKM will be used as a service principal, then add the CCKM app as Key Vault contributor in Access Control IAM and Access Policies. **For Azure Stack using Azure Active Directory (AAD):** In the Azure portal, create an application for each portal (tenant or administrator portal). Set the application's required permissions and get a key. Note the Application ID and client secret. In the Azure Stack, in the administrator portal and tenant portal, add respective application as Owner in the IAM access control of the Subscription. **For Azure Stack using Active Directory Federation Service (ADFS):** Create two service principals using a PowerShell script. Add respective service principal as owner to IAM access control of the subscription in each of the Azure Stack portals (tenant and administrator portal).	**For Salesforce and Salesforce Sandbox:** Create a connected app Note the Consumer Key and Consumer Secret. If you are using both Salesforce and Salesforce Sandbox, create a connected app for each cloud service.	n/a
Before using CCKM	Set user permissions correctly in Azure. Set up Azure Log Analytics service to enable the CCKM Reports functionality. **Note**: Azure Stack does not support Log Analytics. Skip this step for Azure Stack.	Have: - Salesforce Shield Platform Encryption service enabled. -At least one valid Salesforce certificate. -A Salesforce account with “Manage Encryption Keys” permissions enabled.	AWS users should have: - Access key ID and secret access key - IAM GetUser permission for

		- If you are using Salesforce cache-only keys, ensure your Salesforce account also has “Customize Application permission” enabled. Also ensure the “Allow Cache-Only Keys with BYOK” option within the Security > Platform Encryption > Advanced Settings page in Saleforce is also enabled.	self - Proper KMS permissions

3.3 Create DSM Domain and Account for CCKM

When you install the CCKM appliance, take note of the IP address.

Ask the DSM administrator to:

1. Go to the dashboard of the DSM Management Console to obtain the fully qualified domain name (FQDN) of the DSM and give it to you.
2. Create a DSM administrator account of type 'All admin'. Provide the user name and password for this account.

3.4 Set up MongoDB

The CCKM uses MongoDB to store system and user configuration information. Two possible MongoDB configurations are described below.

Note: Supported versions of MongoDB are versions **3.4.10**, **4.0.6**, and **4.0.10**.

Note: Be sure to back up your MongoDB database(s) on a regular basis. For more information on how to back up MongoDB, see section 14.3, Backup and Restore for MongoDB.

Note: Although the CCKM includes a version of MongoDB, the software is supported by MongoDB. Contact the MongoDB Support team with any inquiries or issues.

3.4.1 Scenario 1: Use built-in MongoDB for test purposes

No pre-setup is required. When prompted by the CCKM Configuration Wizard, if you choose this option, then:

Database IP Address: *localhost*
Login: Automatically filled in by the wizard
Password: Automatically filled in by the wizard
[**Port number**: By default, is set to 27017]

3.4.2 Scenario 2: Configure an existing MongoDB

To configure an existing MongoDB instance (either single or replica set), it is necessary to 1) create two databases *vcg_db* and *kmaas*, along with the DB username and password, 2) create a user document in *vcg_db* for the CCKM Web UI admin.

The following are MongoDB commands. (Replace `mypassword` with the password you choose):

```
use vcg_db
db.createUser( { user: “vcg_user”, pwd: “mypassword”, roles: [ “dbOwner” ] } )
use kmaas
db.createUser( { user: “kmaas_user”, pwd: “mypassword”, roles: [ “dbOwner” ] } )
use vcg_db
db.user.insert({'_cls' : 'User.CustUser', 'username' : 'admin', 'password' : '',
  'is_staff' : false, 'is_active' : false, 'is_superuser' : true, 'locked' :
false,
  'try_count': 0, 'locked_at': new Date(), 'last_login' : new Date(),
  'date_joined' : new Date(), 'user_permissions' : [ ]})
```

3.4.3 Scenario 3: Configure an existing MongoDB with LDAP

Before configuring mongoDB with LDAP, request of your IT administrator to set up user accounts in the LDAP server for the usernames of “vcg_user” and “kmaas_user” using the same password for both. In addition, both user accounts must be configured to belong to user group of "dbOwner".

To configure an existing MongoDB instance with LDAP, it is necessary to 1) create a database *vcg_db* 2) create a user document in *vcg_db* for the CCKM Web UI admin.

The following are MongoDB commands:

```
use vcg_db
db.user.insert({'_cls' : 'User.CustUser', 'username' : 'admin', 'password' : '',
  'is_staff' : false, 'is_active' : false, 'is_superuser' : true, 'locked' :
false,
  'try_count': 0, 'locked_at': new Date(), 'last_login' : new Date(),
'date_joined' : new Date(), 'user_permissions' : [ ]})
```

Login into mongoDB database with admin account -

To create vcg_user and kmaas_user in the mongoDB database –

```
db.getSiblingDB("$external").createUser({ user: "vcg_user", roles: [ {
role: "dbOwner", db: "vcg_db" } ] })

db.getSiblingDB("$external").createUser({ user: "kmaas_user", roles: [ {
role: "dbOwner", db: "kmaas" } ] })

Login into mongoDB database using vcg_user. Below are the instructions -

mongo --authenticationMechanism PLAIN --authenticationDatabase '$external' -u
vcg_user -p <password> vcg_db
```

use vcg_db

```
db.user.insert({'_cls' : 'User.CustUser', 'username' : 'admin', 'password' : '',

  'is_staff' : false, 'is_active' : false, 'is_superuser' : true, 'locked' : fals
e,

  'try_count': 0, 'locked_at': new Date(), 'last_login' : new Date(),

  'date_joined' : new Date(), 'user_permissions' : [ ]})
```

To Verify the kmaas_user can login into mongoDB. Below is the command –

```
mongo --authenticationMechanism PLAIN --authenticationDatabase '$external' -u
kmaas_user -p <password> kmaas
```

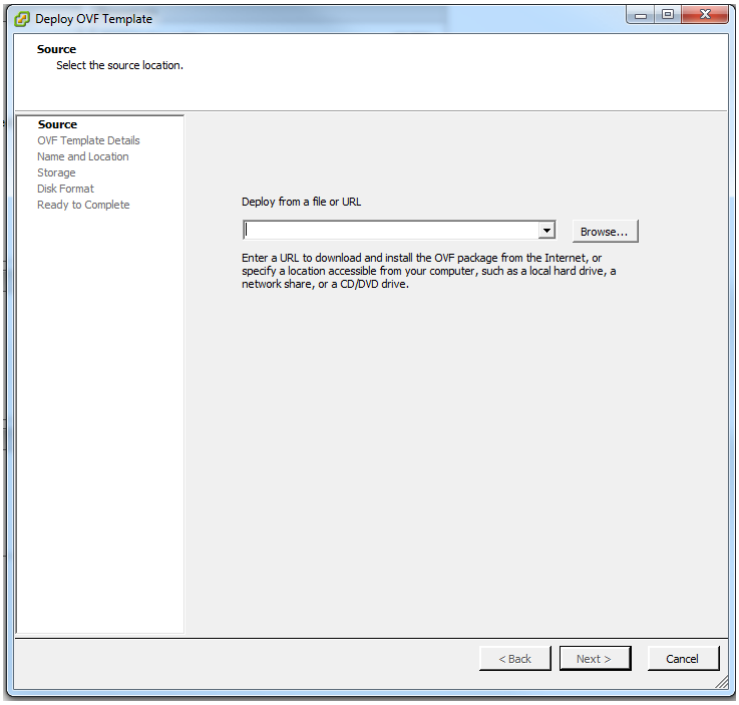
4 Install and Configure CCKM

Install CCKM on either VMWare or AWS.

Note: CCKM is supported on VMware ESXi 6.0 and later versions.

4.1 Install on VMWare ESXi Server

To create the CCKM Server on a VMware ESXi server, load the CCKM server's `.ova` template file in the VMware ESXi server and create the virtual machine (VM). Give the CCKM VM a name and keep the default resource settings.



Take note of the IP address of the VM. This will be the IP Address of CCKM.

4.2 Launch on Azure Marketplace

CCKM can be launched and configured from Azure Marketplace.

4.3 Install on AWS

The administrator should be familiar with AWS in general and AMI installations in particular.

4.3.1 Start the Installation

Contact Thales eSecurity Support to obtain the CCKM Amazon Machine Image (**AMI**).

1 Log in to your AWS account.

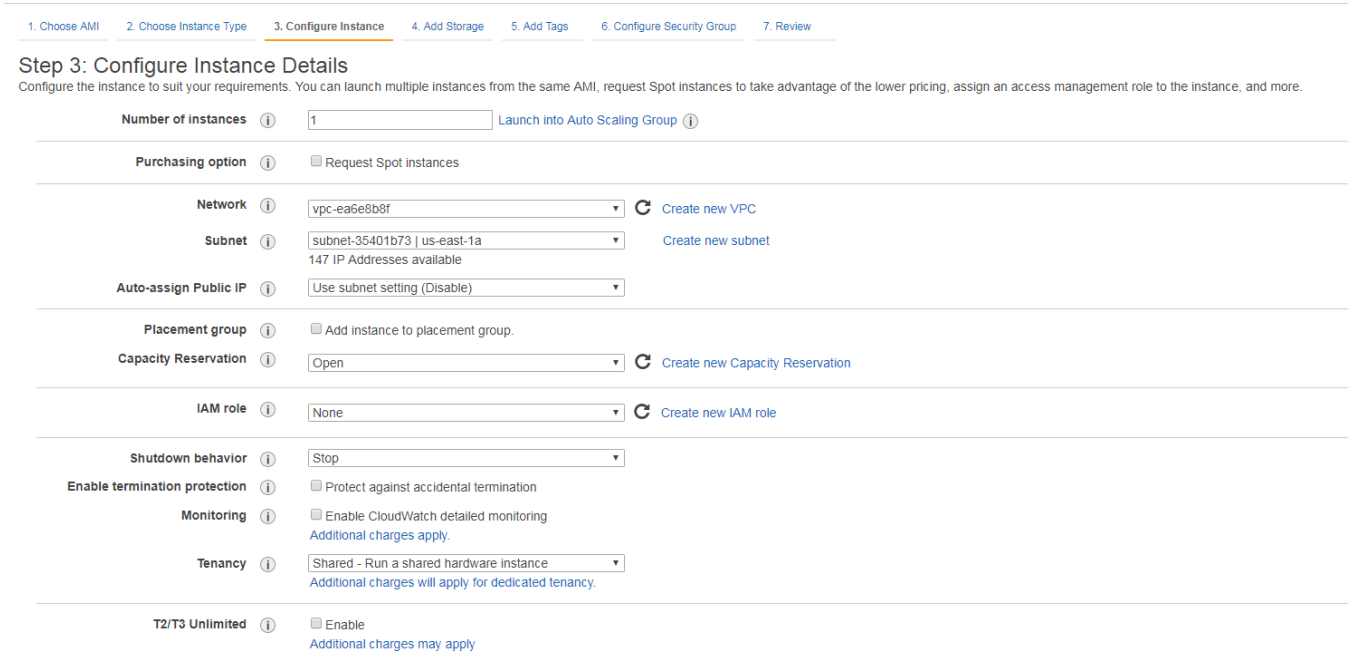
2 Find the AMI in the AWS Dashboard at **EC2 > IMAGES > AMIs > Private images**.

3 Select the correct version of the CCKM AMI and click **Action > Launch**.

4 At **Choose an Instance Type**, select:

- **Family**: General Purpose
- **Type**: m4.xlarge
- **vCPUs**: 4
- **Memory**: 16
- **Network Performance**: High

5 Click **Next**: Configure Instance Details.



4.3.2 Configure Instance Details

Fill out the *Configure Instance Details* page as follows:

1 In the **Network** box, select a VPC.

2 Select your subnet; it must be in the same subnet as the DSM.

3 If you want the CCKM available on a public IP address, then set *Auto-assign Public IP* to **Enable**. This will become the CCKM GUI address.

4 Click **Next: Add Storage**. Go to section 4.3.3, *Complete the Installation*.

4.3.3 Complete the Installation

On the *Add Storage* page:

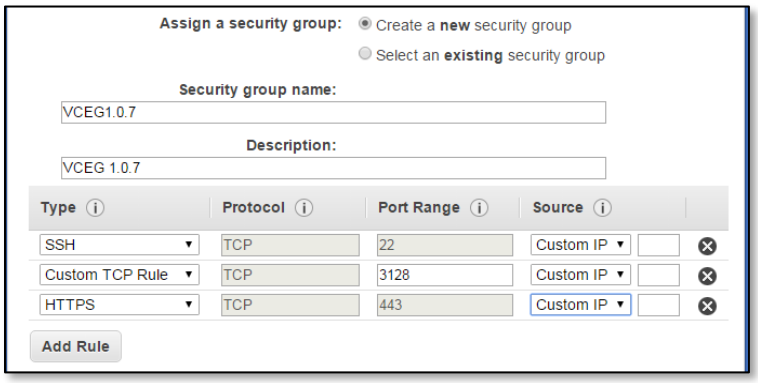
1 Use the default settings for adding storage. Click **Next: Tag Instance**.

2 In the *Tag Instance* page, create a name (key) and value to help organize your Amazon resources. In the *Name* column, type "**Name**". In the *Value* column, type the name you want to use for this instance.
Note the following about tags (from Amazon's AWS documentation):

Tags don't have any semantic meaning to Amazon EC2 and are interpreted strictly as a string of characters. Also, tags are not automatically assigned to your resources. You can edit tag keys and values, and you can remove tags from a resource at any time. You can set the value of a tag to an empty string, but you can't set the value of a tag to null. If you add a tag that has the same key as an existing tag on that resource, the new value overwrites the old value. If you delete a resource, any tags for the resource are also deleted.

3 Click **Next: Security Group**.

4 Select **Create a new security group**, enter a group name, and click **Add Rule** to configure the following sets of parameters:



- **Type:** SSH, **Port:** 22, **Source**: specify the SSH source that can reach your instance
- **Type**: HTTPS, **Port**: 443, **Source**: This port is used to access the CCKM Admin UI
- **Type**: HTTPS, **Port**: 8443, **Source**: This port is used to access the CCKM GUI

Note: Rules with source 0.0.0.0/0 allow all IP addresses to access your instance (not recommended). Do not leave source IP addresses with the default 0.0.0.0/0.

---

5 Click **Review** and Launch.

6 Review parameters and click **Launch**.
A dialog box allows you to define a new public key and private key for EC2.

7 Create a new key/value pair and click **Launch Instances**.

8 You can now use the public or private DNS in your browser to access the CCKM GUI.

4.4 Use SSH to Change the Default Password on the Virtual Machine

After launching CCKM on a VM, it is important to change the default password as a security measure. To do so:

1 SSH to the IP address of the new VM (the IP address of CCKM).

2 Enter the username and password as prompted.
Username: `vormetric`
Password: `vormetric123`

3 Follow the prompt: Enter the old password, and then a new password twice. **Store securely**.

ATTENTION: This username/password combination is used to access the VM via SSH. **There is no way to recover a forgotten password— in this case, the VM would have to be discarded and a new CCKM launched**.

4 The SSH session automatically ends. To validate, launch a new session and log in with the new password.

4.5 Access the CCKM GUI

Open a browser and point to the CCKM IP address with the https prefix (for example: *https://19.4.5.89*).
The configuration wizard is displayed.

Once the CCKM is configured, the CCKM IP address will open to the CCKM GUI Home page.

4.6 CCKM Cluster Configuration

For high availability (HA) and improved performance of the CCKM solution, you can set up a CCKM cluster using at least two CCKM servers, one MongoDB database, and a load balancer. To set up the CCKM cluster, launch two or more CCKM virtual machines (VMs) using the same OVA, AMI, or VHD file for each VM you add to the cluster, configure the VMs to use the same MongoDB database, and then configure the load balancer to operate on the VMs. To access the CCKM Web UI in a cluster configuration after all components are installed and configured, use the DNS name or IP address of the load balancer.

When you configure the database information on the second or additional CCKMs using “Wizard Step 1: Configure Database” of the CCKM Configuration Wizard, enter the same configuration passphrase and database information that you entered (in the **Configuration Passphrase** box) for the first CCKM. This configuration allows the additional CCKM servers to join the first CCKM in the cluster and share the same CCKM configuration in the MongoDB database. When configuring the DSM account information in “Wizard Step 2: Configure DSM” of the CCKM Configuration Wizard, do not enter the DSM account information as you have already associated the DSM to the first CCKM in the cluster when configuring this step earlier. The DSM account information is shared among all of the CCKMs in the cluster. However, you are required to enter a unique CCKM host name (in the **CCKM Hostname** box) for each CCKM in the cluster. Be sure to enter a unique name for each of the CCKMs in Wizard Step 2. For information about wizard steps 1 and 2, refer to Chapter 5, Launch CCKM Configuration Wizard.

When configuring the load balancer to load balance the CCKM servers in the cluster, configure the listeners to listen to and forward traffic from ports **443** and **8443**. For port **8443**, enable application generated cookie stickiness (or session stickiness) on the cookie "JSESSIONID". For health check, set the ping target to https:8443/kmaas/health. Refer to the documentation for the load balancer you are employing in the CCKM cluster for information on how to configure these configurations.

5 Launch CCKM Configuration Wizard

The configuration wizard opens automatically on first access. Use it to associate MongoDB and DSM to the CCKM and to set the CCKM administrator password.

Note: The CCKM Configuration Wizard appears only once and is not available after the configuration is complete.

Note: **After completing the configuration of CCKM configuration wizard, you cannot change the configured username of the DSM Security administrator through the CCKM Admin portal. Changing it will require you to contact Thales eSecurity support for assistance.** However, you can change the configured password of the DSM Security administrator through the CCKM Admin portal by selecting **Settings** from the left-hand navigation bar and then selecting **DSM**. For more information about changing the DSM password from the **Settings** page of the CCKM Admin portal, see section 13.5.2.1, Change DSM Admin Password.

5.1 Wizard Step 1: Configure Database

The CCKM server requires a MongoDB database server to store user setup and cloud usage data. This wizard step provides the steps to connect the CCKM server to the MongoDB database server(s). Prior to performing the steps in this wizard, the MongoDB database server(s) must be deployed. The CCKM solution also includes a built-in version of MongoDB, which is to be used for demo and testing purposes only. See 3.4, Set up MongoDB, for the prerequisites prior to performing the steps in this wizard.

In this wizard, you have the option to perform authentication of the MongoDB users through an AD server using LDAP. Perform the following steps *prior* to enabling this feature in this wizard: In the AD server, create two user accounts (“`vcg_user`” and “`kmass_user`”) with the *same* user password and then add these users to the AD group “`dbOwner`”.

During the configuration of the MongoDB database server in this wizard, it is required that you enter a passphrase. This passphrase is used to derive a master key. The master key, in turn, is used to encrypt all passwords, including the DSM and MongoDB passwords, and other sensitive user credentials that are entered into CCKM. The passphrase is shared among all CCKMs within a multi-CCKM environment and must be entered as part of the configuration of each additional CCKM server.

Note: After the passphrase is saved to CCKM, it cannot be changed or deleted. It is important that you note this passphrase for future use. If you configure additional CCKMs for a multi-CCKM environment, then you are required to enter the same passphrase when configuring these CCKMs.

If you need assistance with recovering your passphrase, contact Thales eSecurity support.

In this wizard, you also have the option to configure the proxy server settings. You can also perform this configuration within the **Proxy Settings** tab from the **Settings** page within the CCKM Admin portal after the CCKM is up and running.

Enter the following information in the database configuration page of the wizard:

If configuring the proxy server settings at this time:

1 Select **Add Proxy Settings**. The **Proxy Settings** dialog box displays.

2 Enter the hostname or IP address of the proxy server in the **Hostname** box.

3 Enter the port number of the proxy server in the **Port** box.

4 In the **Skip Proxy List** box, enter the hostnames (separated by a pipe symbol "|") that are to pass through the proxy server.

5 (Optional) Enter the username of the proxy port in the **Username** box.

6 (Optional) Enter the password for the proxy server in the **Password** box.

7 Click **Add** to add the setting.

If using the built-in MongoDB database server within CCKM (for demo or testing purposes only):

1 Select **Use Local Database**. The wizard automatically fills the information for the following boxes with the default values:
Database IP: *localhost*
Port: 27017
Database Name: vcg_db
Username: vcg_user
Password: ******

2 Enter a passphrase in the **Configuration Passphrase** box. The passphrase must be between 12 and 32 characters long (standard ASCII alphabet characters (a-z, A-Z), integers (0-9), and a limited set of special characters (!@#$%^&*(){}[]). **It is important that you note this passphrase and store it securely**.

3 Click **Check Database Connection** to verify that the connection between the MongoDB database server and the CCKM works.

4 Click **Next**.

If using an external MongoDB database server (single instance):

1 Select **Use Remote Database**.

2 Enter a passphrase in the **Configuration Passphrase** box. The passphrase must be between 12 and 32 characters long (standard ASCII alphabet characters (a-z, A-Z), integers (0-9), and a limited set of special characters (!@#$%^&*(){}[]). **It is important that you note this passphrase and store it securely**.

3 Enter the IP address of the MongoDB database server in the **Database IP** box.

4 Enter the port number of the MongoDB database server in the **Port** box.

5 Optional. Check the **Use LDAP for authentication** box to enable authentication of the MongoDB users through an AD server using LDAP.

6 Enter the name of the MongoDB database in the **Database Name** box.

7 Enter your username and password for the MongoDB database in the **Database Login Credentials** boxes.

8 Click **Check Database Connection** to verify that the connection between the MongoDB database server and the CCKM works.

9 Click **Next**.

If using external MongoDB database servers (replica set):

1 Select **Use ReplicaSet Database**.

2 Enter a passphrase in the **Configuration Passphrase** box. The passphrase must be between 12 and 32 characters long (standard ASCII alphabet characters (a-z, A-Z), integers (0-9), and a limited set of special characters (!@#$%^&*(){}[]). **It is important that you note this passphrase and store it securely**.

3 Enter the name of the MongoDB replica set in the **Replica Set Name** box.

4 For each instance of the MongoDB, enter the IP address, host name and port number of each database servers within the **Database IP**, **Database Hostname**, and **Port** boxes (starting with the information for the primary MongoDB instance).

5 Optional. If you have more MongoDB instances to add, click **Add More DB**.

6 Optional. Check the **Use LDAP for authentication** box to perform authentication of the MongoDB users through an AD server using LDAP.

7 Enter the name of the MongoDB database in the **Database Name** box.

8 Enter your username and password for the MongoDB database in the **Database Login Credentials** boxes.

9 Click **Check Database Connection** to verify that the connection between the (primary) MongoDB database server and the CCKM works.

10 Click **Next**.

5.2 Wizard Step 2: Configure DSM (optional)

In this wizard step, enter the DSM account information gathered in section 3.2, At a Glance: Gather Information and Plan Your Installation.

Note: **After completing the configuration of CCKM configuration wizard, you cannot change the configured username of the DSM Security administrator through the CCKM Admin portal. Changing it will require you to contact Thales eSecurity support for assistance.** However, you can change the configured password of the DSM Security administrator through the CCKM Admin portal by selecting **Settings** from the left-hand navigation bar and then selecting **DSM**. For more information about changing the DSM password from the **Settings** page of the CCKM Admin portal, see section 13.5.2.1, Change DSM Admin Password.

The **DSM Configuration** page is displayed.

1 Enter the following values obtained from the DSM administrator:

- **DSM Hostname**: Enter the Fully Qualified Hostname of the DSM, which you can find on the dashboard of the DSM Management Console. Example: `dsm-a43.vormetric.com`
- **DSM** Credentials**.**
  - **Domain**: The DSM domain name. Ensure this name conforms to DSM domain name restriction.
  - **Username** and **Password**: The DSM Security administrator credentials required to add the CCKM host to that DSM domain. The domain and administrator credentials are created by a DSM administrator of type *All*.
- **CCKM Hostname**: For a multi-CCKM environment, use a different name for subsequent CCKM servers.

- **Static Route**: Optional. To add a static route to the CCKM, select the **Static Route** check box to bring up the following fields:
  IP: Destination
  Gateway: Internet gateway
  Netmask: Netmask of the static route
  net: Type of route. Can be either net or host

2 Click **Next**.

5.3 Wizard Step 3: Set CCKM Root Admin Password

In this wizard step, enter the CCKM root administrator password. The **Set Admin Password** page is displayed.

1 Enter the following values:

- **Login ID:** The default is **admin** and cannot be changed
- **Administrator first and last name**
- **Administrator password** to be used to log into the CCKM GUI and must be 8 to 20 characters, with at least one capitalized letter, one lowercase, and one digit.
- **Confirm password**

2 Click **Finish**.

See also Chapter 13, Maintenance Tools for Administrators, to add/delete administrators, check license and DSM status, check the health monitor, change the root admin password, and more.

---

6 Create an Azure Key Management Blade

Create a connection (key management security blade) between CCKM and an Azure Key Vault. CCKM provides support for Azure clouds (Azure, Azure China, Azure Germany, and Azure Stack). The steps for creating a blade for Azure, Azure Germany, and Azure China differ from those steps to creating a blade for Azure Stack.

6.1 Creating an Azure Key Management Blade using Azure, Azure Germany, or Azure China

This section describes how to create a key management blade for Azure, Azure Germany, and Azure China. The steps to create the connection between CCKM and an Azure Key Vault are the same for Azure, Azure Germany, and Azure China.

Note: The UI within the Azure portal changes periodically. In regards to the documentation relating to Azure App Registrations, this documentation reflects the **legacy App Registrations**, which displays as **App registrations (Legacy)** in the Azure portal.

6.1.1 Prerequisites

Before configuring an Azure blade, in the Azure portal, you must create (or register) the CCKM App and assign required permissions. Then depending on the type of app credential you plan to employ, either create a key (client secret) in the Azure portal or generate a certificate from CCKM, download it, and then upload it to Azure. This entire process generates the connection data needed to configure the blade.

Also before configuring an Azure blade, refer to the following related sections to review information about how Azure allows external applications, such as CCKM, to access and manage resources within Azure and the two types of app credentials employed when registering apps in Azure: |

- 10.1, Accessing and Managing Azure Resources
- 10.2, Registering Apps in Azure

Note: In a production environment, Thales recommends that you DO NOT change the key management blade for Azure, Azure Germany, or Azure China by changing the setting for the Use CCKM as Service Principal check box.

6.1.1.1 Create an Azure Active Directory Application (Multi-tenanted)

To create an Azure Active Directory application in the Azure portal:

1 In the Azure Active Directory > App registrations > New application registration, provide the following parameters:

- **Name:** choose a name for the app that CCKM will use to access Azure.
- **Application type**: Web app/API
- **Sign-on URL:**
  https://{CCKM Instance Host name}:8443/kmaas/login/azure (for Azure)
  https://{hostname}:8443/kmaas/login/azureChina (for Azure China)
  https:// {hostname}:8443/kmaas/login/azureGermany (for Azure Germany)

  If you are using CCKM as a service principal, then in addition to the above URLs, also add the following URLs:

https://{CCKM Instance Host name}:8443/kmaas/auth2/azure (for Azure)
https://{hostname}:8443/kmaas/auth2/azureChina (for Azure China)
https:// {hostname}:8443/kmaas/auth2/azureGermany (for Azure Germany)

https://{CCKM Instance Host name}:8443/kmaas/azureAdminConsent/azure (for Azure)
https://{hostname}:8443/kmaas/azureAdminConsent /azureChina (for Azure China)
https:// {hostname}:8443/kmaas/azureAdminConsent /azureGermany (for Azure Germany)

{hostname} must resolve to the IP address of your CCKM Instance.

Note: If you have deployed a CCKM cluster, then the {hostname} must resolve to the IP address of the load balancer. For more information about CCKM cluster configuration, refer to section 4.6, CCKM Cluster Configuration.

2 Click **Create** to create the app.

3 Access the new app under *App Registrations*.

4 From *App Registrations > {App Name} > Settings*: Note the **Application ID**. This is the "Client ID" used when configuring the CCKM security blade.

5 From App Registrations > {App Name} > Settings > Properties: select **Yes** for **Multi-tenanted**

6 Click **Save**.

6.1.1.2 Set Required Permissions

To set the required permissions on the CCKM app:

1 Access **App Registrations** > **{App Name}** > **Settings** > **Required Permissions** in Azure.

2 Click **+Add** and add the following APIs and their associated permissions:

- **Azure Key Vault | DELEGATED PERMISSIONS | √** Have full access to Azure Key Vault service
- **Windows Azure Service Management API | DELEGATED PERMISSIONS | √** Access Azure Service Management as organization users (preview)
- **Windows Azure Active Directory (Microsoft.Azure.ActiveDirectory) | DELEGATED PERMISSIONS | √** Sign in and read user profile
- **Microsoft Graph | DELEGATED PERMISSIONS | √** Read directory data
  Note: The Microsoft Graph permission is now an optional permission instead of a required one. The Microsoft Graph permission is required to query the names of the Azure applications or services using an Azure key from Azure Logs Analytics when you generate the **Azure Key Service Usage Report** from CCKM. The generated report includes the names of these applications or services and other related information. Without this permission set, the IDs (instead of the names) of the applications or services using an Azure key are included in the report after you run it.

3 Select **Grant Permissions** and select **Yes**. (You must have Azure administrator permissions to Grant Permissions as required by the Microsoft Graph API.)

6.1.1.3 Create a Key

If you will use a secret key or client secret in Azure as part of registering the apps in Azure, create this key in Azure. When setting access to an Azure blade in the CCKM *Step 2 Set Access to the Security Blade* screen, you will provide this secret key.

To create a secret key (or client secret):

1 Access **App Registrations** > **{App Name}** > **Settings** > **Keys**.

2 Enter a key name in **Key description** box.

3 Select a duration period in the **Duration** box.

4 Click **Save.** The **Key Value** column displays the new client secret.
IMPORTANT: Copy this value and store securely; it cannot be retrieved after leaving this view. It will display as Hidden within the **Value** column. (This value is the "client secret" used when configuring the CCKM security blade.)

6.1.2 Select the Blade

In this step, select a blade type and add it to CCKM.

Set access to the security blade:

1 Log in to the CCKM Admin GUI using the credentials defined in section 5.3, Wizard Step 3: Set CCKM Root Admin Password.

2 Select **Blades** from the left-hand navigation bar, select **Add Blade**, and then select the Azure blade (**Azure, Azure China, or Azure Germany)** you wish to set up from the **Security Blade** drop-down list.

3 Add an optional description and click **Next**.
The *Step 2 Set Access to the Security Blade* screen is displayed.

6.1.3 Step 2: Set access to Azure, Azure Germany, or Azure China blade

In this step, set access to Azure, Azure Germany, or Azure China blade:

Note: If you will use CCKM as a service principal, then be sure to, select the **Use CCKM as Service Principal** box

1 Enter:
Client ID: the 'application id' of the Azure app (see section 6.1.1.1, Create an Azure Active Directory Application (Multi-tenanted).

If you using a client secret, select **Use client secret** and enter the client secret in the **Client Secret for the Azure Application** box. This client secret is the key created in the **Keys** section of the Azure app (see section 6.1.1.3, Create a Key).
If you are using a certificate, select **Use certificate** and then click the **Generate and download** button. A message displays indicating a new certificate is generated and ready to download. Download the certificate and save it to a directory within your laptop or desktop. The certificate thumbprint automatically displays in the **Thumbprint** box after you download the certificate.

Note: Upload this certificate to the Azure Portal after completing the configuration of the blade. See section 6.4, Upload a Certificate to Azure Portal, for more information.

Redirect URL: the URL entered as a Sign-on URL (see section 6.1.1.4, Create a Key) or created in the 'Reply URLs' section of Azure.

2 If you will use CCKM as a service principal, select the **Use CCKM as Service Principal** checkbox.

Note: The **Use CCKM as Service Principal** check box acts as a toggle to use CCKM on behalf of users or as a service principal. You can change this setting after completing the configuration of the blade. To use CCKM on behalf of users, ensure this checkbox is deselected. **In a production environment, Thales recommends that you DO NOT change the key management blade for Azure, Azure Germany, or Azure China by changing the setting for the Use CCKM as Service Principal check box.**

3 Click **Next**.
The **Step 3 Configure Security Blade Settings** page is displayed. Proceed to section 6.2, Step 3: Configure Settings.

6.2 Step 3: Configure Settings

CCKM for Azure can send email alerts for user actions performed and scheduled using SMTP. Set optional notification information and log levels here.

Note: A single configured SMTP host is common across all configured blades. Therefore, the SMTP-related configurations that you enter in Step 3 are applicable to all blades that you configure. If you configure the SMTP-related configurations for one blade, skip this optional configuration for the other blades that you plan to configure.

1 Enter Notification information:
SMTP Settings: Optional: enter SMTP host, password, user name, and from address.
Log Level: Select the log level required in log files. The default is ERROR.

2 Click **Next**. The *DSM Certificate* page is displayed.

6.3 Step 4: Upload DSM Certificate

CCKM for Azure performs key operations via REST APIs exposed in the DSM. It requires the DSM certificate to invoke the APIs. You can download the certificate in Base64 format via a browser pointing to the DSM.

1 Click **Choose File**.

2 Upload the certificate in Base64 format.

3 Click **Finish**.

6.4 Upload a Certificate to Azure Portal

If you downloaded a certificate while setting access to Azure, Azure Germany, or Azure China blade (described in section 6.1.3, Step 2: Set access to Azure, Azure Germany, or Azure China blade), then upload this certificate to the Azure portal after completing the configuration of the blade.

To upload a certificate from CCKM to the Azure portal:

1 Access the new app under *App Registrations*.

2 From App Registrations > {App Name} > Settings > Keys, click **Upload Public Key**.

3 From the **Upload a certificate (public key)** selection box, select the certificate you wish to upload to the Azure portal. After it is uploaded, the certificate thumbprint is displayed in the **Thumbprint** column under **Public Keys**.

6.5 Creating a Key Management Blade for Azure Stack with Azure AD

This section describes how to create a key management blade for Azure Stack with Azure Active Directory (AAD).

If you are using AAD as the identity provider, you need to register CCKM application twice (one for the administrator portal and one for user portal) in AAD and provide CCKM with the associated client credentials. The process is similar to adding an Azure blade.

Note: This documentation assumes that you are familiar with using Microsoft Azure Stack. This documentation also assumes that you have installed and configured Azure Stack in your data center**.** For information on Azure Stack, refer to the Microsoft Azure Stack documentation.

Note: The features that allow for the use of CCKM as a service principal to access and manage resources within Azure and the use of a certificate (public key) as an app credential within CCKM and Azure are currently **not** supported in Azure Stack.

6.5.1 Prerequisites

Before configuring an Azure Stack blade with the use of the directory type of AAD, you must create (or register) an App in Azure, assign required permissions, and create a key. This process generates the connection data needed to configure the blade.

6.5.1.1 Create an Azure Active Directory Application

This section provides the steps to create an administrator application and a Tenant application.

Note: If you are enabling the Azure Stack administrator blade and Azure Stack blade, then you will need to register CCKM application twice--the first time for the "azureStackAdmin" login endpoint and the second time for "azureStack" login endpoint.

In the Azure Active Directory> App registrations> New application registration, provide the following parameters:

- **Name:** choose a name for the app that CCKM will use to access Azure Stack.
- **Application type**: Web app/API
- **Sign-on URL:** https://{ hostname }:8443/kmaas/login/azureStackAdmin (for Azure Stack administrator)
  https://{ hostname}:8443/kmaas/login/azureStack (for Azure Stack Tenant)

  {hostname} must resolve to the IP address of your CCKM Instance.

After clicking **Create**, access the new app under *App Registrations*.

From *App Registrations > {App Name} > Settings*: Note the **Application ID**. This is the “Client ID” used when configuring the CCKM security blade.

6.5.1.2 Set Required Permissions

- Access App Registrations > {App Name} > Settings > Required Permissions in Azure.
- **+Add** the following APIs and their associated Permissions:

For Azure Stack (tenant), add:

- **Azure Stack - KeyVault | DELEGATED PERMISSIONS | √** Access Azure Stack - KeyVault

For Azure Stack (administrator), add:

- **AzureStack - KeyVault Internal | DELEGATED PERMISSIONS | √ Access AzureStack KeyVault Internal**
- **AzureStack - Administration | DELEGATED PERMISSIONS | √ Access Azure Stack – Administration**

For both Azure Stack (tenant) and Azure Stack (administrator), add:

- **Windows Azure Service Management API | DELEGATED PERMISSIONS | √** Access Azure Service Management as organization users (preview)
- **Windows Azure Active Directory (Microsoft.Azure.ActiveDirectory) | DELEGATED PERMISSIONS | √** Sign in and read user profile
- **Microsoft Graph | DELEGATED PERMISSIONS | √** Read directory data
  Note: The Microsoft Graph permission is now an optional permission instead of a required one. The Microsoft Graph permission is required to query the names of the Azure applications or services using an Azure key from Azure Logs Analytics when you generate the **Azure Key Service Usage Report** from CCKM. The generated report includes the names of these applications or services and other related information. Without this permission set, the IDs (instead of the names) of the applications or services using an Azure key are included in the report after you run it.

- Under *Required Permissions,* select **Grant Permissions**. (You must have Azure administrator permissions to Grant Permissions, as required by the Microsoft Graph API.)

For AAD administrator app, add:

Azure Stack - Administration | Access Azure Stack - Administration Windows Azure Service Management API | Access Azure Service Management as organization users (preview) AzureStack KeyVault Internal | Access AzureStack KeyVault Internal Microsoft Graph | Read directory data (Delegated Permissions) Windows Azure Active Directory | Sign in and read user profile (Delegated Permissions)

For AAD Tenant app, add:

Azure Stack | Access Azure Stack
Windows Azure Service Management API | Access Azure Service Management as organization users (preview) Azure Stack KeyVault | Access Azure Stack KeyVault Microsoft Graph | Read directory data (Delegated Permissions) Windows Azure Active Directory | Sign in and read user profile (Delegated Permissions)

6.5.2 Select the Blade

1 Log in to the CCKM Admin GUI using the credentials defined in section 5.3, Wizard Step 3: Set CCKM Root Admin Password.

2 Select **Blades** from the left-hand navigation bar, select **Add Blade**, and then select **Azure Stack** from the **Security Blade** drop-down list.

3 Add an optional description and click **Next**.
The *Step 2 Set Access to the Security Blade* screen is displayed.

6.6 Step 2: Set Access to Azure Stack with AAD

This section provides the steps for setting access to Azure Stack with AAD as well as the associated prerequisite steps.

6.6.1 Prerequisites

Part of setting access to Azure Stack with ADD includes entering the required URLs for the full path to the endpoints. To obtain the URLs used in your cloud environment, run the following Azure commands using the Azure CLI in a terminal:

Note: You are required to run these commands for the "azureStackAdmin" cloud environment as an administrator and the "azureStack" cloud environment as a user. Essentially, you will run these commands twice—as an administrator and as a user.

1 In the role of administrator, set the Azure CLI environment for the “azureStackAdmin” cloud environment:
```
az cloud set --name AzureStackAdmin
```

In the role of an Azure user, set the Azure CLI environment for the “azureStack” cloud environment:
```
az cloud set --name AzureStackUser
```

2 Login to Azure CLI using the user name and password of the administrator or user (depending on the role you are using for the login):
```
az login
```

3 Display the configuration details of the cloud environment including the full path to the endpoints:
```
az cloud show
```

Take note of the displayed URLs requested for steps 3, 4, and 5 of section 6.3.2, Set Access to Azure Stack with AAD.

For more information about the Azure cloud-management commands and login command listed in this section, refer to the Azure Command-Line Interface (CLI) documentation within the Microsoft Azure online documentation.

6.6.2 Set Access to Azure Stack with AAD

In this step, set access to Azure Stack with AAD.

1 From **Azure Stack Type**, select **AAD**.

2 Enter the **Tenant Client Credentials**:
Tenant Client ID: The client ID of the Azure app (see section 6.2.1.16.4.1, Create an Azure Active Directory Application).
Tenant Secret Key: The client secret of Azure Stack tenant (see section 6.2.1.1, 6.4.1Create an Azure Active Directory Application).
Redirect URL: the tenant URL entered as a Sign-on URL in section 6.2.1.1 or created in the 'Reply URLs' section of Azure Stack (see section 6.2.1.16.4.1, Create an Azure Active Directory Application).
Enter the **Admin Client Credentials**:
Admin Client ID: The client ID of the Azure administrator (see section 6.2.1.16.4.1, Create an Azure Active Directory Application).
Admin Secret Key: The client secret of Azure Stack administrator.
Redirect URL: the admin URL entered as a Sign-on URL in section 6.2.1.1 or created in the 'Reply URLs' section of Azure Stack.

3 Enter the **Common Portal URIs and Endpoints** for **Azure Active Directory Authority:** the URL of the AD Authority. For example, https://login.windows.net

4 Enter **Tenant Portal URIs and Endpoints**:
Refresh Token URL: the URL of the Refresh token. For example, https://vault.contoso.com/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
Resource Manager URL: the URL of the Resource Manager.
For example, URL https://management.local.azurestack.external
Key Vault DNS Suffix: the URL of the Key Vault DNS Suffix. For example, vault.local.azurestack.external
Graph DNS Suffix: the URL of the Graph DNS Suffix. For example, graph.local.azurestack.external
Active Directory Service Endpoint Resource ID: the URL of the AD Service Endpoint Resource ID.
For example, https://management.contoso.com/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx

5 Enter **Admin Portal URIs and Endpoints**:
Refresh Token URL: the URL of the Refresh token. For example, https://adminvault.contoso.com/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
Resource Manager URL: the URL of the Resource Manager. For example, https://adminmanagement.local.azurestack.external
Key Vault DNS Suffix: the URL of the Key Vault DNs Suffix. For example, adminvault.local.azurestack.external
Graph DNS Suffix: the URL of the Graph DNS Suffix. For example, graph.local.azurestack.external
Active Directory Service Endpoint Resource ID: the URL of the AD Service Endpoint Resource ID. For example, https://adminmanagement.contoso.com/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx

6.7 Creating a Key Management Blade for Azure Stack with ADFS

This section describes how to create a key management blade for Azure Stack with Active Directory Federation Service (ADFS).

If you are using ADFS as the identity provider, you need to create two service principals for CCKM (one for the administrator portal and one for user portal) in ADFS and provide CCKM with the associated client assertion credentials.

Note: This documentation assumes that you are familiar with using Microsoft Azure Stack. This documentation also assumes that you have installed and configured Azure Stack in your data center**.** For information on Azure Stack, refer to the Microsoft Azure Stack documentation.

6.7.1 Prerequisites

Before configuring an Azure Stack blade with the use of the directory type of ADFS, perform the following steps:

- Create two service principals, one for administrator and one for tenant.
  Refer to the following Azure Stack documentation covering how to create the service principals for ADFS:

  https://docs.microsoft.com/en-us/azure/azure-stack/azure-stack-create-service-principals#create-service-principal-for-ad-fs

  Creating the service principals requires you to generate a certificate for the service principals. The certificate will be used as the client assertion credentials for the client assertion authentication. When creating a service principal, ensure to provide the clientRedirectUris. The following are examples of clientRedirectUris for the CCKM administrator and tenant:

  - New-GraphApplication -Name ‘CCKM_Admin’ -ClientRedirectUris "https://<hostname>:8443/kmaas/login/azureStackAdmin" -ClientCertificates $using: <certificate>}
  - New-GraphApplication -Name ‘CCKM_tenant’ -ClientRedirectUris "https://<hostname>:8443/kmaas/login/azureStack" -ClientCertificates $using:cert}

IMPORTANT: Be sure to note the app ID of the service principal and store the certificate securely. This is the client certificate used when configuring the CCKM security blade.

6.7.2 Select the Blade

1 Log in to the CCKM Admin GUI using the credentials defined in section 5.3, Wizard Step 3: Set CCKM Root Admin Password.

2 Select **Blades** from the left-hand navigation bar, select **Add Blade**, and then select **Azure Stack** from the **Security Blade** drop-down list.

3 Add an optional description and click **Next**.
The *Step 2 Set Access to the Security Blade* screen is displayed.

6.8 Step 2: Set access to Azure Stack with ADFS

In this step, set access to Azure Stack with ADFS.

1 From **Azure Stack Type**, select **ADFS.**

2 Enter the **Tenant Client Credentials**:
Tenant Client ID: the App ID of the tenant service principal (see section 6.4.1, Prerequisites).
Tenant Client Certificate: Export the certificate part of the tenant service principal PFX certificate to a PEM file. Then click **Choose File** to select the exported PEM certificate of the Azure Stack tenant service principal.
Tenant Client Certificate Private Key: Export the private key part of the tenant service principal PFX certificate to a PEM file. Then click **Choose File** to select the exported PEM private key of the Azure Stack tenant service principal.
Redirect URL: the tenant URL entered as ClientRedirectURIs when creating the service principal.

3 Enter the **Admin Client Credentials**:
Admin Client ID: the client ID of the Azure administrator app (see section 6.4.1, Prerequisites).
Admin Client Certificate: Export the certificate part of the administrator service principal PFX certificate to a PEM file. Then click **Choose File** to select the exported PEM certificate of the Azure Stack administrator service principal.
Admin Client Certificate Private Key: Export the private key part of the administrator service principal PFX certificate to a PEM file. Then click **Choose File** to select the exported PEM private key of the Azure Stack administrator service principal.
Redirect URL: the administrator URL entered as ClientRedirectURIs when creating the service principal.

4 Enter the **Common Portal URIs and Endpoints** for Azure
Authorization endpoint: the URL of the authorization endpoint. For example, https://adfs.local.azurestack.external/adfs/oauth2/authorize
Token Endpoint URL: the URL of the token endpoint. For example, https://adfs.local.azurestack.external/adfs/oauth2/token
Active Directory Authority: the URL of the AD Authority. For example, https://adfs.local.azurestack.external/adfs

5 Enter **Tenant Portal URIs and Endpoints**:
Refresh Token URL: the URL of the Refresh token. For example, https://vault.adfs.azurestack.local/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
Resource Manager URL: the URL of the Resource Manager. For example, URL https://management.adfs.azurestack.external
Key Vault DNS Suffix: the URL of the Key Vault DNS Suffix. For example, vault.local.azurestack.external
Graph DNS Suffix: the URL of the Graph DNs Suffix. For example, graph.local.azurestack.external
Active Directory Service Endpoint Resource ID: the URL of the AD Service Endpoint Resource ID. For example, https://management.adfs.azurestack.local/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx

6 Enter **Admin Portal URIs and Endpoints**:
Refresh Token URL: the URL of the Refresh token. For example, https://adminvault.adfs.azurestack.local/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
Resource Manager URL: the URL of the Resource Manager. For example, https://adminmanagement.adfs.azurestack.external
Key Vault DNS Suffix: the URL of the Key Vault DNs Suffix. For example, vault.local.azurestack.external
Graph DNS Suffix: the URL of the Graph DNS Suffix. For example, graph.local.azurestack.external
Active Directory Service Endpoint Resource ID: the URL of the AD Service Endpoint Resource ID. For example, https://adminmanagement.adfs.azurestack.local/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx

6.9 Step 3: Configure Settings

CCKM for Azure Stack (with ADFS) can send email alerts for user actions performed and scheduled using SMTP. Set optional notification information and log levels here.

Note: A single configured SMTP host is common across all configured blades. Therefore, the SMTP-related configurations that you enter in Step 3 are applicable to all blades that you configure. If you configure the SMTP-related configurations for one blade, skip this optional configuration for the other blades that you plan to configure.

1 Enter Notification information:
SMTP Settings: Optional: enter SMTP host, password, user name, and from address.
Log Level: Select the log level required in log files. The default is ERROR.

2 Click **Next**. The *DSM Certificate* page is displayed.

6.10 Step 4: Upload DSM Certificate

CCKM for Azure Stack (with ADFS) performs key operations via REST APIs exposed in the DSM. It requires the DSM certificate to invoke the APIs. You can download the certificate in Base64 format via a browser pointing to the DSM.

1 Click **Choose File**.

2 Upload the certificate in Base64 format.

3 Click **Finish**.

7 Create a Salesforce Key Management Blade

Create a connection (key management security blade) between CCKM and Salesforce to permit CCKM access to any of the organization's users with valid Salesforce credentials. This section describes how to create a security blade for Salesforce or Salesforce Sandbox. A Salesforce sandbox is a copy of your production Salesforce environment used primarily for testing, development, or training purposes. The steps to create the connection between CCKM and Salesforce or Salesforce Sandbox are the same.

Note that in this chapter, the information regarding Salesforce is applicable to both Salesforce and Salesforce Sandbox unless otherwise noted. The CCKM UI for Salesforce and Salesforce Sandbox is identical with the exception of the display of the names of the cloud services. Your login credentials for Salesforce and Salesforce Sandbox are unique to each cloud service. If you are configuring for Salesforce Sandbox, be sure to configure within the Salesforce Sandbox cloud service.

7.1 Salesforce Prerequisites

Using the correct permissions in Salesforce, create a Salesforce connected app to which CCKM can connect.

7.1.1 Create a Connected App in Salesforce

1 In Salesforce, ensure you have the permissions required to **Create** connected apps.

2 In the Salesforce app creation interface, supply basic information (app name, description, etc.).

3 Define how the app communicates with Salesforce using the following **API** parameters:

- Enable OAuth Settings: check
- Callback URL for Salesforce: `https://{hostname}:8443/kmaas/login/salesforce`
  Callback URL for Salesforce Sandbox:
  `https://{hostname}:8443/kmaas/login/salesforcesandbox`

  {hostname} must resolve to the IP address of your CCKM Instance.

  Note: If you have deployed a CCKM cluster, then the {hostname} must resolve to the IP address of the load balancer. For more information about CCKM cluster configuration, refer to section 4.6, CCKM Cluster Configuration.
- Selected OAuth Scopes: choose *"Access and manage your data (api)"* and *"Access your basic information (id, profile, email, address, phone)"*.

4 Click Save. The Consumer Key is created and displayed, and the Consumer Secret is created.

Note the consumer key and secret and store them securely; you will use them when configuring the Salesforce (SFDC) or Salesforce Sandbox blade in CCKM.

7.2 Step 1: Select the Blade

1 Log in to the CCKM GUI using the credentials defined in section 5.3, Wizard Step 3: Set CCKM Root Admin Password.

2 Select **Blades** from the left-hand navigation bar, select **Add Blade**, and then select the Salesforce blade (**SFDC** or **SFDC Sandbox)** from the **Security Blade** drop-down list.

3 Add an optional description and click **Next**.
The *Step 2 Set Access to the Security Blade* page is displayed.

7.3 Step 2: Set Access to the Blade

In this step, enter the connection details between CCKM and the Salesforce app you created in section 7.1.1, Create a Connected App in Salesforce, to set access to the Salesforce or Salesforce Sandbox blade.

1 Enter:
Client ID: The value of the 'Consumer Key' of the Salesforce app
Client Secret: The value of the 'Consumer Secret' of the Salesforce app
Redirect URL: The value of the 'Callback URL' of the Salesforce app
For Salesforce: `https://{hostname}:8443/kmaas/login/salesforce`
For Salesforce Sandbox: `https://{hostname}:8443/kmaas/login/salesforcesandbox`

The hostname must resolve to the IP address of your CCKM Instance.

2 Click **Next**.
The *Step 3 Configure Security Blade Settings* page is displayed.

7.4 Step 3: Configure Settings

CCKM for Salesforce can send email alerts for user actions performed and scheduled using SMTP. Set optional notification information and log levels here.

Note: A single configured SMTP host is common across all configured blades. Therefore, the SMTP-related configurations that you enter in Step 3 are applicable to all blades that you configure. If you configure the SMTP-related configurations for one blade, skip this optional configuration for the other blades that you plan to configure.

1 Enter: **SMTP Settings:** Optional: enter SMTP host, password, user name, and from address.
Log Level: Select the log level required in log files. The default is ERROR.

2 Click **Next**. The *DSM Certificate* page is displayed.

7.5 Step 4: Upload DSM Certificate

CCKM for Salesforce performs key operations via REST APIs exposed in the DSM. It requires the DSM certificate to invoke the APIs. You can download the certificate in Base64 format through a browser pointing to the DSM.

1 Click **Choose File**.

2 Upload the certificate in Base64 format.

3 Click **Finish**.

8 Create an AWS Key Security Blade

8.1 Step 1: Select the Blade

1 Log in to the CCKM GUI using the credentials defined in Wizard Step 3: Set CCKM Root Admin Password.

2 Select **Blades** from the left-hand navigation bar, select **Add Blade**, and then select **AWS** from the **Security Blade** drop-down list.

3 Add an optional description and click **Next**.

Note: For AWS, there is no Step 2:Access. The Wizard progresses directly to Step 3.

8.2 Step 3: Configure Settings

CCKM for AWS Key Management can send email alerts for user actions performed and scheduled using SMTP. Set optional notification information and log levels here.

Note: A single configured SMTP host is common across all configured blades. Therefore, the SMTP-related configurations that you enter in Step 3 are applicable to all blades that you configure. If you configure the SMTP-related configurations for one blade, skip this optional configuration for the other blades that you plan to configure.

1 Enter: SMTP Settings: Optional: enter SMTP host, password, user name, and from address.
Log Level: Select the log level required in log files. The default is ERROR.

2 Click **Next**. The *DSM Certificate* page is displayed.

8.3 Step 4: Upload DSM Certificate

CCKM for AWS performs key operations via REST APIs exposed in the DSM. It requires the DSM certificate to invoke the APIs. You can download the certificate in Base64 format via a browser pointing to the DSM.

1 Click **Choose File**.

2 Upload the certificate in Base64 format.

3 Click **Finish**.

9 CCKM Overview

CipherTrust Cloud Key Manager (CCKM) is designed to address the security of encryption keys in cloud services environments. Holistic data security views, key visibility in hybrid Cloud deployments, ease of key management across public clouds, compliance requirements (such as PCI, FIPS 140-2 Level 3), automation of key lifecycle management, and data loss prevention are among the top requirements as well as the top challenges.

CCKM supports Salesforce, Azure Key Vault, and AWS Key management enabling enterprises who use the services to bring their own keys to those clouds, back up keys on premise, destroy cloud keys in specific situations, and manage the life cycle of cloud keys. CCKM uses the Vormetric Data Security Manager (DSM) to generate keys, store keys, and back up cloud keys. DSM is available in various form factors that satisfy FIPS 140-2 compliance at Level 1, Level 2, and/or Level 3 standards. The CCKM provides a web GUI for all functions.

CCKM provides support for Azure clouds (Azure, Azure China, Azure Germany, or Azure Stack). **Note that in this chapter, the information regarding Azure is applicable to all Azure clouds unless otherwise noted.** The CCKM UI for Azure, Azure China, Azure Germany, and Azure Stack is identical with the exception of the display of the names of the cloud services. And for Azure Stack, there are two portals--one for the administrator and one for the Tenant.

Note: CCKM is deployed on premise or in AWS. It is also possible to license CCKM as a hosted cloud service directly from Thales eSecurity. The hosted SaaS version is available as a multi-tenant offering.

This guide assumes that an organization has deployed the CCKM solution themselves, but if a SaaS version has been licensed from Thales eSecurity, the CCKM GUI will look the same.

9.1 CCKM Administration Privileges

The CCKM Administrator has root privileges on the server where the CCKM solution is deployed, and this administrator manages the tasks of adding and configuring key security blades. Unless the CCKM administrator also has login credentials to Salesforce, Azure, or AWS Key Management, he or she will not be able to access the CCKM portal pages directly.

To access the CCKM portal, a user (also known as a “Cloud administrator”) must have valid login credentials for the organization’s designated cloud services. No specific “CCKM Administrator” is created. Administrators have access from the portal to all cloud services for which they have valid credentials.

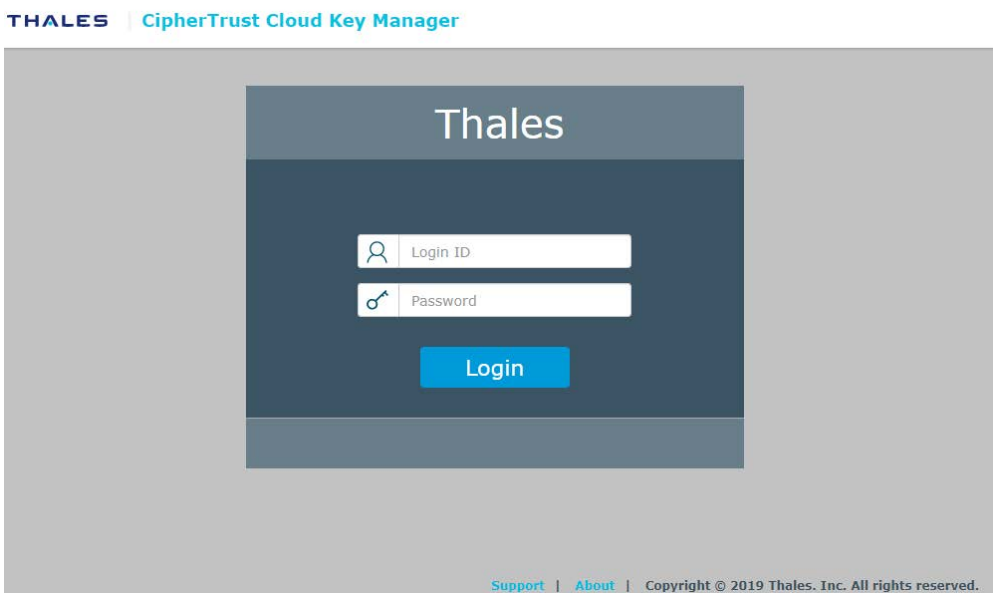
If an organization has licensed the SaaS CCMK option, then Thales eSecurity Support will provide a URL for accessing the CCKM portal, and any administrator with valid credentials for Salesforce, Azure, or AWS Key Management can access the portal pages.

9.1.1 Accessing CCKM as an Administrator

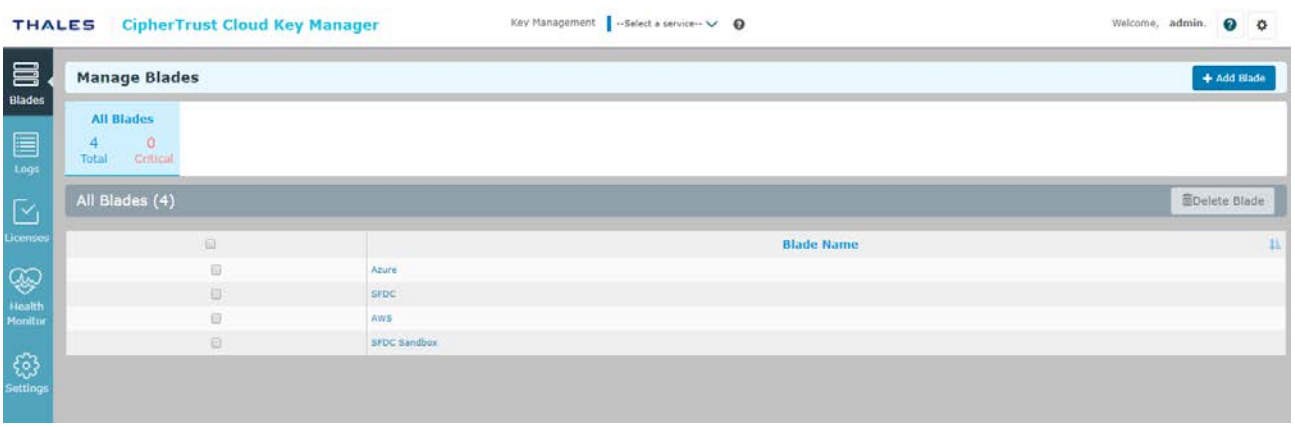
CCKM can be accessed from within the CCKM GUI or from the CCKM Home page.

From the CCKM GUI:

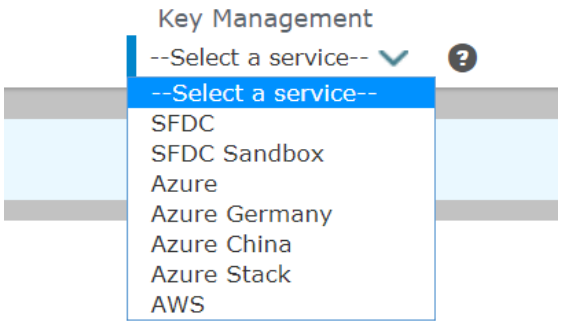
1 Go to the CCKM GUI by entering “https://<IP address of CCKM Instance>” in a browser. The CCKM Admin Login page is displayed:



2 Log in as a CCKM administrator using your Administrator credentials. The CCKM Admin portal is displayed:



3 Select an installed Key Management service link from the **Key Management** drop-down list at the top of the page.

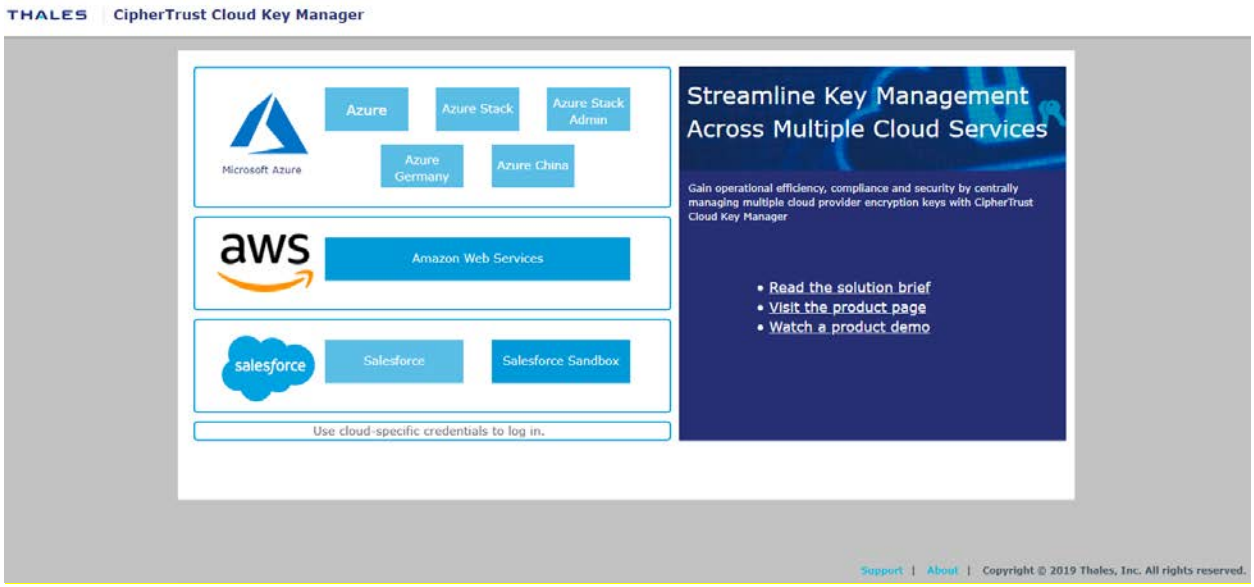


The login page for the selected Key Management service is displayed.

4 Enter your login credentials to log into CCKM. The CCKM Home page is displayed.

From the CCKM IP address:

1 Enter “https://<CCKM IP address>:8443/kmaas” in a browser.
The CCKM login page is displayed.



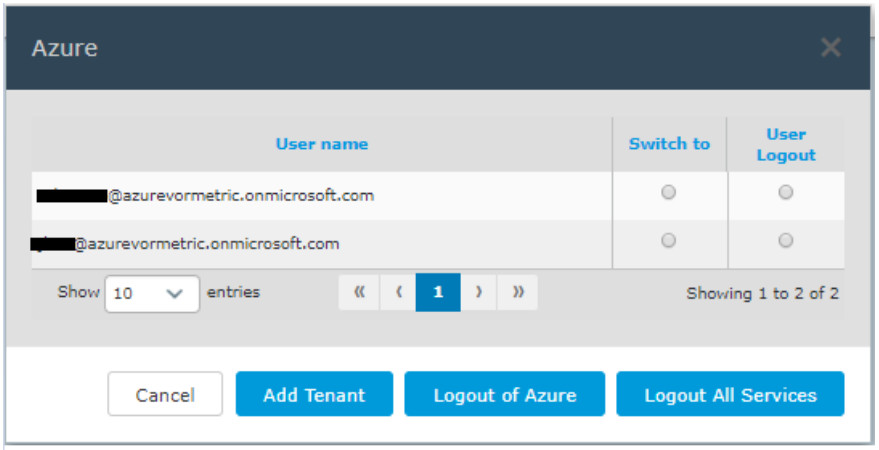
2 Select the installed Key Management service you wish to use. The login page for the selected Key Management service is displayed.

3 Enter your login credentials to log into CCKM. The CCKM Home page is displayed.

9.2 Support for Multi-accounts per Cloud Service

CCKM supports login and key management for multiple accounts or multiple users per Cloud service through a single CCKM UI (Multi-account Login/Logout) and browser window.

The following **Multi-account Login/Logout** window reflects the Azure account (varies depending on the cloud service account into which you are currently logged):



You can perform the following using the **Multi-account Login/Logout** window.

- Logon to multiple accounts for a specific Cloud service (AWS/SFDC/Azure)
- Logout of specific or all accounts within one cloud service or all cloud services
- Perform key management for multiple accounts from your current CCKM browser window

Accessing the Multi-account Login/Logout window

To access the **Multi-account Login/Logout** window for a specific cloud service, you must have at least one user already logged on to an account. Once you are logged onto an account, from the service **Dashboard**, click on the cloud service icon (Context Switcher) at the top of right of the page. The **Multi-account Login/Logout** window displays. The following sections provide instructions on how to use this window.

Switching to another user account (already logged on)

(Applicable for a given cloud service on to which you are currently logged) To access another user account, which is already logged on, select the **Switch to** button next to the user name (account) for which to log on. You are then switched to that account with the user name displayed at the top of the **CCKM Home** page.

Logging on to another user account

(Applicable for a given cloud service on to which you are currently logged) To log on to another user account, click one of following buttons depending on the service you are currently logged and then and enter the user credentials in the login dialog boxes of the cloud service:

- **Add Tenant** (applicable to Azure, Azure China, Azure Germany, and Azure Stack)
- **Add OrgID** (applicable to Salesforce and Salesforce Sandbox)
- **Add Account** (applicable to AWS)

Once the user authentication is completed, you are logged on to the given user account and the username is displayed at the top of the **CCKM Home** page.

Logging out of a user account

(Applicable for a given cloud service on to which you are currently logged) To logout of a specific user account, select the **User Logout** button next to the user name (account) for which to log out. The account is then logged out.

Logging out of all user accounts from a *specific* cloud service or *all* cloud services

To logout of all of the user accounts for a *specific* cloud service, click one of the following depending on the service to which you are currently logged:

- **Logout of Azure (Logout of Azure Germany, Logout of Azure China, or Logout of Azure Stack)**
- **Logout of Salesforce (Logout of Salesforce Sandbox)**
- **Logout of AWS**

To logout of all cloud services, click the **Logout All Services** button. All services are then logged out.

9.3 User Handling

User Authentication

For AWS users, the user provides their AWS long-term credentials (for example, access key Id and secret access key) to CCKM. CCKM uses these credentials to obtain a one-time, temporary credential for the user session. CCKM does not save user's long-term credentials.

For Azure and Salesforce users, the CCKM itself does not maintain user credentials. It uses OAuth2 to authenticate users.

During login, users are redirected from the CCKM portal to the Salesforce or Azure login page to enter their respective username/password combinations. Once authenticated, the CCKM portal obtains an OAuth2 access token from the cloud service and then maintains a user session between CCKM and the user's browser. While the user session is active, CCKM uses the access token to call Azure or Salesforce REST APIs, and will allow the logged-in user to access and manage keys within those services per the granted permission.

User Authorization

In Azure, user authorization defines what a user can do on Azure keys. For example, whether they can view, upload, update, delete, or restore keys in a key vault. This is managed within Azure Key Vault's Access Control (IAM) and Access Policies. To allow a user to manage keys in a key vault, the key vault owner must give the user "Key Vault Contributor" role in Access Control, and appropriate permissions as needed on keys in Access Policies.

In Salesforce, user authorization defines whether the user has "Manage Encryption Keys" permission or not. Only users that have Manage Encryption Keys permission can manage Tenant Secrets.

In AWS, user authorization defines what a user can do with AWS KMS. For a user to manage AWS CMKs, he or she should have proper KMS permissions. In addition, to use CCKM, the user should have IAM GetUser permission.

User Audit

User operations, including login and logout (successful or failed) are captured in CCKM audit logs. Currently there are no access policies on CCKM audit logs, so any logged-in user for a specific organization and service can view all audit logs, including operations made by other users in the past, within the scope of that organization.

9.4 Functionality

CCKM offers similar functionality for each cloud service it supports:

- **Life cycle management** of keys, key versions, and attributes:
    - View Keys
    - Update Keys
    - Upload Keys
    - Rotate Keys
    - Delete Keys
- **Disaster recovery** of keys
    - Backup Keys
    - Restore Keys
- **Hybrid key management**
    - On-premise Keys storage
    - Key synchronization with Azure Key Vault
- **Compliance Management**
    - Key storage within on premise FIPS 140-2 Level 3 storage
- **Certificate handling (Salesforce)**
    - View certificates
    - Synchronize certificates
    - Use certificates to wrap keys for secure key upload
- **Key visibility reporting**
    - Combined Key Activity Reconciliation Report
      Reconciliation before and after key import the cloud service
      Note: This report is not supported for Azure Stack.
    - Key Activity Report
      Insight into who, what and when keys were accessed, created, deleted across PaaS
      Note: This report is not supported for Azure Stack.
    - Key Aging Report
      Insight into who owns the keys and when keys expire

- Key Service Usage Report
  Insight into applications consuming the keys across IaaS/PaaS/SaaS
  Note: This report is not supported for Azure Stack, Salesforce, and Salesforce Sandbox.
- Cloud Key Manager User Action Report
  Insight into aggregated user activities such as delete, create etc. of keys

- Connection to a Remote Syslog Server for syslog messages

9.5 Supported Browsers for CCKM Portal

The CCKM portal supports:

- **Chrome** 51.0.2704 (64-bit) or later
- Firefox 45.0 or later
- Internet Explorer 11 or later

10 CCKM for Azure Key Management

This chapter is for Azure Key Vault users who access the CCKM.

CCKM provides support for Azure clouds (Azure, Azure China, Azure Germany, or Azure Stack). **Note that in this chapter, the information regarding Azure is applicable to all Azure clouds unless otherwise noted.** The CCKM UI for Azure, Azure China, Azure Germany, and Azure Stack is identical with the exception of the display of the names of the cloud services. And for Azure Stack, there are two portals--one for the administrator and one for the Tenant.

10.1 Accessing and Managing Azure Resources

Note: This section is only applicable to Azure, Azure China, and Azure Germany. It is **not** applicable to Azure Stack.

Azure allows external applications, such as CCKM, to access and manage resources within Azure in the following ways:

- On behalf of a user
- As a service principal

Both ways require that the application be first registered in Azure. For applications on behalf of a user, it is required that the Azure users be created and given the appropriate access controls and policies on the resources. The application obtains an access token through app+user authenticate. What an application can do with the Azure resources depends on what the user can do.

For as-service-principal applications, the application must be given the appropriate access controls and policies on the resources. The application obtains an access token through an *app-only authentication*. What an application can do with Azure resources depends on whether the service principal has proper permissions on the resources.

10.2 Registering Apps in Azure

When registering apps in Azure, the following are the two types of app credentials employed:

- Secret (password)—The user will generate a secret key in Azure when registering the app, and then copy the secret key and provide it to the app. When making OAuth authentication calls, the app will send the secret key to Azure.

- Certificate (public key)—the user will create a private key and public key pair locally, create a certificate for the public key, and then provide the certificate to Azure when registering the app. For the private key, the app will create a client assertion and send it to Azure when making OAuth authentication calls.

CCKM currently supports both types of app credentials for Azure, Azure China, and Azure Germany.

10.3 Prerequisites on Azure

The Azure administrator must ensure the following prerequisites are met:

- Each Azure key vault must belong to an Azure Resource Group.
- Key vault owner must give designated users or the CCKM app (if CCKM is used as a service principal) the "Key Vault Contributor" role in Azure Access Control, and appropriate permissions as needed on keys in Access Policies.

  Note: changing key vault Access Control and/or Access Policies will not affect the active CCKM user sessions. To reflect the change, a user must re-log in to CCKM to establish a new user session.

- CCKM Reports draw on the Azure Log Analytics service, which must be pre-configured if the CCKM Reports functionality is desired (see section 10.9, Reports). At minimum, you must have:
  - An Azure *Log Analytics* service created and associated with the relevant key vaults.
  - In the *Log Analytics > {Log service name} > Access Control (IAM)* section of the Azure portal, set "**Log Analytics Contributor**" permissions for any user who should be able to generate reports in CCKM. Other users can view, but not create reports.

  Note: Azure Stack does not currently support the *Log Analytics* service.

  For more information about how to configure Log Analytics in Azure key vaults, see the latest version of the following Microsoft Azure documentation:

  https://docs.microsoft.com/en-us/azure/log-analytics/log-analytics-azure-key-vault

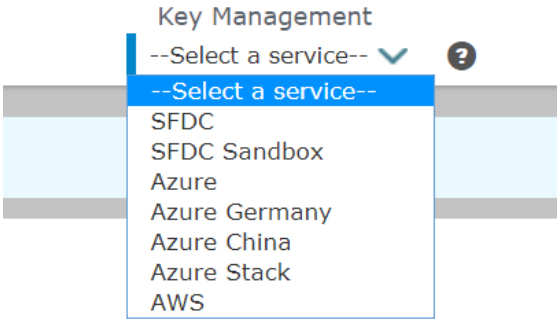
10.4 Access CCKM for Azure Key Management

This section describes the two ways in which to access the CCKM service for Azure Key Management.

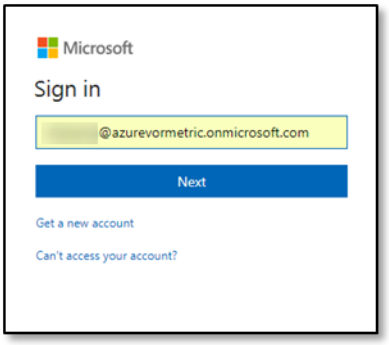
10.4.1 Accessing CCKM using User Login

To access the CCKM service for Azure Key Management using your Microsoft user login credentials:

1 In the CCKM Admin portal, select **Key Management** > **Azure** | **Azure Germany** | **Azure China** | **Azure Stack**.



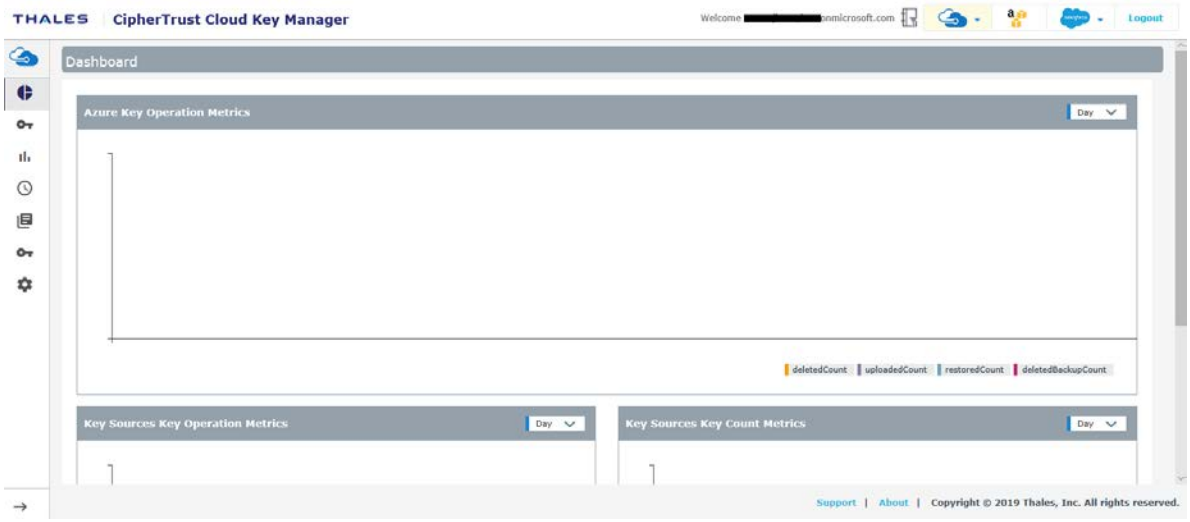
The Microsoft login page opens at *https://login.microsoftonline.com/*



OR

2 Access the CCKM login page (see section 9.1.1, Accessing CCKM as an Administrator).

3 Use your Azure credentials to log in. The Azure CCKM dashboard is displayed.



10.4.2 Accessing CCKM using Service Principal

Note: This section only applies to Azure, Azure Germany, and Azure China using CCKM as a service principal.

To log into CCKM for Azure using CCKM as a service principal, you are redirected to a CCKM login dialog box for Azure. If this is the first time you are logging into CCKM as a service principal, you (as an Admin user) are required to provide the admin consent to grant permissions to the CCKM app to access your Azure resources (your organization data). To do this, click the "admin consent" link (indicated by “here” in blue text at the top CCKM login dialog box) after which the CCKM Admin Consent dialog box is displayed. In this box, enter your Azure tenant name, tenant ID, and a new tenant password. After you click on the "Admin Consent" button, you are redirected to the Azure consent page where you can provide your consent (after reviewing the page) by clicking **Accept**. As an Admin user, you can confirm that CCKM has been granted the permissions as a service principal by logging into the Azure portal and verifying that the CCKM app is listed in the **Enterprise Applications** > **All Applications** page.

To access the CCKM service for Azure Key Management using CCKM as a service principal:

1 In the CCKM Admin portal, select **Key Management** > **Azure** | **Azure Germany** | **Azure China**.

You are redirected to the CCKM login dialog box for Azure to use CCKM as a service principal.

2 If this is the first time you are logging into CCKM using CCKM as a service principal, click on the "admin consent" link (indicated by “here” in blue text at the top CCKM login dialog box) to provide your admin consent to grant permissions to the CCKM app to access your Azure resources. (You must be logged in as an Admin user to provide your admin consent.) The CCKM Admin Consent dialog box is displayed.

Skip to Step 10, if this is not the first time you are logging into CCKM using CCKM as a service principal.

3 In the **Tenant Name** box, enter the tenant name. To locate your tenant name in the Azure portal (after logging in as an admin user), click **Directory + subscription** icon and view A**ll Directories.** If more than one directory is listed, find the domain name that is associated with the admin username within the list. This domain name is the tenant name.

4 In the **Tenant ID** box, enter the tenant ID. To locate your tenant ID in the Azure portal (after logging in as an admin user), click **Directory + subscription** icon and view A**ll Directories.** If more than one directory is listed, find the domain name that is associated with the admin username within the list. The value that displays under the domain name is your tenant ID. This value is equivalent to the **Directory ID** in Azure. Alternatively, you can locate the tenant ID within in the **Directory ID** box in **Directory properties** of the **Azure Active Directory** > **Properties** when logged in as an admin user.

5 To locate your tenant ID in the Azure portal (after login as admin user), go to **Azure Active Directory** > **Properties**. In **Directory properties**, locate the value displayed in the **Directory ID** box. This value is your tenant ID.

6 In the **Tenant Password** box, enter the tenant password. The password must be at least 8 characters long and contain one upper case letter, one lower case letter, one digit, and one special character from a limited set of special characters "!@#$%^&*()".

7 In the **Confirm Tenant Password** box, reenter the tenant password.

8 Click **Admin Consent** to provide your consent to Microsoft. You are redirected to the Microsoft login page to select an account to use for the login.

9 Select the account. The Microsoft consent prompt is displayed.

10 Review the information and then click **Accept** to accept the request to grant permissions to the CCKM app to access your Azure resources as a service principal.

11 In the CCKM Azure login dialog box, in the **Tenant Name** box, enter the tenant name.
Note: Ensure the tenant name you enter in this step matches the name entered in step 3.

12 In the **Tenant Password** box, enter the tenant password.
Note: Ensure the tenant password you enter in this step matches the password entered in step 5.

13 Click **Login**. The Azure CCKM dashboard is displayed.

10.5 Azure CCKM Portal

The Azure CCKM portal contains the following menu options in the left-hand navigation bar:

- **Home:** Home page displays key operation and count metrics.
- **Keys:** Keys page displays key management overview and provides the Upload and Synchronize functionality, and the Delete, Restore, and Auto Rotate key operations available in CCKM.
- **Reports:** All Reports overview page links to detailed individual reports.
- **Schedule:** In Schedule, you can schedule automatic key rotation, key rotation based on key expiration, and key synchronization.
- **Logs:** All Logs page lists individual logs that can be searched, sorted, and viewed.
- **Key Sources:** In Key Sources, you can create keys to upload to Azure. Currently DSM is the default key source where the keys are securely generated and stored. Also, supported in **Key Sources** is BYOK for all Azure Clouds (including Azure Stack AAD), and PFX file upload (all Azure Clouds including Azure Stack AAD and Azure Stack ADFS).

  Note: BYOK for Azure Stack ADFS is currently not supported.
- **Settings:** The **Settings Management** page allows you to configure the email settings, key alerts, remote syslog servers to which to send syslog messages, and proxy server settings. You can also view the release number of the CCKM you are running from this page. If you are using CCKM as a service principal, then you can change your tenant password or revoke your admin consent from this page.

To display the context of which Key Management service you are currently logged into within CCKM, the icon associated with the Key Management service displays at the top of the left-hand navigation bar.

The **arrow** button at the bottom of the navigation bar allows you to expand the bar to view the names of the menu options and collapse it to view only the icons associated with the menu options. Click the **arrow** button to either expand or collapse the navigation bar.

On the top-right side of the Azure CCKM portal, the following options are available:

- **Azure Directory Context Switcher:** This context switcher allows you to switch to another Azure Active Directory to which you belong. For more information about this context switcher, see section 10.7, *Azure Directory Context Switcher (Support for Guest User for AAD B2B Collaboration*).
  Note: This context switcher is not applicable to a CCKM that is used as a service principal.
- **Azure Cloud Service Context Switcher:** This context switcher allows you to login into another Azure cloud service different from the current Azure cloud service into which you are currently logged. If you select the Azure cloud in which you are currently logged, the **Multi-account Login/Logout** window for the Azure account(s) displays. For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service.*

- **AWS Login:** If you are not already logged into AWS, clicking on the **AWS Login** icon brings you to the **AWS Password Login** page where you can enter your AWS credentials to access the CCKM AWS cloud service (from the current cloud service in which you are currently logged). If you are already logged into AWS, clicking on the **AWS Login** icon brings you to the **Multi-account Login/Logout** window for the AWS account(s). For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service*.
- **Salesforce Cloud Service Context Switcher:** This context switcher allows you to login into another Salesforce cloud service different from the current Salesforce cloud service into which you are currently logged. This is either Salesforce or Salesforce Sandbox. If you select the Salesforce cloud in which you are currently logged, the **Multi-account Login/Logout** window for the Salesforce account(s) displays. For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service.*
- **Logout:** Logout from CCKM.

10.6 Home

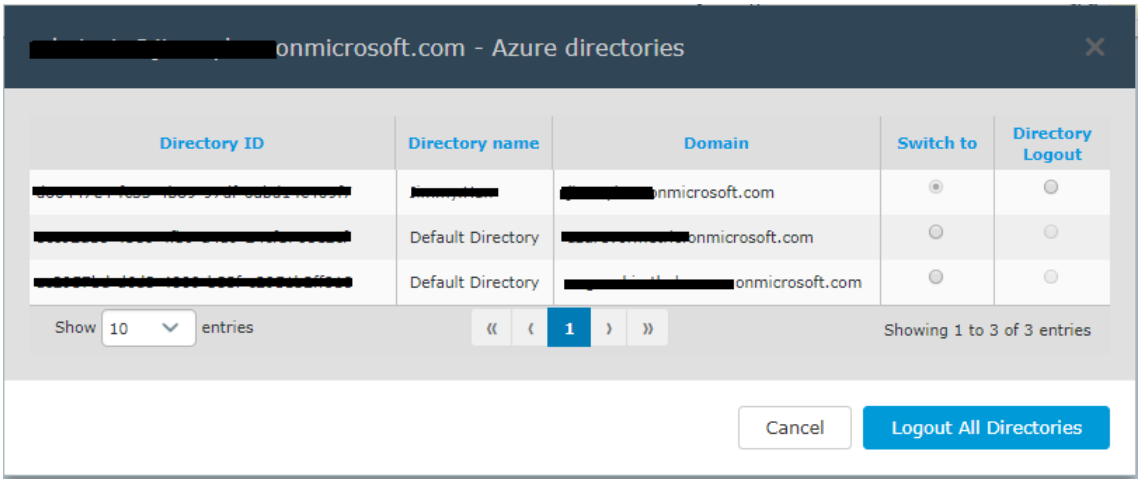
The Home page provides a dashboard to display key operation and count metrics.

10.7 Azure Directory Context Switcher (Support for Guest User for AAD B2B Collaboration)

CCKM supports the ability to allow Azure guest users to take part in B2B collaboration for Azure tenants to share user authentication and applications with partners. Prior to using this feature in CCKM, the guest user account must already be set up in the Azure Active Directory as a guest user with the appropriate access policies applied and the account activated in Azure. Once you are logged into CCKM Azure, from the **CCKM Directory Context Switcher** window, you have the ability to view all of the Azure directories to which you belong, including your default directory and your business partner's directory to which you are set up as a guest user. Using the **CCKM Directory Context Switcher** window, you can switch to any of the other directories to which you belong by logging into the directory from the same window. Once logged into a given directory, you can access and manage the resources and keys that reside in the given directory.

Note: The **Azure Directory Context Switcher** is not applicable to a CCKM that is used as a service principal.

The following **CCKM Directory Context Switcher** window displays the Azure directories in which a given user belongs:



You can perform the following actions using the **CCKM Directory Context Switcher** window:

- Logon to multiple directories for the Azure cloud service
- Logout of a specific directory or all directories
- Perform key management for multiple directories from your current CCKM browser window

10.7.1 Accessing the CCKM Directory Context Switcher Window

To access the **CCKM Directory Context Switcher** window after you are logged into CCKM Azure, click on the **CCKM Directory Context Switcher** icon ( ) at the top of right of the page. The **CCKM Directory Context Switcher** window displays. The following sections provide instructions on how to use this window.

10.7.2 Logging Out of a User Directory

To logout of a specific directory, select the **Directory Logout** button next to the domain name associated with the directory for which to log out. The **Log out CipherTrust Cloud Key Manager** window displays. Click **Continue with Log Out** to log out. You are logged out of the directory and then the CCKM login page is displayed.

10.7.3 Switching to Another User Directory

To access another directory (you have not already logged into), select the **Switch to** button next to the domain name associated with the directory for which to log on. The Microsoft login page displays. Enter your user credentials for the given directory in the login dialog boxes. Once you successfully log into Microsoft, the Azure CCKM portal is displayed.

After you have logged into a directory, you are not prompted to log in again to switch to that directory.

10.7.4 Logging Out of all User Directories

To logout of all directories for Azure, click **Logout of All Directories**. The **Log out CipherTrust Cloud Key Manager** window displays. Click **Continue with Log Out** to log out. You are logged out of all the directories and then the CCKM login screen is displayed.

10.8 Key Sources

The **Key Sources** page displays all currently posted keys in Azure. Two tabs are available at the top of the page:

- **DSM Key**: Use this tab to generate your keys from DSM. The generated keys are stored within the DSM.
- **nShield Key**: Use this tab to generate your keys from nShield Connect HSM. The generated keys are stored as blobs within the MongoDB.

The **Key Sources** page displays all currently posted keys in your key vault for Azure. Keys are listed by name, key type (the cloud service in which this key is used), the algorithm of the key, the creation date of the key, key description and actions you can take regarding that key. Note that the key algorithm currently supported for an Azure key in CCKM is RSA.

A **Search** box is available at the top left of the **Keys Sources** page allowing for a search of a specific key by entering any one of the following parameters:

- Name (key name)
- Key Type
- Algorithm
- Description
- Created By (a key created by a specific user)

10.8.1 Add (Generate) a Key

From the **Key Sources** page, you can add (or generate) keys in CCKM using either DSM or nShield Connect HSM as the key generator. These keys are of the RSA algorithm type. From the **Key Sources** page, the **DSM Key** tab also allows you also generate a .byok key from an external HSM (such as nShield Connect HSM) and import it into a Premium Azure Key Vault from CCKM. From **DSM Key** tab, you can also import an external password-protected PFX key into a Standard or Premium Azure Key Vault. Once imported from CCKM, the .byok and PFX key is stored in DSM securely as blobs.

10.8.1.1 Add (Generate) a DSM Key

1 In the CCKM portal, in the left navigation, select **Key Sources**.

2 Select the **DSM Key** tab.

3 Select **Add Key** > **Azure Key**.

4 In the **Add DSM Key** dialog box, select the name of the user of the key from the **User** drop-down menu. **Service** is fixed to Azure.

5 In the **Source Key Tier**, select one of the following:

- **DSM**—Generate a key from DSM and import it into a Standard or Premium Azure Key Vault.
- **HSM.byok**—Generate a .byok key from an external Thales HSM and import it into a Premium Azure Key Vault from CCKM. Once imported from CCKM, this .byok key is stored in the DSM securely as a blob. The .byok key is user and region specific. Be sure to select the correct username and region when creating the .byok key.
- **PFX**—Import an external password-protected PFX key into a Standard or Premium Azure Key Vault. Once imported from CCKM, this PFX key is stored in the DSM securely as a blob.

6 In the **Name** box, enter the name of the new key.

7 Enter a description of the key in the **Description** box.

8 From the **Algorithm** box, select the algorithm (**RSA 2048** or **RSA 4096**) to apply to the key.

9 Click **Save** to save the new key to CCKM. The new key is added to the key list.
(Reset clears the screen before it has been saved.)

10.8.1.2 Add (Generate) an nShield Key

In the CCKM portal, in the left navigation, select **Key Sources**.

1 Select the **nShield Key** tab.

2 Select **Add Key** > **Azure Key**.

3 In the **Add nShield Key** dialog box, select the name of the user of the key from the **User** drop-down menu. **Service** is fixed to Azure.

4 In the **Name** box, enter the name of the new key.

5 Enter a description of the key in the **Description** box.

6 From the **Algorithm** box, select the algorithm (**RSA 2048** or **RSA 4096**) to apply to the key.

7 Click **Save** to save the new key to CCKM. The new key is added to the key list.
(Reset clears the screen before it has been saved.)

10.8.2 Delete a Key

Deleting a key from the **Key Sources** page deletes the key from CCKM. If you delete a key from **Key Sources** that has already been uploaded to Azure, the key will not be deleted from Azure.

1 In the CCKM portal, in the left navigation, select **Key Sources**.

2 Select either the **DSM Key** or **nShield Key** tab depending on where the key you wish to delete resides.

3 Click **Delete** next to the target key.

4 The **Delete Key** dialog box with a Warning display asking you to confirm the key deletion by entering the supplied phrase.

5 Enter or Copy/paste the confirmation phrase.

6 Click **Delete** to confirm.

10.8.3 Manage Keys in the Key Sources List Page

In addition to adding and deleting keys, the **Key Sources** page allows you to sort the existing keys by Name, Key Type, Algorithm and Created Date.

10.9 Keys

The **Keys** page allows Cloud administrators to view all the keys and manage them within the boundary of granted permissions. Those keys or key vaults that do not have List permission for the respective Cloud administrator will not be listed.

A **Search** box is available at the top left of the **Keys** page allowing for a search of a specific key by entering any one of the following parameters:

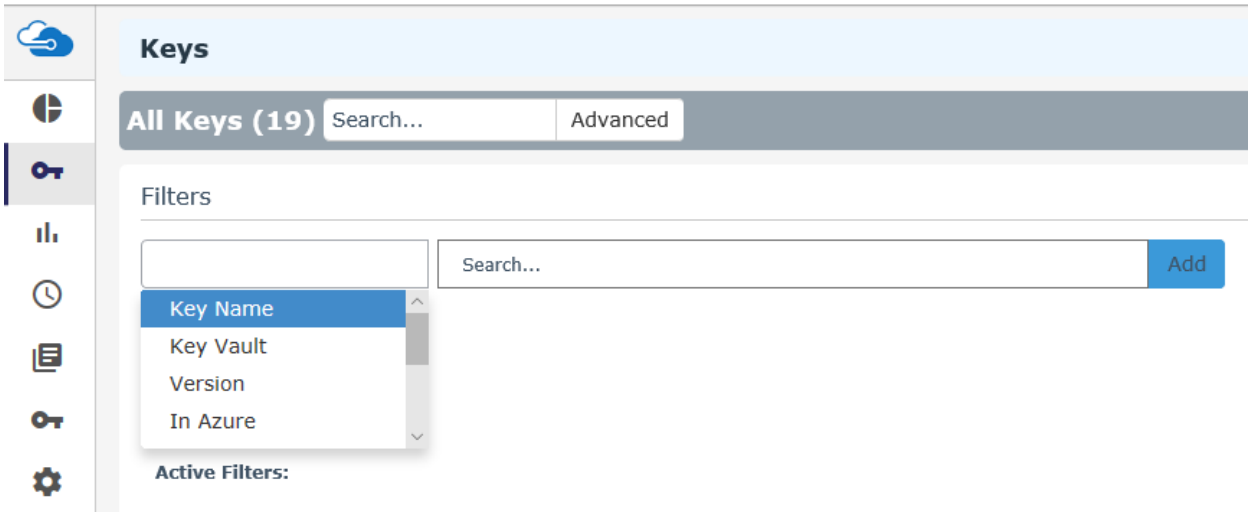
- Key ID
- Version Key ID
- Keyvault name
- Keyvault location
- Source key
- Subscription ID

The **Advanced** tab allows you to filter your search on any or a combination of these columns displayed on the **Keys** page with the exception of the **Count** and **Action** columns:

- Key Name
- Key Vault
- Version
- In Azure—Available options from the drop-down menu are In Azure, Soft Deleted, and Deleted.
- Backup— Available options from the drop-down menu are Backed Up and Not Backed Up.
- Enabled— Available options from the drop-down menu are Enabled and Disabled.
- Location
- Key Material Origin
- Auto Rotate— Available options from the drop-down menu are ON and OFF.

Each of these columns are described in more detail below in this section.

To select to filter based on a column or combination of these columns from the **Keys** page, click **Advanced**. The **Filters** section displays. From the column filter drop-down menu, select the name of the column on which to filter your search. For example, select **Key Name**. In the **Search** box, enter the value on which to search (for example, **MK-createkey** as the key name) and then click **Add**. For the **In Azure**, **Backup**, **Enabled** and **Auto Rotate** columns, a drop-down menu is available from which to select a value for the search. Click the down arrow, select a value from the available options, and then click **Add**. The specified search filter is then added to **Active Filters**. In the example, **Key Name: MK-createkey** is added to **Active Filters**. The results of the search displays in the **Keys** table. You can narrow your search by selecting more columns to add to the filter. To clear the filtered results, from **Active Filters**, click on the “X” (or the **Delete** button) for each of the named filters.



At the top right of the **Keys** page, the following buttons are available:

- **Export Keys**: Allows you to export the contents of the Keys list into a report in a CSV format.
- **New Key**: Allows you to upload an external BYOK key (DSM, nShield Connect, HSM .byok or PFX key) or create a native key. For information on how to upload an external BYOK key or create a native key, see section 10.7.1, Create a Key.
- **Synchronize**: Allows you to download all the keys that were created in Azure (to which you have access) into CCKM. For information on how to upload a key, see section 10.7.9, Synchronize Keys.

The following columns display on the **Keys** page:

- Key Name: The name of the key.
- Key Vault: The name of the Azure Key Vault. If the key vault is a premium key vault, then (PREMIUM) will be appended to the key vault name.
- Version: The version number of the key.
- Count: The version count.
- In Azure: Indicates the status of the key in Azure, which can be one of the following:
    - A green check icon indicates the key exists in Azure.
    - An orange exclamation icon indicates the key vault or key was deleted from Azure with the Soft Delete feature enabled.
    - A red exclamation icon indicates the key vault or key was deleted from Azure *without* the Soft Delete feature enabled.
- Backup: Indicates whether a backup of the key exists in CCKM. The status can be one of the following:
    - A green check icon indicates the backup exists in CCKM.
    - A red exclamation icon indicates the backup key does not exist in CCKM.
- Enabled: Indicates whether the key is enabled for use.
- Location: Indicates the Azure region in which the key is stored. For example, West US or Central West US.

- Key Material Origin:
    - If the key was uploaded from CCKM, then the value is **Internal(<source_key_name>)**.
    - If the key was not uploaded from CCKM as a BYOK key, then the value is **EXTERNAL**.
- Auto-Rotate: Indicates whether the key is set to be automatically rotated.
- Actions: Allows you to select any of the following actions on a given key:
    - Auto rotate
    - Delete
    - Add Version
    - Recover
    - Purge
    - Delete Backup
    - Restore

10.9.1 Create a Key

You have two options to create a new key:

- Upload an external key (BYOK)
- Create a native key

10.9.1.1 Upload your Own Key (BYOK)

Upload the RSA keys created from the **Key Sources** page of CCKM to the Azure Key Vault. These are your BYOK keys.

As part of upload process, Azure generates a corresponding backup key, which CCKM downloads and stores in DSM or MongoDB depending on your configuration of the key provider.

Note: You must have the import permissions for the Azure Key Vault to which you wish to upload the key.

To upload your BYOK key to CCKM:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 On the **All Keys** list page, select the **New Key** button and then **Upload**. The **upload a Key** dialog box displays.

3 From **Source Key Tier**, select one of the following:

- **DSM**—Generated a key from the DSM and imported it into a Standard or Premium Azure Key Vault.
- **HSM.byok**—Generated a .byok key from an external Thales HSM and imported it into a Premium Azure Key Vault from CCKM. Once imported from CCKM, this key is stored in the DSM securely as a blob.
- **PFX**—Imported an external password-protected PFX key into a Standard or Premium Azure Key Vault. Once imported from CCKM, this PFX key will be stored in the DSM securely as a blob.

- **nShield**—Generated a key from the nShield Connect and imported it into a Standard or Premium Azure Key Vault.

4 From **Source Key** box, select the source key.

5 From **Key Vault Tier** box**,** select the key vault tier of **Standard** or **Premium** depending on the subscription service type you have purchased.

6 From the **Key Vault Region** box, select the region where your key vault resides and whether you want to use the Azure Standard or Premium subscription service. Based on the Standard or Premium Key Vault Tier selection, the regions are filtered accordingly displaying only the regions, which are either Standard or Premium.

7 From the **Key Vault** box, select the name of the Azure Key Vault to where you intend to upload the key.

8 Add the name of the key in the **Key Name** box.

9 From the **Key Destination** box, select either **Software** or **HSM** (Premium subscription only) depending on which destination you want to use.

10 Click **Enabled**, if the key is enabled. Note that if the key is disabled, the key cannot be used for cryptographic operations.

11 Enter the activation date of the key in the **Activation Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

12 Enter the expiration date of the key in the **Expiration Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

13 From **Key Operations**, select the cryptographic operations the key will perform. The options are:

- Encrypt
- Decrypt
- Sign
- Verify
- Wrap Key
- UnwrapKey

14 (Optional) From **Tags**, enter the tag name and value of the key.

15 Click **Upload to Azure**.
The uploaded key is displayed in the list.

10.9.1.2 Create a Native Key

Create a native key in an Azure key vault and select the key types and sizes natively available in Azure. There is no source key tier as the Azure key vault will be the source of the key material. After CCKM creates a native key, it displays as **EXTERNAL** under the **Key Material Origin** column within the **All Keys** list page.

To create a native key:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 On the **All Keys** list page, select the **New Key** button and then **Create**. The **Create Key** dialog box displays.

3 From the **Key Vault Tier** box**,** select the key vault tier of **Standard** or **Premium** depending on the subscription service type you have purchased.

4 From the **Key Vault Region** box, select the region where your key vault resides. Based on the Standard or Premium Key Vault Tier selection, the regions are filtered accordingly displaying only the regions, which are either Standard or Premium.

5 From the **Key Vault** box, select the name of the Azure key vault to where you intend to create the key.

6 Add the name of the key in the **Key Name** box.

7 From the **Key Destination** box, select either **Software** or **HSM** (Premium subscription only) depending on which destination you want to use.

8 From **Key Type**, select **RSA** or **EC** for your key type.

9 If you selected **RSA** for key type, then select **2048**, **3072**, or **4096** from **Key Size** for the key size to apply to the new key. If you selected **EC** for key type, then select **P-256**, **P-384**, **P-521**, or **SECP256K1** from **Elliptical Curve Name** for the name of the elliptical curve to apply to the new version of the key.

10 Click **Enabled**, if the key will be enabled. Note that if the key is disabled, the key cannot be used for cryptographic operations.

11 Enter the activation date of the key in the **Activation Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

12 Enter the expiration date of the key in the **Expiration Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

13 From **Key Operations**, select the cryptographic operations the key will perform. The options are:

- Encrypt
- Decrypt
- Sign
- Verify
- Wrap Key
- UnwrapKey

14 (Optional) From **Tags**, enter the tag name and value of the key. Up to 15 tags are supported.

15 Click **Create**.
The new key is displayed in the list. As part of the create process, Azure generates a corresponding backup key, which CCKM downloads and stores in the DSM.

10.9.2 Delete a Key

From the **Keys** page, deleting a key deletes it from Azure Key Vault (but a backup key remains in the CCKM). After deletion, two new actions are displayed within the Action column: *Restore* and *Delete Backup*.

Note: This Delete action deletes ALL versions of the Azure key.

1 In the CCKM portal, in the left navigation, select **Keys**.

2 In the resulting *All Keys* list, find the name of the key and click **Delete** from the **Action** column. A warning message is displayed.

3 Enter or copy/paste the confirmation phrase.

4 Click **Delete**.
The Action column for key should now show *Restore* and *Delete Backup* options.

10.9.3 Restore a Key (to Azure)

In the event that someone deletes an Azure key directly in the Azure portal, it is possible to restore it from the CCKM as follows:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 From the **All Keys** list page, select **Restore** from the **Action** column of a deleted key.

3 The key and all its versions and metadata are restored to Azure Key Vault. The Success message is displayed, and the key is listed at the top of the **All Keys** page.

Note: If anyone deletes the backup key from CCKM, it will **not** be possible to restore the key to Azure.

10.9.4 Delete a "Soft-deleted" Key

From the **Keys** page, soft deleting a key deletes it from Azure Key Vault removing it from displaying in the CCKM UI. However, the key still exists in the Azure Key Vault and CCKM. After a soft delete, two new actions are displayed within the Action column: *Recover* and *Purge*.

To delete a key in a soft-delete state:

1 In the left navigation pane of the CCKM portal, select **Keys**.

2 From the **All Keys** list page, find the name of the key and click **Soft Delete** from the **Action** column.
A warning displays indicating that this action will delete the key from your Azure key vault and that the operation may take some time to complete. The message also indicates to type the confirmation phrase (I wish to delete the key) in the text box if you wish to delete the key.

3 Enter or copy/paste the confirmation phrase in the text box.

4 Click **Delete**.
The key and all its versions and metadata are deleted from the Azure Key Vault. The Success message is displayed, and the key is no longer listed within the **All Keys** page.

10.9.5 Recover a "Soft-delete" Key (to Azure)

From the **Keys** page, recovering a key that is in a "soft-delete" state allows you to restore the key to the Azure Key Vault. After the key is recovered, the key displays in the Azure Key Vault and is listed within the **All Keys** page.

To recover a key in a soft-delete state:

1 In the left navigation pane of the CCKM portal, select **Keys**.

2 From the **All Keys** list page, find the name of the key and select **Recover** from the **Action** column.
A warning displays indicating that this action will recover your key from your Azure key vault and that the operation may take some time to complete.

3 Click **Recover**.
The key and all its versions and metadata are restored to Azure Key Vault. The Success message is displayed, and the key is listed at the top of the **All Keys** page.

10.9.6 Purge a “Soft-deleted” Key

From the **Keys** page, purging a key that is in a “soft-delete” state allows you to delete it permanently from the Azure Key Vault. However, you still can restore this same key to the Azure Key Vault using the **Restore** action. See section 10.7.7, Restore a “Soft-delete” Key (to Azure), for more information.

To purge a key in a soft-delete state:

1 In the left navigation pane of the CCKM portal, select **Keys**.

2 From the **All Keys** list page, find the name of the key and click **Purge** from the **Action** column.
A warning displays indicating that this action will purge the key from the Azure key vault. The message also indicates to type the confirmation phrase (I wish to purge the Azure key) in the text box if you wish to purge the key.

3 Enter or copy/paste the confirmation phrase in the text box.

4 Click **Purge**.
The key and all its versions and metadata are deleted from the Azure Key Vault. The Success message is displayed, and the key is no longer listed within the **All Keys** page.

10.9.7 Restore a “Soft-delete” Key (to Azure)

From the **Keys** page, restoring a key that was previously set to a “soft-delete” state and then purged allows you to restore the key from the DSM and upload it to the Azure Key Vault.

To restore a key that was previously set to a “soft-delete” state and then purged:

1 In the left navigation pane of the CCKM portal, select **Keys**.

2 From the **All Keys** list page, find the name of the key and select **Restore Key** from the **Action** column.
A warning displays indicating that this action will restore your key to your Azure key vault and that the operation may take some time to complete.

3 Click **Restore**.
The key and all its versions and metadata are restored to Azure Key Vault from the DSM. The Success message is displayed, and the key is listed at the top of the **All Keys** page.

10.9.8 Rotate a Key

CCKM allows for a manual or autorotation of a key. You manually rotate a key by adding a new version from the **Add a version** dialog box. Auto rotating a key is adding a new version of the key at a scheduled time. Key autorotation requires that you first schedule the autorotation in the **Schedules** page prior to enabling this feature in the **Auto-rotate Key** dialog box. Note that the key you select to auto rotate must be enabled for it to be used for cryptographic operations after the rotation.

Note: Any key that is shared among different users within the *same* Azure cloud and is scheduled for a key rotation by one user cannot be scheduled for autorotation again by another user. Any key that is scheduled for autorotation displays the status of “AUTO” with an orange check icon next to it within the **Auto Rotate** column of the **Keys** page. If you attempt to schedule autorotation using **Rotate Key** within the **Keys** page with a key already scheduled, a message displays indicating that the key is already scheduled for rotation with the scheduled interval, date, and time at which the rotation is to take place. In addition, the name of the user responsible for scheduling the rotation displays.

10.9.8.1 Manually Rotate a Key (Add a New Version of a Key)

To rotate a key manually (add a new version of a key):

1 In the CCKM portal, in the left navigation, select **Keys**.

2 In the *All Keys* list, find the name of the key to which to add a new version.

3 Select **Add Version** from the **Actions** drop-down menu. The **Add a Version** dialog box is displayed.

4 From **Source Key Tier**, select one of the following as the source of your key to which to add a new version:

- **DSM**—Key was generated from DSM and imported into a Standard or Premium Azure Key Vault.
- **HSM.byok**—Your byok key was generated from an external Thales HSM and imported into a Premium Azure Key Vault from CCKM.
- **PFX**—Key was imported as an external password-protected PFX key into a Standard or Premium Azure Key Vault.
- **nShield**—Key was generated from nShield Connect and imported into a Standard or Premium Azure Key Vault.
- **Native**—Key is generated natively in the designated Azure Key Vault using the CCKM UI. Note that this option is equivalent to generating a key in the Azure portal.

5 From the **PFX-File Password** box, enter your password for your PFX file. This box only displays, if you selected PFX as your source key tier.

6 From **Source Key** box, select the source key to which to add a new version. Based on the selected source key tier and source key, the **Key Vault Tier**, **Key Vault Region**, and **Key Name** boxes are prepopulated with the values associated with your selections.

7 From the **Key Destination** box, select either **Software** or **HSM** (Premium subscription only) depending on which destination you want to use.

8 Click **Enabled**, if the key is enabled. Note that if the key is disabled, the key cannot be used for cryptographic operations. This option may already be selected depending on the selected source key.

9 From **Key Type**, select **RSA** or **EC** for your key type. **Key Type** only displays, if you selected **Native** as your source key tier. If you select **RSA**, then select **2048**, **3072**, or **4096** from **Key Size** for the key size to apply to the new version of the key. If you select **EC**, then select **P-256**, **P-384**, **P-521**, or **SECP256K1** from **Elliptical Curve Name** for the name of the elliptical curve to apply to the new version of the key.

10 Enter the activation date of the new version of the key in the **Activation Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

11 Enter the expiration date of the new version of the key in the **Expiration Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

12 From **Key Operations**, click the cryptographic operations the new version of the key will perform. The options are:

- Encrypt
- Decrypt
- Sign
- Verify
- Wrap Key
- UnwrapKey

13 (Optional) From **Tags**, enter the tag name and value of the new version of the key.

14 Click **Upload to Azure**.
The new version of the key is displayed in the list. As part of the process of adding a new key version of a key, Azure generates a corresponding backup key blob, which CCKM downloads and stores in DSM.

10.9.8.2 Enable Autorotation of Key

To enable autorotation of a key (add a new version of the key at a scheduled time):

1 In the CCKM portal, in the left navigation, select **Keys**.

2 In the All Keys list, find the name of the key for which to enable autorotation.

3 Select **Auto Rotate** from the **Actions** drop-down menu. The **Auto-rotate Key** dialog box is displayed. The **Key Algorithm** box is populated with the algorithm used in the selected key to auto rotate.

4 For **Enable Auto Rotation on this key**, select **On** to enable the autorotation of this key.
Note: If the toggle for **Enable Auto Rotation on key expiration** is displayed, skip it. This option displays only if the key is scheduled for autorotation before it expires. This option is not applicable to enabling autorotation of the key.

5 Select **Change key algorithm**, if you choose to use a different key algorithm than the one displayed in **Key Algorithm** for the new version of the key. If you choose to use a different key algorithm, **Source Key Tier**, **Key Destination**, and **Key Type** display for configuration.

6 From **Source Key Tier**, select one of the following as the new source key tier of the new version of the key:

- **DSM**—Key is to be generated from DSM and imported into a Standard or Premium Azure Key Vault.
- **nShield**—Key is to be generated from nShield Connect and imported into a Standard or Premium Azure Key Vault.
- **Native**—Key is to be generated natively in the designated Azure Key Vault using the CCKM UI. Note that this option is equivalent to generating a new key in the Azure portal.

7 From the **Key Destination** box, select either **Software** or **HSM** depending on whether you want to use the Azure Standard or Premium subscription service. If you are using the Standard subscription service for the selected key to be auto rotated, then Software is selected. If you are using the Premium subscription service for the selected key to be auto rotated, then HSM is selected. However, you have the option to select Software to use the Standard subscription service for the new version of the key.

8 From **Key Type**, select **RSA** or **EC** for the key type of the new version of the key.

- If your source key tier is set to **DSM**, then the key type of **RSA** along with the key sizes of **2048** and **4096** for **Key Size** are your options.
- If your source key tier is set to **nShield**, then the key type of **RSA** along with the key sizes of **2048** and **4096** for **Key Size** are your options.
- If your source key tier is set to **Native**, then the key types of **RSA** and **EC** are your options. For **RSA**, the key sizes **2048**, **3072**, and **4096** for **Key Size** are your options. For **EC**, the name of the elliptical curves **P-256**, **P-384**, **P-521**, and **SECP256K1** for **Elliptical Curve Name** are your options.

9 Click **Submit**.

10.9.9 Synchronize Keys

Synchronizing allows you to download all the keys that were created in Azure (to which you have access) into CCKM. You have the option to schedule key synchronization. Perform this scheduling in the **Schedules** page.

The story below illustrates a use of synchronization and restoration:

> Sally is the Azure Cloud administrator with access to many vaults. She has synchronized keys in CCKM. Now, innocent, ignorant rogue Charlie logs into Azure portal and deletes all the keys in his vault, which also belonged to Sally. Now, if Sally synchronizes keys again from CCKM portal, she will see those keys as deleted in CCKM portal. However, she will also see options to restore the deleted keys back into the relevant Azure Key Vault. This is because CCKM also maintains Azure backup keys as part of basic key management.

To synchronize your keys:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 Click **Synchronize**. A warning box is displayed; click **Synchronize** again to proceed with the synchronization. New keys that are created in Azure are retrieved and displayed in the CCKM UI.

10.9.9.1 Enable Autorotation before Key Expiration

To enable autorotation of a key before it expires, you must first set up a key expiration schedule in the **Schedules** page. Once the key is enabled for autorotation, the key will be rotated before it expires. After the key is auto rotated, the key will have a new expiration date based on the key-expiration schedule.

To enable autorotation of a key before it expires:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 In the All Keys list, find the name of the key for which schedule a key rotation.

3 Select **Auto Rotate** from the **Actions** drop-down menu. The **Auto-rotate Key** dialog box is displayed. The **Key Algorithm** box is populated with the algorithm used in the selected key to auto rotate.

4 For **Enable Auto Rotation on key expiration**, select **On** to enable the autorotation of this key before it expires. **Note**: Skip the **Enable Auto Rotation on this key** as it is not applicable to enabling autorotation of this key before it expires.

5 Select **Change key algorithm**, if you choose to use a different key algorithm than the one displayed in **Key Algorithm** for the new version of the key. If you choose to use a different key algorithm, **Source Key Tier**, **Key Destination**, and **Key Type** display for configuration.

6 From **Source Key Tier**, select one of the following as the new source key tier of the new version of the key:

- **DSM**—Key is to be generated from DSM and imported into a Standard or Premium Azure Key Vault.
- **nShield**—Key is to be generated from nShield Connect and imported into a Standard or Premium Azure Key Vault.
- **Native**—Key is to be generated natively in the designated Azure Key Vault using the CCKM UI. Note that this option is equivalent to generating a new key in the Azure portal.

7 From the **Key Destination** box, select either **Software** or **HSM** depending on whether you want to use the Azure Standard or Premium subscription service. If you are using the Standard subscription service for the selected key to be auto rotated, then Software is selected. If you are using the Premium subscription service for the selected key

to be auto rotated, then HSM is selected. However, you have the option to select Software to use the Standard subscription service for the new version of the key.

8 From **Key Type**, select **RSA** or **EC** for the key type of the new version of the key.

- If your source key tier is set to **DSM**, then the key type of **RSA** along with the key sizes of **2048** and **4096** for **Key Size** are your options.
- If your source key tier is set to **nShield**, then the key type of **RSA** along with the key sizes of **2048** and **4096** for **Key Size** are your options.
- If your source key tier is set to **Native**, then the key types of **RSA** and **EC** are your options. For **RSA**, the key sizes **2048**, **3072**, and **4096** for **Key Size** are your options. For **EC**, the name of the elliptical curves **P-256**, **P-384**, **P-521**, and **SECP256K1** for **Elliptical Curve Name** are your options.

9 Click **Submit**.

10.9.10 Export Keys

Export Keys allows you to export the contents of the Keys list into a report in a CSV format.

To export your Keys list:

1 In the left navigation bar, click **Keys**.

2 Click **Export Keys**.
This will download all of the keys' versions in a report in a CSV format.

10.9.11 Backup Key

When a key is uploaded to Azure, Azure creates a backup key, which CCKM automatically downloads and stores in DSM (if you have DSM configured as your key provider) or in MongoDB (if you have nShield Connect configured as your key provider). If you have both DSM and nShield Connect configured as your key provider, then the backup key is stored only in DSM. In addition, when Azure keys are synchronized, the backup key of any key that was created directly in Azure Portal is downloaded. Backup keys are encrypted blobs and stored securely in DSM or MongoDB (depending on your configuration of the key provider). A backup key contains information about the Azure key and all its versions. This key can be used to restore a key that has been deleted.

10.9.11.1 Delete a Backup Key

Delete a backup key when you definitely do not want the key and have no corresponding data that will ever need decryption.

For a key that has been deleted:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 From the **All Keys** list page, select **Delete Backup** from the action column of a deleted key.

3 The **Delete Key** warning page is displayed. Type or Copy/paste the confirmation phrase (I wish to delete the Azure backup key).

4 Click **Delete**.
The key is removed from the list.

10.10 Schedules

The **Schedules** page allows Cloud administrators to view and schedule the following job types:

- Key Rotation
- Key Synchronization
- Key Expiration

At the top right of the **Schedules** page, the **Add Schedule** button is available from which to add a scheduled job type.

Note: If you are using CCKM as a service principal, you are not required to enter a username and password to schedule key rotation, key synchronization, or key expiration for Azure Key Management. The username and password boxes do not display in the dialog box for each of these job types.

10.10.1 Schedule Key Rotation

CCKM allows for a scheduled rotation (autorotation) of an Azure key. A new version of the key is added to the Azure key during a scheduled rotation. Scheduled rotation requires that this feature be configured in the **Schedules** page prior to enabling the feature in the **Auto Rotate Key** dialog box (from the **Actions** column of the *All Keys* list). Your Azure login password is required as part of the configuration and is saved to DSM (if you have DSM configured as your key provider) or to MongoDB (if you have nShield Connect configured as your key provider). If you have both DSM and nShield configured as your key provider, then the Azure login password that you enter is saved only to DSM. During a scheduled rotation of a key, CCKM uses the associated username and password to rotate the key. If the password expired or changed, then the new password must be reentered. Otherwise, the scheduled rotation will fail.

You can pause or delete a scheduled key rotation in the **Schedules** page. Note that when a key rotation schedule is deleted from CCKM, then the Azure user password is also deleted from DSM or MongoDB (depending on your configuration of the key provider).

10.10.1.1 Set up a Key Rotation Schedule

To set up a key rotation schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

Note: If you are using CCKM as a service principal, the **Username** box does not display as you are not required to enter a username when using CCKM as a service principal

3 From the **Job Type** drop-down menu, select **KEY_ROTATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 For **Schedule**, set a schedule for the rotation by selecting **Basic** (to enter a date and interval for the schedule) or **Advanced** (to enter a Cron Expression for the schedule). If you selected **Basic**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

Set the repeat interval of the key rotation by entering a number in **Repeat Interval** and then selecting Day, Week, or Month from the drop-down menu. The supported ranges are 1 through 30 days, 1 through 4 weeks, and 1

through 12 months. You can increase or decrease the number you enter. If you selected **Advanced**, enter the cron expression in the **Cron Expression** box.

7 Enter your Azure user password in the **User Password** box.

Note: If you are using CCKM as a service principal, the **User Password** box does not display as you are not required to enter a password when using CCKM as a service principal.

8 Click **Save** to save the setting.

10.10.1.2 Pause a Key Rotation Schedule

To pause a key rotation schedule, select **Yes** for **Pause** within the **Add Schedule** dialog box and then click **Save**. The setting is saved. To resume the rotation schedule after pausing it, select **No** for **Pause** and then click **Save.** The setting is saved.

10.10.1.3 Delete a Key Rotation Schedule

To delete a key rotation schedule, select **Delete** from the **Actions** drop-down menu from the **Schedules** page. Note that Azure user password associated with the given key rotation schedule is also deleted from either DSM or MongoDB (depending on your configuration of the key provider).

10.10.2 Schedule Key Synchronization

CCKM allows for a scheduled synchronization of Azure keys. Synchronizing downloads any keys that were created in Azure, outside of the CCKM portal, into your CCKM. Scheduled key synchronization requires that this feature be configured in the **Schedules** page. You can also pause or delete a scheduled key synchronization in the **Schedules** page. Note that when a key synchronization schedule is deleted from CCKM, then the Azure user password is also deleted from DSM (if you have DSM configured as your key provider) or from MongoDB (if you have nShield Connect configured as your key provider). If you have both DSM and nShield configured as your key provider, then the Azure user password is deleted from DSM.

10.10.2.1 Set up a Key Synchronization Schedule

To set up a key synchronization schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

Note: If you are using CCKM as a service principal, the **Username** box does not display as you are not required to enter a username when using CCKM as a service principal.

3 From the **Job Type** drop-down menu, select **KEY_SYNCHRONIZATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 For **Schedule**, set a schedule for the key synchronization by selecting **Basic** (to enter a date and interval for the schedule) or **Advanced** (to enter a Cron Expression for the schedule). If you selected **Basic**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

Set the repeat interval of the key synchronization by entering a number in **Repeat Interval** and then selecting Day, Week, or Month from the drop-down menu. The supported ranges are 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter. If you selected **Advanced**, enter the cron expression in the **Cron Expression** box.

7 Enter your Azure user password in the **User Password** box.

Note: If you are using CCKM as a service principal, the **User Password** box does not display as you are not required to enter a password when using CCKM as a service principal.

8 Click **Save** to save the setting.

10.10.3 Schedule Key Expiration

CCKM allows for a scheduled key expiration of Azure keys in the **Schedules** page. You can also pause or delete a scheduled key expiration in the **Schedules** page. Note that when a key expiration schedule is deleted from CCKM, then the Azure user password is also deleted from DSM (if you have DSM configured as your key provider) or from MongoDB (if you have nShield Connect configured as your key provider). If you have both DSM and nShield configured as your key provider, then the Azure user password is deleted from DSM.

10.10.3.1 Set up a Key Expiration Schedule

To set up a key expiration schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.
Note: If you are using CCKM as a service principal, the **Username** box does not display as you are not required to enter a username when using CCKM as a service principal.

3 From the **Job Type** drop-down menu, select **KEY_EXPIRATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 From **Start Date**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

7 From **Expire a new key after**, enter a number and select the interval unit of Hour, Day, Week, Month, or Year from the drop-down menu after which the new key expires. The supported ranges are 6 through 24 hours, 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter.

8 Enter your Azure user password in the **User Password** box.

Note: If you are using CCKM as a service principal, the **User Password** box does not display as you are not required to enter a password when using CCKM as a service principal.

9 Click **Save** to save the setting.

10.11 Reports

CCKM includes the following reports:

- **Combined Key Activity Reconciliation Report**
  Provides the history of the key in a selected period. Click **Run Report** to generate a new list of events connected to a key.
- **Azure Key Activity Report**
  Illustrates the activity type performed by a specific user and at the stated time.
- **Azure Key Service Usage Report**
  Indicates the relationship between the application and the key, including:
    - Key name
    - Key Description
    - Requesting Service: Name or ID of the service using a key
    - Service Platform
    - Request Timestamp: Time when the application used the key.
- **Azure Key Aging Report**
  Provides information about key expiration using the following columns:
    - Key Name
    - Key description
    - Key destination: Indicates services, such as Azure
    - Key owner: Person who created the key. In Azure, there is no owner associated with the key.
    - Key expiration: Date and time at which this key expires.
- **Cloud Key Manager User Action Report**
  Lists the users and their actions in creating and deleting keys in the CCKM portal.

10.12 Logs

CCKM maintains logs for events, such as login, key generation, synchronizing of keys, and deletion of keys, and displays these logs in the **All Logs** page. For each logged event listed on the **All Logs** page, the following information is displayed:

- Event name
- Severity level
- Date and time of the event
- Event message
- User (name) associated with the event

You can search for a specific logged event from the **All Logs** page using the **Search** box. Allowable filters on the search are event name, severity level, event message, and user (name).

10.13 Settings

To access the **Settings** page in CCKM, select **Settings** from the left-hand navigation. From the **Settings** page, you can:

- Enter the emails of the recipients who are to receive alerts and reminders.
- Set up key alerts.
- Configure remote syslog servers to which to send syslog messages.
- (Applicable to CCKM when used as a service principal) Change your tenant password or revoke your admin consent.
- View the release number of the CCKM you are running.

Note that for Azure, each of the supported clouds (Azure, Azure China, and Azure Germany) must be set up independently of each other, as these are separate and distinct key management services within Azure.

10.13.1 Adding Recipients' Emails (for Alerts and Reminders)

Use the **General** settings tab to enter the designated recipients' emails for email alerts and reminders to be sent from the system. Enter the recipients' email addresses separated by a comma in the **Alert Format** box.

10.13.2 Setting Keys Alerts

Use the **Keys Alert** tab from the **Settings** page to enable whether to send key alerts to specified recipients. When a key is deleted, restored, or uploaded in Azure, a key alert can be sent. You can also to set up a reminder to manually rotate (or rekey) a key along with the frequency (in days) at which the reminder is sent. The recipients' emails are set up in the **General** tab. By default, no alerts or reminders are sent.

To set up key alerts or a reminder to manually rotate (or rekey) a key:

1 In the **Settings** page, select the Keys Alert tab.

2 Under **Alerts**, click On for each of the alert types and reminder you wish to enable:

- Rekey Alert (reminder)
- Delete Azure Key
- Restore Azure Key
- Upload Azure Key

3 In the **Days** box, enter the number of days for the frequency an email reminder to rotate a key is sent.

4 Click **Save** to save the setting.

10.13.3 Configuring Syslog Server

Use the **Syslog** tab from the **Settings** page to configure a remote syslog server to which to send syslog messages. Note that the default port number of 514 and the facility name of “LOCAL1” for the syslog server display in the **Port** and **Facility** boxes, respectively. These boxes are not available for modification.

10.13.3.1 Configure Remote Syslog Server

To configure a remote syslog server:

1 In the **Settings** page, select the **Syslog** tab. The **Syslog** dialog box displays.

2 Enter the hostname or IP address of the syslog server in the **Hostname** box.

3 (Optional) Enter a description of the syslog server in the **Description** box.

4 Click **Save** to save the setting.

10.13.3.2 Delete Remote Syslog Server Configuration

To delete the configuration of a remote syslog server, click **Delete** from the **Syslog** dialog box. No syslog messages are sent to the remote syslog server after the deletion of this configuration.

10.13.4 Configuring Tenant Password and Revoking Admin Consent (for Service Principal)

Note: The **Tenant** tab is available in the **Settings** page only if you have configured to use CCKM as a service principal.

The **Tenant** tab in the **Settings** page allows you to change your tenant password that is associated with the use of CCKM as a service principal.

The **Tenant** tab also allows you to revoke the admin consent, which you (as the Admin user) had provided during the initial login into CCKM for Azure Key Management (using CCKM as a service principal). See section 10.3.2, Accessing CCKM using Service Principal, for more information about providing your admin consent. If you revoke the admin consent and then logout of CCKM, you will receive an error message when you attempt to log back into CCKM. The message indicates that the login failed due to having entered an incorrect tenant name *or* the given tenant not having granted permissions to CCKM (to be used as a service principal). You must provide the admin consent again to login into CCKM. Click on the "admin consent" link (indicated by “here” in blue text at the top CCKM login dialog box) after which the CCKM Admin Consent dialog box is displayed.

10.13.4.1 Change Tenant Password

Note: The **Tenant** tab is available in the **Settings** page only if you have configured to use CCKM as a service principal.

To change a tenant password that is associated with CCKM used as a service principal:

1 In the **Settings** page, select the **Tenant** tab. The **Tenant** dialog box displays.

2 Enter the new tenant password in the **New Tenant Password** box.

3 Reenter the new tenant password in the **Confirm Tenant Password** box.

4 Click **Save** to save the setting.

10.13.4.2 Revoke Admin Consent

Note: The **Tenant** tab is available in the **Settings** page only if you have configured to use CCKM as a service principal.

To revoke your admin consent that is associated with CCKM used as a service principal:

1 In the **Settings** page, select the **Tenant** tab. The **Tenant** dialog box displays.

2 Deselect the **Granted Permission to this CCKM** box to revoke your admin consent. By default, this box is checked.

3 Click **Save** to save the setting.

10.13.5 About

Use the **About** tab to view the release number of the CCKM you are running.

11CCKM for Salesforce

This chapter is for Salesforce and Salesforce Sandbox users who access the CCKM.

CCKM provides support for Salesforce and Salesforce Sandbox. The Salesforce Sandbox is a copy of your production Salesforce environment used primarily for testing, development, or training purposes. **Note that in this chapter, the information regarding Salesforce is applicable to both Salesforce and Salesforce Sandbox unless otherwise noted.** The CCKM UI for Salesforce and Salesforce Sandbox is identical with the exception of the display of the names of the cloud services. Your login credentials for Salesforce and Salesforce Sandbox are unique to each cloud service. If you are configuring for Salesforce Sandbox, be sure to configure within the Salesforce Sandbox cloud service.

11.1 Prerequisites on Salesforce

To use CCKM for Salesforce, you must:

- Have Salesforce's Shield Platform Encryption Service enabled
- Have at least one valid certificate in Salesforce with the following settings:
    - **Key Size:** 4096
    - **"Use Platform Encryption"** selected
    - **"Exportable Private Key"** deselected

- **Be familiar with different types of tenant secrets, Salesforce 'Bring Your Own Key' options, and certificates** in Salesforce.
- **Have access to a Salesforce account with "Manage Encryption Keys" permission enabled**, in order to handle tenant secrets. If you are using Salesforce cache-only keys, ensure your account also has "**Customize Application" permission** enabled. Also ensure the "**Allow Cache-Only Keys with BYOK"** option within the **Security > Platform Encryption > Advanced Settings** page in Saleforce is also enabled. (See section 11.1.1, Enable "Manage Encryption Keys" Permission in Salesforce, for more information.)

11.1.1 Enable the permissions of "Manage Encryption Keys" and "Customize Application" (optional) in Salesforce

You must have the "**Manage Encryption Keys"** permission enabled to perform operations related to tenant secrets. If you are using Salesforce cache-only keys, then you must also have the "**Customize Application"** permission enabled to perform operations related to cache-only keys.

1 Log in with the administrator privileges to your Salesforce account.

2 Click **Setup**.

3 In the left navigation, expand **Manage Users**.

4 Click **Permission Sets**.
Be sure that *Session Activation Required* is not selected.

5 Click **New**, and then select **New Permission Set**.

6 Click **System Permissions**, then click **Edit**.

7 Check **Manage Encryption Keys**. If you are using Saleforce cache-only keys, then also check **Customize Application**. Click **Save**.

8 Select **Manage Assignments > Add Assignments** and select the user account(s) which you want to enable to import and export keys.

For more information about managing permissions on encryption keys in Salesforce, refer to the "Permission Sets" documentation within the Salesforce Help documentation.

11.2 Salesforce Cache-only Keys

CCKM provides support for Salesforce's cache-only key service as part of its extended BYOK capability. With the use of a cache-only key, Salesforce calls the CCKM REST APIs to fetch a tenant secret (or cache-only key) from CCKM. This tenant secret is used for encrypt and decrypt operations in Salesforce. With the initial versions of Salesforce BYOK capability, the upload of the tenant secret and other key operations in Salesforce are initiated from CCKM using the available Salesforce Rest APIs.

In the API call from Salesforce to CCKM to fetch a tenant secret for the use of Salesforce's cache-only key service, a named credential is employed. The named credential is a combination of a URL of a CCKM callout endpoint along with the configured authentication parameters (CCKM username and password) associated with the URL. When the CCKM receives the API call and successfully authenticates it, CCKM passes the tenant secret to Salesforce through a secure and authenticated connection. For more information about Salesforce cache-only keys, refer to the Salesforce Help documentation.

Note: In this documentation and in the context of support for Salesforce's cache-only key service, the term "tenant secret" is used interchangeably with "cache-only key".

11.2.1 Configuring Cache-only Keys

The following is a summary of the steps to take to configure the use of cache-only keys within CCKM *and* Salesforce:

1 From the CCKM **Settings** page, configure the username and password to associate with the URL of the CCKM Rest API endpoint (or callout endpoint). See section 11.10.3.3, Configure Cache-only Key Endpoint, for more information.

2 Set up a named credential in Saleforce for use with the cache-only keys. See section 11.2.2, Set up Named Credential in Salesforce, for more information.

3 Generate a key in DSM for use with the cache-only keys. See section 11.4.1.1, Add (Generate) a DSM Key, for more information.

4 Configure a cache-only key in CCKM. See section 11.5.1.3, Create a Cache-only Key, for more information.

11.2.2 Set up Named Credential in Salesforce

Before setting up a named credential in Salesforce, ensure you have a CCKM deployed on a publicly accessible URL, and you have configured the username and password to be associated with the URL of the CCKM Rest API endpoint (or callout endpoint) in the **Endpoint for Cache-only Key** tab from the CCKM **Settings** page. This information is used to set up the named credential in Salesforce, which in turn provides Salesforce a method for fetching the cache-only key from CCKM through a secure and authenticated connection.

For detailed information about how to set up a named credential in Salesforce, refer to the Salesforce Help documentation.

To set up a named credential in Salesforce, do the following:

1 Login into Salesforce.

2 Go to the **Setup** page.

3 In **Quick find** box, enter “Named Credentials” to search on this topic and then select **Named Credentials** in the results. The **Named Credentials Setup** page displays.

4 Select New Named Credentials.

5 In the **Label** box, enter the name of the named credential to use for CCKM. It is recommended you include “CCKM” in the name to reflect CCKM-associated information in the labels within Salesforce list views and reports. The name you provide in the **Label** box is automatically populated in the **Name** box.

6 In the **URL** box, enter the URL of the CCKM Rest API endpoint. To obtain the information to enter in this box, go to the **Endpoint for Cache-only Key** tab of the CCKM **Settings** page, and copy the URL from the **Call out URL** box and paste it into this box. Replace “<cckm_hostname>” in the URL with the name of the CCKM hostname you are using.

7 Under **Authentication**, do the following:

8 Leave the **Certificate** box blank.

9 In the **Identity Type** drop-down menu, select **Named Principal**.

Note: Named Principal is currently the only option supported for the Salesforce identify type within CCKM.

10 In the **Authentication protocol** drop-down menu, select **Password Authentication**.

Note: Password Authentication is currently the only option supported for authentication protocol within CCKM.

11 In the **Username** box, enter the username to associate with the URL of the CCKM Rest API endpoint. This username must match the username configured in the **Endpoint for Cache-only Key** tab of the CCKM **Settings** page.

12 In the **Password** box, enter the password to associate with the URL of the CCKM Rest API endpoint. This password must match the password configured in the **Endpoint for Cache-only Key** tab of the CCKM **Settings** page.

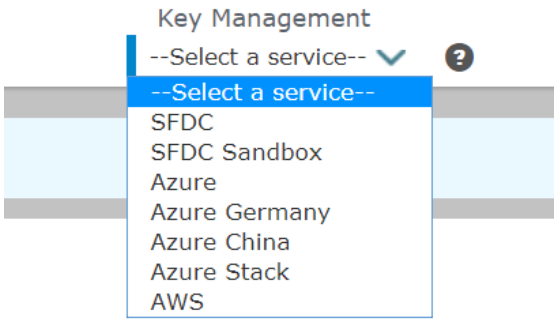
13 Under the **Callout Options**, ensure the **Generate Authorization Header** checkbox is selected. This checkbox is selected by default.

14 Click **Save**.

11.3 Access CCKM for Salesforce

To access the CCKM service for Salesforce or Salesforce Sandbox Key Management from CCKM:

1 In the CCKM Admin portal, select **Key Management** > **SFDC** (Salesforce Dot Com) | **SFDC Sandbox**:



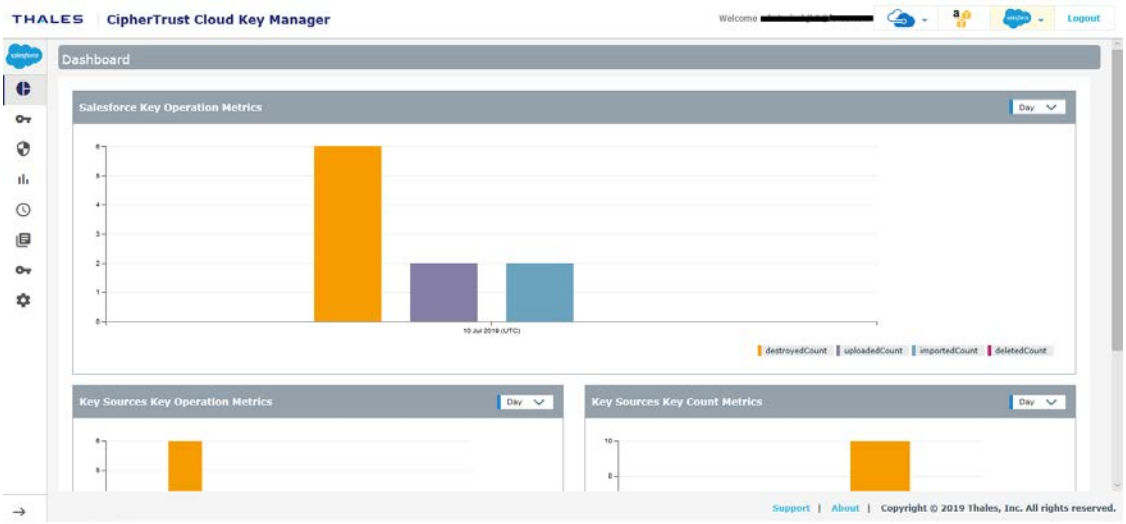
The Salesforce login page opens at *https://login.salesforce.com/*.
The Salesforce Sandbox login page opens at *https://test.salesforce.com/*.

2 Enter your Salesforce credentials or click on the saved user name.
The Salesforce Login page is displayed. Enter your Salesforce password and click **Log In**.

OR

1 Access the CCKM login page (see section 9.1.1, Accessing CCKM as an Administrator).

2 Use your Salesforce credentials to log in. The first time you access the CCKM, the tenant secrets and certificates are synchronized and a progress page is displayed. Once synchronization is complete, click **Continue**. The CCKM Home page is displayed.



The CCKM portal for Salesforce contains the following menu options in the left-hand navigation bar:

- **Home:** Home page displays key operation and count metrics
- **Tenant Secrets:** Salesforce terminology for “keys”. The All Tenant Secrets page displays key management overview and provides the Upload and Synchronize functionality, and the Create, Destroy, Import, Delete Backup key operations available in CCKM.
- **Certificates:** The All Certificates page displays certificate management overview, plus Synchronize functionality.
- **Reports:** All Reports overview page links to detailed individual reports.
- **Schedule:** In Schedule, you can schedule automatic key rotation, key rotation based on key expiration, and key synchronization.

- **Logs:** All Logs page lists individual logs that can be searched, sorted, and viewed.
- **Key Sources:** In **Key Sources**, you can create tenant secrets to upload to Salesforce. Currently DSM is the default key source where the keys are securely generated and stored.
- **Settings:** Settings Management page allows you to configure alert email settings, tenant secret alerts, remote syslog server to which to send syslog messages, and proxy server settings. You can also view the release number of the CCKM you are running from this page.

To display the context of which Key Management service you are currently logged into within CCKM, the icon associated with the Key Management service displays at the top of the left-hand navigation bar.

The **arrow** button at the bottom of the navigation bar allows you to expand the bar to view the names of the menu options and collapse it to view only the icons associated with the menu options. Click the **arrow** button to either expand or collapse the navigation bar.

On the top-right side of the CCKM portal for Salesforce, the following options are available:

- **Azure Cloud Service Context Switcher:** This context switcher allows you to login into another Azure cloud service different from the current Azure cloud service into which you are currently logged. If you select the Azure cloud in which you are currently logged, the **Multi-account Login/Logout** window for the Azure account(s) displays. For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service.*
- **AWS Login:** If you are not already logged into AWS, clicking on the **AWS Login** icon brings you to the **AWS Password Login** page where you can enter your AWS credentials to access the CCKM AWS cloud service (from the current cloud service in which you are currently logged). If you are already logged into AWS, clicking on the **AWS Login** icon brings you to the **Multi-account Login/Logout** window for the AWS account(s). For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service*.
- **Salesforce Cloud Service Context Switcher:** This context switcher allows you to login into another Salesforce cloud service different from the current Salesforce cloud service into which you are currently logged. This is either Salesforce or Salesforce Sandbox. If you select the Salesforce cloud in which you are currently logged, the **Multi-account Login/Logout** window for the Salesforce account(s) displays. For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service.*
- **Logout:** Logout from CCKM.

11.4 Key Sources

The **Key Sources** page displays all currently posted tenant secrets or keys in Salesforce. Two tabs are available at the top of the page:

- **DSM Key**: Use this tab to create your keys (including cache-only keys) from DSM.
- **nShield Key**: This tab is currently unavailable for use, as CCKM does not currently support generating nShield Connect keys for uploading to Salesforce.

The **Key Sources** page displays all currently posted tenant secrets in your key vault for Salesforce.
Keys are listed by name, key type (the cloud service in which this key is used), the algorithm of the key, the creation date of the key, key description, and actions you can take regarding that key. Note that the key algorithm currently supported for a Salesforce key in CCKM is AES.

A **Search** box is available at the top left of the **Keys Sources** page allowing for a search of a specific key by entering any one of the following parameters:

- Name (key name)
- Key Type
- Algorithm
- Description
- Created By (a key created by a specific user)

Note: The Salesforce term “tenant secrets” is used interchangeably with “keys” in CCKM. Manage tenant secrets from two places in the CCKM portal: the **Tenant Secrets** page and the **Key Sources** page.

11.4.1 Add (Generate) a Key

From the **Key Sources** page, you can add (or generate) a key in CCKM using a DSM as the key generator including a key to be used as a cache-only key. These keys are of the AES algorithm type.

Note: The **nShield Key** tab is currently unavailable for use, as CCKM does not currently support generating nShield Connect keys for uploading to Salesforce.

11.4.1.1 Add (Generate) a DSM Key

1 In the CCKM portal, in the left navigation, select **Key Sources**.

2 Select the **DSM Key** tab.

3 Select **Add Key** > **Salesforce Key**.

4 In the **Add DSM Key** dialog box, select the name of the user of the key from the **User** drop-down menu. **Service** is fixed to Salesforce Key and (key) **Algorithm** is fixed to AES256.

5 In the **Name** box, enter the name of the new key.

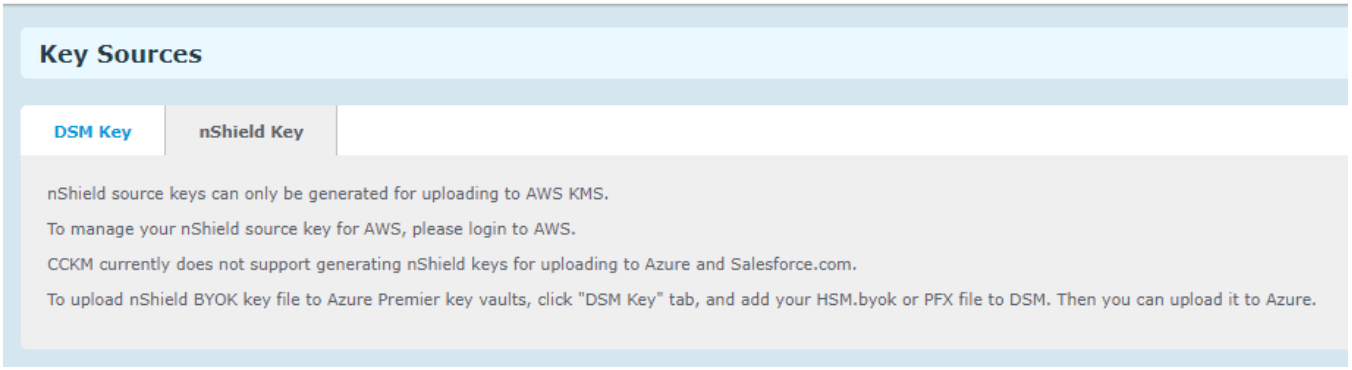
6 Enter a description of the key in the **Description** box.

7 From the **Algorithm** box, select the algorithm to apply to the key.

8 Click **Save** to save the new key to CCKM. The new key is added to the key list.
(Reset clears the screen before it is saved.)

11.4.1.2 Add (Generate) an nShield Key

The **nShield Key** tab is currently unavailable for use, as CCKM does not currently support generating nShield Connect keys for uploading to Salesforce.com. The following message displays in this tab:



11.4.2 Delete a Key

Deleting a key from the **Key Sources** page deletes the key from CCKM. If you delete a key from the **Key Sources** page that has already been uploaded to Salesforce, the key will **not** be deleted from Salesforce.

1 In the CCKM portal, in the left navigation, select **Key Sources**.

2 Click **Delete** next to the target key. The **Delete Key** dialog box with a Warning display asking you to confirm the key deletion by entering the supplied phrase.

3 Enter or copy/paste the confirmation phrase.

4 Click **Delete** to confirm.

11.4.3 Manage Keys in the Key Sources List Page

In addition to adding and deleting keys, the **Key Sources** page allows you to sort the existing keys by Name, Key Type, Algorithm and Created Date.

11.5 Tenant Secrets

The **Tenant Secrets** page allows Cloud administrators to view all the keys and manage them within the boundary of granted permissions. Those keys or key vaults that do not have List permission for the respective Cloud administrator will not be listed.

Within the **Tenant Secrets** page, the following types of Salesforce tenant secrets are available to specify the kind of data to encrypt using a tenant secret:

- Data in Salesforce: Encrypts data using the probabilistic encryption scheme.
- Data in Salesforce (Deterministic): Encrypts data using the deterministic encryption scheme.
- Search Index: Encrypts search index files.
- Analytics: Encrypts Einstein Analytics data.
- Event Bus (Developer Preview): Encrypts data changes and the associated change event.

You specify the key type on the **Tenant Secrets** page when creating a new key using the **Create Tenant Secret** dialog box or uploading an existing key using the **Upload a Tenant Secret** dialog box. The existing key is one you previously created from the **Key Sources** page. For more information about the types of tenant secrets, refer to the appropriate Salesforce documentation.

The option to opt out of using the Salesforce key derivation mode when uploading tenant secrets from the **Tenants Secret** page is available in support of this Salesforce feature. The key derivation mode in CCKM allows you to derive a data encryption key based on the tenant secret you create or upload to Salesforce from the **Tenants Secret** page. Opting out of this mode means the tenant secret that you create or upload will be the key used to encrypt and decrypt your data.

From the **Tenant Secrets** page, a **Filter by** box is also available allowing you to filter on the types of tenant secrets to display in the list of tenant secrets.

A **Search** box is available at the top left of the **Tenant Secrets** page allowing for a search of a specific key by entering any one of the following parameters:

- Key ID
- Version Key ID
- Keyvault name
- Keyvault location
- Source key
- Subscription ID

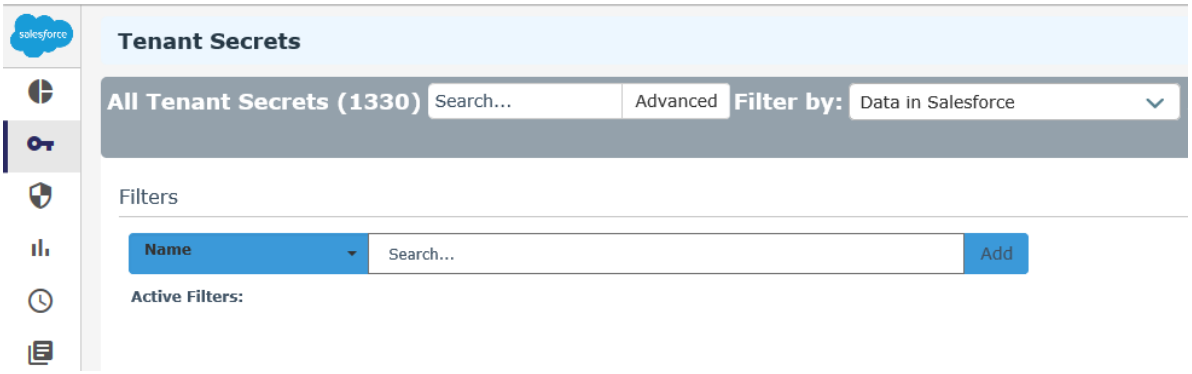
The **Advanced** tab allows you to filter your search on the each of the columns displayed on the **Tenant Secrets** page with the exception of the **Action** column:

- Key Name
- Version—Min and Max
- Status— Available options from the drop-down menu are ACTIVE, ARCHIVED, and DESTROYED.
- Generated By
- Generated On—Enter START DATE and END DATE.
- Key Derivation
- Backup—Available options from the drop-down menu are Backed Up and Not Backed UP.
- Uploaded—Available options from the drop-down menu are Uploaded and Not Uploaded
- Key Material Origin

In addition, you can also search on **Key ID**. A **Key ID** for a given key is only available in the **Tenant Secret** box. To view the Key ID of a key, from the **Tenant Secrets** page, select the name of the key listed under the **Name** column. The **Tenant Secret** box displays and includes the name of the key, the SFDC Key ID, key version, description of the key, and other properties.

Each of these columns are described in more detail below in this section.

To select to filter based on a specific column or Key ID from the **Tenant Secrets** page, click **Advanced**. The **Filters** section displays. From the filter drop-down menu, select the name of the column on which to filter your search or Key ID. For example, select **Key Name**. For the **Status**, **Backup,** and **Uploaded** columns, a drop-down menu is available from which to select a value for the search. Click the down arrow, select a value from the available options, and then click **Add**. For the **Version** column, enter a minimum and maximum version number for the search and then click **Add**. For the **Generated On** column, enter a start and end date for the search and then click **Add**. The specified search filter is then added to **Active Filters**. In the example, **Key Name: MK-createkey** is added to **Active Filters**. The results of the search displays in the **Tenant Secrets** table. You can narrow your search by selecting more columns to add to the filter. To clear the filtered results, from **Active Filters**, click on the “X” (or the **Delete** button) for each of the named filters.



At the top right of the **Tenant Secrets** page, the following buttons are available:

- Export Secrets: Allows you to export the contents of the Tenants Secrets list into a report in a CSV format. For information on how to export your tenant secrets, see section 11.5.8, Export Tenant Secrets.
- New Key: Allows you to upload an external key (BYOK) or create a native key. For information on how to upload an external key or create a native key, see section 11.5.1, Create a Tenant Secret/Key.
- Synchronize: Allows you to download any keys that were created in Salesforce, outside of the CCKM portal, into CCKM. For information on how to synchronize your keys, see section 11.5.7, Synchronize Tenant Secrets.

The following columns display in the **Tenant Secrets** page:

- Name: The name of the key.
- Version: The version number of the key.
- Status:
  - ACTIVE
  - DESTROYED
  - ARCHIVED
- Generated By: The username associated with generating the key.
- Generated On: The date on which the key was generated.
- Key Derivation: Indicates whether the tenant secret is in key derivation mode. A green check icon indicates that it is in this mode.
- Backup: Indicates whether a backup of the key exists in CCKM. The status can be one of the following:
  - A green check icon indicates the backup exists in CCKM.
  - A red exclamation icon indicates the backup key does not exist in CCKM.
- Uploaded: Indicates whether the key is uploaded to Salesforce. A green check icon indicates that it is uploaded. Cache-only keys that are uploaded to Salesforce also display with the green check icon.
- Key Material Origin:
  - If the key was uploaded from CCKM, then the value is **Internal(<source_key_name>)**. For a cache-only key, the value is **FETCHED(<source_key_name>)**.
  - If the key was not uploaded from CCKM as a BYOK key, then the value is **EXTERNAL**.

- Actions: Allows you to select any of the following actions on a given key:
    - Destroy
      Note: For a cache-only key, **Destroy** is the only action available within the **Actions** column. After the key is destroyed, the **Actions** column displays as blank with no action that can be taken.
    - Edit key
      Note: This action is *only* applicable to a cache-only key.
    - Import
    - Delete Backup

In this section, create, upload, destroy, import, synchronize, and manage tenant secrets, as well as delete a backup key are described.

11.5.1 Create a Tenant Secret/Key

You have three options to create a new tenant secret or key:

- Upload an external key (BYOK)
- Create a native key
- Create Cache-only key

11.5.1.1 Upload your Own Tenant Secret

Salesforce supports only one active tenant secret per organization/OrgID. When you upload your own tenant secret, that key becomes active and the previously active key is automatically archived. Use **Upload** as part of your organization's key rotation policy. You can only upload a tenant secret once every 24 hours, as per Salesforce's policy.

Note: A valid certificate is required, as described in the *Prerequisites on Salesforce*. If no certificate is listed in the certificate drop-down, you must go to Salesforce, create an appropriate certificate, and re-synchronize with CCKM.

To upload your BYOK tenant secret to CCKM:

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**. The *Tenant Secrets* list page displays.

2 From the *Tenant Secrets* list page, within the **New Key** box, select **Upload**. The **Upload a Tenant Secret** dialog box displays.

3 From the **Type** box, select the type of tenant secret to upload. The following are the options (in the order of selection):

- Data in Salesforce
- Search Index
- Data in Salesforce (Deterministic)
- Analytics
- Event Bus

4 From the **Tenant Secret** drop-down menu, select the tenant secret to upload. Enter the first three characters of the name of the tenant secret to bring it up for selection.

5 Select whether to apply the Salesforce key derivation mode to the tenant secret. If you choose to apply it, ensure that the **Use Salesforce key derivation** checkbox is selected. By default, the checkbox is selected. Otherwise, deselect the checkbox to opt out of applying the key derivation mode to the tenant secret.

6 From the **Certificate** drop-down menu, select a valid certificate (used to wrap the key for secure upload).

7 Click **Upload to Salesforce** to upload the tenant secret to Salesforce.

The uploaded key is displayed in the list. As part of Upload process, Salesforce generates a corresponding backup key, which CCKM downloads and stores in the DSM.

11.5.1.2 Create a Tenant Secret/Key

Create a native key in Salesforce using CCKM. After CCKM creates a native key, it displays the key as **EXTERNAL** under the **Key Material Origin** column within the **Tenant Secrets** > **All Tenant Secrets** list page in CCKM.

To create a native tenant secret:

1 From the *Tenant Secrets* list page, within **New Key** box, select **Create**.

2 From the **Type** box of the **Create a Tenant Secret** dialog box, select the type of tenant secret to create.

3 Select whether to apply the Salesforce key derivation mode to the tenant secret. If you choose to apply it, ensure that the **Salesforce key derivation** checkbox is selected. By default, the checkbox is selected. Otherwise, deselect the checkbox to opt out of applying the key derivation mode to the tenant secret.

4 Click **Create** to create the new tenant secret.

11.5.1.3 Create a Cache-only Key

In CCKM, create a tenant secret or cache-only key for use with Salesforce's cache-only key service. In creating this key, select the key you generated using a DSM. (See section 11.4.1.1, Add (Generate) a DSM Key for more information.) Also, select a named credential that matches the named credential you previously defined in Salesforce. (See section 11.2.2, Set up Named Credential in Salesforce for more information.) After this key is created, the name of the key is displayed in the **Name** column, a green checkmark icon is displayed in the **Uploaded** column, and a status of FETCHED(<key_name>) is displayed in the **Key Material Origin** column of the **Key Sources** page. In addition, a log message in the **Logs** page is recorded and displayed indicating a call-out connection to Salesforce has been created.

To create a cache-only key:

1 From the *Tenant Secrets* list page, within **New Key** box, select **Cache Only**. The **Create cache-only key** dialog box displays.

2 From the **Type** box, select the type of cache-only key to create. The following are the options (in the order of selection):

- Data in Salesforce
- Search Index
- Data in Salesforce (Deterministic)
- Analytics
- Event Bus

3 From the **Tenant Secret** drop-down menu, select the key you generated using a DSM. Enter the first three characters of the name of the key to bring it up for selection.

4 In the **Certificate** box, select a valid certificate (used to wrap the key for secure upload).

5 In the **Named Credentials**, select the named credential to associate with this cache-only key.

6 Click **Create cache-only key** to create the new cache-only key.

11.5.2 Destroy a Tenant Secret

On the *Tenant Secret* page, destroying a tenant secret permanently deletes it from Salesforce (but a backup key will remain in CCKM). After deletion, two new actions are displayed in the Action column: *Import* and *Delete Backup*.

Note: You can destroy only an archived tenant secret. If you want to destroy the currently active tenant secret, first change its status to ARCHIVED by uploading a new tenant secret (see section 11.5.1.1, Upload your Own Tenant Secret).

---

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**.

2 Find the name of the tenant secret and click **Destroy** under the **Actions** column. The **Destroy Tenant Secret** dialog box displays with a warning asking you to confirm the key destruction by entering the supplied phrase.

3 Enter or copy/paste the confirmation phrase.

4 Click **Destroy**.
The *Status* column for this tenant secret shows **DESTROYED**, and *Import* and *Delete Backup* actions are listed under the **Actions** column.

Note: If you simultaneously deleted the backup key using the **check box** option, the **Action** column will be blank.

11.5.3 Destroy a Cache-only Key

On the *Tenant Secret* page, destroying a cache-only key removes it from the Salesforce cache. However, a backup of the key still exists in the DSM. After the key is destroyed, no actions are displayed under the **Action** column for the key.

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**.

2 Find the name of the key and click **Destroy**. The **Destroy Tenant Secret** dialog box displays with a warning asking you to confirm the key destruction by entering the supplied phrase.

3 Enter or copy/paste the confirmation phrase.

4 Click **Destroy**.
The *Status* column for this key shows **DESTROYED**.

11.5.4 Edit a Cache-only Key

If a certificate configured for use with a cache-only key has expired or you want to use another named credential that is configured with another CCKM hostname in Salesforce other than the one currently in use, edit the cache-only key configuration within the *Tenant Secret* page.

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**.

2 Find the name of the key and click **Edit Key**. The **Edit <key_name> Cache-only Key** dialog box displays.

3 To update the certificate replacing the current certificate with another one, in the **Certificate** box, select the certificate you wish to use.

4 To update the named credential replacing the current named credential with another one, in the **Named Credentials** box, select the named credential you wish to use.

5 Click **Update cache-only key**.

11.5.5 Import a Tenant Secret

In the event that someone deletes a Salesforce tenant secret directly in Salesforce, it is possible to restore it from the CCKM as follows:

Note: The Import action is available for tenant secrets with the status DESTROYED.

---

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**.

2 From the list, select **Import** in the **Action** column of a Destroyed key.

3 The key and metadata are restored to Salesforce. The Success message is displayed and the key is listed at the top of the All Tenant Secrets page with the status **ARCHIVED**.

Note: If anyone deletes the backup key from CCKM, it will not be possible to import/restore the key to Salesforce.

11.5.6 Rotate a Tenant Secret

CCKM allows for a manual or a scheduled rotation of a Salesforce tenant secret by the tenant secret type. When you rotate a tenant secret, that key becomes active and the previously active key is automatically archived. Scheduling a rotation of a key is adding a new version of the key at a scheduled time. Scheduling the rotation of a tenant secret requires that you first configure this feature from the **Schedules** page prior to enabling it using the **Auto-rotate secret type** toggle at the top of the **Tenant Secrets** page.

Note: If you provide your Salesforce user password when configuring the scheduled rotation of a tenant secret, which is an optional step, the password is saved to DSM.

Note: If you have not scheduled a key rotation from the **Schedules** page prior to enabling this feature on the **Tenant Secrets** page, then the **Auto-rotate secret type** toggle is disabled.

11.5.6.1 Manually Rotate a Tenant Secret (Add a New Version of a Tenant Secret)

Creating and uploading a tenant secret in CCKM is equivalent to manually rotating a tenant secret in CCKM.

11.5.6.2 Enable a Scheduled Tenant Secret Rotation

Enabling a scheduled rotation of a tenant secret requires that you first schedule to rotation in the **Schedules** page. Be sure to schedule the rotation of a tenant secret prior to enabling this feature.

During the scheduled date and time for the rotation, the tenant secret is rotated. The new tenant secret becomes active and the previously active tenant secret is automatically archived.

Note: A tenant secret that is a cache-only key remains as a cache-only key after rotation.

To enable a scheduled rotation of a tenant secret:

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**. The *Tenant Secrets* list page displays.

2 From the *Tenant Secrets* list page, within the **Filter by** box, select the type of tenant secret to enable for a scheduled rotation. The following are the options:

- Data in Salesforce
- Search Index
- Data in Salesforce (Deterministic)
- Analytics
- Event Bus

3 Click on the **Auto-rotate secret type** toggle to turn it to **On**. By default, this toggle is set to **Off**.

11.5.7 Synchronize Tenant Secrets

Synchronization downloads tenant secrets that were directly created in the Salesforce portal (not in CCKM). It does not upload anything from CCKM to Salesforce. Periodically synchronize CCKM with Salesforce to get any newly created tenant secrets from Salesforce.

To synchronize tenant secrets:

1 In the left navigation bar, click **Tenant Secrets**.

2 Click **Synchronize**.
Any new tenant secrets created in Salesforce are retrieved and displayed in the UI.

11.5.8 Export Tenant Secrets

Export Secrets allows you to export the contents of the Tenant Secrets list into a report in a CSV format.

To export your Tenant Secrets list:

1 In the left navigation bar, click **Tenant Secrets**.

2 Click **Export Secrets**.
This will download all of the tenant secrets in a report in a CSV format.

11.5.9 Backup Key

When a tenant secret is uploaded to Salesforce, Salesforce creates a backup key, which CCKM automatically downloads and stores in DSM. In addition, when Salesforce tenant secrets are synchronized, a backup key of any tenant secret that was created directly in Salesforce is downloaded. Backup keys are encrypted blobs, and stored securely in DSM. A backup key contains information about the tenant secret. It can be used to restore a key that has been deleted in Salesforce.

11.5.9.1 Delete a Backup Key

Delete a backup key when you definitely do not want the key and have no corresponding data that will ever need decryption. For a tenant secret that has been deleted:

1 In the CCKM portal, in the left navigation, select **Tenant Secrets**.

2 From the list, select **Delete Backup** from the **Action** column of a Destroyed tenant secret. The **Destroy Key** warning page is displayed.

3 Enter or copy/paste the confirmation phrase.

4 Click **Delete**.
The key is removed from CCKM.

11.6 Certificates

The **All Certificates** list page allows you to view all Salesforce certificates, including Name, Type, Key Size and Expiration Date. Use the **Synchronize** button to download newly created certificates from Salesforce.

Note: See *Prerequisites on Salesforce* for the minimum required parameters for creating Salesforce certificates compatible with CCKM.

11.6.1 Synchronize Certificates

Synchronization downloads certificates that were directly created in the Salesforce portal (not in CCKM). Periodically synchronize CCKM with Salesforce to get any newly created tenant certificates from Salesforce.

1 In the left navigation bar, click **Certificates**.

2 Click **Synchronize**.
Any new certificates created in Salesforce are retrieved and appear in the UI.

11.7 Reports

Before the Salesforce keys can be represented in the CCKM reports, Event monitoring and Event log files must be activated in Salesforce. In addition, the log files validity must be set to 30 days in Salesforce.

Note: Currently, there is no support for cache-only keys within CCKM reports.

The **All Reports** page displays links to the following detailed individual reports

1 **Key Activity Report**
Displays the key activity type (such as TS (tenant secret) stored or TS generated) performed by a specific user along with the associated key ID, key version number, key name, and time at which the activity took place.

2 **Cloud Key Manager User Action Report**
Lists the users and their actions in creating, uploading, importing, destroying, and deleting keys in the CCKM portal.

For each report displayed on the **All Reports** page, the report name, its description, the email address of the user who had run it (Created By), and the date on which it was previously run (Last Run) is displayed. For a specific report you select to view, the date on which it was previously run (Last Run), the start and end dates selected for the report, the email address of the user who had run it (Created By), and the total number of records available in the report (Report Count) is displayed. In addition, an **Actions** column is displayed, which provides the **Delete Report** link to allow for the deletion of the selected report from CCKM.

11.7.1 Generate a Report

To generate a report:

1 In the CCKM portal, in the left navigation, select **Reports**.

2 Under **Report Name**, select the report you wish to run.

3 From **Date Range**, click **Last day** to run the report for the activities from the previous day or **Specific** to run the report using a date range. If you choose to run by a date range, enter the start and end dates in the format MM/DD/YY or select the dates from the calendar button at the right of the box.

4 Click **Run Report.** The report displays.

5 To download the report, click **Download** at the top right of the page and select the report format (PDF or CSV) to download.

11.7.2 Delete a Report

To delete a report:

1 From All **Reports** page, select the specific report you wish to delete. The report displays.

2 From the **Actions** column, click **Delete Report**. The **Delete Current Report** dialog box with a warning that you are about to delete a current report run on a specified date and time.

3 Click **Delete** to confirm. The report is deleted from CCKM.

11.8 Schedules

The **Schedules** page allows Cloud administrators to view and schedule the following job types:

- Key Rotation
- Key Synchronization

At the top right of the **Schedules** page, the **Add Schedule** button is available from which to add a scheduled job type.

Note: Currently, there is no support for cache-only keys within CCKM schedules.

11.8.1 Schedule Key Rotation

You can schedule a key rotation of a Salesforce tenant secret in CCKM. When you rotate a tenant secret, that key becomes active and the previously active key is automatically archived. Scheduled rotation is configured in the **Schedules** page. Note that if you provide your Salesforce user password when configuring a scheduled key rotation, which is an optional step, the password is saved to DSM.

You can pause or delete a scheduled key rotation in the **Schedules** page. Note that when a key rotation schedule is deleted from CCKM, the password is also deleted from DSM.

11.8.1.1 Set up a Key Rotation Schedule

To set up a key rotation schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the Schedules page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

3 From the **Job Type** drop-down menu, select **KEY_ROTATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For Pause, select **No**.

6 For **Schedule**, set a schedule for the rotation by selecting **Basic** (to enter a date and interval for the schedule) or **Advanced** (to enter a Cron Expression for the schedule). If you selected **Basic**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

   Set the repeat interval of the key rotation by entering a number in **Repeat Interval** and then selecting Day, Week, or Month from the drop-down menu. The supported ranges are 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter. If you selected **Advanced**, enter the cron expression in the **Cron Expression** box.

7 Select the wrapping certificate to use to wrap the key from the **Wrapping Certificate** drop-down menu.

8 (Optional) Enter your Salesforce user password in the **User Password** box.

9 Click **Save** to save the setting.

11.8.1.2 Pause a Key Rotation Schedule

To pause a key rotation schedule, select **Yes** for **Pause** within the **Add Schedule** dialog box and then click **Save**. The setting is saved. To resume the rotation schedule after pausing it, select **No** for **Pause** and then click **Save.** The setting is saved.

11.8.1.3 Delete a Key Rotation Schedule

To delete a key rotation schedule, select **Delete** from the **Actions** drop-down menu from the **Schedules** page. Note that password associated with the given key rotation schedule is also deleted from DSM.

11.8.2 Schedule Key Synchronization

CCKM allows for a scheduled synchronization of Salesforce keys. Synchronizing downloads any keys that were created in Salesforce, outside of the CCKM portal, into your CCKM. Scheduled key synchronization requires that this feature be configured in the **Schedules** page. You can also pause or delete a scheduled key synchronization in the **Schedules** page. Note that when a key synchronization schedule is deleted from CCKM, then the password is also deleted from DSM.

11.8.2.1 Set up a Key Synchronization Schedule

To set up a key synchronization schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

3 From the **Job Type** drop-down menu, select **KEY_SYNCHRONIZATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 For **Schedule**, set a schedule for the key synchronization by selecting **Basic** (to enter a date and interval for the schedule) or **Advanced** (to enter a Cron Expression for the schedule). If you selected **Basic**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

   Set the repeat interval of the key synchronization by entering a number in **Repeat Interval** and then selecting Day, Week, or Month from the drop-down menu. The supported ranges are 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter. If you selected **Advanced**, enter the cron expression in the **Cron Expression** box.

7 Enter your Salesforce user password in the **User Password** box.

8 Click **Save** to save the setting.

11.9 Logs

CCKM maintains logs for events, such as login, key generation, synchronizing of tenant secrets, calls from Salesforce to CCKM for cache-only keys, and deletion of keys, and displays these logs in the **All Logs** page. For each logged event listed on the **All Logs** page, the following information is displayed:

- Event name
- Severity level
- Date and time of the event
- Event message
- User (name) associated with the event

You can search for a specific logged event from the **All Logs** page using the **Search** box. Allowable filters on the search are event name, severity level, event message, and user (name).

11.10 Settings

To access the **Settings** page in CCKM, select **Settings** from the left-hand navigation. From the **Settings** page, you can:

- Enter the emails of the recipients who are to receive alerts and reminders.
- Set up key alerts.
- Configure a remote syslog server to which to send syslog messages.
- Configure the username and password to associate with the CCKM Rest API endpoint (or callout endpoint) for use with cache-only keys.
- View the release number of the CCKM you are running.

11.10.1 Adding Recipients' Emails (for Alerts and Reminders)

Use the **General** settings tab to enter the designated recipients' emails for email alerts and reminders to be sent from the system. Enter the recipients' email addresses separated by a comma in the **Alert Format** box.

11.10.2 Enabling Alerts and Reminders

Use the **Tenant Secret Alerts** tab from the **Settings** page to specify whether to send an alert to the specified user email(s) when a Salesforce key is deleted, restored, or uploaded in Salesforce. You can also use this tab to set up a reminder to manually rotate (or rekey) a key along with the frequency (in days) at which the reminder is sent to the specified user email(s). The user emails are set up in the **General** tab. By default, no alerts or reminders are sent.

To set up key alerts or a reminder to manually rotate (or rekey) a key:

1 In the **Settings** page, select the **Tenant Secret Alerts** tab.

2 Under **Alerts**, click **On** for each of the alert types and reminder you wish to enable:

- Rekey Alert (reminder)
- Destroy Tenant Secret
- Import Tenant Secret
- Upload Tenant Secret

3 In the **Days** box, enter the number of days for the frequency a reminder email is sent to rotate a key.

4 Click **Save** to save the setting.

11.10.3 Configuring Syslog Server

Use the **Syslog** tab from the **Settings** page to configure a remote syslog server to which to send syslog messages. Note that the default port number of 514 and the facility name of "LOCAL1" for the syslog server display in the **Port** and **Facility** boxes, respectively. These boxes are not available for modification.

11.10.3.1 Configure a Remote Syslog Server

To configure a remote syslog server:

1 In the **Settings** page, select the **Syslog** tab. The **Syslog** dialog box displays.

2 Enter the hostname or IP address of the syslog server in the **Hostname** box.

3 (Optional) Enter a description of the syslog server in the **Description** box.

4 Click **Save** to save the setting.

11.10.3.2 Delete Remote Syslog Server Configuration

To delete the configuration of a remote syslog server, click **Delete** from the **Syslog** dialog box. No syslog messages are sent to the remote syslog server after the deletion of this configuration.

11.10.3.3 Configure Cache-only Key Endpoint

Use the **Endpoint for Cache-only Key** tab from the **Settings** page to configure the CCKM username and password to associate with the URL of the CCKM Rest API endpoint (or callout endpoint) for use with cache-only keys.

To configure the username and password to associate with the URL of the CCKM Rest API endpoint:

1 In the **Settings** page, select the Endpoint for Cache-only Key tab. The Cache-only Key Configuration page displays.

2 The URL of the CCKM Rest API endpoint automatically displays in the **Call out URL** box.

Note: When defining the named credential for use with cache-only keys within Salesforce's **Named Credentials Setup** page, copy the URL from the **Call out URL** box and paste it into the **URL** box in **Named Credentials Setup** page. Then replace "<cckm_hostname>" in the URL with the name of the CCKM hostname you are using.

3 In the **Username** box, enter the username to associate with the URL of the CCKM Rest API endpoint. This username must match the username configured in Salesforce's **Named Credentials Setup** page.

4 In the **Password** box, enter the password to associate with the URL of the CCKM Rest API endpoint. This password must match the password configured in Salesforce's **Named Credentials Setup** page.

5 Click **Save Settings** to save the setting.

11.10.4 About

Use the **About** tab to view the release number of the CCKM you are running.

12 CCKM for AWS Key Management

CipherTrust Cloud Key Manager for AWS offers risk management and assessment of AWS customer-managed Customer Master Keys (CMKs) across AWS services. It provides operational visibility of AWS CMKs across AWS services, as well as tools to manage the complete lifecycle of keys from on premise or from the cloud.

CCKM for AWS Key Management involves a few concepts specific to AWS:

- **Source key:** Source keys are generated in the CCKM/DSM or nShield Connect HSM and can be uploaded to AWS.
- **AWS regions:** Source keys can be uploaded to only one AWS region (chosen from a drop-down menu).
- **Customer Master Key (CMK):** There are two types of CMKs—AWS-managed or customer-managed. The CMK origin can be AWS_KMS (generated in KMS), or EXTERNAL (imported by user). CCKM can manage the lifecycle of all keys, but owns the key material of only EXTERNAL keys imported by CCKM itself. CMKs are region-specific, meaning CMK operations and APIs work on CMKs in only one region at a time.
- CMKs can be disabled and enabled. CMKs cannot be deleted immediately, but can be scheduled for deletion 7 to 30 days in advance. CMKs can be identified by Key Id, Key ARN, or alias.
- **Alias:** Use a reader-friendly alias when uploading a source key to AWS to make searching for key names more intuitive. Multiple aliases can be assigned to the same CMK.
- **Delete key:** Unlike some other key management cloud services, the AWS CMK APIs do not provide a direct mechanism for creating backup keys. Therefore, the process of key deletion in the CCKM has several phases.
    - A source key that has been created in the DSM/CCKM, but not uploaded or used in AWS, can be fully deleted from the **Key Sources** page by clicking **Delete Key**.
    - Once a source key has been uploaded to AWS, deleting the source key will only remove it from the **Key Sources** page, but not delete it from DSM. CCKM will keep the source key in DSM as a backup. So, for the uploaded AWS key, the user can do “Delete > Delete Key Material Now” to remove the key material from AWS, and then later do “Import” to re-import (restore) the key material in AWS.

      Once a source key has been uploaded to AWS, to delete it from DSM you must delete both the uploaded AWS and the source key from **Key Sources** page. First, schedule deletion of the uploaded AWS key. After the AWS key is deleted on AWS, do a “Synchronize” on CCKM. Then delete the source key from “Key Sources” page. This action will remove the key from both AWS and DSM completely and permanently.

- **CMK key rotation:** The rotation of CMKs is encouraged as part of cryptographic best practices. Note that CCKM for AWS currently supports key rotation for CCKM keys only. In CCKM, Key rotation involves creating and uploading new external CMK and moving the alias(es) from the old (previous key) to this newly uploaded key (current key). This key rotation does not affect the components of the current CMK, such as the key ID, key ARN, and region. The old key can be continued to be used to decrypt the data that was previously encrypted by this key.

  CCKM allows for a manual or a scheduled rotation of a CMK. Scheduled rotation requires that the key rotation feature be configured in the **Settings** page within CCMK prior to enabling this feature in the **Rotate AWS Key** page.

  Note that by default all CKMs, which the CCKM creates, allow both encrypt and decrypt operations. However, in CCKM, when enabling key rotation (either manually or scheduled), there is an option (on by default) to prevent the current key from encrypting any new data going forward. During the configuration of key rotation,

ensure the **Disable Encrypt permission on Current key** check box is cleared to keep the encryption permission on the current CMK key. **Note that for native AWS CMK rotation, the current key, once rotated out, can only be used to decrypt the previous data it encrypted. It cannot be used to encrypt any new data.**

12.1 Prerequisites on AWS

You must have:

- KMSFullAccess
- IAM GetUser permissions
- CloudWatch Logs permissions (Limited: List, Read All Resources)

12.2 Access AWS Key Management

You can either enter your AWS secret key ID and access key when prompted, or upload an AWS credentials file on the login page. AWS Credentials file should be `.json` format (can be written in notepad and saved for upload).

Example:

```
{
"aws_access_key_id":"ABCDEFG1234hiJKLMN",
"aws_secret_access_key":"fgreg1234567tg\refg35"
}
```

12.3 Access CCKM for AWS

To access the CCKM service for AWS Key Management from CCKM:

1 In the CCKM portal, select **Key Management** > **AWS**.

Or enter https://<CCKM IP address>:8443/kmaas

The CCKM login page is displayed.

2 Click **Log in to AWS**. The AWS Password Login page is displayed.

Enter your AWS key credentials in the **AWS Access Key ID** and **AWS Secret Access Key** boxes and click **Log In**.
The first time you access the CCKM, the keys are synchronized and a progress page is displayed. Once synchronization is complete, click **Continue**. The CCKM Home page is displayed.

The CCKM portal for AWS contains the following menu options in the left-hand navigation bar:

- **Home:** Home page displays key operation and count metrics.
- **Keys:** Keys page displays key management overview and provides the Upload and Synchronize functionality, and the Delete, Restore, and Auto Rotate key operations available in CCKM.
- **Reports:** All Reports overview page links to detailed individual reports.
- **Schedule:** In Schedule, you can schedule automatic key rotation, key rotation based on key expiration, and key synchronization.
- **Logs:** All Logs page lists individual logs that can be searched, sorted, and viewed.
- **Key Sources:** In **Key Sources**, you can create keys to upload to AWS. CCKM can be configured to interconnect with either DSM or nShield Connect HSM as the key source where the keys are securely generated and stored. If you are using nShield Connect, the generated keys are stored as blobs within the MongoDB.
- **Settings:** Settings Management page allows you to configure alert email settings, key alerts, remote syslog server to which to send syslog messages, and proxy server settings. You can also view the release number of the CCKM you are running from this page.

To display the context of which Key Management service you are currently logged into within CCKM, the icon associated with the Key Management service displays at the top of the left-hand navigation bar.

The **arrow** button at the bottom of the navigation bar allows you to expand the bar to view the names of the menu options and collapse it to view only the icons associated with the menu options. Click the **arrow** button to either expand or collapse the navigation bar.

On the top-right side of the CCKM portal for AWS, the following options are available:

- **Azure Cloud Service Context Switcher:** This context switcher allows you to login into another Azure cloud service different from the current Azure cloud service into which you are currently logged. If you select the Azure cloud in which you are currently logged, the **Multi-account Login/Logout** window for the Azure account(s) displays. For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service.*
- **AWS Login:** If you are not already logged into AWS, clicking on the **AWS Login** icon brings you to the **AWS Password Login** page where you can enter your AWS credentials to access the CCKM AWS cloud service (from the current cloud service in which you are currently logged). If you are already logged into AWS, clicking on the **AWS Login** icon brings you to the **Multi-account Login/Logout** window for the AWS account(s). For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service*.
- **Salesforce Cloud Service Context Switcher:** This context switcher allows you to login into another Salesforce cloud service different from the current Salesforce cloud service into which you are currently logged. This is either Salesforce or Salesforce Sandbox. If you select the Salesforce cloud in which you are currently logged, the **Multi-account Login/Logout** window for the Salesforce account(s) displays. For more information about the **Multi-account Login/Logout** window, refer to section 9.2, *Support for Multi-accounts per Cloud Service.*
- **Logout:** Logout from CCKM.

12.4 Key Sources

The **Key Sources** page displays all currently posted keys in AWS. Two tabs are available at the top of the page:

- **DSM Key**: Use this tab to create your keys from DSM.
- **nShield Key**: Use this tab to create your keys from nShield Connect HSM. The generated keys are stored as blobs within the MongoDB.

Keys are listed by name, key type (the cloud service in which this key is used), the algorithm of the key, the creation date of the key, key description, and actions you can take regarding that key. Note that the key algorithm currently supported for an AWS key in CCKM is AES256.

A **Search** box is available at the top left of the **Keys Sources** page allowing for a search of a specific key by entering any one of the following parameters:

- Name (key name)
- Key Type
- Algorithm
- Description
- Created By (a key created by a specific user)

12.4.1 Add (Generate) a Key

From the **Key Sources** page, you can add (or generate) keys in CCKM using either DSM or nShield Connect HSM as the key generator. These keys are of the AES256 algorithm type.

12.4.1.1 Add (Generate) a DSM Key

1 In the CCKM portal, in the left navigation, select **Key Sources**.

2 Select the **DSM Key** tab.

3 Select **Add Key** > **AWS Key**.

4 In the **Add DSM Key** dialog box, select the name of the user of the key from the **User** drop-down menu. **Service** is fixed to AWS and (key) **Algorithm** is fixed to AES256.

5 In the **Name** box, enter the name of the new key.

6 Enter a description of the key in the **Description** box.

7 Click **Save** to save the new key to CCKM. The new key is added to the key list.
(Reset clears the screen before it has been saved.)

12.4.1.2 Add (Generate) an nShield Key

In the CCKM portal, in the left navigation, select **Key Sources**.

1 Select the **nShield Key** tab.

2 Select **Add Key** > **AWS Key**.

3 In the **Add nShield Key** dialog box, select the name of the user of the key from the **User** drop-down menu. **Service** is fixed to AWS and (key) **Algorithm** is fixed to AES256.

4 In the **Name** box, enter the name of the new key.

5 Enter a description of the key in the **Description** box.

6 Click **Save** to save the new key to CCKM. The new key is added to the key list.
(Reset clears the screen before it has been saved.)

12.4.2 Delete a Key

Deleting a key from the **Key Sources** page deletes the key from CCKM. If you delete a key from **Key Sources** that has already been uploaded to AWS, the key will not be deleted from AWS.

1 In the CCKM portal, in the left navigation, select **Key Sources**.

2 Select either the **DSM Key** or **the nShield Key** tab depending on where the key you wish to delete resides.

3 Click **Delete** next to the target key.

4 The **Delete Key** dialog box with a Warning display asking you to confirm the key deletion by entering the supplied phrase.

5 Enter or Copy/paste the confirmation phrase.

6 Click **Delete** to confirm.

12.4.3 Manage keys in the Key Sources List Page

In addition to adding and deleting keys, the **Key Sources** page allows you to sort the existing keys by Name, Key Type, Algorithm and Created Date.

12.5 Keys

The **AWS Keys** page allows Cloud administrators to view all of the keys and manage them within the boundary of granted permissions. The keys or key vaults that do not have List permission for the respective Cloud administrator are not listed.

At the top right of the **Keys** page, the following buttons are available:

- Export Keys: Allows you to export the contents of the Keys list into a report in a CSV format.
- Alias View: Allows you to view the Keys list by the key alias names.
- New Key: Allows you to upload an external key (BYOK) or create a native key. For information on how to upload an external key or create a native key, see section 11.5.7, Synchronize Tenant Secrets.
- Synchronize: Allows you to download any keys that were created in AWS, outside of the CCKM portal, into CCKM. For information on how to upload a key, see section 10.7.9, Synchronize Keys.

In this section, create, delete, schedule delete, import, update, rotate, synchronize, exports tenant secrets are described.

You can view the **AWS Keys** page by the **Keys** view or the **Alias** view. Here is the **Keys** view with **Key ID** as the first column.

To view the **AWS Keys** page by Alias view, click **Alias View**. Any alias you assigned to the source keys when uploading these keys to AWS is shown in the **Alias** column, the first column listed within this view.

A **Search** box is available at the top the **AWS Keys** page allowing for a search of a specific key by entering any one of the following parameters:

- **Key ID**: The globally unique identifier for the CMK.
- **Source Key:** The name of the source key. This is the key created in DSM and displayed in the CCKM UI.
- **Region:** The AWS Region to which the key is uploaded.
- **ARN**: Amazon Resource Name of the CMK.
- **ARN description:** The description given to the CMK's ARN.
- **Origin:** The source of the key material. The valid values are:
    - AWS_KMS: Key material was created by AWS KMS.
    - EXTERNAL: Key material was imported from CCKM or another key management system.
- **Key Manager:** The manager of the CMK. The valid values are:
    - AWS: AWS-managed CMK.
    - CUSTOMER: Customer-managed CMKs that you create, use, and manage.
- **Key State:** The state of the CMK. The valid values are:
    - Enabled
    - Disabled

- PendingDeletion
- PendingImport

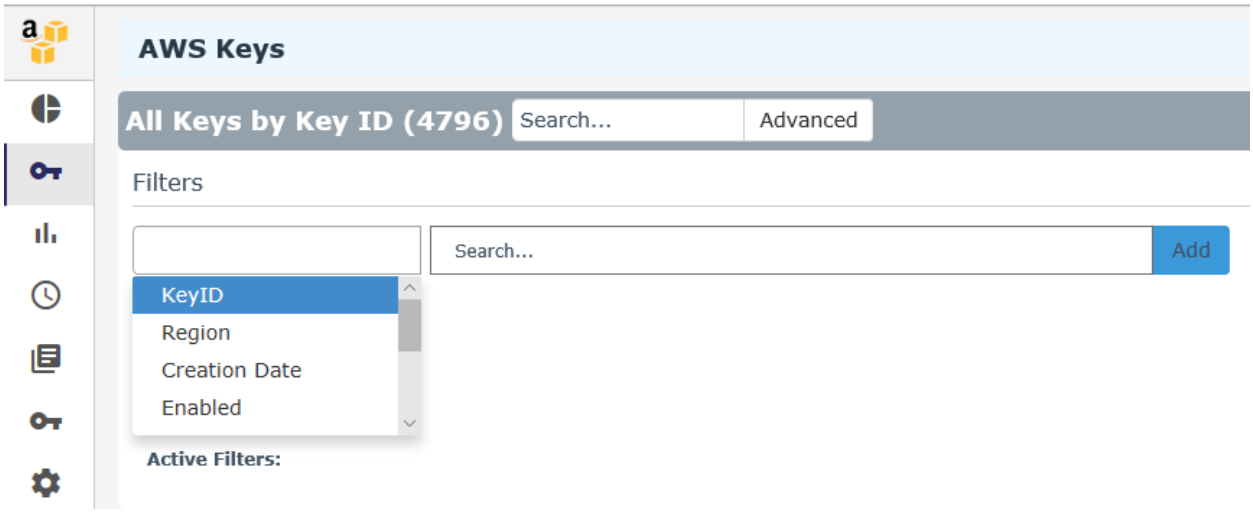
- **Key Usage:** The cryptographic operations for which the CMK is used. The only valid value is ENCRYPT_DECRYPT.
- **Expiration Model:** Specifies whether the key material of the CMK expires. This value is present only when the source of the key material was imported from CCKM or another key management system. The valid values are:
  - KEY_MATERIAL_EXPIRES
  - KEY_MATERIAL_DOES_NOT_EXPIRE
- **Tags:** The label assigned to the key, which consists of a customer-defined key and an optional value. For example, department finance.
- **Key Aliases:** The alias or aliases assigned to the key.

The **Advanced** tab allows you to filter your search on the each of the columns displayed on the **AWS Keys** page with the exception of the **Action** columns:

- Key ID
- Region
- Creation Date—Enter START DATE and END DATE.
- Enabled— Available options from the drop-down menu are ENABLED and DISABLED.
- Key State— Available options from the drop-down menu are Enabled, Disabled, PendingImport, PendingDeletion
- Scheduled Deletion Date—Select a START DATE and END DATE.
- Key Material Expiration Date—Select a START DATE and END DATE.
- Key Material Origin
- Rotation Status— Available options from the drop-down menu are ACTIVE and ARCHIVED.

Each of these columns are described in more detail below in this section.

To select to filter based on a specific column from the **AWS Keys** page, click **Advanced**. The **Filters** section displays. From the column filter drop-down menu, select the name of the column on which to filter your search. For example, select **Region**. In the **Search** box, enter the value on which to search (for example, **US East (Ohio)** as the region name) and click **Add**. For the **Enabled**, **Key State,** and **Rotation Status** columns, a drop-down menu is available from which to select a value for the search. Click the down arrow, select a value from the available options, and then click **Add**. For the **Creation Date**, **Scheduled Deletion Date**, and **Key Material Expiration Date**, select a start and end date for the search and then click **Add**. The specified search filter is then added to **Active Filters**. In the example, **Region: US East (Ohio)** is added to **Active Filters**. The results of the search displays in the **AWS Keys** table. You can narrow your search by selecting more columns to add to the filter. To clear the filtered results, from **Active Filters**, click on the "X" (or the **Delete** button) for each of the named filters.



The following columns display on the **AWS Keys** page:

- **Key ID**: The globally unique identifier for the CMK.
- **Region:** The AWS Region to which the key is uploaded.
- **Creation Date**: Indicates the date on which the key was created.
- **Enabled**: Indicates whether the key is enabled for use.
- **Key State:** The state of the CMK. The valid values are:
    - Enabled
    - Disabled
    - PendingDeletion
    - PendingImport
- **Scheduled Deletion Date**: Indicates the date on which the key is scheduled to be deleted.
- **Key Material Expiration Date**: Indicates the date on which the key material is set to expire.
- **Key Material Origin**:
    - If the key was uploaded from CCKM, then the value is **INTERNAL(<source_key_name>)**.
    - If the key was not uploaded from CCKM as a BYOK key, then the value is **EXTERNAL**.
    - If the key material was created by AWS KMS, then the value is **KMS**.
- **Rotation Status**: Indicates whether the key is in active rotation schedule or if it has been rotated in the past.
    - Auto—if the autorotation is currently enabled.
    - Active—if the key is the most latest in the rotation sequence.
    - Archived—if the key has been rotated out.

- **Actions**: Allows you to select any of the following actions on a given key:
    - Rotate
    - Delete
    - Schedule Delete
    - Cancel Schedule Delete
    - Import
    - Update

12.5.1 Create a Key

You have two options to create a new key:

- Upload an external key (BYOK)
- Create a native key

12.5.1.1 Upload your Own Key (BYOK)

Upload the AES256 keys created from the **Key Sources** page of CCKM to AWS. These are your BYOK keys.

To upload your BYOK key to CCKM:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 On the **All Keys** list page, select the **New** button and then **Upload**. The **Upload a Key** dialog box displays.

3 From the **Region** box, select the AWS Region to which you will upload the key.

4 From **Key Provider** box, select either **DSM** or **nShield** depending on which key provider you used.

5 From **Source Key** box**,** select one of the available source keys created in CCKM. The keys available in the drop-down menu reflect either DSM or nShield source keys depending on your key provider selection.

6 In the **Alias** box, assign an alias for intuitive key name searching and for use in key rotation. Required by AWS Management Console.

Note: In order to rotate keys, you must assign an alias, when either uploading or using the **Update Key** function.

7 (Optional) Enter the expiration date of the key material in the **Key Material Expiration Date** box in the format MM/DD/YY or select a day from the calendar button at the right of the box.

8 **Description:** (optional) Enter a description for the key you are uploading.

9 (Optional) From **Tags**, enter the tag name and value of the key.

10 (Optional) From **Allow Access for External Accounts,** click **+** and enter the account ID of an external AWS account with which the key is shared. This step is applicable only if you are to share this key with an external AWS account.

11 Click **Upload to AWS**.
(Reset clears the page before it is uploaded, if necessary.)
The uploaded key is displayed in the list.

12.5.1.2 Create a Native Key

Create a native key in AWS KMS using CCKM. After CCKM creates a native key, it displays the key as **KMS** under the **Key Material Origin** column within the **Keys** > **All Keys** list page in CCKM.

To create a native key:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 On the **All Keys** list page, select the **New Key** button and then **Upload**. The **Create Key** dialog box displays.

3 From the **Region** box, select the AWS Region to which you will add the new key.

4 In the **Alias** box, assign an alias for intuitive key name searching and for use in key rotation. Required by AWS Management Console.

Note: In order to rotate keys, you must assign an alias, when either uploading or using the **Update Key** function.

5 (Optional) In the **Description** box, enter a description of the key.

6 (Optional) From **Tags**, enter the tag name and value of the key.

7 (Optional) From **Allow Access for External Accounts,** click **+** and enter the account ID of an external AWS account with which the key is shared. This step is applicable only if you are to share this key with an external AWS account.

8 Click **Create**.
(Reset clears the page before it is created, if necessary.)
The new key is displayed in the list.

12.5.2 Delete a Key

From the **Keys** page, deleting a key deletes it from AWS. AWS requires a minimum wait period of seven days before it fully deletes the entire key from AWS.

To delete a key:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 Click **Delete** next to the target key.
In the Delete Key dialog, choose: a) **Delete key material now** or
b) **Schedule to delete** key after <enter between 7 -30> number of days.

If you chose a) **Delete key material now**, then the **Action** column for the key will include a **Schedule Delete** link. While the key material will have been deleted from AWS, you still must enter a time of 7-30 days after which the remaining key metadata will be removed. Until that scheduled deletion, you can still Import the key again to AWS. The Action column will also show an **Import Key** link, whereby you could re-import the key to AWS before the scheduled deletion is completed.

If you chose b) **Schedule to delete key after <7-30> number of days**, then nothing is deleted from AWS until the designated time frame, then the key material and metadata will be deleted simultaneously and cannot be reimported. During the "Pending" period, you can **Cancel Delete** from the Action column.

3 Type or copy/paste the confirmation phrase in the Warning box.

4 Click **Delete** to confirm.
The full deletion will occur after the 7-30 period has passed. Until then, the Action list for the key will include **Import** key and **Schedule Delete** (if applicable).

12.5.3 Schedule Delete

See section 12.4.2, Delete a Key. If you chose “Delete key material now,” use the **Schedule Delete** link to a Pending period and finalize the deletion.

12.5.4 Import a Key

See section 12.4.2, Delete a Key. If the source key originated in CCKM and the key material has been deleted from AWS but the full deletion has not completed, use **Import** to re-import the key from CCKM to AWS.

12.5.5 Update Key

Use **Update** to do the following:

- Create and/or assign an alias to a key.
- Adjust the Enabled status, Description, or Tags.
- Add the account ID of an external AWS account with which the key is shared.
- Add an AWS key policy used to share the key with an external AWS account.

Note: Keys cannot be updated when they are in a PendingImport or PendingDeletion state.

To update a key:

1 On the **Keys** list page (Keys View or Alias View), click **Update** in the Action column for a particular key.

2 From the **Key Properties** tab, select an available alias or enter a new alias name in the **Alias** dialog. Click **+.**

3 Adjust the **Enabled** check box, **Description**, or **Tags**.

4 Click **Update**.

5 (Optional) From the **Key Policy** tab, in **Default View**, in **Allow Access for External Accounts,** click **+** and enter the account ID of an external AWS account with which the key is shared. This step is applicable only if you are to share this key with an external AWS account and you have not yet customized the AWS key policy.

You also have the option to enter an AWS key policy that allows you to share this key with an external AWS account. From the **Key Policy** tab, within **Key Policy** in **Policy View**, enter the AWS key policy copied from the associated JSON file.

6 Click **Save Accounts**.

12.5.6 Rotate Key

CCKM allows for a manual rotation and an autorotation of a key. You manually rotate a key by adding a new key (with the same alias name) using the **Rotate Now** tab in the **Rotate AWS Key** dialog box. Auto rotating a key is adding a new key (with the same alias name) at a scheduled time. Key autorotation requires that you first schedule the autorotation in the **Schedules** page prior to enabling this feature in the **Auto Rotate** tab in the **Rotate AWS Key** dialog box. Note that only source keys that have not yet been uploaded to AWS can be used as new key material in a key rotation.

Note: Any key that is shared among different IAM users within a *single* AWS account and is scheduled for autorotation by one IAM user cannot be scheduled for autorotation again by another IAM user. Any key that is scheduled for autorotation displays the status of “AUTO” within the **Rotation Status** column of the **Keys** page. If you attempt to

schedule autorotation using **Rotate Key** within the **All Keys** list with a key already scheduled, a message displays indicating that the key is already scheduled for rotation with the scheduled interval, date, and time at which the rotation is to take place. In addition, the name of the user responsible for scheduling the rotation displays.

12.5.6.1 Rotate a Key Manually

To rotate a key manually:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 From the ***All Keys*** list, choose the **Alias** or **Keys** view.

3 Find the row that holds the key you wish to rotate and select **Rotate Key** from the **Actions** column.
The **Rotate AWS Key** dialog box is displayed.

4 Select the **Rotate Now** tab.

5 From **Key Provider** box, select either **DSM** or **nShield** depending on the source of the key you are to rotate.

6 Select a new source key from the **Source Key** list.
Only source keys that have not yet been uploaded to AWS are listed. The **Aliases** box displays the alias associated with the selected source key.

7 Deselect the **Disable Encrypt permission on Current key** check box, if you want to continue using the key that has been rotated out to encrypt new data. By default, this check box is selected. Best practice is not to use the old key that has been rotated out to encrypt new data. Strictly maintain the rotated-out key so that KMS can continue using it to decrypt the previously encrypted data by that key.

8 (Optional) Enter a description of the new CMK key in the **Description** box.

9 Click **Rotate**.

12.5.6.2 Enable Autorotation of Key

To enable autorotation of a key:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 Choose **Alias** or **Keys** view. The **Rotate Key** link is displayed in the **Actions** column of applicable keys.

3 Select **Rotate Key** from the drop-down menu. The **Rotate AWS Key** dialog box is displayed.

4 Select the **Auto Rotate** tab.

5 For **Enable Auto Rotation on this key**, select **On** to enable the autorotation of this key.
Note: If the toggle for **Enable Auto Rotation on key expiration** is displayed, skip it. This option displays only if the key is scheduled for autorotation before it expires. This option is not applicable to enabling autorotation of the key.

6 From **Key Provider** box, select either **DSM** or **nShield** depending on the source of the key you are to rotate.

7 Deselect the **Disable Encrypt permission on Current key** check box, if you want to continue using the key that has been rotated out to encrypt new data. By default, this check box is selected. Best practice is not to use the old key that has been rotated out to encrypt new data. Strictly maintain the rotated-out key so that KMS can continue using it to decrypt the previously encrypted data by that key.

8 Click **Submit**.

12.5.7 Synchronize Keys

First login to the CCKM AWS automatically synchronizes keys from all AWS regions. If you make a key status change in AWS, you must synchronize to see the latest status in the CCKM GUI. If you schedule key deletion, then after the designated time period has passed, it will delete from AWS. Click **Synchronize** to remove it from the CCKM/DSM permanently.

1 In the CCKM portal, in the left navigation, select **Keys**.

2 Click **Synchronize**. The warning box is displayed; click **Synchronize** again.
Any new keys created in AWS are retrieved and appear in the UI

12.5.7.1 Enable Autorotation before Key Expiration

To enable autorotation of a key before it expires, you must first set up a key expiration schedule in the **Schedules** page. Once the key is enabled for autorotation, the key will be rotated before it expires. After the key is auto rotated, the key will have a new expiration date based on the key-expiration schedule.

To enable autorotation of a key before it expires:

1 In the CCKM portal, in the left navigation, select **Keys**.

2 Choose **Alias** or **Keys** view. The **Rotate Key** link is displayed in the **Actions** column of applicable keys.

3 Select **Rotate Key** from the drop-down menu. The **Rotate AWS Key** dialog box is displayed.

4 Select the **Auto Rotate** tab.

5 For **Enable Auto Rotation on key expiration**, select **On** to enable the autorotation of this key before it expires.
Note: Skip the **Enable Auto Rotation on this key** as it is not applicable to enabling autorotation of this key before it expires.

6 From **Key Provider** box, select either **DSM** or **nShield** depending on the source of the key you are to rotate.

7 Deselect the **Disable Encrypt permission on Current key** check box, if you want to continue using the key that has been rotated out to encrypt new data. By default, this check box is selected. Best practice is not to use the old key that has been rotated out to encrypt new data. Strictly maintain the rotated-out key so that KMS can continue using it to decrypt the previously encrypted data by that key.

8 Click **Submit**.

12.5.8 Export Keys

Export Keys allows you to export the contents of the Keys list into a report in a CSV format.

To export your **Keys** list:

1 In the left navigation bar, click **Keys**.

2 Click **Export Keys**.
This will download all of the keys in a report in a CSV format.

12.6 Reports

It is necessary to set up the CloudTrail and CloudWatch services in AWS before AWS keys will be represented in the CCKM reports. You also have the option to set these services up in CCKM using the **General** tab in the **Settings** page. Refer to section 12.6.1*,* Configure CloudTrail and CloudWatch in AWS to Use CCKM reports*,* for more information.

Choose **Reports** in the left-hand navigation to access the following reports:

- Combined Key Activity Reconciliation Report
  Click **Run Report** to generate a new list of events connected to a key.
- **Key Activity Report**
  Illustrates the activity type performed by a specific user and at the stated time.
- **Key Service Usage Report**
  Indicates the:
  - Key name
  - Origin: AWS KMS or External
  - Requesting Service: Name or ID of the service using a key
  - Region
  - Request Timestamp: Time when the application used the key.
- Key Aging Report
  Provides information about key expiration using the following columns:
  - Key Name
  - Region
  - Scheduled key deletion date
  - Key expiration: Date and time at which this key expires.
- Cloud Key Manager User Action Report
  Lists the users and their actions in creating and deleting keys in the CCKM portal.

12.6.1 Configure CloudTrail and CloudWatch in AWS to Use CCKM reports

CCKM uses AWS CloudTrail and CloudWatch services to generate reports.

12.6.1.1 Overview

AWS CloudTrail

- Actions taken by a user, role, or an AWS service are recorded as events in CloudTrail.
- Actions performed on or by AWS KMS keys are also recorded in CloudTrail.
- CCKM uses these audit records to generate 'Key reconciliation report', 'Key service usage report' and 'Key activity report'.

AWS CloudWatch:

- CloudWatch allows the querying and filtering of CloudTrail logs through log groups.
- CCKM uses CloudWatch log group to fetch KMS audit logs and generate reports.

12.6.1.2 Configure CloudTrail

1 Log in the AWS Management Console.

2 **Important: Select any region. Note down this region.**
This region will be provided in the CCKM in the last steps.

3 Go to the **CloudTrail** section in **Management Tools**, select **Trail**, and click **Create Trail**. Create trail in CloudTrail with following settings:

a. Apply trail to all regions
b. 'Read/Write events' in Management events
c. Skip data events section.
d. Select any Storage location as per your convenience for logs collected by CloudTrail.

https://docs.aws.amazon.com/awscloudtrail/latest/userguide/cloudtrail-create-and-update-a-trail.html

12.6.1.3 Configure CloudWatch Log Group

Monitor the new trail using a CloudWatch log group

1 Configure CloudWatch Logs to receive your logs from CloudTrail so that you can monitor for specific log events.

2 Edit trail created in the previous section.

3 Enter a new or existing CloudWatch log group name. **Note down this log group name.**

https://docs.aws.amazon.com/awscloudtrail/latest/userguide/send-cloudtrail-events-to-cloudwatch-logs.html

12.7 Schedules

The **Schedules** page allows Cloud administrators to view and schedule the following job types:

- Key Rotation
- Key Synchronization
- Key Expiration

At the top right of the **Schedules** page, the **Add Schedule** button is available from which to add a scheduled job type.

12.7.1 Schedule Key Rotation

CCKM allows for a scheduled rotation of an AWS key. A new version of the key is added to the AWS key during a scheduled rotation. Scheduled rotation requires that this feature be configured in the **Schedules** page prior to enabling the feature in the **Auto Rotate Key** dialog box (from the **Actions** column of the *All Keys* list). Your AWS login password is required as part of the configuration and is saved to DSM. During a scheduled rotation of a key, CCKM uses the associated username and password to rotate the key. If the password expired or changed, then the new password must be reentered. Otherwise, the scheduled rotation will fail.

You can pause or delete a scheduled key rotation in the **Schedules** page. Note that when a key rotation schedule is deleted from CCKM, then the password is also deleted from DSM.

12.7.1.1 Set up a Key Rotation Schedule

To set up a key rotation schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

3 From the **Job Type** drop-down menu, select **KEY_ROTATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 For **Schedule**, set a schedule for the rotation by selecting **Basic** (to enter a date and interval for the schedule) or Advanced (to enter a Cron Expression for the schedule).

7 If you selected **Basic**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

   Set the repeat interval of the key rotation by entering a number in the Repeat Interval and then selecting Day, Week, or Month from the drop-down menu. The supported ranges are 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter.

8 If you selected **Advanced**, enter the cron expression in the **Cron Expression** box.

9 Enter your AWS key credentials in the **AWS Access Key ID** and **AWS Secret Access Key** boxes or click **Choose file** from **Upload Credentials** to upload an AWS credentials file that contains both the AWS Access Key ID and AWS Secret Access Key.

10 Click **Save** to save setting.

12.7.1.2 Pause a Key Rotation Schedule

To pause a key rotation schedule, select **Yes** for **Pause** within the **Add Schedule** dialog box and then click **Save**. The setting is saved. To resume the rotation schedule after pausing it, select **No** for **Pause** and then click **Save.** The setting is saved.

12.7.1.3 Delete a Key Rotation Schedule

To delete a key rotation schedule, select **Delete** from the **Actions** drop-down menu from the **Schedules** page. Note that password associated with the given key rotation schedule is also deleted from DSM.

12.7.2 Schedule Key Synchronization

CCKM allows for a scheduled synchronization of AWS keys. Synchronizing downloads any keys that were created in AWS, outside of the CCKM portal, into your CCKM. Scheduled key synchronization requires that this feature be configured in the **Schedules** page. You can also pause or delete a scheduled key synchronization in the **Schedules** page. Note that when a key synchronization schedule is deleted from CCKM, then the password is also deleted from DSM.

12.7.2.1 Set up a Key Synchronization Schedule

To set up a key synchronization schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

3 From the **Job Type** drop-down menu, select **KEY_SYNCHRONIZATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 For **Schedule**, set a schedule for the key synchronization by selecting **Basic** (to enter a date and interval for the schedule) or **Advanced** (to enter a Cron Expression for the schedule). If you selected **Basic**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

   Set the repeat interval of the key synchronization by entering a number in **Repeat Interval** and then selecting Day, Week, or Month from the drop-down menu. The supported ranges are 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter. If you selected **Advanced**, enter the cron expression in the **Cron Expression** box.

7 Enter your AWS key credentials in the **AWS Access Key ID** and **AWS Secret Access Key** boxes or click **Choose file** from **Upload Credentials** to upload an AWS credentials file that contains both the AWS Access Key ID and AWS Secret Access Key.

8 Click **Save** to save setting.

12.7.3 Schedule Key Expiration

CCKM allows for a scheduled key expiration of AWS keys in the **Schedules** page. You can also pause or delete a scheduled key expiration in the **Schedules** page. Note that when a key expiration schedule is deleted from CCKM, then the password is also deleted from DSM.

12.7.3.1 Set up a Key Expiration Schedule

To set up a key expiration schedule:

1 In the CCKM portal, in the left navigation, select **Schedules**. The **Schedules** page displays.

2 On the **Schedules** page, select the **Add Schedule** button. The **Add Schedule** dialog box displays including your username in the **Username** box.

3 From the **Job Type** drop-down menu, select **KEY_EXPIRATION**.

4 (Optional) Enter a description of the schedule in the **Description** box.

5 For **Pause**, select **No**.

6 From **Start Date**, select a start date and time from a date and time selector by clicking the calendar button to the right of the box. The time is based on a 12-hour clock and the A.M./P.M. modifiers.

7 From **Expire a new key after**, enter a number and select the interval unit of Hour, Day, Week, Month, or Year from the drop-down menu after which the new key expires. The supported ranges are 6 through 24 hours, 1 through 30 days, 1 through 4 weeks, and 1 through 12 months. You can increase or decrease the number you enter.

8 Enter your AWS key credentials in the **AWS Access Key ID** and **AWS Secret Access Key** boxes or click **Choose file** from **Upload Credentials** to upload an AWS credentials file that contains both the AWS Access Key ID and AWS Secret Access Key.

9 Click **Save** to save setting.

12.8 Logs

CCKM maintains logs for events, such as login, key generation, synchronizing of keys, and deletion of keys, and displays these logs in the **All Logs** page. For each logged event listed on the **All Logs** page, the following information is displayed:

- Event name
- Severity level
- Date and time of the event
- Event message
- User (name) associated with the event

You can search for a specific logged event from the **All Logs** page using the **Search** box. Allowable filters on the search are event name, severity level, event message, and user (name).

12.9 Settings

To access the **Settings** page in CCKM, select **Settings** from the left-hand navigation. From the **Settings** page, you can:

- Enter the emails of the recipients who are to receive alerts and reminders.
- Set up key alerts.
- Enable reporting for AWS keys using an existing AWS CloudTrail and CloudWatch log group or a new trail and log group that you create (from the **Settings** page).
- Configure a remote syslog server to which to send syslog messages.
- View the release number of the CCKM you are running.

12.9.1 Adding Recipients' Emails (for Alerts and Reminders)

Use the **General** settings tab to enter the emails of the recipients who are to receive alerts and reminders from the system. Enter the recipients' email addresses separated by a comma in the **Alert Format** box and then click **Save** to save setting.

12.9.2 Enabling Alerts and Reminders

Use the **Keys Alert** tab from the **Settings** page to specify whether to send an alert to the specified user email(s) when an AWS key is deleted, restored, or uploaded in AWS. You can also use this tab to set up a reminder to manually rotate (or rekey) a key along with the frequency (in days) at which the reminder is sent to the specified user email(s). The user emails are set up in the **General** tab. By default, no alerts or reminders are sent.

To set up key alerts or a reminder to manually rotate (or rekey) a key:

1 In the **Settings** page, select the Keys Alert tab.

2 Under **Alerts**, click On for each of the alert types and reminder you wish to enable:

- Rekey Alert (reminder)
- Delete AWS Key
- Import AWS Key
- Upload AWS Key

3 In the **Days** box, enter the number of days for the frequency a reminder email is sent to rotate a key.

4 Click **Save** to save the setting.

12.9.3 Enabling Reporting for AWS Keys

Use the **General** tab within the **Settings** page to enable CCKM reporting for AWS keys using an existing AWS CloudTrail applicable to an AWS region and a CloudWatch log group to which the trail belongs. You can also use the **General** settings tab to create a new CloudTrail and CloudWatch log group to which the trail belongs. You first select an AWS region in which to create it and then add the name of a new CloudWatch log group. Thereafter, you configure the other required configurations.

12.9.3.1 Enabling Reporting for AWS Keys Using an Existing CloudTrail and CloudWatch Log Group

To enable CCKM reporting for AWS keys using an existing CloudTrail and CloudWatch log group:

1 In the **Settings** page, select the **General** tab. The **General** page displays.

2 From the **Select CloudTrail Region** drop-down menu, select the AWS region to which the trail applies.

3 From the **Select CloudWatch Log Group Name** drop-down menu, select the name of the log group to which trail belongs.

4 Click **Save** to save setting.

12.9.3.2 Enabling Reporting for AWS Keys Using a new CloudTrail and CloudWatch Log Group

To enable CCKM reporting for AWS keys using a new CloudTrail and CloudWatch log group to which the trail belongs:

1 In the **Settings** page, select the **General** tab. The **General** page displays.

2 From the **Select CloudTrail Region** drop-down menu, select the AWS region in which to create the new trail.

3 From the **Select CloudWatch Log Group Name** drop-down menu, enter the name of the new CloudWatch Log Group to create. This log group receives your logs from CloudTrail so that you can monitor for specific log events.

Entering a name of the new log group triggers the display of other configuration boxes associated with creating a new trail and log group.

4 From the **Stack Name** box, enter the name of the new stack to create. The AWS CloudFormation stack automates the creation of all resources required for creating a new trail (bucket, trail, and log group). If a resource fails, the stack rolls back to the previous version. From AWS CloudFormation, you can view information about any failure during the process of the creating the stack using the name of the stack. You can also delete the stack from CloudFormation without any impact to your AWS resources.

5 From the **S3 Bucket Name** box, enter the name of an S3 bucket to designate as the storage of the trail log files. The name must be globally unique. If you do not enter a bucket name, the default bucket name of **cckm-bucket** is used.

6 From the **Cloud Trail Name** box, enter the name of the new trail. If you do not enter a trail name, the default trail name of **cckm-bucket** is used.

7 From the **CCKM IAM Role Name** box, enter the name of the AWS IAM Role used for CCKM. If you enter an existing IAM role name, then CCKM uses this name to apply to the configuration. If you enter a new IAM Role name, CCKM creates this role with the required permissions to create the resources for the trail. If you do not enter an IAM role name for CCKM, the default role name of **AWS-<*region-name*>-cckm-cloudtrail-to-cloudwatch** is used.

Note: If you do not have the permissions to create a new IAM role, contact your AWS administrator to create the IAM role with permissions to create and delete log groups, trails and buckets.

8 Enter your AWS key credentials in the **AWS Access Key ID** and **AWS Secret Access Key** boxes or click **Choose file** from **Upload Credentials** to upload an AWS credentials file that contains both the AWS Access Key ID and AWS Secret Access Key.

9 Click **Save** to save setting.

12.9.4 Configuring Syslog Server

Use the **Syslog** tab from the **Settings** page to configure a remote syslog server to which to send syslog messages. Note that the default port number of 514 and the facility name of “LOCAL1” for the syslog server display in the **Port** and **Facility** boxes, respectively. These boxes are not available for modification.

12.9.4.1 Configure Remote Syslog Server

To configure a remote syslog server:

1 In the **Settings** page, select the **Syslog** tab. The **Syslog** dialog box displays.

2 Enter the hostname or IP address of the syslog server in the **Hostname** box.

3 (Optional) Enter a description of the syslog server in the **Description** box.

4 Click **Save** to save the setting.

12.9.4.2 Delete Remote Syslog Server Configuration

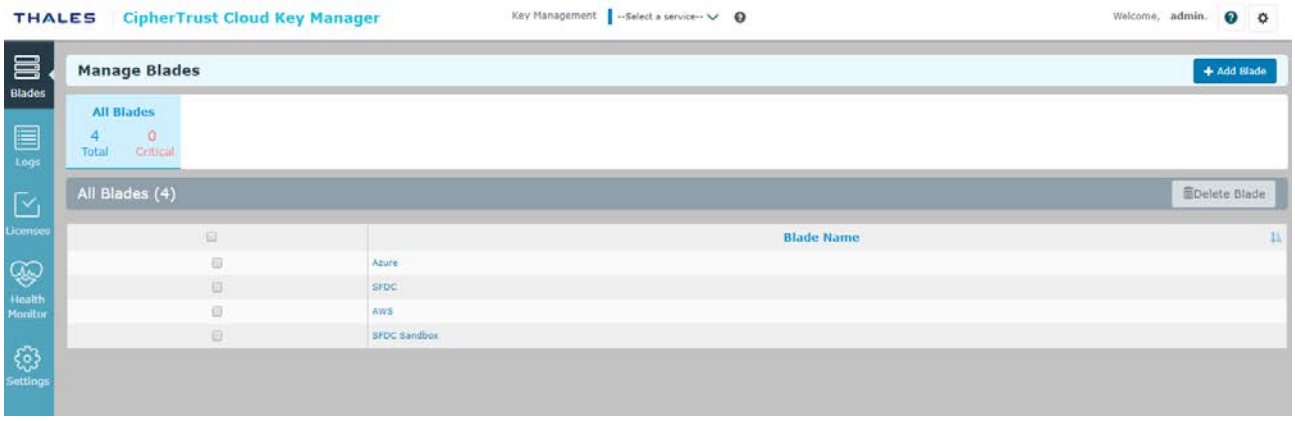
To delete the configuration of a remote syslog server, click **Delete** from the **Syslog** dialog box. No syslog messages are sent to the remote syslog server after the deletion of this configuration.

12.9.5 About

Use the **About** tab to view the release number of the CCKM you are running.

13 Maintenance Tools for Administrators

The CCKM Admin portal contains the following options in the left-hand navigation bar:



- **Blades:** From the **Blades** page, you can create a connection (key management security blade) between CCKM and a given cloud service to permit CCKM access to any of the cloud-service users with valid user credentials. From this page, you can also delete a security blade.
- **Logs:** The **Logs** page lists individual logs that can be searched, sorted, and viewed.
- **Licenses:** In **Licenses**, you can view all of your available licenses for your installed and configured Key Management services.
- **Health Monitor:** The **Health Check Monitor** page provides a snapshot of the status of all components in the CCKM environment.
- **Settings:** The **Settings** page allows you to configure WebAPP, DSM, nShield Connect, proxy settings, and user management.

Log in to CCKM Admin portal to access the administrator maintenance tools. To do so, log in to the CCKM GUI using the credentials defined in section 5.3, Wizard Step 3: Set CCKM Root Admin Password. For more information on how to log into the CCKM GUI, see section 9.1.1, Accessing CCKM as an Administrator.

13.1 Manage Blades

Select **Blades** from the left-hand navigation bar within the CCKM Admin portal to access the **Blades** page.

For information on how to create a key management security blade, see the following sections depending on the cloud service for which you wish to create a blade:

- Chapter 6, Create an Azure Key Management Blade
- Chapter 7, Create a Salesforce Key Management Blade
- Chapter 8, Create an AWS Key Security Blade

13.2 Review Logs (for Data Storage Services)

Select **Logs** from the left-hand navigation bar within the CCKM Admin portal to access the **Logs** page.

This page lists audit logs, describing user access, set to Low, Med, or High alert levels.

The CCKM administrator can use these logs to review user and file activity in the system. Note that these logs differ from the logs used by Thales/Vormetric support for deep troubleshooting (see section 15.2, Troubleshooting Logs Files for Thales eSecurity Support Team), and that the CCKM has its own logs, detailed in the appropriate CCKM sections of this guide.

From the **Logs** page, you can:

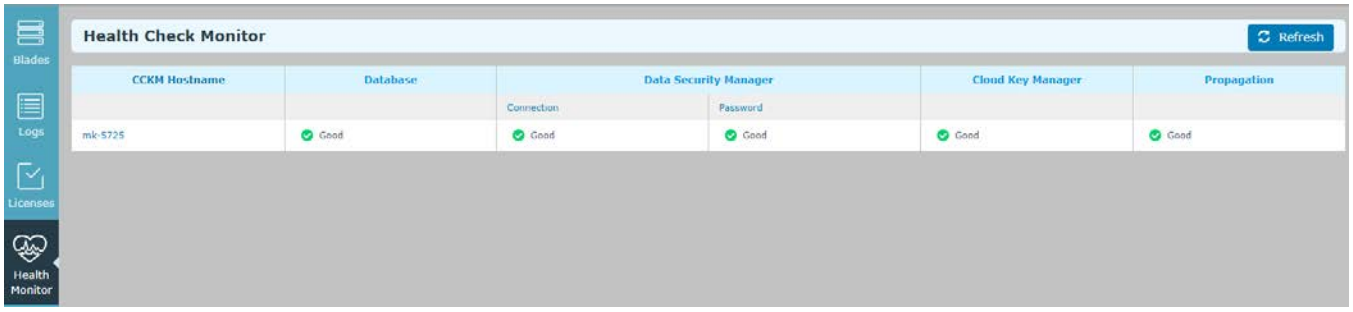
- **Search** by log name or category filter
- **Search** for logs by **Date Range**
- Sort by **Date**, **Event Message**, **Log Type**, **Category**, **Action**, **Where** the activity occurred (which CCKM server), and **Severity** of the alert/Fineness of log capture level.
- Check **More** details to find, for example, uploaded file size, user ID, or timestamp.
- Select a log and **Refresh** it.

13.3 Review Licenses

Select **Licenses** from the left-hand navigation bar within the CCKM Admin portal to access the **Licenses** page where you can view all of the licenses for your installed and configured Key Management services.

13.4 Review Health Monitor

Select **Health Monitor** from the left-hand navigation bar within the CCKM Admin portal to obtain a snapshot of the status of all components in the CCKM environment.



On the **Health Check Monitor** page, the following columns display reflecting each of the major components of the CCKM solution and the status for each component with either a status of “Good” or “Bad”:

- **CCKM Hostname**—The host name of the CCKM instance for which the information is displayed on the **Health Check Monitor** page.
- **Database**—This column indicates the status of the connection between CCKM and the MongoDB server(s). A status of “Bad” indicates the connection between CCKM and the MongoDB server(s) is down or one or more of the MongoDB servers are down.

- **Data Security Manager, Connection**—This column indicates the status of the connection between the CCKM and DSM. A status of “Bad” indicates the network connection between the CCKM and DSM may be down or the DSM may be down. If a status of “Bad” is displayed for this column, check to see whether the DSM is running and reachable from CCKM.
- **Data Security Manager, Password**—This column indicates whether the DSM user password configured in the CCKM matches the DSM user password that is configured within DSM. To check whether this password in CCKM is the same as the one in DSM, the CCKM Health Check Monitor periodically logs into the DSM using the password in CCKM. If the monitor fails to login into the DSM twice, a status of “Bad” is displayed in this column. To change the status to “Good”, you must ensure the DSM user password configured in CCKM is the same as the one configured in the DSM. If the password is not the same, then change it from within CCKM so that these passwords match. For information on how to change the DSM user password from within CCKM, see section 13.5.2, Review DSM Settings or Change DSM Password.

  Note: At a scheduled time (a configured setting in DSM), the DSM Administrator is required to change the DSM user password in the DSM based on security regulations. This change would require an update of this password within CCKM. This is a case when the password in both DSM and CCKM can get out of sync.
- **Cloud Key Manager**—This column indicates the status of the CCKM (software) server running the CCKM user portal. A status of “Bad” indicates the CCKM server is down, which means the CCKM user portal is also down. To investigate this issue, check to see whether the kmaas (CCKM) process is running. If the process is not running, restart it by entering `sudo systemctl restart kmaas` at the command prompt. Also, check the kmaas log file (/var/log/vormetric/kmaas.log) for any error messages. Address the error message(s).
- **Propagation**—This column indicates the status of the CCKM propagation, which is a CCKM mechanism for passing configuration changes to other CCKMs within a multi-CCKM deployment. If you only have a single CCKM running, then the status for this column is “Good”. A status of “Bad” indicates the CCKM propagation is not working. Manually run the Propagate script to aim to resolve the issue. To do so, log into the CCKM appliance console and run the following command as a root user:

  ```
  /bin/bash /opt/vormetric/webapp/postinst/conf/propagate.sh
  ```

  If running this command does not resolve the issue, contact Thales eSecurity Support for assistance on resolving the issue.

13.5 Settings

Select **Settings** from the left-hand navigation bar within the CCKM Admin portal to access the **Settings** page. From the **Settings** page, you can:

- Manage WebApp SSL settings
- Review DSM settings or change the DSM password
- Configure nShield Connect (for Azure and AWS Cloud Services)
- Set or modify the proxy server settings
- (As a CCKM root administrator) manage admin users

Each of these settings are described in the sections that follow.

13.5.1 Manage WebApp SSL Settings

Select **Settings** and then **WebApp** to view the **Configure SSL** page. From the **Configure SSL** page, you can upload a CA-signed SSL certificate and/or key, if you want to replace the self-signed certificate that is delivered with the CCKM.

13.5.2 Review DSM Settings or Change DSM Password

Select **Settings** and then **DSM** to review the DSM properties:

- **Hostname:** the fully qualified hostname for the DSM
- **DSM Domain:** domain name for the DSM
- **DSM Username:** username for DSM admin
- **DSM Password:** password for the DSM admin

13.5.2.1 Change DSM Admin Password

Changing your DSM administrator password in CCKM is required after the DSM administrator has changed the password on the DSM according to its security regulations. Then the CCKM administrator must update the DSM administrator password within the **Settings** > **DSM**.

1 From the **DSM** tab in **Settings**, enter a new DSM administrator password and re-enter to confirm.

2 Click **Update**.
The password is changed.
Note: Use the **Reset** button to clear anything you have entered but not saved while updating the password.

Note: If you do not change your DSM administrator password in the CCKM *after* the DSM administrator has changed this password in the DSM, then the CCKM Health Check Monitor will eventually display a status of "Bad" in the **Data Security Manager, Password** column of the **Health Check Monitor** page. For more information about CCKM Health Check Monitor, see section 13.4, Review Health Monitor.

13.5.3 Configure nShield Connect (for Azure and AWS Cloud Services)

If you are using CCKM to manage keys in Azure or AWS, support for the use of nShield Connect as a key provider is available. For Azure and AWS, CCKM also provides support for the use of the nShield Connect HSM to generate source keys. This section covers how to configure the connection from CCKM to nShield Connect HSM(s), a remote file system (RFS), and the Security World for nShield. This configuration is available within the **Settings** page > **nShield** tab of CCKM. The prerequisites steps you are required to perform are also provided in this section.

After completing the steps successfully in this section, you can proceed to generating an nShield Connect key within the CCKM **Source Key** page.

Note: This documentation assumes that you are familiar with using nShield Connect. This documentation also assumes that you have installed nShield Connect, and the associated RFS and Security World. For information on how to install an nShield Connect, and the associated RFS and Security World, refer to the *nShield® Connect Installation Guide*. For information on how to configure nShield Connect, and the associated RFS and Security World, refer to the *nShield Connect User Guide for Unix*.

13.5.3.1 Prerequisites

Prior to configuring the nShield Connect and the associated RFS and Security World within the **Settings** page, perform the following steps:

- Install and configure nShield Connect, RFS, and Security World.
- Configure CCKM as an nShield Connect client within the nShield Connect HSM by specifying the IP address of CCKM.

For more information regarding these prerequisite steps, refer to Table 1: Setup for CCKM to Connect to DSM, nShield Connect, and MongoDB.

13.5.3.2 Configuring nShield

After successfully configuring the nShield Connect and the associated RFS and Security World in the **Settings** page, CCKM runs a number of nShield Connect commands to enroll the CCKM as a new nShield client to the nShield Connect HSM(s), and updates the Security World configuration from the RFS computer. The output of these commands (provided in links) display on the right side of the **nShield Info** page. Be sure to click on each link within the **nShield Info** page to verify that each command ran successfully. If an error message displays, be sure to address the error. For common issues that may arise during the configuration of the connection from CCKM to nShield Connect and the associated RFS and Security World, see section 13.1.4.3, Troubleshooting nShield Configurations.

To configure nShield Connect and the associated RFS and Security World in the **Settings** page > **nShield** tab, perform the following steps:

1 From **nShield 1 IP Address**, enter the IP address of the nShield Connect.

2 From **nShield 2 IP Address**, enter the IP address of the nShield Connect that is designated as nShield Connect 2 (if you have configured more than 1 nShield Connect).

3 Select **Add More nShield IP** to add and configure more nShield Connect HSMs.

4 From **nShield RFS IP Address**, enter the IP address of the remote file system (RFS).

5 Click **Add to Security World** to enroll CCKM to Security World. A set of links to the configuration command output display. Click on each link to verify that the commands ran successfully. If an error displays, address it before proceeding to the next step.

6 (**Only perform this step, if all of the commands from the previous steps ran successfully.**) Click **Save** to save the configurations. This step will restart the hardserver (Security World Software server) on which the CCKM is installed and configured.

Note: If an error occurred during any of the previous steps, and you click **Save**, you will **not** be able to generate an nShield Connect key within the CCKM **Source Key** page when you attempt to generate it. An error message displays. For common issues that may arise during the configuration of the connection from CCKM to nShield Connect HSM(s) and the associated RFS and Security World, see section 13.1.4.3, Troubleshooting nShield Configurations.

13.5.3.3 Troubleshooting nShield Configurations

Table 3: Basic Troubleshooting nShield Configurations describes a few of the common issues that may arise during the configuration of the connection from CCKM to nShield Connect HSM(s), and the associated RFS and Security World within the **Settings** page and the steps to take to attempt to resolve these issues.

For more information on troubleshooting the configuration for nShield Connect, and the associated RFS and Security World, refer to the *nShield Connect User Guide for Unix*.

Table 3: Basic Troubleshooting nShield Configurations

Issue	How to Resolve
You entered an incorrect IP address for an nShield Connect in Step 1 or Step 2 of section 13.5.3.2, Configuring nShield.	Verify the electronic serial number (ESN) of each nShield Connect. To verify this, ensure that the output of the "rfs-sync --setup" command indicates "Configuration successfully written". From the **nShield 1 IP Address** box of the **Settings** page > **nShield** tab, enter the correct IP address of the nShield Connect and then click **Add to Security World** again.
A network connection from CCKM to nShield Connect cannot be detected.	Verify the ESN of each nShield Connect. To verify this, ensure that the output of the "rfs-sync --setup" command indicates "Configuration successfully written". Also, ensure that the output of the "rfs-sync --update" command includes "Updated module ..." or "Finished synchronization".
You entered an incorrect IP address for the RFS computer in Step 4 of section 13.5.3.2, Configuring nShield.	The output of the "rfs-sync --update" command does not include "Updated module ...". From the **nShield RFS IP Address** box of the **Settings** page > **nShield** tab, enter the correct IP address of the RFS computer and then click **Add to Security World** again.

Only after you have verified that the output of all commands (displayed on the right side of the **nShield Info** page) are correct should you proceed to saving the nShield Connect configurations in step 6 of section 13.5.3.2, Configuring nShield. Saving the configuration restarts the hard server (Security World Software server) on which the CCKM is installed and configured.

13.5.4 Set Proxy Settings

Select the **Proxy Settings** tab from the **Settings** page to set (or modify) the proxy server settings.

Note: In the case of a multi-instance CCKM, it may take up to 5 minutes for the proxy setting changes to become effective. If you edit the proxy settings, it is recommended that you logout and then login again to ensure the settings become effective.

To set the proxy settings:

1 In the **Settings** page, select the **Proxy Settings** tab. The **Modify Proxy Settings** dialog box displays.

2 Enter the hostname or IP address of the proxy server in the **Hostname** box.

3 Enter the port number of the proxy server in the **Port** box.

4 In the **Skip Proxy List** box, enter the hostnames (separated by a pipe symbol “|”) that are to pass through the proxy server.

5 (Optional) Enter the username of the proxy port in the **Username** box.

6 (Optional) Enter the password for the proxy server in the **Password** box.

7 Click **Save** to save the setting.

13.5.5 Manage Admin Users

Select the **User Management** tab from the **Settings** page to manage the admin users.

The CCKM root administrator has the ability to create, enable, and disable additional administrators from the **User Management** tab. These administrator users cannot create users but otherwise have the same access to CCKM as the root admin. Administrator users cannot be deleted.

13.5.5.1 Create Admin User

1 Log in to CCKM as root administrator (login admin, password as defined), select the **Settings** icon, and choose the **User Management** tab.

2 Click **Add User**.
The **Add User** dialog box is displayed.

3 Enter the required and optional fields:

- **Username:** required
- **Password/Confirm password:** required. Must be 8 to 20 characters, with at least one capitalized letter, one lowercase, and one digit.
- **First Name, Last Name, Email:** optional

4 Select the **Active** check box (unless this user should be disabled).

5 Click **Submit**.

13.5.5.2 Disable Admin User

There are two places from which to disable an administrator user:

1 From the **User Management** page, click **Disable** by the user's name.

2 From the **User Management** page, click **Update** by the user's name.
On the resulting page, you can change First Name, Last Name, email, and select or clear the **Active** check box.

14 Backup and Restore

14.1 Backup and Restore Recommendation

1 Both MongoDB and DSM should be backed up at the same time.

2 CCKM should be inactive when this backup is performed to avoid data corruption.

3 Daily backups of MongoDB and DSM are recommended.

14.2 Backup, Restore, and Failover for DSM

CCKM requires system-level backup of the DSM. The DSM documentation includes details on each topic: backup, restore, and failover.

Note: In a High Availability (HA) setup of CCKM, if the primary DSM server goes down, the administrator must manually designate the secondary server as primary. Until then, `WRITE` calls to the server will fail. (`GET` calls will still succeed.)

For details, refer to the following chapters in the Vormetric Data Security Manager (DSM) Administrator's Guide:

- "Backing Up and Restoring the DSM"
- "Configuring High Availability," specifically the section "Converting a Failover DSM to a Primary DSM."

14.3 Backup and Restore for MongoDB

1 Make sure not to use CCKM during backup.

2 Use following commands to backup data using mongodb dump utility:

```
mongodump -u kmaas_user -p {password} --db kmaas --out kmaas.dump
mongodump -u vcg_user -p {password} --db vcg_db --out vcg.dump

tar cvf backup.tar kmaas.dump vcg.dump
```

Move this tar file to reliable storage.

3 Use following commands to restore data to new MongoDB instance.

- Deploy new mongodb instance.
- Configure mongodb by following same steps that you used to configure earlier mongodb
  - Steps for configuring MongoDB
- Empty contents of vcg_db and kmaas databases.
  - You can use db.dropDatabase() to remove db from MongoDB.

- Create '{'_cls' : 'User.CustUser',"username" : "admin", "is_active" : false}' in vcg_db
  - Use following command on mongodb console

```
use vcg_db
db.user.insert({'_cls' : 'User.CustUser',"username" : "admin",
"is_active" : false})
```

- Use following command to restore previously backed-up data.

```
tar xvf backup.tar
mongorestore -u kmaas_user -p {password} --db kmaas kmaas.dump/kmaas
mongorestore -u vcg_user -p {password} --db vcg_db vcg.dump/vcg_db
```

15 Troubleshooting

15.1 How to Enable SSH

If SSH is not enabled by default, it can be enabled in CCKM before and after configuring CCKM. Before running the CCKM configuration wizard, but after the pre-setup steps for MongoDB, run the following REST API to turn on SSH:

```
curl -k -X POST https://$ceg/vcg/rest/v1.0/vcg/configuration/ssh/start
```

After Configuring Gateway, run the following command to get the token:

```
curl  -k –X POST https://$ceg/vcg/api-token-auth -d
“username=admin&password=$ADMIN_PASSWORD”
```

Look at the output and find the token. Then run the following curl command to enable SSH. Replace "$token" with the token got from the last command

```
curl –k –X POST https://$ceg/vcg/rest/v1.0/vcg/configuration/ssh/start -H
“Authorization: Token $token”
```

15.2 Troubleshooting Logs Files for Thales eSecurity Support Team

The following table lists logs that Thales eSecurity Support may request that you collect and send in case of troubleshooting needs.

Log File Name	Generator	Description	Frequency
/var/log/vormetric/kmaas.log *	CCKM	All logs created in CCKM (including audit logs)	Whenever CCKM is used.
/var/log/vormetric/webapp.log	Web GUI / WebApp	Logs from the web application. Configuration changes and user initiated action, REST API logs	request frequency
/var/log/vormetric/vor_health_check.log	Health Check Module	Logs capturing the status of all the modules (c-icap, database, webapp)	2 minutes.

* This file is set to finer or grosser levels using the Log Level (Debug/Info/Error) dropdowns from the **Logs** page of the CCKM UI.

15.3 Restrictions

The following restrictions apply to the current version of CCKM:

- Azure key vault name and key name combined cannot exceed 64 characters.
- The DSM domain name cannot exceed 22 characters.

Appendix A, CCKM CLI Commands

The CCKM Command Line Interface (CLI) enables you to configure the CCKM network and perform other system-level tasks using the CCKM CLI commands.

This appendix provides a high-level overview of each of the command categories that are available in the CCKM. For information about the details of each command and its options, run the command with the "--help" option within the CCKM CLI.

A.1 CCKM CLI Navigation

The following are the supported categories of CCKM CLI commands:

- Maintenance
- Network
- System
- User
- Applog

As a CCKM CLI administrator, log on to the CLI, and then enter a command category by typing the category name at the command line prompt. For example, type `system` to enter the system category. While in the category, you can execute the commands for that category.

Enter the entire category name, command, or argument, or enter just enough characters to identify uniquely the category, command, or argument. You can use the <Tab> key to complete a category, command, or argument. Enter enough characters to uniquely identify a category, command, or argument, and then press the <Tab> key. The CLI will complete it for you.

For example, at the top level, enter `n` and press <Tab>, the CLI expands it to `network`.

Inside the network category, you can enter `d`<Tab> and it expands to `dns`. Type `n`<Tab> and it expands to `netstat`. Note that there are other commands in the network category that start with n, such as nslookup, and ntpservice. To ensure to expand to netstat command, enter `ne` to identify it uniquely.

Other supported CLI navigation methods are:

- Enter a question mark (?) to display the next command or argument that is expected. Think of it is as a shorthand form of help.
- Enter "`up`" to return to the top level so that you can enter another category. You can enter another category only from the top level.
- Enter "`exit`" at any time to end the current CLI session.

A.2 Maintenance Commands

The CCKM maintenance commands enable you to upgrade the CCKM and restore the most recent backup of the CCKM data to the CCKM. Table 4 lists the supported CCKM maintenance commands:

Table 4: CCKM Maintenance Commands

Command	Description
restore	Restores the most recent backup of the CCKM data to the CCKM.
upgrade	Upgrades the CCKM. **Note**: A backup of the CCKM data is saved to the CCKM as part of the upgrade process.
quit|q|up|<ctrl-d>	Quits or returns to the previous menu.
Exit	Exits the application.

Note: Before running the restore command, you are required to backup the CCKM data. Otherwise, the restore command fails if the backup data is not available. To perform a backup of the CCKM data, use the upgrade command, which saves a backup to the CCKM server. You cannot access or download this backup.

Note: You can also use the backup command within the database. However, the location of this backup is not saved to the CCKM server. If you run the restore command on the CCKM using only this method of backing up, the restore command will fail because the backup is not saved to the CCKM.

Warning: **The restore command rollbacks any data changes made since the backup was taken.**

A.3 Network Commands

The CCKM network commands are used to set, modify, or delete IP addresses on the CCKM, set up DNS servers, and perform other network-related tasks. Table 5 lists the supported CCKM network commands:

Table 5: CCKM Network Commands

Command	Description
checkport	Checks port connection status of a host.
dns	Shows or configures the DNS settings for the CCKM.
ip	Shows or configures the CCKM network interface settings.
netstat	Prints network connections.
nslookup	Queries Internet name servers.
ntpservice	Configures NTP service settings.
ping	Pings an IP address, host name, or FQDN.
route	Sets a static route.
service	Starts, stop, or restart a network service or gets the status of a network service.
set	Configures a network device interface.

setup	Sets up a network configuration.
show	Shows network device configuration.
traceroute	Traces route to an IP address or hostname.
quit|q|up|<ctrl-d>	Quits or returns to the previous menu.
exit	Exits the application.

A.4 System Commands

The CCKM system commands enable you to configure the CCKM host settings (such as, hostname or timezone), and view and manage other common system administration tasks for the server (such as, reboot or shut down, or view currently logged-in users). Table 6 lists the supported CCKM system commands:

Table 6: CCKM System Commands

Command	**Description**
date	Prints or sets the system date and time.
host	Performs DNS lookups.
hostname	Shows hostname of the CCKM.
hosts	Updates the host file.
lastlogin	Shows the last logged-in date for each user on the system.
reboot	Reboots the CCKM appliance.
security	Configures the CCKM security features including the certificates.
server	Manages the CCKM server.
shutdown	Stops the CCKM software and powers off the virtual appliance.
timezone	Sets or shows the system timezone.
uptime	Shows how long the CCKM has been running.
version	Shows the CCKM version information.
who	Shows information about the users who are currently logged into the CCKM appliance using CLI.
quit|q|up|<ctrl-d>	Quits or returns to the previous menu.
exit	Exits the application.

A.5 User Commands

The CCKM user commands enable you to add, modify, delete CCKM CLI users, and display information about these users. Table 7 lists the supported CCKM user commands:

Table 7: CCKM User Commands

Command	Description
add	Adds a new CLI user.
delete	Deletes a CLI user.
modify	Changes the password of a CLI user.
show	Lists all of the configured CLI users.
quit|q|up|<ctrl-d>	Quits or returns to the previous menu.
Exit	Exits the application.

A.6 Applog Commands

The CCKM applog commands enable you to upload debug logs from the CCKM to the remote servers using the secure copy protocol (SCP) to transfer the log files between the two hosts. Table 9 lists the supported CCKM applog commands.

Table 8: CCKM Applog Commands

Command	Description
upload	Uploads logs to remote server using SCP.
quit|q|up|<ctrl-d>	Quits or returns to the previous menu.
Exit	Exits the application.