Keyfactor Command 11.0

Release Notes

Table of Contents

List of Tables

List of Figures

1.0 Introduction

The *Keyfactor Command Documentation Suite* includes:

- *Keyfactor Command Reference Guide*
- *Keyfactor API Reference Guide*
- *Keyfactor Command Server Installation Guide*
- *Keyfactor Orchestrators Installation and Configuration Guide*
- *Keyfactor Command Release Notes & Upgrading*

In addition, Keyfactor offers documentation for products that are not part of the *Keyfactor Command Documentation Suite*, including the *Keyfactor Command Upgrade Overview* and installation guides for third-party CA gateways that interface with Keyfactor, which are available upon request.

2.0 Release Notes & Upgrading

The Keyfactor Command suite of documentation is released as both major releases, with version numbers ending in zero, and minor releases, with incremental fixes and updates following the major release. When reviewing release notes, be sure to review those for both the minor releases and their corresponding major release.

Upgrade instructions are included for Keyfactor-hosted and self-hosted installation (see Upgrade Overviews below).

2.1 Upgrade Overviews

The *Keyfactor Command Upgrade Overview* is provided in two formats for different users. Follow the appropriate upgrade instructions for your configuration.

- Upgrade Overview - Keyfactor-Hosted below

  Use this guide if you do any of the following:

  - Consume Keyfactor Command certificate lifecycle automation as a service hosted by Keyfactor
  - Consume a managed PKI hosted by Keyfactor

- Upgrade Overview - Self-Hosted on page 6

  Use this guide if you have any of the following:

  - Have deployed Keyfactor Command on premise in your data center or cloud
  - Have deployed the Keyfactor CA Policy Module and associated CA policy handlers on premise in your data center or cloud

2.1.1 Upgrade Overview - Keyfactor-Hosted

The Keyfactor Command solution by Keyfactor allows organizations to issue and manage certificates across enterprise infrastructures. For a comprehensive description of the components that make up Keyfactor Command, see Logical Architecture in the *Keyfactor Command Server Installation Guide* and Installing Orchestrators in the *Keyfactor Orchestrators Installation and Configuration Guide*. There are also Keyfactor installation guides for third-party CA gateways that interface with Keyfactor Command. For an overview of the key new features in the latest version of Keyfactor Command, see the Release Notes.

This document provides guidance to help you prepare for and complete an upgrade. The Keyfactor Command server software will be upgraded for you, and in most cases, a Keyfactor Solution Architect will assist you with the upgrade and walk you through the process. Please contact your Client Success representative for assistance.

Before upgrading to a major version, Keyfactor recommends first upgrading to the final incremental version of the previous major version—completing both the software installation and configuration

with the configuration wizard—for the optimal upgrade experience. For example, if you are currently on version 9.4 and want to upgrade to version 10.0 or later, you should upgrade first to version 9.10.1 (the final incremental version of 9.x) before upgrading. Contact your Customer Success Manager for more information.

2.1.1.1 Upgrading

Most Keyfactor Command upgrades are brief with a minimum of changes to existing user accounts, groups, CA templates, firewall settings, etc. The prerequisites have not materially changed from previous versions and the current version can generally be installed using the same hardware and existing instances of the supporting software. The upgrade process is often completed within three to four hours, including the time spent by your Keyfactor representative to upgrade your hosted environment.

The overall task flow consists of the following steps:

Upgrade of the Server Software

The Keyfactor Command server software will be installed and configured for you. Once this is complete, you may upgrade any orchestrators and gateways in your environment.

Update Windows Orchestrators

Support for the Keyfactor Windows Orchestrator has been deprecated in Keyfactor Command release 11.0. All uses of the Keyfactor Windows Orchestrator should be updated to the Keyfactor Universal Orchestrator. The Keyfactor Universal Orchestrator replaces the Keyfactor Windows Orchestrator and runs on both Windows or Linux servers. As of this release, the following functions that were part of the Keyfactor Windows Orchestrator are supported in the Keyfactor Universal Orchestrator with custom extensions:

- Interact with F5 devices for certificate management
- Interact with NetScaler devices for certificate management
- Interact with Amazon Web Services (AWS) resources for certificate management
- Interact with Windows certificate stores and IIS

For more information about using custom extensions with the Keyfactor Universal Orchestrator, see *Installing Custom-Built Extensions* in the *Keyfactor Orchestrators Installation and Configuration Guide*.

> ⚠ **Important:** The Keyfactor Universal Orchestrator is only compatible with Keyfactor Command version 9.0 or later. The current version of the Keyfactor Universal Orchestrator is 11.0 and requires .NET 6.

If you're upgrading from a version of Keyfactor Command prior to 8.0, you will need to update any Windows Orchestrators (a.k.a. Windows Agents) that are used for SSL scanning to support the current scanning architecture. Install and configure the Keyfactor Universal Orchestrator software (see Universal Orchestrator in the *Keyfactor Orchestrators Installation and Configuration Guide*).

Note: The orchestrator endpoint location changed for Keyfactor Command release 6 and may need to be modified in your orchestrator endpoint configuration—from CMSAgents to KeyfactorAgents.

Cloud Gateway

The latest version of the Keyfactor Cloud Gateway—used to support management of certificates in the hosted Keyfactor Command environment—is 22.2 released in late 2022. If you are already using this version, no configuration changes need to be made. Restart the gateway service to refresh the connection to the upgraded Keyfactor Command instance.

If you're using a recent version of the gateway (20.6 or newer), you don't need to upgrade the gateway unless the gateway contains a change that's needed in your environment. See the gateway release notes in the *Keyfactor Cloud Gateway Installation & Configuration Guide* to review some of the recent changes.

In most cases, the Keyfactor gateway software can be installed over the existing software installation without uninstalling the previous version. Review the configuration for your gateway, and then install and configure the software as per the *Keyfactor Cloud Gateway Installation & Configuration Guide*, retaining the same installation location.

EJBCA CA Gateway

If you're using an EJBCA gateway and wish to make use of the new feature in Keyfactor Command for native support of EJBCA CAs, you will need to follow the EJBCA gateway upgrade process to unlink the EJBCA certificates in your Keyfactor Command database from your EJBCA gateway CA to enable them to be relinked to a native CA configured in Keyfactor Command. For more information, contact Keyfactor support.

Other CA Gateways

In most cases, the Keyfactor gateway software can be installed over the existing software installation without uninstalling the previous version. Review the configuration for your gateway, and then install and configure the software as per the Keyfactor gateway guide for the particular gateway, retaining the same installation location. The gateway configuration wizard has significantly changed in recent releases for many of the gateways, which may require modification to your configuration.

Tip: New versions of CA gateways are not necessarily released at the same time as new versions of Keyfactor Command and so gateways may not need upgrading at the same time as Keyfactor Command.

API

Important: The Classic API, also known as the CMS API or CMSAPI, has been deprecated in Keyfactor Command version 11.0. Customers should migrate all uses of the Classic API to the

Keyfactor API prior to upgrading to Keyfactor Command version 11.

Please see the latest Release Notes & Upgrading if you are using any custom scripts that leverage the Keyfactor API.

Post-Install Configuration and Testing

See Post-Upgrade Steps below.

The bulk of the time upgrading will be spent verifying that all functions and configurations have correctly carried over and the upgraded instance is performing correctly.

2.1.1.2 Post-Upgrade Steps

There is no particular order in which the tasks on the following pages must be accomplished.

Tip: If, following the upgrade, you open a page in the Keyfactor Command Management Portal and find it unexpectedly blank or otherwise displaying incorrectly, try refreshing the page with a CTRL-F5. If this doesn't resolve the problem, try clearing the browser cache and then reloading the page. It may be helpful to advise all end users to do this following an upgrade.

Testing

Once everything is up and running again, confirm that the following features are operating correctly:

- Does the Keyfactor Command Management Portal load correctly?
- Run a report in the Keyfactor Command Management Portal to confirm that the connectivity to LogiAnalytics is operating correctly.
- Issue a certificate in the Keyfactor Command Management Portal to confirm connectivity to CAs.

Post-Install Configuration

If you are upgrading from any release of Keyfactor Command version 6 or greater, you may want to make some additional configuration changes post-installation:

- Upgrade any Keyfactor CA gateways in your environment that are based on the AnyGateway. The AnyGateway must be upgraded to at least 22.1 to be compatible with Keyfactor Command 10.0 and later.
- Consider whether you wish to implement Keyfactor Command workflows and whether a Keyfactor Command-level workflow could replace CA-level manager approval for any templates that are configured to require CA-level manager approval.
- Review the new enrollment default and policy settings for enrollment. Enrollment defaults and polices can be defined at two levels:

- System-wide settings apply to all enrollments done through Keyfactor Command unless they are overridden by template-specific settings.
- Template-level settings allow you to modify any established enrollment defaults or policies on a per-template basis.

There are several settings available for configuration as part of the template policies:

- Allow Wildcards
- Allow Public Key Reuse
- Enforce RFC 2818 Compliance
- Supported Key Types

Enrollment defaults allow you to pre-polulate the subject fields in PFX Enrollment and CSR Generation. Users are allowed to override these at enrollment.

- If you're using certificate metadata or regular expressions, optionally define these for each template. Certificate metadata fields and regular expressions can be defined at two levels:
  - System-wide settings apply to all enrollments done through Keyfactor Command unless they are overridden by template-specific settings.
  - Template-level settings allow you to modify any established certificate metadata or regular expressions on a per-template basis (for instance, for a metadata field, whether the field is required, what default value it should provide, or whether to hide the field during enrollment, regardless of system-wide setting).
- The enrollment configuration will have been carried over in the upgrade, however you may want to confirm the configuration of Certificate Authority and Template enrollment (PFX, CSR, and CSR generation) and make any changes.
- Review any template that is configured to require manager approval at the CA level and confirm that a Keyfactor Command private key retention policy in place.
- Review the new reports in the Keyfactor Command Report Manager and add them to the menu or favorite them, if desired.

2.1.2 Upgrade Overview - Self-Hosted

The Keyfactor Command solution by Keyfactor allows organizations to issue and manage certificates across enterprise infrastructures. For a comprehensive description of the components that make up Keyfactor Command, see Logical Architecture in the *Keyfactor Command Server Installation Guide* and Installing Orchestrators in the *Keyfactor Orchestrators Installation and Configuration Guide*. There are also Keyfactor installation guides for third-party CA gateways that interface with Keyfactor Command. For an overview of the key new features in the latest version of Keyfactor Command, see the Release Notes.

This document provides guidance to help you prepare for and complete an upgrade. In most cases, a Keyfactor Solution Architect will assist you with the upgrade and walk you through the process. Please contact your Client Success representative for assistance.

Before upgrading to a major version, Keyfactor recommends first upgrading to the final incremental version of the previous major version—completing both the software installation and configuration

with the configuration wizard—for the optimal upgrade experience. For example, if you are currently on version 9.4 and want to upgrade to version 10.0 or later, you should upgrade first to version 9.10.1 (the final incremental version of 9.x) before upgrading. Contact your Customer Success Manager for more information.

Important: Keyfactor Command version 10.0 and later require an encrypted connection to the SQL server. Upgrades will fail if the SQL server is not correctly configured to support this. See System Requirements on page 11.

2.1.2.1 Preparing

This section describes the steps that need to be taken prior to a Keyfactor Command upgrade to complete the prerequisites, create any required supporting components, and gather the necessary information to complete the Keyfactor Command upgrade process.

The following are some key things to be aware of and preparation steps that need to be addressed in order to upgrade to version 11.0:

- If you're migrating from Active Directory to Keyfactor Identity Provider as an identity provider in version 11, be aware that it's important to leave Basic Authentication and Windows Authentication disabled in IIS once you enable the OAuth option in the Keyfactor Command configuration. Attempting to use OAuth with Basic Authentication and/or Windows Authentication enabled can result in unexpected access to Keyfactor Command by users with lingering Active Directory permissions in Keyfactor Command from before the upgrade.
- Select substitutable special text tokens provide the ability for parts of the Keyfactor Command system to look up data in Active Directory to use in alerts and workflows. Lookups for these locate the object in Active Directory identified by the user or computer account that requested the certificate from the CA and substitute the contents of the referenced attribute. For example, an alert might reference *requester:displayname* to look up the requester's display namein Active Directory.

  The substitutable special text tokens that begin *requester:* or *principal:* (but not the standalone *requester* token) are only supported when using Active Directory as an identity provider. They cannot be used with other identity providers since there is no Active Directory to query. Customers wishing to use similar functionality with a different identity provider will need to do API calls to query the identity provider, populate the data into a metadata field, and use the metadata field when populating alert and workflow messages.
- In Keyfactor Command version 11.0, the route information for each virtual application (the host name, web site name, use ssl setting, and virtual directory name) has been moved from being stored in the database to being stored in local files on the Keyfactor Command server (one appsettings.json file for each virtual application). On upgrade, the information is copied from the database to each local file. However, if you encounter any issues with the upgrade process, you may be required to re-enter this data. Be sure to make note of the values before beginning your upgrade.

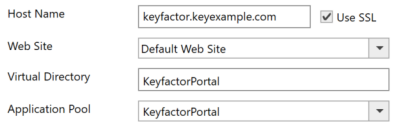

Figure 1: Configuration Wizard Route Information for the Keyfactor Portal

- Keyfactor Command version 10.0 and later by default connects to SQL with an encrypted connection using an SSL certificate configured on your SQL server. Customers should acquire and install an SSL certificate for the SQL server before upgrading to Keyfactor Command version 10.0 or later (see Using SSL to Connect to SQL Server in the *Keyfactor Command Server Installation Guide*). If you would prefer not to use an encrypted channel for your connection to SQL, see Configurable SQL Connection Strings in the *Keyfactor Command Server Installation Guide*.
- Upgrade to SQL Server 2016 CU2 or higher and adjust the database compatibility level if needed. For more information, see System Requirements on page 11.
- As of Keyfactor Command version 10.0, Windows Server 2016 is no longer supported. The installer will not check your server version nor prevent installation, but the product will not function properly in some instances. Customers should upgrade to Windows Server 2019 or higher before upgrading to Keyfactor Command version 10.0 or later. If you choose to use Server 2016, any PFXs will need to be configured to use SHA1 and 3DES for encryption for use by Keyfactor Command.
- Before upgrading to a major version, Keyfactor recommends first upgrading to the final incremental version of the previous major version—completing both the software installation and configuration with the configuration wizard—for the optimal upgrade experience. For example, if you are currently on version 9.4 and want to upgrade to version 10.0 or later, you should upgrade first to version 9.10.1 (the final incremental version of 9.x) before upgrading. Contact your Customer Success Manager for more information.
- If you have any saved certificate collections containing any of the following deprecated certificate search fields, these collections will need to be removed or updated to remove use of these fields that are no longer in version 10.0 and later:
  - *KeyfactorRequestId*
  - *RequestResolutionDate*
  - *CARequestId*

  These certificate search fields parsers have been removed to allow for native EJBCA support in Keyfactor Command as of version 10.0.
- If you have the CA Policy module version 7.0 installed on the same server as the Keyfactor Command Management Portal, you'll need to upgrade the module to version 7.1 or later before running the Keyfactor Command version 10.0 or later upgrade.
- Be Aware of the following certificate store changes in Keyfactor Command v11:
  1. The following store types have been deprecated and will no longer be shipped with Keyfactor Command: All F5 store types, AWS, NetScaler.

     These store types will not be created with new databases. Upon upgrading, they will be removed if the customer does not have any stores or containers defined for the type (per

type. i.e. if an AWS store or container is defined and the other types are not, the AWS type will not be removed on upgrade, but the other types will be removed).

2. The built-in functions for IIS and FTP certificate store management in the Keyfactor Universal Orchestrator have been deprecated in Keyfactor Command version 11.0.

   Customers should migrate all uses of IIS certificate store management to the Keyfactor Universal Orchestrator with the IIS custom extension publicly available at:

   https://github.com/Keyfactor/iis-orchestrator

   For more information, see *Installing Custom-Built Extensions* in the *Keyfactor Orchestrators Installation and Configuration Guide*.

   For customers who have FTP certificate store type, when you upgrade Keyfactor Command to version 11 your FTP stores and related data will not be removed on upgrade. In order to manage the FTP certificate store, you will need to use an older version of the UO that still has the FTP extension.

Licensing

You will receive a new license file for the new version of Keyfactor Command. Before upgrading, locate your existing license file so that, should you need to revert to your existing software version, you will easily be able to do so without requesting a new license file from Keyfactor. (License files have the file extension '.cmslicense'.)

As you begin the upgrade, have both your new license file and your existing license file on hand.

If you need assistance with a license, send a request to support@keyfactor.com.

Users, Service Accounts and Groups

Review the Active Directory service accounts and groups used by your Keyfactor Command implementation. You will need to have these accounts and groups available during the upgrade process, along with the passwords for the service accounts. For a full overview of the required service accounts and groups, see Create Active Directory Service Accounts for Keyfactor Command and Create Active Directory Groups to Control Access to Keyfactor Command Features in the *Keyfactor Command Server Installation Guide*. The most common service accounts are:

Keyfactor Command Service Account

In many environments, a single service account is used for most Keyfactor Command functions, including the application pool service account and the service account for the Keyfactor Command Service[1]. In some environments, separate service accounts are used for these functions.

[1]If you're running the Certificate Management System rather than Keyfactor Command, this service will be called the CMS Timer Job Service.

Keyfactor Command LogiAnalytics Service Account

Keyfactor Command uses a the reporting engine LogiAnalytics. This reporting engine uses the same service account the application pool is configured to use.

Keyfactor Command Orchestrator Service Accounts

If you are using orchestrators, you will need the account(s) the orchestrators are configured to run as and the account(s) used to connect to the Keyfactor Command Orchestrators site.

Keyfactor Command Policy Module

If you are using the Keyfactor Command policy module with any of the standard policy handlers or any custom policy handlers, you will need to have access to upgrade these on the CA if you will be upgrading these at the same time.

CA Gateways

Important: All CA Gateways must be upgraded to AnyGateway v22.1 to work with Keyfactor Command v10.

If you are upgrading any of the CA Gateways, you will need to have the correct credentials to connect to the cloud-based certificate authority. The format of these varies depending on the CA provider. Some providers use a username and password while others use client certificate authentication. Some support the choice of either.

If you are unable to locate the existing passwords for your service accounts, you will need to reset the passwords so that the accounts will have known values in preparation for the upgrade. These password changes will need to be coordinated with your existing Keyfactor Command installation to avoid a service interruption. On your Keyfactor Command server(s), the password for the Keyfactor Command service account (assuming you are using just one) will need to be changed:

- In IIS for the CMS/Keyfactor Command application pool.
- In the Services MMC for the Keyfactor Command Service[1].
- Via the Keyfactor Command Configuration Wizard for the LogiAnalytics connectivity.
- Via the Keyfactor Command Orchestrator Configuration Wizard for any orchestrators running in the environment.

Password updates for the Keyfactor Command service accounts can be done via the Keyfactor Command Configuration Wizard during the upgrade process and do not need to be done ahead of the upgrade. The password(s) should be changed in Active Directory as close to upgrade time as possible to limit down time in the existing Keyfactor Command implementation.

If possible, identify the user account that was used to do the original installation of Keyfactor Command (the “installer” account) and use this same account to perform the upgrade. If you are

[1]Or CMS Timer Job Service for older versions of the software.

upgrading under a different account than this, the permissions required in SQL will be different. See SQL Permissions below.

SQL Permissions

The user who upgrades Keyfactor Command must have permissions to administer the SQL server and update databases. The user may need to be able to add users (logins), depending on the features used. Full sysadmin permissions in SQL are needed if you're upgrading from a previous version of Keyfactor Command and the user running the install is not the same user who installed the previous version of Keyfactor Command. If the user is the same, only the dbcreator, public and securityadmin roles are needed.

Once Keyfactor Command has been upgraded, these permissions can be removed for the user.

Connecting to SQL over SSL

By default, Keyfactor Command connects to SQL using an encrypted connection. This requires configuration of an SSL certificate on your SQL server.

If your SQL server is not configured correctly for SSL, you'll see an error message similar to the following when you try to make a connection from Keyfactor Command:

```
Unable to establish a connection to the database server. Please ensure that the server
name is correct and sufficient privileges have been granted to the connection account.:
Encountered an invalid or untrusted certificate and could not connect to the database.
TLS encryption is enabled by default. Please visit 'Planning and Preparing --> SQL
Server' In the Keyfactor Installing Server guide to resolve this.
```

To acquire a new SSL certificate or check for an existing certificate, see Using SSL to Connect to SQL Server in the *Keyfactor Command Server Installation Guide*.

If you would prefer not to use an encrypted channel for your connection to SQL, see Configurable SQL Connection Strings in the *Keyfactor Command Server Installation Guide*.

System Requirements

For a full list of the requirements, see System Requirements in the *Keyfactor Command Server Installation Guide*.

Operating System

Keyfactor Command server is supported on Windows Server 2019 or 2022.

PKI Architecture

Please visit Confirm the Architecture on the next page and review the implications of upgrading with regard to the PKI architecture elements.

SQL Server

As of Keyfactor Command version 10.0, Microsoft SQL Server 2017, 2019 or 2022 is required and connectivity to the SQL server requires TLS encryption. For information about configuring TLS for SQL server, see Using SSL to Connect to SQL Server in the *Keyfactor Command Server Installation Guide* and:

> https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/enable-encrypted-connections-to-the-database-engine?view=sql-server-ver15

.NET Framework

Microsoft .NET 4.7.2 or greater must be installed on the Keyfactor Command server(s) prior to installation of the latest Keyfactor Command software.

For Windows Server 2019 and Windows Server 2022, .NET is a standard Windows feature added through the Windows Server Manager tool. It can be updated to .NET 4.7.2 or greater with a downloadable update package or through Windows update. For information on checking the .NET version, see Install IIS and .NET on the Keyfactor Command Server in the *Keyfactor Command Server Installation Guide*.

PowerShell Requirement

More recent versions of Keyfactor Command make use of the Active Directory tools for PowerShell to do group membership queries in Active Directory in some functions (e.g. when using a group to create a mapping between a Linux logon for SSH and one or more SSH keys). If this feature is not already installed on your Keyfactor Command server, you will need to install it before upgrading the Keyfactor Command software. The *Active Directory module for Windows PowerShell* is installed as a feature as part of the *Remote Server Administrator Tools*. You may install this through the Roles and Features wizard or using the following PowerShell command:

```
Install-WindowsFeature RSAT-AD-PowerShell
```

Download the Software

Your Keyfactor contact should provide you with a link to download the updated software versions. Be sure to download all the files you will need ahead of the actual upgrade date. This includes the main Keyfactor Command server software as well as the software for the Keyfactor CA Policy Module, any orchestrators (e.g. Keyfactor Universal Orchestrator, Keyfactor Java Agent) or gate-

ways (e.g. AnyGateway), that you will be upgrading at the same time or new software you will be deploying.

Before upgrading to a major version, Keyfactor recommends first upgrading to the final incremental version of the previous major version—completing both the software installation and configuration with the configuration wizard—for the optimal upgrade experience. For example, if you are currently on version 9.4 and want to upgrade to version 10.0 or later, you should upgrade first to version 9.10.1 (the final incremental version of 9.x) before upgrading. Contact your Customer Success Manager for more information.

Configuration File

Keyfactor Command can use a file to pass the configuration information into the configuration wizard, which saves a significant amount of typing when you do your initial installation. You may have been provided one of these already configured for your initial implementation, or you may have created one after typing in all the configuration information during the initial implementation. If you can locate this file, it can save some time in the upgrade process. You won't want to import the existing file again, as the file structure may change between versions and importing the file again will overwrite any changes you might have made to the configuration since your initial install, but you can refer to the file for previous configuration information.

The configuration files generally have a .cmscfg extension. When creating the file, you have the option to encrypt and password protect the file. If the file has been password protected, sensitive information in the file, such as any service account passwords, will be encrypted, but the remainder of the file will be human readable. You will need to know the password used to protect the file in order to use the file in its complete state.

Confirm the Architecture

Before you start your upgrade, make sure you have a clear picture of your Keyfactor Command architecture and all the parts that make up the environment, and carefully consider the following.

Roles

Identify all the servers that play a role in the Keyfactor Command environment, including whether you have duplicates of any server roles to support high availability, and make note of what role or roles will need upgrading on each one. Think about whether you want to make any changes to the architecture at this time, such as adding high availability, or consolidating roles.

Certificate Authorities

Keyfactor Command includes a constraint (introduced in version 9.0) that prevents any two certificate authorities from having the same logical name and host name combination. Think about the logical name and host name of the CAs that will be implemented with Keyfactor Command and check for duplicates.

Important: During upgrade, if duplicates are found, then among the duplicates, if there is only one that has any information tied to it, such as certificates, API applications, etc., then all of the others will be removed by the upgrade script. If more than one of the duplicates has any information associated with it, then the upgrade script will stop with an error. In that instance, you will need to manually fix the data before upgrading can proceed.

Templates

Keyfactor Command 10.0 and later upgrades will fail if the database has duplicate templates, defined as:

- Duplicate CommonName and Forest, or
- Duplicate OID and Forest

This should be a rare case. If it does occur, contact Keyfactor support. Support will be able to identify the duplicate templates, save the desired templates, and remove the duplicates.

Backup

Immediately before starting the upgrade, make a backup of these items:

- Your Keyfactor Command SQL database
- Your SQL server Service Master Key (SMK) and/or Database Master Key (DMK), if needed (see Important note)

  If you plan to migrate your Keyfactor Command implementation to a different SQL server during the upgrade, you need a thorough understanding of how Keyfactor Command uses the SMK and DMK. Review the data in SQL Encryption Key Backup in the *Keyfactor Command Reference Guide* and make appropriate plans before beginning your upgrade. If you plan to stay on the same SQL instance for the upgrade, you don't necessarily need to backup the SMK or DMK immediately before starting the upgrade. These can just be backed up as part of your normal disaster recovery planning process. Failing to back up the SMK and/or DMK will result in data loss and require manual re-entry of any secret data into Keyfactor Command in the event that the Keyfactor Command database needs to be restored from a backup to a SQL instance other than the original installed instance of SQL server.

  Note: For more information about how Keyfactor Command uses the SMK and DMK and how to back these up, see SQL Encryption Key Backup in the *Keyfactor Command Reference Guide*. For more details on the mechanics of SQL Server Encryption and related disaster recovery procedures, see the SQL Server documentation.

- If you're using Keyfactor Command encryption, backup your encryption certificate, with private key (see below).

  More recent versions of Keyfactor Command allow you to encrypt select sensitive data stored in the Keyfactor Command database using a separate encryption methodology. This Keyfactor

Command encryption utilizes a Keyfactor-defined certificate on top of the SQL server encryption noted above. This additional layer of encryption (Keyfactor Command encryption) protects the data in cases where the SQL Server master keys cannot be adequately protected. More information is provided in SQL Server in the *Keyfactor Command Server Installation Guide*.

- Backup the NLog configuration file for each application to be upgraded. The location of this file varies depending on the application in question. For older versions of the Keyfactor Command server, it can be found in one of these locations:

  ```
  C:\Program Files\Common Files\Certified Security Solutions\Certificate Management System\NLog.config
  C:\Program Files\Common Files\Keyfactor\Keyfactor Platform\NLog.config
  ```

  More recent versions of Keyfactor Command separate the NLog configuration into multiple files, with these locations, by default:

  ```
  C:\Program Files\Keyfactor\Keyfactor Platform\Configuration\NLog_Configuration.config
  C:\Program Files\Keyfactor\Keyfactor Platform\KeyfactorAPI\NLog_KeyfactorAPI.config
  C:\Program Files\Keyfactor\Keyfactor Platform\Service\NLog_TimerService.config
  C:\Program Files\Keyfactor\Keyfactor Platform\WebAgentServices\NLog_Orchestrators.config
  C:\Program Files\Keyfactor\Keyfactor Platform\WebAPI\NLog_ClassicAPI.config
  C:\Program Files\Keyfactor\Keyfactor Platform\WebConsole\NLog_Portal.config
  ```

- Make a backup of the Logi configuration file, which is found here by default:

  ```
  C:\Program Files\Keyfactor\Keyfactor Platform\Logi\_Definitions\_Settings.lgx
  ```

- If you have any custom extension handlers (e.g. auto-registration, alert events), make a backup of these.
- If you've have any other text-based configuration files that have been modified (this is most common for users who have enabled a third-party PAM provider such as CyberArk), make a backup of these.
- If you're using a custom logo for your Management Portal, make a backup of this image. This file can be found here, by default:

  ```
  C:\Program Files\Keyfactor\Keyfactor Platform\WebConsole\Images\Banner.png
  ```

- Review the authentication settings you have configured in IIS for each of the Keyfactor Command applications under the Default Web Site (or other web site if you've installed elsewhere) and make notes as to how they are configured so that you can confirm that the configuration is the same following upgrade.
- If you're using a virtualization solution for your Keyfactor Command application server(s), backup each virtual server as an image.
- If you are using a version of Keyfactor Command older than 6 and have existing SSL scans, export them to a file using the following script. Replace the bold italicized parts with the information relevant to your environment. This step is not necessary if you're upgrading from release

version 6 or later.

```
$connectionString = "Data Source=SQLServerName;Integrated
Security=SSPI;Initial Catalog=KeyfactorDB"
$connection = new-object
System.Data.SqlClient.SqlConnection($connectionString)
$connection.Open()
# Password can be read from an encrypted file which can be secured as follows:
# Create a password file while logged in as the service account that will run this script:
# $credential = Get-Credential
# $credential.Password | ConvertFrom-SecureString | Set-Content
C:\Keyfactor\PowerShell\encrypted_password1.txt
# use the code below for the credentials
#$password = Get-Content C:\Keyfactor\PowerShell\encrypted_password1.txt | ConvertTo-SecureString
#comment out the below line if using secure credentials
$password = "Password" | ConvertTo-SecureString -AsPlainText -Force
#Update with the credentials for your environment
$username = "domain\administrator"
$credential = New-Object System.Management.Automation.PSCredential($username, $password)
$passphrase = $username + ":" + $password
$fileName = "DiscoveryGroupsExported.txt"
#The name of the agent in your environment
$AgentName = "kyfagent1.domain.com"
#Update with the URLs for your environment.
$kyfAgentUrl = "http://kyfagent1.domain.com/CMSAPI/SSL/1/Agents"
$kyfGroupUrl = "http://kyfagent1.domain.com/CMSAPI/SSL/1/AddEndpointGroup"
$kyfEndpointUrl = "http://kyfagent1.domain.com/CMSAPI/SSL/1/AddEndpoint"
$Bytes = [System.Text.Encoding]::Unicode.GetBytes($passphrase)
$EncodedText = [Convert]::ToBase64String($Bytes)
$headers = @{"Authorization" = "Basic $EncodedText";
       "Content-Type"  = "application/json;" }
$responseAgent = Invoke-RestMethod -Method Get -Uri $kyfAgentUrl -Header
$headers -Credential $credential
if ($responseAgent)
{
  write-host $responseAgent.Name.ToLower()
  if ($responseAgent.Name.ToLower() -eq $AgentName.ToLower())
  {
    $AgentGUID = $responseAgent.Guid
  }
  write-host $AgentGUID
  $kyfEndpointGroups = "http://ky-
fagent1.domain.com/CMSAPI/SSL/1/EndpointGroups?agentId=$AgentGUID"
  write-host $kyfEndpointGroups
  $responseEndpointGroups = Invoke-RestMethod -Method Get -Uri $kyfEndpointGroups -Header
$headers -Credential $credential
```

```
  if ($responseEndpointGroups)
  {
    foreach($res in $responseEndpointGroups)
    {
      write-host $res.Name
      $GroupName = $res.Name
      #write-host $res.guid
      $GroupGuid = $res.Guid
      $sql = "SELECT VALUE, TypeID FROM cms_agents.SslEndpointGroupItems WHERE GroupID =
'$GroupGuid'"
      #write-host $sql $command = new-object System.Data.SqlClient.SqlCommand($sql,$connection)
      $reader = $command.ExecuteReader()
      while ($reader.Read())
      {
        $value = ""
        $type = ""
        $value = $reader["Value"]
        $typeId = $reader["TypeId"]
        #Add-Content $filename "$GroupName,$GroupGuid,$value,$typeId"
      }
      $reader.Close()
    }
  }
}
else
{
  write-host "Agent not found."
}
$connection.Close()
```

The resulting text file will contain the network definitions you currently have and can be opened in Excel. When Keyfactor Command has been upgraded, you can copy and paste from the file into the newly defined *Networks* that replace the previous *Discovery* and *Monitoring* groups.

2.1.2.2 Upgrading

Most Keyfactor Command upgrades are brief with a minimum of changes to existing user accounts, groups, CA templates, firewall settings, etc. The prerequisites have not materially changed from previous versions and the current version can generally be installed using the same hardware and existing instances of the supporting software. The upgrade process is often completed within three to four hours.

Before upgrading, please be sure you have reviewed and addressed the important preparation steps (see Preparing on page 7).

Important: The MicrosoftECCurveUpgradeModule may fail due to a pre-version 10.0 issue in which Certificate Request Contents were truncated to 4k characters when saved to the database. If your upgrade fails when the MicrosoftECCurveUpgradeModule is run, contact Keyfactor Support to obtain assistance with the scripts that will have to be run to fix this issue.

Important: During the upgrade process Keyfactor Command prevents duplicate template records from being inserted into the database. Duplicate templates could be found, for example, if there are templates in different forests with the same name. If you receive an error message during upgrade, and the log shows a list of the duplicate templates, contact Keyfactor Support. We will be able to support you through the process of resolving the issue and completing the upgrade.

Before upgrading to a major version, Keyfactor recommends first upgrading to the final incremental version of the previous major version—completing both the software installation and configuration with the configuration wizard—for the optimal upgrade experience. For example, if you are currently on version 9.4 and want to upgrade to version 10.0 or later, you should upgrade first to version 9.10.1 (the final incremental version of 9.x) before upgrading. Contact your Customer Success Manager for more information.

The overall task flow consists of the following steps:

Upgrade of the Server Software

In most cases the Keyfactor Command server software can be installed over the existing software installation without uninstalling the previous version. Install the software retaining the same installation location (see Installing in the *Keyfactor Command Server Installation Guide*). In the configuration wizard, populate the fields while referring to your configuration file open in a text editor (see Configuration File on page 13). Use the existing IIS application pool.

Update Windows Orchestrators

Support for the Keyfactor Windows Orchestrator has been deprecated in Keyfactor Command release 11.0. All uses of the Keyfactor Windows Orchestrator should be updated to the Keyfactor Universal Orchestrator. The Keyfactor Universal Orchestrator replaces the Keyfactor Windows Orchestrator and runs on both Windows or Linux servers. As of this release, the following functions that were part of the Keyfactor Windows Orchestrator are supported in the Keyfactor Universal Orchestrator with custom extensions:

- Interact with F5 devices for certificate management
- Interact with NetScaler devices for certificate management
- Interact with Amazon Web Services (AWS) resources for certificate management
- Interact with Windows certificate stores and IIS

For more information about using custom extensions with the Keyfactor Universal Orchestrator, see *Installing Custom-Built Extensions* in the *Keyfactor Orchestrators Installation and Configuration Guide*.

Important: The Keyfactor Universal Orchestrator is only compatible with Keyfactor Command version 9.0 or later. The current version of the Keyfactor Universal Orchestrator is 11.0 and requires .NET 6.

If you're upgrading from a version of Keyfactor Command prior to 8.0, you will need to update any Windows Orchestrators (a.k.a. Windows Agents) that are used for SSL scanning to support the current scanning architecture. Install and configure the Keyfactor Universal Orchestrator software (see Universal Orchestrator in the *Keyfactor Orchestrators Installation and Configuration Guide*).

Note: The orchestrator endpoint location changed for Keyfactor Command release 6 and may need to be modified in your orchestrator endpoint configuration—from CMSAgents to KeyfactorAgents.

Keyfactor CA Policy Module

Important: The most recent versions of the Keyfactor CA Policy Module software need to be upgraded using the below method and PowerShell script and can't be installed over an existing implementation of the Keyfactor CA Policy Module as an upgrade method.

To upgrade a Keyfactor CA Policy Module:

1. Make a note of all your existing policy module configuration, including which policy handlers are enabled and what configurations are set within each handler. **During the upgrade process, you will uninstall the policy module, which will remove your configuration.** The upgrade script should successfully restore the configuration as part of the upgrade process, but you will want to have a complete record of the configuration as a backup.
2. On the Keyfactor CA Policy Module server, open a PowerShell window using the "Run as administrator" option.
3. In the PowerShell window, change to the directory in which you placed the upgrade script included with the latest version of the Keyfactor CA Policy Module and execute it in *archive* mode. For example:

   ```
   .\Keyfactor-CA-Modules-Upgrade-Script.ps1 -Mode archive -InformationAction Continue
   -ErrorAction Stop
   ```

   Note: This step is creating a backup of your policy module configuration before you uninstall the old policy module. It will create an output file, *Keyfactor-CA-Policy.dat*, in the current directory.

Tip: Additional options are available in the upgrade script and can be viewed using the *-full* switch with *Get-Help*. For example:

```
Get-Help .\Keyfactor-CA-Modules-Upgrade-Script.ps1 -full
```

4. Unload the existing policy module in the CA MMC, and close the MMC.
5. **Uninstall the existing policy module**.
6. Install the latest version of the Keyfactor CA Policy Module but do not configure it (see Installing the Keyfactor CA Policy Module Handlers in the *Keyfactor Command Server Installation Guide*). Be sure to install all the same policy handlers that were installed previously.

   Execute the upgrade script included with the latest version of the Keyfactor CA Policy Module again, but this time in *restore* mode. For example:

   ```
   .\Keyfactor-CA-Modules-Upgrade-Script.ps1 -Mode restore -InformationAction Continue
   -ErrorAction Stop
   ```

   Note: This step takes the backup of your policy module configuration from the first run of the upgrade script and restores the information to the correct locations so that you will not need to re-configure the policy module. Be sure that the output file from the first run of the upgrade script, *Keyfactor-CA-Policy.dat*, is in the current directory.

7. Open the CA MMC and load the Keyfactor CA Policy Module (see Installing the Keyfactor CA Policy Module Handlers in the *Keyfactor Command Server Installation Guide*).
8. Open the Properties for the policy module and, if you've received a new license, install the new license on the License tab. On the Custom Handlers tab, review all the configuration to confirm that it has been correctly restored by the upgrade script.

Tip: New versions of the policy module are not necessarily released at the same time as new versions of Keyfactor Command and so the policy module may not need upgrading at the same time as Keyfactor Command.

EJBCA CA Gateway

If you're using an EJBCA gateway and wish to make use of the new feature in Keyfactor Command for native support of EJBCA CAs, you will need to follow the EJBCA gateway upgrade process to unlink the EJBCA certificates in your Keyfactor Command database from your EJBCA gateway CA to enable them to be relinked to a native CA configured in Keyfactor Command. For more information, contact Keyfactor support.

Other CA Gateways

In most cases, the Keyfactor gateway software can be installed over the existing software installation without uninstalling the previous version. Review the configuration for your gateway, and then

install and configure the software as per the Keyfactor gateway guide for the particular gateway, retaining the same installation location. The gateway configuration wizard has significantly changed in recent releases for many of the gateways, which may require modification to your configuration.

Tip: New versions of CA gateways are not necessarily released at the same time as new versions of Keyfactor Command and so gateways may not need upgrading at the same time as Keyfactor Command.

API

Important: The Classic API, also known as the CMS API or CMSAPI, has been deprecated in Keyfactor Command version 11.0. Customers should migrate all uses of the Classic API to the Keyfactor API prior to upgrading to Keyfactor Command version 11.

Please see the latest Release Notes & Upgrading if you are using any custom scripts that leverage the Keyfactor API.

Replacing or Re-Updating Customized Files

Files such as the nlog.config file or customized files for third-party PAM integration (e.g. web.config customizations for CyberArk) may have slight changes in the latest version as compared to the previous version, so you should not just copy your old, customized versions of those files over the current stock versions of these files. You will need to compare the files and make your customizations in the current versions of the files.

Post-Install Configuration and Testing

See Post-Upgrade Steps below

The bulk of the time upgrading will be spent verifying that all functions and configurations have correctly carried over and the upgraded instance is performing correctly.

2.1.2.3 Post-Upgrade Steps

The recommended best practices for after you finish running the Keyfactor Command configuration wizard(s) are:

- The server should be rebooted to assure that the services have a clean start. If this is not possible:
  - Restart Keyfactor Command Service
  - Restart IIS
- Advise users to clear the cache on their web browser and reload the Keyfactor Command Management Portal.

There is no particular order in which the tasks on the following pages must be accomplished.

Tip: If, following the upgrade, you open a page in the Keyfactor Command Management Portal and find it unexpectedly blank or otherwise displaying incorrectly, try refreshing the page with a CTRL-F5. If this doesn't resolve the problem, try clearing the browser cache and then reloading the page. It may be helpful to advise all end users to do this following an upgrade.

Testing

Once everything is up and running again, confirm that the following features are operating correctly:

- Does the Keyfactor Command Management Portal load correctly?
- Run a report in the Keyfactor Command Management Portal to confirm that the connectivity to LogiAnalytics is operating correctly.
- Issue a certificate in the Keyfactor Command Management Portal to confirm connectivity to CAs and that Kerberos authentication is operating correctly (assuming the environment is configured for Kerberos authentication).
- Check the Keyfactor Command log files to confirm that no errors are appearing and that logging is occurring correctly.

Post-Install Configuration

If you are upgrading from any release of Keyfactor Command version 6 or greater, you may want to make some additional configuration changes post-installation:

- Upgrade any Keyfactor CA gateways in your environment that are based on the AnyGateway. The AnyGateway must be upgraded to at least 22.1 to be compatible with Keyfactor Command 10.0 and later.
- Consider whether you wish to implement Keyfactor Command workflows and whether a Keyfactor Command-level workflow could replace CA-level manager approval for any templates that are configured to require CA-level manager approval.

  Note: To prevent REST requests from being made to inappropriate locations by malicious users, if you plan to implement REST type workflows, configure a system environment variable of KEYFACTOR_BLOCKED_OUTBOUND_IPS on your Keyfactor Command server pointing to the IP address or range of addresses in CIDR format that you wish to block. Both IPv4 and IPv6 addresses are supported. More than one address or range may be specified in a comma-delimited list. For example:

  ```
  192.168.12.0/24,192.168.14.22/24
  ```

  When a REST request is made where the URL is either configured to a blocked IP address or resolves via DNS to a blocked IP address, the REST request will fail.

- Review the new enrollment default and policy settings for enrollment. Enrollment defaults and polices can be defined at two levels:

- System-wide settings apply to all enrollments done through Keyfactor Command unless they are overridden by template-specific settings.
- Template-level settings allow you to modify any established enrollment defaults or policies on a per-template basis.

There are several settings available for configuration as part of the template policies:

- Allow Wildcards
- Allow Public Key Reuse
- Enforce RFC 2818 Compliance
- Supported Key Types

Enrollment defaults allow you to pre-polulate the subject fields in PFX Enrollment and CSR Generation. Users are allowed to override these at enrollment.

- If you're using certificate metadata or regular expressions, optionally define these for each template. Certificate metadata fields and regular expressions can be defined at two levels:
  - System-wide settings apply to all enrollments done through Keyfactor Command unless they are overridden by template-specific settings.
  - Template-level settings allow you to modify any established certificate metadata or regular expressions on a per-template basis (for instance, for a metadata field, whether the field is required, what default value it should provide, or whether to hide the field during enrollment, regardless of system-wide setting).
- Review any alert PowerShell event handlers you may have configured to ensure they are in the path (or subdirectory thereof) as defined in the *Extension Handler Path* application setting value. Changes as of version 9.0 will cause PowerShell event handlers to fail if not located in the defined directory. For more information, see Adding PowerShell Handlers to Alerts in the *Keyfactor Command Reference Guide*.
- The enrollment configuration will have been carried over in the upgrade, however you may want to confirm the configuration of Certificate Authority and Template enrollment (PFX, CSR, and CSR generation) and make any changes.
- Review any template that is configured to require manager approval at the CA level and confirm that a Keyfactor Command private key retention policy in place.
- Update any monitoring or other processes that reference the log files to point to the new log file location.
- Review the new reports in the Keyfactor Command Report Manager and add them to the menu or favorite them, if desired.

If you are upgrading from a release prior to Keyfactor Command version 6.1, please contact support (support@keyfactor.com) for upgrade assistance.

2.1.2.4 Troubleshooting

Typically, an upgrade completes with few hiccups and the new version of Keyfactor Command comes up without incident. If this doesn't happen, start by checking the log file(s) for any errors. By default, these are located in C:\Keyfactor\logs. It is sometimes helpful to enable debug or trace level

logging. This is done by editing the nlog.config file for each application. For more information, see Editing NLog in the *Keyfactor Command Reference Guide*.

Error During Upgrade

If you encounter an error during upgrade, this can be the result of a number of different things. Often, it's related to connectivity to SQL or issues on the SQL server. Check the *Command_Configuration_Log.txt* for messages related to upgrading and upgrade failures. This will point you in the right direction to begin troubleshooting.

The following error message indicates that the referenced upgrade script failed because it took longer to run than the allowed limit for SQL tasks:

```
2022-12-07 10:19:07.5078 Keyfactor.Sql.Management.Upgrade.UpgradePlan [Error] - Failed
to run upgrade module CSS.CMS.Install.Upgrade.Scripts.EJBCA_Resolved_Request_Contents_
Removal.sql: Execution Timeout Expired. The timeout period elapsed prior to completion
of the operation or the server is not responding.
```

The Keyfactor Command upgrade process includes multiple scripts, each doing different tasks, and each script is run in batches to limit the time and load of any one SQL request, but it's still possible to encounter a batch that exceeds the limit with vary large or complex databases. To resolve this particular issue, you can increase the timeout limit and restart the upgrade. You do not need to restore and start over.

To increase the timeout limit:

1. On the Keyfactor Command server, open a text editor (e.g. Notepad) using the "Run as administrator" option.
2. In the text editor, browse to open the *CSS.CMS.Install.ConfigurationWizard.exe.config* file (*ConfigurationWizardConsole.exe.config* if you're doing a command-line install) in the Configuration directory under the installed directory. By default, this is:

   ```
   C:\Program Files\Keyfactor\Keyfactor
   Platform\Configuration\CSS.CMS.Install.ConfigurationWizard.exe.config
   ```

3. Locate the appSettings line that contains *Keyfactor.Sql.DbCommandTimeout*. This will look something like:

   ```
   <add key="Keyfactor.Sql.DbCommandTimeout" value="1800" />
   ```

4. The timeout value is set in seconds, so the default of 1800 seconds is 30 minutes. Set it to a new, longer value to allow the upgrade to complete. Don't set it to a value that's too high, as you do want the upgrade to time out if there's some fundamental problem communicating with SQL.
5. Save the file, close the configuration wizard, open the configuration wizard again (you should find it on the menu), and begin the upgrade again.

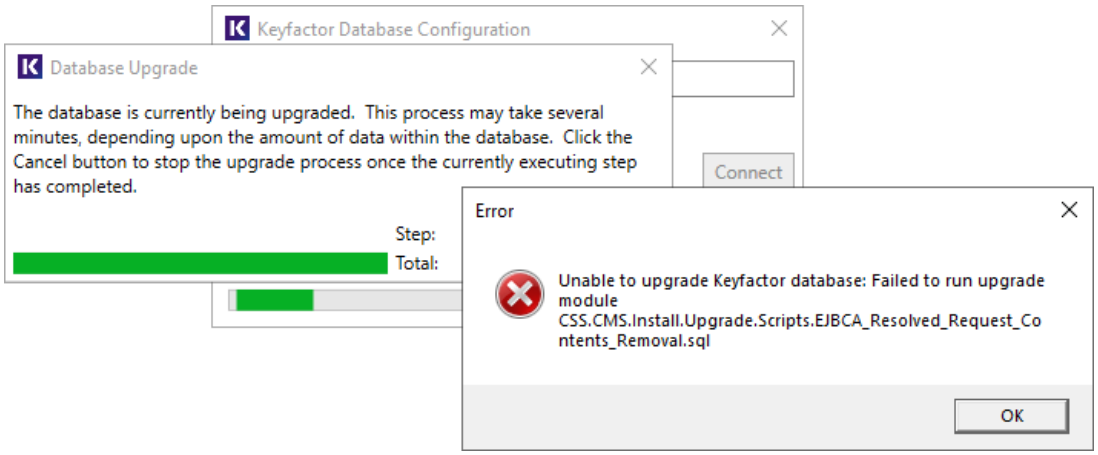

Figure 2: Error During Upgrade

Note: In previous versions of Keyfactor Command, the timeout was controlled with the *command timeout* setting in the connection string of the *SharedSqlConnectionStrings.config* file and had a default of 360 seconds (6 minutes).

Management Portal Doesn't Load After Upgrade

If the Keyfactor Command Management Portal appears to partially load or does not appear to include expected updates after the upgrade, try clearing the browser cache, closing the browser, and opening a fresh browser session. Try using CTRL-F5 to request the page again without cached content. In some upgrade cases, with Internet Explorer, the Certificate Search page only partially loads. With some browsers, opening the Developer Tools with the F12 key and clearing the cache will resolve the problem.

Certificate Enrollment Fails

If the certificate enrollment fails, this is often an indication that there is a Kerberos authentication problem. Confirm that the service principal name (SPN) is set correctly for the application pool service account and that Kerberos constrained delegation is configured correctly from the Keyfactor Command server(s) to the CA(s). For more information, see Configure Kerberos Authentication in the *Keyfactor Command Server Installation Guide*.

Event Handlers Don't Run

If your alert PowerShell event handlers or renewal event handlers do not run correctly, be sure that you have updated them to the correct new location. Scripted alert handlers will fail to run if not in the path (or a subdirectory of it) specified by the *Extension Handler Path* application setting. By default, this is C:\Program Files\Keyfactor\Keyfactor Platform\ExtensionLibrary\. For more information, see Adding PowerShell Handlers to Alerts in the *Keyfactor Command Reference Guide*.

500 Error on the Dashboard or in Reports

If you receive a 500 error loading the dashboard and running reports but the remainder of the Management Portal seems to be operating correctly, check to be sure that the IP address(es) configured in the Configuration Wizard on the Dashboard and Reports tab have been entered correctly.

Underlying Connection Closed

If you receive an error when opening the Management Portal that "the underlying connection was closed" please be sure you have all the latest Windows updates installed.

Please refer to the Keyfactor Command Release Notes for known issues.

If you need further assistance, please contact support. During normal business hours, support can be reached at support@keyfactor.com or (877)-715-5448.

2.2 Major Release 11.0 Notes

October 2023

We're thrilled to announce Keyfactor Command 11.0, which includes some major new features and updates such as support for OAuth 2.0, Universal Orchestrator Container, and certificate renewal tracking.

Please refer to Upgrade Overviews on page 2 for important information about the upgrade process.

Important: The MicrosoftECCurveUpgradeModule may fail due to a pre-v10 issue in which Certificate Request Contents were truncated to 4k characters when saved to the database. If your upgrade fails when the MicrosoftECCurveUpgradeModule is run, contact Keyfactor Support to obtain assistance with the scripts that will have to be run to fix this issue.

Tip: We encourage customers to contact their customer success manager to discuss the new features and functionality in Keyfactor Command 11, and to schedule an upgrade. Please refer to Upgrade Overviews on page 2 for important information about the upgrade process.

Highlights

New Security Permission Model

For release 11.0 of Keyfactor Command, a new permission structure has been introduced. Users of Keyfactor Command through the Management Portal will see some changes, including:

- The permissions tabs on the Security Roles Role Information For: *Role Name* page have a new look and method for setting permissions using check boxes in a tree structure.

- The Security Identities page has been removed.
- A new page for Claims has been added. This page allows the user to see the roles a claim is in.
- The security permissions for PKI Administration (*PKI Management: Read* and *PKI Management: Modify*) have been replaced with two new security permissions: *Certificate Authorities* (*Read* and *Modify*) and *Certificate Templates* (*Read* and *Modify*). The upgrade process will appropriately translate the old security permission to the new ones.

Users of Keyfactor Command through the Keyfactor API will need to understand the new model.

On upgrade, existing security roles will be converted to the new permission structure.

The version one permission model was largely replaced in Keyfactor Command version 11.0, but is retained for backwards compatibility for use with select Keyfactor API endpoints.

Support for OAuth 2.0

Release 11.0 of Keyfactor Command introduces support for open authorization (OAuth) 2.0 compliant identity providers as an alternative to Microsoft Active Directory. Keyfactor offers the Keyfactor Identity Provider, which is a lightweight application that is easily installed in the same environment as Keyfactor Command and can be used to provide standalone authentication with user and group management for Keyfactor Command. It may be used directly or federated to another open authorization (OAuth) 2.0 compliant identity provider (e.g. Okta, Ping Identity). Keyfactor Identity Provider runs in a Linux-based container (e.g. Docker).

The Management Portal has been updated to include a new system setting, *Identity Providers*, that provides the ability to view and edit the identity provider configured for your Keyfactor Command implementation. Active Directory as an identity provider cannot be edited. New identity providers should be added through the Keyfactor Command Configuration Wizard. Authentication with only one identity provider at a time for a given Keyfactor Command server is supported.

When using Keyfactor Identity Provider, there is now a login page to present the user with a login prompt where he or she can choose to login directly to Keyfactor Identity Provider or click a link that will redirect to the login page of the federated identity provider.

The logout functionality for users authenticated with Keyfactor Identity Provider will log the user out and redirect them to the login page. Users authenticated via Active Directory will receive a message instructing them to “Close your browser to sign out” as per earlier versions of Keyfactor Command.

Identity provider secrets can be managed by a PAM provider, if desired. Please see the *Identity Provider Operations* in the *Keyfactor Command Reference Guide* for more information.

Important: Select substitutable special text tokens provide the ability for parts of the Keyfactor Command system to look up data in Active Directory to use in alerts and workflows. Lookups for these locate the object in Active Directory identified by the user or computer account that requested the certificate from the CA and substitute the contents of the referenced attribute. For example, an alert might reference *requester:displayname* to look up the

⚠ requester's display namein Active Directory.
The substitutable special text tokens that begin *requester:* or *principal:* (but not the standalone *requester* token) are only supported when using Active Directory as an identity provider. They cannot be used with other identity providers since there is no Active Directory to query. Customers wishing to use similar functionality with a different identity provider will need to do API calls to query the identity provider, populate the data into a metadata field, and use the metadata field when populating alert and workflow messages.

Certificate Renewal Tracking

When certificates are renewed in Keyfactor Command via the Management Portal or Keyfactor API, a new *Keyfactor Renewal* record is created. This can be used as a filter for *Ignore Renewed Certificates* in certificate collections. A history record will now show on the original and renewed certificates showing the thumbprint it was renewed from/to.

Keyfactor Universal Orchestrator in a Container

The Keyfactor Universal Orchestrator can now be deployed as a Linux container using a containerization platform such as Docker or Kubernetes. This will make deployments faster and less error prone and can fit into the CI/CD pipeline.

Updates

Changes & Improvements

- **Script/Handler Library in Command DB**
  - POST Extension/Script API endpoint added for registering PowerShell scripts for alerts for certificate requests, SSH and workflow, that results in their storage in the Keyfactor Command database (there is no Management Portal equivalent for security reasons). Alert and custom workflow scripts are now loaded from the Keyfactor Command database instead of the file system and available from dropdowns on any dialogs that use scripts, so users can select from the list of registered scripts. The upgrade process will retrieve scripts from the extension library folder and import into Keyfactor Command DB.
  - When configuring alerts and workflows, script selection now appears as a dropdown in the Management Portal. Only those scripts that have been defined in the Keyfactor Command database and configured for the selected category appear in the dropdown.
  - When upgrading from a version of Keyfactor Command prior to version 11, the upgrade process will search the file location defined in the *Application Settings > Console Tab > Extension Handler Path* setting and add all the files found in that directory to the database with the naming convention of *foldername (_subfolder name, if applicable)_filenname* so it is clear which scripts were imported from which location. The upgrade process will also identify which, if any, of the categories the script is configured for and add that information to the database with the script so that existing scripts will work upon upgrade. The extension

library directory has an added file called *Migration-<date>.txt* that informs the user that this directory has had its scripts migrated to the database.

- **.NET Updates**

  Updates have been made to allow the Keyfactor Command solution to multi target to .NET Framework (NET FX) 4.7.2 and .NET Core 6.0 to allow progress towards the goal of allowing Keyfactor Command to run on Linux-based operating systems and achieving cross-platform containerization. Eventually, the platform will not use NET FX.

- **System Alert Bar Design Change**

  System alerts are now indicated by a bell icon in the header. Clicking on the bell will open a menu of alerts:

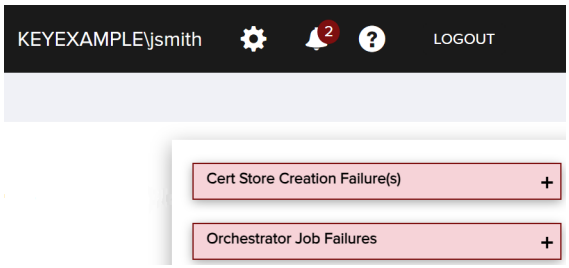

Figure 3: System Alerts

- **One-Click Renewal Enhancements**

  More control was added for allowing one-click renewal on certificates. If you wish to use *One-Click Renewal* for certificates, the **Allow One-Click Renewals** option must be enabled in both the templates and CAs to which you want *One-Click Renewal* to apply (see *Certificate Template Operations* and *Certificate Authority Operations* in the *Keyfactor Command Reference Guide*). For more information about one-click renewals, see *Certificate Operations: Renew* in the *Keyfactor Command Reference Guide*.

- **Enrollment Enhancements**

  Added an information message on the PFX Enrollment page to let the user know the required password length.

  Added three new download options to PFX Enrollment: PEM, DER, and P7B.

  Added the option to select an algorithm and key size when using CSR Generation and PFX Enrollment.

  Added an auto-select option to the enrollment pages. When a template is selected that has 2 or more available CAs, the default option in the CA drop down will be auto-select, which will use the first eligible and reachable CA to do the enrollment.

  PFX Enrollment, CSR Enrollment, and CSR Generation now support RSA key size 3072.

- **New Email Metadata Type**
  For new databases the standard *Email-Contact* metadata field will be created as the new email metadata type field. For existing databases, any *Email-Contact* data will be unaffected, but the new metadata type will be available.

- **Alerts**

  Expiration alerts will no longer determine the time range based on the *LastExpiration* service settings field. Instead, each expiration alert will have its own last execution time. When created, the value will be set to the current UTC time. The next time it is executed, the time span will be from the time in the LastExecution field and the current UTC time. After execution, the time will update to the current UTC time. Previously, alerts would update the LastRun field in the service settings. This no longer happens. The LastRun field was removed. Users can still see when alerts were last run by looking at the event logs.

- **Installation and Configuration Changes**

  The configuration wizard service tab has been changed. Individual Keyfactor Command Service job check boxes have been removed, leaving only the **Start service on bootup** checkbox. By checking **Start service on bootup** all jobs default to **true**. A given job can be disabled by changing this setting to **false** in the appsettings.json file (see *Keyfactor Command Service Job Settings* in *Keyfactor Command Reference Guide* for more information).

  The configuration wizard settings *{filename}.cmsfg* file in Keyfactor Command version 11 has a new <AdminUsers> section, where the settings for the **Administrative Users** tab of the configuration wizard are saved/defined. This new section contains the additional parameters for the new identity provider and security protocol in v11. In previous Keyfactor Command versions, the **Administrative Users** were saved/defined in the *<Console>* section, *<AdminUser>* parameter. Running the configuration wizard for v11 with an older version *{filename}.cmsfg* will populate the configuration wizard **Administrative Users** tab from the details in the *<Console>* section, *<AdminUser>* parameter. Once populated, additional information will need to be provided on the **Administrative Users** section for v11.

- **Supported Browsers**

  The supported browsers for Keyfactor Command have been updated to:

  - Chrome: 99.0.4844.74
  - Firefox: 98.0
  - Microsoft Edge: 99.0.1150.30

- **Certificate Search Updates**

  Added download options to the Certificate Search page. The options are now: PEM, DER, P7B, PFX, ZIP PEM, and JKS. (Previously, the options were PEM, SER, P7B, and PFX.)

  Users without renew certificate permission will no longer see the Renew button.

  On the Certificate Store - View Inventory dialog when you click **Query Certificate** on a selected inventory item, Keyfactor Command will include the *Revoked* and *Expired* check boxes if either condition is true for that certificate.

  When a certificate collection is saved, the collection record will include the *Revoked* and *Expired* check box state and these will load in the appropriate state when the collection is loaded.

- **Certificate Templates**

On the Certificate Templates grid, the Key Type column is renamed to Key Types and displays all the template's key algorithms and key sizes in a comma-separated list with format *KeyType KeySize*. For example, a template that supports all key types would look like this: RSA 2048, RSA 4096 and ECC p-256, ECC p-384. Currently, the supported key types are RSA, ECC, ECDSA, Ed448, Ed25519. There is also a new section *Key Types and Sizes* on the *Template > Details tab* which list all key type and size information for the template.

Support is now provided in Keyfactor Command for a single certificate template containing multiple key types, sizes and/or curves.

- **Certificate Store Types**

  Certificate Store Type Entry Parameters can now be added, edited, or deleted regardless of whether the certificate store type is in use or not. It should be noted that the parameter type (String, Bool, Multiple Choice, or Secret) cannot be changed.

  Certificate Store Type Custom Fields can be added, edited, and deleted even if the type is in use. The custom field's type, however, cannot be changed after it is created. If you want to change that, you must delete the property and add it back with a different type.

  Certificate Store Type Short Name and Name fields can be edited. The Custom Capability field length has been expanded to 64 characters.

- **Keyfactor Command Service**

  New Keyfactor Command Service related application settings have been added around lock acquisition for ensuring that only one Keyfactor Command Service job runs at a time across multiple servers. On the Console > General tab:

  - Lock Timeout (seconds) - The amount of time to attempt to acquire a lock.
  - Lock Heartbeat Interval (seconds) - How often to update the lock to keep it alive while running a long running Keyfactor Command Service job.
  - LockHoldTimeout (seconds) - How long to wait after the last successful heartbeat interval before the lock is considered to be lost and can be acquired by another machine.

- **Certificate Stores**

  The maximum length for certificate store reenrollment jobs has been increased from 256 chars to 850 chars.

- **Custom Extension Handling Updates**

  To reduce the number of assemblies listed in the shared assemblies file, a new extension registration option, *LoadInUpstreamContext*, is available. When set to *True* it causes all of the extension's assemblies to be shared with the calling assembly. The *SendEmailOrchestratorJobCompleteHandler* should be updated to add and set *LoadInUpstreamContext* to *True* (see Editing Job Completion Handlers 1).

- **Logging**

  Improved logging for EJBCA Gateway configuration connection errors.

  Correlation IDs are now returned in the API 's HTTP response header as *X-Keyfactor-Correlation-Id*. These match the correlation GUID tracked in the log files.

- **Improved CA Synchronization Performance**

  Improved performance for synchronizing CAs that have a large number of duplicate keys.

- **Dashboard**

  The ability to authenticate to the Dashboards and Reports Engine, Logi Analytics, no longer requires IP addresses.

  The error message that displays in the case of an error condition has been simplified to remove any sensitive information that might reveal inappropriate information to an attacker (e.g. server hostname or IP address). The message will be displayed as: *Your logon credentials failed. Please contact Keyfactor support at support@keyfactor.com for assistance with report customization or migration to the new platform.*

- **SQL Updates**

  Previously, SQL retry logic was controlled by parameters in the web.config files. With the migration to .NET 6, these settings have been moved to the appsettings.json file located in the Configuration folder of the KeyfactorAPI and/or WebConsole folders. Additionally, to allow for the rest of the product to also use the same retry logic, the parameters in the web.config files for projects that are still in .NET framework have been moved to the appsettings section of the web.config. See *Keyfactor Command Changing SQL Retry Settings* in the *Keyfactor Command Reference Guide* for more information.

- **Orchestrator Management**

  The WebAgentServices\Configuration\appsettings.json file has some added configuration settings for managing orchestrator behavior. For more information, see *Keyfactor Command Web Agent Services* in the *Keyfactor Command Reference Guide*.

  For Keyfactor Command instances configured to use Keyfactor Identity Provider, the Keyfactor Orchestrator API calls support the use of bearer tokens.

- **PAM**

  Added an optional capability to the Universal Orchestrator to ignore certificate store passwords from Keyfactor Command and obtain them from a client-side PAM provider instead. For more information, see *Installing Custom PAM Provider Extensions* in the *Keyfactor Command Reference Guide*.

  In previous versions of Keyfactor Command, once a PAM provider had been associated with an orchestrator whose certificate stores were configured to retrieve their secrets from a PAM provider, it was not possible to edit the PAM provider configuration. In Keyfactor Command v11, the PAM configuration can be edited even if the configuration is in use. Note that the PAM configuration can not be deleted if it is in use.

  PAM provider role based access controls have been updated. Previously, certificate store containers were used to provide the security for PAM providers. Since PAM is now being leveraged across Keyfactor Command, this has been removed and replaced with a new security role permission for PAM providers.

- **CA Policy Module**

When upgrading from a previous version of the Keyfactor CA Policy Module, refer to *Upgrading: Keyfactor CA Policy Module* in the *Keyfactor Command Upgrade Overview* for important upgrade guidance and an upgrade script.

- **Workflow**

  Added a new REST API step in the workflows to support the use of an OAuth bearer token.

  In addition to the tokens in the dropdown in Workflows, any data in the current data bucket can be referenced by entering an appropriate reference string. For example, to return the CSR for an enrollment request you can use $(CSR). Refer to the CurrentStateData field in the response to the GET /Workflow/Instances/{instanceId} API method for information on all the data found in the current (as opposed to initial) data bucket (see *GET Workflow Instances Instance ID* in the *Keyfactor API Reference Guide*).

  The PowerShell *Set Variable Data* workflow step now allows support for the ConvertFrom and ConvertTo cmdlets.

Fixes

- Denied alerts configured with metadata fields as substitutable special text tokens in the message were not getting the metadata values inserted on alert generation.
- The GET /OrchestratorJobs/JobStatus/Data endpoint was returning a 404 error even with a valid jobHistoryId.
- Template names were being treated as case sensitive during enrollment API calls. In addition resolution this issue, logging improvements were made to assist in troubleshooting similar items in the future.
- The CSR generation page would not allow input of subject information if the RFC 2818 compliance policy was turned on at either the template or system-wide settings level.
- Deleting a CA and then re-adding it was causing the CA sync to get stuck.
- OCSP files could not be uploaded in the Management Portal.
- Editing certificate metadata caused the certificate import date to change.
- Metadata fields could not be deleted if they had custom settings defined on a template.
- Pending certificate requests without CNs could not be approved.
- The dashboard was not accurately reflecting the counts of certificate collections that included the revoked and expired certificates.
- The certificate store type “Depends On” radio button would not display the available fields when selected.
- The Metadata, Enrollment RegExes, and Enrollment Defaults tabs on Certificate Templates were incorrectly displaying multi-select checkboxes.
- The Dashboard could hang when trying to remove a widget.
- An LDAP CRL could hang when loading the Dashboard Revocation Monitoring widget.

- Reports that have templates as parameters were not allowing the template parameter to be selected.
- When saving an HTTPS CA, on the Authorization methods tab, there is a database field length limit that applies to the secret supplied for the authentication certificate. Errors encountered with a lengthy secret for the certificate have been resolved by saving only the end-entity certificate to the database, rather than the whole chain.
- Certificates that were part of a certificate renewal chain could not be deleted. Now, whenever a certificate is deleted that is a part of a certificate renewal chain, the certificates on either end of the deleted certificate(s) will have their certificate histories updated to show that either a certificate before or after the certificate was deleted in the renewal chain of that certificate.
- Prior to Command v11, it was possible to create a Keyfactor schedule via the API that was impossible to display via the Management Portal schedule picker. For example, an interval that is not on the interval dropdown (e.g., every 107 minutes). Now, the API will also not allow interval values to be saved that cannot be represented in the Management Portal. Previously saved schedules that don't meet these restrictions can still be parsed and used, but not edited and re-saved, as a result errors may be encountered on re-saving invalid schedules that were created via the API.

Deprecation

> ⚠ **Important:** The Classic API, also known as the CMS API or CMSAPI, has been deprecated in Keyfactor Command version 11.0. Customers should migrate all uses of the Classic API to the Keyfactor API prior to upgrading to Keyfactor Command version 11.

- The Keyfactor Java Agent will be deprecated in a future release of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at: https://-github.com/Keyfactor/remote-file-orchestrator.

  For more information, see *Installing Custom-Built Extensions* in the *Keyfactor Orchestrators Installation and Configuration Guide*.
- The following store types have been deprecated and will no longer be shipped with Keyfactor Command:
  - All F5 store types
  - AWS
  - NetScaler
  - IIS certificate store types

  These store types will not be created with new databases. Upon upgrading, they will be removed if the customer does not have any stores or containers defined for the type (per type. i.e. if an AWS store or container is defined and the other types are not, the AWS type will not be removed on upgrade, but the other types will be removed). If you would like to manage these types of stores, you will create new certificate store types as per the appropriate Keyfactor-built custom

extension for managing the selected store type (see *Installing Custom-Built Extensions* in the *Keyfactor Orchestrators Installation and Configuration Guide*).

- The built-in functions for IIS and FTP certificate store management in the Keyfactor Universal Orchestrator have been deprecated in Keyfactor Command version 11.0.

  Customers should migrate all uses of IIS certificate store management to the Keyfactor Universal Orchestrator with the IIS custom extension publicly available at: https://-github.com/Keyfactor/iis-orchestrator.

  For more information, see *Installing Custom-Built Extensions* in the *Keyfactor Orchestrators Installation and Configuration Guide*.

  For customers who have FTP certificate store type, when you upgrade Keyfactor Command to version 11 your FTP stores and related data will not be removed on upgrade. In order to manage the FTP certificate store, you will need to use an older version of the UO that still has the FTP extension.
- The prescript and postscript functionality of the Keyfactor Universal Orchestrator has been replaced by other functionality in Keyfactor Command such as that provided by Keyfactor Command workflows (see *Workflow Definitions* in *Keyfactor Command Reference Guide*). As a result, prescript and postscript functionality has been deprecated and will be removed from a future release.
- Windows Orchestrator has been deprecated in v11. The reason for this change is that there has not been a new version since Keyfactor Command v8 and our policy is to support orchestrators for the latest two (2) versions of the products. Additionally, all Windows Orchestrator functionality is available in the new Universal Orchestrator and it's associated extensions.
- In preparation for separating the vSCEP web application into its own solution, the configuration wizard no longer provides the option to install vSCEP. Customers wishing to upgrade and also still use vSCEP should install vSCEP on a separate server. The separate vSCEP application will be released at some point in the future.

Future Changes

Known Issues/Limitations

- Updating a workflow step's type without changing the unique name will result in an error. To get around this issue, when changing a workflow step's type also change the step's unique name. Another work-around is to delete the step you no longer want and add the new step you do want.
- If the .NET Hosting Bundle is installed before IIS is installed, you may have to repair the hosting bundle after IIS is installed. The official Microsoft page for the hosting bundle specifically mentions this scenario is at: https://learn.microsoft.com/en-us/aspnet/core/host-and-deploy/iis/hosting-bundle?view=aspnetcore-7.0#install-the-net-core-hosting-bundle.
- Although permissions for a container can be viewed or modified using the permission option on the certificate store containers tab, Keyfactor recommends best practice is to manage permissions as part of the overall permission configuration on the Security Roles and Claims page, as additional system-wide configuration settings are available there that cannot be viewed or modified from certificate store containers - permissions page.

- Any renewal instances that were in-flight within the workflows that have been started in version 10 will not create a new renewal link in the certificate's renewal chain unless the workflow instance was created after an upgrade.
- In Keyfactor Command v11, the v1 security role endpoints only work for Active Directory users. If you move to OAuth and have any integrations against certain security roles APIs, they will not work. Security API endpoints that support OAuth have been added as version v2 of the endpoints.
- The Management Portal and the Keyfactor API do not display duplicate SANs. For certificates that have duplicate SANs, the Management Portal and Keyfactor API only show the unique list and count of the certificate SANs. For example, a certificate with four SANs, of which three are the same (duplicates), would only show two unique SANS on the Certificate Details page and the GET Certificates Keyfactor API method.
- SSH management in Keyfactor Command with the Keyfactor Bash Orchestrator is only supported when using Active Directory as an identity provider (see *Selecting an Identity Provider for Keyfactor Command* in the *Keyfactor Command Server Installation Guide*). The SSH option in the Management Portal will only appear when Keyfactor Command is installed using Active Directory as an identity provider (and with a license that supports SSH).
- Orchestrator auto-registration in Keyfactor Command is only supported when using Active Directory as an identity provider (see *Selecting an Identity Provider for Keyfactor Command* in the *Keyfactor Command Server Installation Guide*). If you need auto-registration with Keyfactor Identity Provider authentication, see *Custom Auto-Registration Handlers* in the *Keyfactor Command Reference Guide*.
- The configuration options on the Keyfactor Universal Orchestrator for remote CA management require that the *AdditionalCertificateAuthoritiesAllowed* value be set to **true** regardless of whether you populate the *CertificateAuthorities* section with your CA information.
- In version 11, if you attempt to access the Management Portal when the SQL server is offline, you will get a blank page with a 503 unavailable error. Note that an IIS reset will be needed to connect to the Management Portal when you restart the SQL server. In version 10 and prior, you would get a screen that said that there was an issue.
- When using Keyfactor Command on a non-domain-joined server, there are some areas in the Keyfactor Command Management Portal that may cause confusion since they reference Active Directory. These places are not going to be addressed for 11:
  - Certificate authority import may show an empty dropdown.
  - The orchestrator auto-registration Validate Users function will return a *could not be resolved* error.
- When querying a workflow by ID with the *contains* or *starts with* operators and a partial id value an error is reported that it is an *Invalid Id value*.
- In the configuration wizard under Authentication, there is a field for *Identity Provider Hint*. This field has been deprecated and will be removed in a future version of Keyfactor Command.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

To reflect the changes made during the .NET 6 migration, in Keyfactor Command version 11, the *Keyfactor API Reference and Utility* (Swagger) link from the Management Portal is changed as follows:

- The URL for the *Keyfactor API Reference and Utility* changed from https:/*keyfactor.keyexample.com*/**KeyfactorAPI**/ref/index#

  to https://*keyfactor.keyexample.com*/**KeyfactorAPI**/swagger/index.html

  Tip: Where *keyfactor.keyexample.com* is the Keyfactor Command Management Portal server and **KeyfactorAPI** is the virtual directory assigned to the Keyfactor Command Management Portal in the configuration wizard.

- There is no longer an individual URL link to each API endpoint within the *Keyfactor API Reference and Utility* (such as *https://keyfactor.keyexample.com/KeyfactorAPI/ref/index#!/Certificate/Certificate_QueryCertificates*) at the bottom of each *Keyfactor API Reference Guide* documentation page—just the one link to the main page of the *Keyfactor API Reference and Utility*.
- At the top of the *Keyfactor API Reference and Utility* under the title there is a link to allow easy access to the raw OpenAPI JSON.

Important: The Classic API, also known as the CMS API, was deprecated in Keyfactor Command version 11. All uses of the Classic API should be migrated to the Keyfactor API prior to upgrading to Keyfactor Command version 11.

Table 1: API Change Log

Endpoint	Methods	Action	Notes
AppSetting	GET, PUT	Added	
AppSetting/{id}	GET	Added	
AppSetting/{id}/Set	PUT	Added	
AppSetting/{name}/Set	PUT	Added	
CertificateAuthority/SourceCount	GET	Added	
CertificateAuthority/ConfigurationTenants	GET	Added	
CertificateAuthority/HealthMonitoring/Schedule	GET	Added	

Endpoint	Methods	Action	Notes
CertificateAuthority/AlertRecipients/CAHealthRecipients	GET	Added	
CertificateAuthority/AlertRecipients/CAHealthRecipients	POST	Added	
CertificateAuthority/AlertRecipients/CAThresholdRecipients	GET	Added	
CertificateAuthority/AlertRecipients/CAThresholdRecipients	POST	Added	
CertificateAuthority/AlertRecipients/CAHealthRecipients/{id}	DELETE	Added	
CertificateAuthority/AlertRecipients/CAHealthRecipients/{id}	GET	Added	
CertificateAuthority/AlertRecipients/CAHealthRecipients/{id}	PUT	Added	
CertificateAuthority/AlertRecipients/CAThresholdRecipients/{id}	DELETE	Added	
CertificateAuthority/AlertRecipients/CAThresholdRecipients/{id}	GET	Added	
CertificateAuthority/AlertRecipients/CAThresholdRecipients/{id}	PUT	Added	

Endpoint	Methods	Action	Notes
CertificateAuthority/Import	POST	Added	
CertificateAuthority/ConfigurationTenants	GET	Changed	The endpoint is now renamed to GET /CertificateAuthority/AvailableForests and the definition is changed to: Returns a list of available forests that are in Active Directory.
Certificates/CSV	GET	Added	
Certificates/IdentityAudit/{id}	GET	Added to V2 definitions	This API endpoint is available in both the V1 and V2 definitions in the Keyfactor API Reference and Utility and acts exactly the same in both.
CertificateCollections/{id}/Permissions	POST	Removed	Instead use POST Security/Roles/{id}/Permissions/Collection.
CertificateCollections/{id}	DELETE	Added	
CertificateCollections/NavItems	GET	Added	
CertificateCollections/CollectionList	GET	Added	
CertificateCollections/{id}/Favorite	PUT	Added	
CertificateStores/Server	GET, POST, PUT	Deprecated	
CertificateStoreTypes	GET	Changed	The API will return ALL certificate store types if at least one of these conditions are met: • The end-user has one of the /certificate_

Endpoint	Method-s	Action	Notes
			stores/read/ global permissions. • The end-user has permission to at least one certificate store container.
ComponentInstallation/{id}	DELETE	Added	
ComponentInstallation/	GET	Added	
EventHandlerRegistration/{id}	GET, DELETE, PUT	Added	
EventHandlerRegistration/	GET, POST	Added	
Extensions/Scripts/{id}	DELETE, GET	Added	
Extensions/Scripts	GET, POST, PUT	Added	
IdentityProviders/{id}	GET, PUT	Added	
IdentityProviders	GET	Added	
IdentityProviders/Types	GET	Added	
Permissions	GET	Added	
PermissionSets/{id}	GET, DELETE	Added	
PermissionSets	GET, POST, PUT	Added	
Scheduling	POST	Added	
Security/Containers/{id}/Roles	GET,	Added	

Endpoint	Method-s	Action	Notes
	POST		
Security/Audit/Collections/{id}	GET	Added	
Security/Claims/{id}	GET, DELETE	Added	
Security/Claims	GET, POST, PUT	Added	
Security/Claims/Roles	GET	Added	
Security/Identities	GET	Changed	The non-working query string field has been removed.
Security/Roles/{id}/Permissions/PamProviders	GET, PUT	Added	
Security/Roles (V1) Security/Roles/{id} (V1) Security/Roles/{id}/Identities(V1) Security/Roles/{id}/copy(V1)	GET, POST, PUT	Deprecated in V1	All SecurityRoles API endpoints (except DELETE /{id}) have been deprecated from the V1 API, as they only work against Active Directory users. There are new Security/Roles endpoints in the V2 API
Security/Roles(V2) Security/Roles/{id}(V2)	GET, POST, PUT	Added in V2	Security/ Roles API endpoints have been recreated in V2 API to work with both OAUTH and AD users.
Templates/{id}	GET	Changed	Now returns an object with a TemplatePolicy property and a KeyAlgorithms property that show the policies and algorithms the template supports.
Templates/Import	GET, POST	Changed	Now supports multiple algorithms.

Endpoint	Method-s	Action	Notes
Templates/Settings	GET, PUT	Changed	The Template Policy property used to update global application settings now contains four properties: ECDSA, RSA, Ed448, and Ed25519. These replace the AllowEd448, AllowEd25519, RSAValidCurves, and ECCValidCurves.

2.3 Major Release 10.0 Notes

September 2022

We're thrilled to announce Keyfactor Command 10.0, which includes some major new features and updates to improve the user experience, enhance automation, and provide native integration with EJBCA.

Important: The MicrosoftECCurveUpgradeModule may fail due to a pre-v10 issue in which Certificate Request Contents were truncated to 4k characters when saved to the database. If your upgrade fails when the MicrosoftECCurveUpgradeModule is run contact Keyfactor Support to obtain assistance with the scripts that will have to be run to fix this issue.

Tip: We encourage customers to contact their customer success manager to discuss the new features and functionality in Keyfactor Command 10, and to schedule an upgrade. Please refer to the *Keyfactor Command Upgrade Overview* for important information about the upgrade process.

Highlights

Workflow Builder

Workflows in Keyfactor Command allow for automation and governance of certificate enrollment and revocation. The workflow builder makes it easy to define workflows within the Keyfactor Command Management Portal to automate event-driven tasks when a certificate is requested (including renewals) or revoked. The workflows can be built with multiple steps between the start and end of the operation that offer a simple way to send notifications, submit approvals, and configure end-to-end automation throughout the environment. This provides for operational agility in an intuitive and easy-to-user tool. Supported built-in steps that can be used in the workflow builder include one or more approval steps supporting one or more approvers, calls to REST APIs, calls to PowerShell,

sending emails, and updating enrollment requests with changes to the submitted subject or SANs, if needed. Custom steps can also be built to address specific needs. The workflow builder provides an easy-to-use experience to create rich workflows with multiple steps.

EJBCA Integration with Keyfactor Command

EJBCA is a robust and highly scalable certificate authority. Keyfactor Command now natively integrates with EJBCA version 7.8.1 or higher without the need for a gateway, providing a simpler architecture. The Certificate Authorities area of Keyfactor Command now allows an administrator to enter connection information to an EJBCA CA to manage certificates and support enrollment. With native EJBCA integration, Keyfactor Command offers an alternative to Microsoft CAs. EJBCA is a much more scalable CA with options for multiple CAs on a single server and high availability configuration options that the Microsoft CA lacks. It can also handle a much larger number of certificates than the Microsoft CA.

CA Gateway 22.1 required for Keyfactor Command v10

Upgrade to AnyGateway 22.1 if using gateways on Keyfactor Command v10.

Expanded Template Functionality

- System-wide settings for enrollment templates have moved from the application settings to the templates page.
- Templates can be configured to set policies for the following at both the template level and the system-wide configuration level:
  - Allow Wildcards
  - Allow Public Key Reuse
  - Enforce RFC 2818 Compliance
  - Supported Key Types
- Added a new configuration tab at both the template level and the system-wide configuration level called *Enrollment Defaults* that allows for defining default values for select certificate subject parts that will auto-populate on the PFX Enrollment and CSR Generation pages.
- *Template RegExes* has been renamed to *Enrollment RegExes*. Regular expressions for certificate subject values can be defined at both the template level and the system-wide configuration level.
- Metadata can be configured on a per-template basis to control which fields are shown during enrollment and what default values they have.
- When enrolling with the template, the key size of the request is validated against the template key size. This allows for a key size to be set on a template in Keyfactor Command for validation purposes that can be different than the CA template key size setting.

- If a CSR Enrollment request is made with a key size that is not valid, per the template policy settings, an error will be displayed when you click the **Enroll** button (for example, the CSR has a key size of 2048 but the template policy supports only 4096).
- For PFX Enrollment, the request will contain the minimum settings from the Keyfactor Command presiding template settings.

- During the upgrade process Keyfactor Command prevents duplicate template records from being inserted into the database. Duplicate templates could be found if there are templates in different forests with the same name. If you receive an error message during upgrade, contact Keyfactor Support. We will be able to support you through the process of resolving the issue and completing the upgrade. See the *Keyfactor Command Upgrade Overview* for more information.

Keyfactor API Endpoints

The Keyfactor API now has endpoints for most of the functionality found in the product. See the API Endpoint Change Log on page 53 for information on new and updated API endpoints.

Updates

Changes & Improvements

- **CARecordID Replaces CARequestID**

  The field CARecordID has been added and the field CARequestID has been removed.
- ***Forest* has been Renamed *Configuration Tenant***
  - To broaden Keyfactor Command's compatibility with certificate authorities, the Microsoft-centric term **forest** has been renamed to **configuration tenant**. For EJBCA, there should be one configuration tenant per EJBCA server install. For Microsoft, there should be one per forest. Note that configuration tenants cannot be mixed, so Microsoft and EJBCA CAs cannot exist on the same configuration tenant.
  - Added the ability to search templates by configuration forest and key type. The option to search by forest has been retained for backwards compatibility.
- **SQL Server Connection over SSL**

  As of Keyfactor Command version 10.0, by default Keyfactor Command connects to SQL using an encrypted connection using an SSL certificate configured on your SQL server. Customers should acquire and install an SSL certificate for the SQL server before upgrading to Keyfactor Command 10.0 (see *Using SSL to Connect to SQL Server* in the *Keyfactor Command Server Installation Guide*). If you would prefer not to use an encrypted channel for your connection to SQL, see *Configurable SQL Connection Strings*.
- **SQL Encryption Key Backup**

  When Keyfactor Command is installed, the option is presented to make a backup of the SQL database master key (DMK). In previous versions of Keyfactor Command, this option backed up the service master key (SMK) instead. For more information about how Keyfactor Command uses

the DMK and SMK, see *SQL Encryption Key Backup* in the *Keyfactor Command Reference Guide*.

- **SQL Server 2022 Compatibility**

  Keyfactor Command is compatible with SQL Server 2022.

- **Certificate Requests**
  - The Certificate Requests page is now sorted in descending order by submission date by default. This has been done to cause the more recent requests to appear at the top of the page.
  - The Certificate Requests page is now separated into tabs for pending, external validation, and denied/failed certificate requests.
  - The Denied/Failed tab on the Certificate Requests page now includes only certificate requests denied through Keyfactor Command (see *Viewing Certificate Requests* in the *Keyfactor Command Reference Guide*).
  - The Revoked view filter has been removed from the Certificate Requests page since the expectation is that Keyfactor Command workflows will be used for enrollments and the history can be viewed as part of that (see *Workflow Instances* in the *Keyfactor Command Reference Guide*).
- **Alerts**
  - When an alert is copied, " - Copy" is appended to the display name to prevent alert display names being duplicated.
  - To aid in clarity, changed the wording on templates when configuring alerts from *None* to *All Templates*.
- **SMTP Application Settings**

  When making changes to the SMTP configuration, the test email can be sent without saving the configuration changes.

- **Certificate Authorities**
  - Added an option to delegate enrollment requests to the Authorization Methods tab. This is in addition to the option to delegate management functions. This allows Keyfactor Command to delegate the authenticated user's credentials to the CA during enrollment to provide end-to-end authentication without unpacking the credentials at the Keyfactor Command layer. If this is not enabled the Restrict Allowed Requesters setting will be used instead. Please see the *Certificate Authority Operations: Adding or Modifying a CA Record Authorization Methods Tab* in the *Keyfactor Command Reference Guide* for more information.
  - When configuring a new certificate authority in the Management Portal, there is now an option to test the connection to the CA before saving the configuration, and CAs will be tested and must be verified and valid to be saved.
  - Updated the CA synchronization so that it logs a message if it could not chain a certificate up to a CA in the system instead of throwing an error.

- Added a new application setting, *CA Sync Consecutive Error Limit*, which controls the number of times an error can occur before the synchronization job is abandoned.
- There is no longer the need to register offline CAs, as the root/policy CA certificates can be imported from the issuing CA sync without them. Additionally, the new CA validation makes it impossible to save offline CAs.

- **Certificate Stores**
  - Added the ability for users with only container-level permission to create and use certificate stores in the container, including certificate store types that have a server component. Users will not be able to access certificate stores outside of the containers they have permissions to manage. (Previously, users needed to have Certificate Store Manage permissions in order to change client machine credentials as certificate store servers was shared across all certificate stores with the same type and server name. Now, certificate store servers are partitioned by container.)
  - Added the ability to import PEM certificates that have comments in them when doing an inventory of an F5 REST certificate store.
  - On the Discover tab the label for *Approve* has been changed to *Manage* for clarity.
- **Dashboard and Reporting**
  - The Risk header can now be hidden via security role permissions.
  - Some cosmetic updates have been made to the Risk header.
  - The Collections Dashboard widget is limited to only displaying the first 25 collections configured to be on the dashboard. It sorts the list alphabetically.
  - The stale date is visible in the CRL Monitoring Dashboard widget as a new column and is called *Next Publish by Date*. The stale date should not be used for calculating the status of the CRL. A stale CRL is a valid state and not something that needs to be warned on. If a CRL is stale, the system will check how far it is from expiration and if it is within the warning period it will have a status of *Warning,* or *Valid* if outside the warning period.
  - Keyfactor Command v10 ships with a newer version of Logi Analytics (v14) which drives the Reports and Dashboards. This version provides a number of improvements and fixes some security vulnerabilities.
  - CRL dates are always shown in UTC on the Revocation Monitoring Dashboard.
  - A new report—SSH Key Usage—shows a table which displays a list of SSH keys that have not been used to log on in the given minimum number of days.
  - The Risk header on the dashboard has been updated to avoid awkward text formatting and scrolling when resizing the page.
  - The Risk header titles have been updated for consistency and clarity. Titles referring to expiring certificates are now all in the “Expiring” tense and consistent with each other. *Weak Keys* has been renamed to *Certs with Weak Keys*.

- The *Certificate Count by Template* report has been updated so that it takes the same parameters as the *Certificate Count per User by Template* report for consistency. This included changing the Evaluation Date to *Start Date* and adding an *End Date* field.
- All reports have been updated to reference UTC time to avoid confusion about which time zone is being applied.
- The *PKI Status for Collection* report has been updated to provide clarity on the meaning of *Total Active Certificates*.

- **Agent, Orchestrators, and Orchestrator Management**
  - The Orchestrator Details dialog has been updated to show more information about the orchestrator:
    - Legacy Thumbprint
    - Current Thumbprint
    - Last Thumbprint Used
    - Last Register Status
    - Certificate Rotation Status
  - The Job History now shows the time the job completed.
  - The default value for the *Registration Handler Timeout (seconds)* application setting has been extended to 90 seconds for new implementations only. Keyfactor recommends any existing customers using or planning to use custom registration handlers consider extending this timeout to at least 60 seconds.
  - SSL scan job parts are now grabbed more deterministically to help keep the job assignments more predictable. For more information, see *SSL Network Operations* in the *Keyfactor Command Reference Guide*.
  - The SSL Scan Now option now allows you to select whether to start a discovery job, a monitoring job, or both (see *SSL Network Operations: Initiating a Manual Scan* in the *Keyfactor Command Reference Guide*).
  - The Keyfactor Universal Orchestrator now does CRL checking when contacting Keyfactor Command over an encrypted channel (when you configure the orchestrator with a URL referencing https) both when certificate authentication is used and when basic authentication is used. Previously this was only done when certificate authentication was used. If you attempt to connect your orchestrator using SSL and do not have a valid CRL available to the orchestrator, you will get an error message similar to the following:

    ```
    The remote certificate is invalid because of errors in the certificate chain:
    RevocationStatusUnknown, OfflineRevocation
    ```

    For troubleshooting information, see *Troubleshooting* in the *Keyfactor Orchestrators Installation and Configuration Guide*.

- **Reenrollment**

  A certificate authority and template can now be specified when scheduling a reenrollment job.

- **Certificate Metadata**
  - A certificate metadata field now cannot be deleted if it is in use in a certificate collection definition.
  - When creating a new certificate metadata type, different fields will be displayed depending on the value selected in the Data Type dropdown field. For more information, see *Metadata Field Operations: Adding or Modifying a Metadata Field* in the *Keyfactor Command Reference Guide*.
- **Security Identities and Roles**
  - A search bar has been added to search for the collections and containers in the security roles dialog.
  - Improvements were made to performance when loading a large number of security roles in the portal.
  - When copying a security role, a new disclaimer will appear to advise the user that copying a security role will also assign the new role to all the same security identities as the target role.
  - The security roles dialog has been updated to be a tabbed dialog box.
- **UI Changes**
  - Some edit dialogs have been changed to use sliding panels to accommodate two different views within the same page rather than pop up windows.
  - Added scroll bars to the certificate details pop ups.
  - Added the ability to copy data from grid information (e.g. SSL location information when expanding the certificate locations). Information in a grid field can be **copied** to the clipboard by highlighting text in a grid field and clicking **Ctrl**+**C**.
  - Performance improvements have been made in loading large data sets in the Management Portal results grids.
- **System Alerts**

  The alerts that are displayed in the UI for notification of things like failed orchestrator jobs have been renamed *System Alerts* for clarity.
- **Logging**
  - The Keyfactor API and Orchestrator API logs on the Keyfactor Command server and the log for the Keyfactor Universal Orchestrator include a correlation ID that helps to identify log messages that originated from the same request. The correlation ID is a randomly generated GUID that often appears just after the date in the log entry and is the same for all log messages for the given request until the request completes.
  - Lowered the logging level for the user's authentication from Info to Trace to avoid cluttering log files.
- **Mac Auto-Enrollment**

The Mac auto-enrollment process now identifies all the CAs that have the auto-enrollment template(s) available for enrollment and makes a determination as to whether the enrolling user has permissions to enroll on a CA and whether that CA is online before submitting a request to the CA. Previously, a CA was selected randomly among the CAs that had the template(s) available without regard to the user's permissions on the CA or the availability of the CA.

- **Auditing**

  Orchestrator reset, approval, disapproval will now properly audit under the new *Orchestrator* category and their respective operation.

- **Installation**
  - On installation, Keyfactor Command creates an initial record in the DatabaseUpgradeLog table that indicates the exact version of Keyfactor Command that created the database. This can be helpful for troubleshooting.
  - If you are upgrading from an older version of Keyfactor Command the installation directory changed, as of Keyfactor Command v9, to C:\Program Files\Keyfactor. Move any scripts or files that are held in the old directory structure to the new location.

- **Policy Modules**

  The policy modules have been migrated to leverage .NET Core.

- **Custom Registration Handlers**

  A custom registration handler can now be designed to enroll against a specific certificate authority and template combination. The registration handler chooses which combination to use. If no combination is requested by the registration handler, then the certificate authority and template from the application settings are used. For more information, see *Register a Client Certificate Renewal Extension* in the *Keyfactor Orchestrators Installation and Configuration Guide*.

- **Application-Level Encryption Certificate Thumbprint**

  The reference thumbprint for the application-level encryption certificate, if configured, is now stored in the registry on the Keyfactor Command server(s) instead of the SQL database to provide a further level of separation from SQL.

Fixes

- **Keyfactor Command**
  - Revocation Monitoring Dashboard panel no longer stalls as perpetually "Loading" for OCSP endpoints.
  - Certificate subjects for PFX enrollment via the legacy API have been fixed so they can be formatted according to the API.CertEnroll.Pkcs12CertificateSubjectFormat app setting.
  - Fixed an issue when parsing the CSR so that CSRs containing IP or Email SANs no longer cause excess warnings in CA syncs, and IP and Email SANs show up in the pending request details.

- Fixed an issue where synching external certificates would cause an “object reference not set to an instance of an object” error.
- Fixed an issue with revocation monitoring alerts reporting time in the local time zone instead of UTC. Emails now have the time in UTC. The time is explicitly labeled UTC.
- Fixed an issue where special characters like apostrophes would appear HTML-encoded in the collection name.
- Fixed an issue in certificate enrollment where SANs for IPv4 and IPv6 addresses were not being validated properly.
- Fixed an issue where an untrusted certificate chain would prevent the certificate details dialog from opening. An error will still occur if a certificate chain is attempted to be downloaded and the chain build fails, but will not prevent the dialog from opening.
- Fixed an issue where the Identity Audit table wasn’t populating from the Certificate Search page.
- Fixed an issue where unscheduling an orchestrator management job failed to cancel the previously staged job.
- Fixed an issue in enrollment where the subject incorrectly added an extra quotation mark when the subject format default was set in certain ways.
- Fixed an issue where SQL would timeout when deleting over 1,000 certificates from the Keyfactor Command Management Portal.
- Fixed an issue where the gateway configured to run as a domain service account and running on the same server as Keyfactor Command caused RPC errors.
- Fixed an issue where the gateway configured to run as a domain service account caused RPC errors.
- Lowered the logging level for the user’s authentication from Info to Trace to avoid cluttering log files.
- Fixed an issue where PEM files with headers could not convert to DER with BouncyCastle 1.9.0 and Keyfactor.PKI.dll v4.x.
- Fixed an issue for certificate store types with the *Advanced>Supports Custom Alias* setting set to **Forbidden**, so that the custom alias should only show on the Add to Certificate Store page when the **Overwrite** checkbox is checked.
- Fixed an issue where using *Delete All* on the Certificate Search page would not delete revoked and expired certificates.
- Fixed an issue in the *Issued Certificates Per Certificate Authority* report that was caused by having templates with the same name in separate forests.
- Fixed an issue with certificate store inventories where a certificate store that had completed an inventory scheduled for an interval would fail if it then was scheduled to run immediately.

- **Keyfactor Agents and Orchestrators**
  - Fixed an issue so that CRLs are now checked regardless of the authentication method being used by the orchestrator.

- Fixed an issue where permissions were not being set correctly on the appsettings.json and orchestratorsettings.json file that prevented the files being read or updated if the service was running as the Network Service.
- Fixed an issue where a misconfigured orchestrator using certificate authentication would renew certificate multiple times.
- Fixed an issue where an orchestrator's registration session was still allowed even when denied by a registration handler and added an auditing event for the orchestrator session registration.

Deprecation

- **Windows Server 2016**

  As of Keyfactor Command version 10.0, Windows Server 2016 is no longer supported.

- **Deprecated Certificate Search Fields**

  The *KeyfactorRequestId*, *RequestResolutionDate*, and *CARequestId* certificate search fields parsers are deprecated due to native EJBCA support in Keyfactor Command as of v10. Any certificate collections using them must be changed before upgrading to v10+.

- **Archive Key on Templates**

  As of Keyfactor Command v10 we no longer support enrolling for certificates that have the archive key option turned on in the template to enable the certificate to store the private key for the certificate in the CA. Attempting to enroll using a template that has this option turned on will result in the following error:

  ```
  The certificate request failed with the reason 'The request is missing a required private key for archival by the server.'
  ```

- **CA Policy module v7.0**

  You will need to upgrade the CA Policy module to v7.1 before running the Keyfactor Command 10.0 upgrade.

- **Reports**

  The Resolution Date field has been removed from the *Certificate Count by User By Template* report.

Future Changes

- **Microsoft .NET Runtime version 3.1**

  By the end of 2022, Microsoft will no longer be supporting .NET Runtime version 3.1. Currently both Microsoft .NET Runtime version 6.0 (x64) and version 3.1 are supported by Keyfactor.

  If you wish to continue using older versions of the Universal Orchestrator but the newer .NET Runtime, you can update the .NET Runtime version on the orchestrator server without

needing to reinstall the orchestrator (see *System Requirements* in the *Keyfactor Orchestrators Installation and Configuration Guide*).

- **Intune Portal/SCEP Change-over**

  Intune portal change-over will be required for SCEP when the old APIs are shut off by Microsoft's deprecation of ADAL at the end of the year.

Known Issues/Limitations

- When editing a template, changes will be lost without warning if the *Save* button isn't clicked before navigating away. This is slated to be fixed in a future release.
- When editing a template, the checkboxes for the Metadata, Enrollment RegExes, and Enrollment Defaults tabs do not allow for multi-edit. This will be fixed in a future release.
- When copying a security role, the identities associated with the security role will also be copied.
- The Condition Variable field in a step of the workflow builder accepts input values that are not valid. Only true, false and variables that will evaluate to true or false are supported.
- For most certificate stores, the *Client Machine* is the machine where the store is located, and the *Orchestrator* drop-down selects the orchestrator/agent. However, for the Java Keystore, the *Client Machine* field is actually the agent and there is no orchestrator dropdown. This will be made more clear in a future release.
- When creating a new certificate store type, the *Depends On Other* option may not be available when creating the parameter. The workaround is to save the certificate store type and then use edit to update the parameter.
- Using the browser back button after generating a report creates a nested instance of Keyfactor Command in Firefox.
- Occasionally, removing a widget from the Dashboard causes the dashboard to hang. Refreshing the browser should resolve this issue.
- The -ne operator in certificate search does not return NULL results for Boolean metadata fields. For a metadata field such as *Unit* use an advanced search such as *Unit -ne "false" OR Unit eq NULL* to get the desired results.
- The *Certificate Count Grouped by Single Metadata Field* report falsely reports no results if using the default metadata value. This will be fixed in a future release.
- The *PKI Status for Collection* report click-throughs do not retain the *Include Unknown* certificates option when clicking through to the certificate search results page. This will be fixed in a future release.
- SMTP Sender information isn't correctly saved by the Configuration Wizard. This will be fixed in a future release. It is recommended to check the SMTP Configuration page upon upgrade.
- Alert tests do not show certificate information if there is no recipient configured to receive an email even if **Send Alerts** is not selected. This will be fixed in a future release. The workaround is to add an email recipient when running the tests.
- Adding multiple enrollment fields at the same time is only saving the last field entered. This will be fixed in a future release. Workaround is to add and save each enrollment field one at a time.

- The *Certificates in Collection* report falsely reports ECC certificates with a certificate state of *Denied* rather than *Active*, revoked certificates with a certificate state of *Active* rather than *Revoked*, and shows a incorrectly shows a revocation reason of *Unspecified* for certificates with an *Active* certificate state. This will be fixed in a future release.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 2: API Change Log

Endpoint	Methods	Action	Notes
/Agents/{id}	GET	Add	
/Agents/Reset	POST	Add	
/AgentBlueprint	GET	Add	
/AgentBlueprint/{id}	GET, DELETE	Add	
/AgentBlueprint/{id}/Jobs	GET	Add	
/AgentBlueprint/{id}/Stores	GET	Add	
/AgentBluePrint/ApplyBlueprint	POST	Add	
/AgentBluePrint/GenerateBluePrint	POST	Add	
/Alerts/Denied	GET, PUT, POST	Add	
/Alerts/Denied/{id}	GET, DELETE	Add	
/Alerts/Expiration	GET, PUT, POST	Add	
/Alerts/Expiration/{id}	GET, DELETE	Add	
/Alerts/Expiration/Schedule	GET, PUT	Add	
/Alerts/Expiration/Test	POST	Add	
/Alerts/Expiration/TestAll	POST	Add	

Endpoint	Methods	Action	Notes
/Alerts/IssuedAlerts	GET, PUT, POST	Add	
/Alerts/IssuedAlerts/{id}	GET, DELETE	Add	
/Alerts/Issued/Schedule	GET, PUT	Add	
/Alerts/KeyRotation	GET, PUT, POST	Add	
/Alerts/KeyRotation/{id}	GET, DELETE	Add	
/Alerts/KeyRotation/Schedule	GET, PUT	Add	
/Alerts/KeyRotation/Test	POST	Add	
/Alerts/KeyRotation/TestAll	POST	Add	
/Alerts/Pending	GET, PUT, POST	Add	
/Alerts/Pending/{id}	GET, DELETE	Add	
/Alerts/Pending/Schedule	GET, PUT	Add	
/Alerts/Pending/Test	POST	Add	
/Alerts/Pending/Test/{id}	POST	Add	
/CertificateAuthorities	GET	Update	Schedules are now included in the results.
/CertificateAuthorities	POST	Update	Ability to turn off schedules, sessions are abandoned properly, and threshold monitoring schedule is included.
/CertificateAuthorities/{id}	PUT	Update	Ability to turn off schedules, sessions are abandoned properly, and threshold monitoring schedule is included.

Endpoint	Methods	Action	Notes
/CertificateAuthorities/{id}	DELETE	Update	Deletion is now prevented if schedules are associated.
/CertificateCollections	POST	Update	Query parameter no longer needed when a valid CopyFromId is provided.
/CertificateCollections/{id}/Permissions	POST	Deprecated	Replaced by /Security/Roles/{id}/Permissions/Collection.
/Certificates/Analyze	POST	Add	
/Certificates/IdentityAudit/{id}	GET	Add	
/CertificateStoreContainers	POST	Add	
/CertificateStoreContainers/{id}	PUT, DELETE	Add	
/CertificateStores/Server	GET, POST, PUT	To Be Deprecated	Server usernames, server passwords, and the UseSSL flag are managed by the /CertificateStores API endpoints directly as JobProperties using the Properties parameter, replacing the deprecated /CertificateStores/Server API endpoints.
/CertificateStores	GET, POST, PUT	Updated	Server usernames, server passwords, and the UseSSL flag are managed by the /CertificateStores API endpoints directly as JobProperties using the Properties parameter, replacing the deprecated /CertificateStores/Server API endpoints.
/Enrollment/PFX (v2)	POST	Add	
/Enrollment/Settings/{id}	GET	Add	
/JobTypes/Custom	POST	Update	DefaultValue property is no

Endpoint	Methods	Action	Notes
			longer required, validation is now performed on the JobTypeFields/DefaultValue property, validation prevents names containing spaces.
/JobTypes/Custom/{id}	DELETE	Update	Includes validation so that deletion is prevented if at least one associated approved orchestrator implements the capability.
/MacEnrollment	GET, PUT	Add	
/Monitoring/Revocation	GET, POST	Update	Renamed from /Workflow/RevocationMonitoring
/Monitoring/Revocation/{id}	GET, PUT, DELETE	Update	Renamed from /Workflow/RevocationMonitoring/{id}
/Monitoring/Revocation/Test	POST	Add	
/Monitoring/Revocation/TestAll	POST	Add	
/Orchestrators/JobHistory	GET	Update	Added JobId field.
/Orchestrators/ScheduledJobs	GET	Add	
/OrchestratorJobs/Reschedule	POST	Add	
/OrchestratorJobs/Unschedule	POST	Add	
/OrchestratorJobs/Acknowledge	POST	Add	
/Security/Identities/{id}	GET	Add	
/Security/Roles/{id}/Identities	GET, POST	Add	
/Security/Roles/{id}/Containers	GET, POST	Add	
/Security/Roles/{id}/Copy	POST	Add	
/Security/Roles/{id}/Permissions	GET	Add	
/Security/Roles/{id}/Permissions/Global	GET, POST, PUT	Add	

Endpoint	Methods	Action	Notes
/Security/Roles/{id}/Permissions/Collections	GET, POST, PUT	Add	Replaced the /CertificateCollections/{id}/Permissions endpoint functionality.
/Security/Roles/{id}/Permissions/Containers	GET, POST, PUT	Add	Returns only containers that have a permission set for the selected security role.
/SMTP	GET, PUT	Add	
/SMTP/Test	POST	Add	
/Templates	GET, PUT	Update	Includes template-specific policy information.
/Templates/{id}	GET	Update	Includes template defaults.
/Templates/Settings	GET, PUT	Update	Includes global template policies.
/Template/SubjectParts	GET	Add	
/Templates/Global/Settings	GET, PUT	Add	
/Templates/Import	POST	Add	
/Workflow/Certificates/Pending	GET	Update	Now supports query fields of Requester and RequestType.
/Workflow/Definitions/Steps/{extensionName}	GET	Add	
/Workflow/Definitions/{definitionId}	GET, PUT, DELETE	Add	
/Workflow/Definitions	GET, POST	Add	
/Workflow/Definitions/Steps	GET	Add	
/Workflow/Definitions/Types	GET	Add	
/Workflow/Definitions/{definitionId}/Steps	PUT	Add	
/Workflow/Definitions/{definitionId}/Publish	POST	Add	

Endpoint	Methods	Action	Notes
/Workflow/Instances/{instanceId}	GET, DELETE	Add	
/Workflow/Instances	GET	Add	
/Workflow/Instances/My	GET	Add	
/Workflow/Instances/AssignedToMe	GET	Add	
/Workflow/Instances/{instanceId}/Stop	POST	Add	
/Workflow/Instances/{instanceId}/Signals	POST	Add	
/Workflow/Instances/{instanceId}/Restart	POST	Add	

2.3.1 Hot Fix Release 10.4.6 Notes

September 2023

Note: Keyfactor Command 10.4.6 is a hot fix release with a few fixes following the Keyfactor Command 10.4 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Update: The data field sizes for the certificate subject field were increased to support re-enrollment jobs for certificates with long subjects.
- Fix: CA synchronizations were hanging with the following error when a corrupted certificate request (for example, a Pending or External Validation record with no CSR) was encountered:

```
Keyfactor.CertificateAuthorityClient.Microsoft.MicrosoftClient [Error] - Value
cannot be null.
Parameter name: inArray
```

- Fix: In an environment with a large numbers of certificate store containers where the orchestrator user did not have global certificate store read permissions, SSL scan times could be excessively long during the certificate import step. This was due to frequent permission check queries on the containers. This hotfix removes unneeded checks. A workaround is to grant the account running the orchestrator the *Certificate Store Management: Read* permission.
- Fix: Workflow instance details for an enrollment request from an enrollment request with multiple SANs of the same type were only displaying the last SAN in the Management Portal. This was limited to the Management Portal as the Keyfactor API returned the correct data.
- Fix: CSR enrollment with a CSR that included SAN data was also adding any SANs that were provided separately on the CSR enrollment page of the Management Portal, rather than replacing the SANs from the CSR with those entered on the CSR enrollment page. The proper behavior is that the enrollment should use *only* the SANs entered on the CSR enrollment page of the Management Portal if they are provided. If they are not provided on the CSR enrollment page of the Management Portal, the SANs in the CSR will be used on the certificate.
- Fix: SANs entered on the CSR enrollment page of the Management Portal or outside the CSR through the Keyfactor API were not displayed in the workflow instance details page.
- Fix: If the *Allow CSR SAN Entry* application setting was set to false, the Keyfactor API still allowed SANs to be sent in with the certificate request outside of the CSR.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.
- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 3: API Change Log

Endpoint	Methods	Action	Notes
/Enrollment/CSR	POST	Fixed	Includes SANs entered outside the CSR only when the *Allow CSR SAN Entry* application setting is set to true. SANs entered outside the CSR replace SANs in the CSR rather than appending to SANs from the CSR.
/Workflow/Instances	GET	Fixed	Includes SANs entered outside the CSR in workflow instance details.
/Workflow/Instances/AssignedToMe	GET	Fixed	Includes SANs entered outside the CSR in workflow instance details.
/Workflow/Instances/My	GET	Fixed	Includes SANs entered outside the CSR in workflow instance details.
/Workflow/Instances/{instanceId}	GET	Fixed	Includes SANs entered outside the CSR in workflow instance details.

2.3.2 Hot Fix Release 10.4.5 Notes

September 2023

Note: Keyfactor Command 10.4.5 is a hot fix release with a few fixes following the Keyfactor Command 10.4 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://key-factor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Update: PFX Enrollment, CSR Enrollment, CSR Generation, and certificate renewal all now allow 3072-bit RSA keys.
- Fix: An agent registering with Keyfactor Command was causing SQL locking errors in environments with a large number of scheduled jobs.
- Fix: SANs were not displaying on the pending certificate requests page when a CSR enrollment was done while the *Keyfactor SAN Attribute Policy Handler* was installed on the Microsoft CA and configured for the template that was being used to enroll.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.
- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 4: API Change Log

Endpoint	Methods	Action	Notes
/CSRGerneration/Generate	POST	Update	3072-bit RSA keys are supported.
/Enrollment/CSR	POST	Update	3072-bit RSA keys are supported.
/Enrollment/PFX	POST	Update	3072-bit RSA keys are supported.
/Enrollment/Renew	POST	Update	3072-bit RSA keys are supported.

2.3.3 Hot Fix Release 10.4.4 Notes

August 2023

Note: Keyfactor Command 10.4.4 is a hot fix release with a few fixes following the Keyfactor Command 10.4 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Fix: Expiration alert event handlers would fail if they referenced CN or DN in the handler parameters and encountered a null-valued CN or DN certificate during processing of certificates.
- Fix: One-click certificate renewal was not scheduling a job to add the new certificate to the certificate store when the renewal was done by selecting *Continue* instead of *Configure*. The certificate renewal step was completing with both *Continue* and *Configure*.
- Fix: Queries to display a warning about failed or possibly errored orchestrator jobs were impacting SQL performance.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.
- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

No API endpoint changes were made in this release.

2.3.4 Hot Fix Release 10.4.3 Notes

July 2023

Note: Keyfactor Command 10.4.3 is a hot fix release with a few fixes following the Keyfactor Command 10.4 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://key-factor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Fix: Certificate authorities of EJBCA version 8 could not be added to the certificate authorities page due to a failed version check.
- Fix: One-click renewal was encountering an error when trying to renew against EJBCA version 8.
- Fix: Importing templates to Keyfactor Command from EJBCA version 8 failed.
- Fix: External validation certificates being enrolled from public certificate authorities were sometimes resulting in the following error in the Management Portal with no errors in the log:

  ```
  Cannot convert unidentified or null to object
  ```

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.
- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 5: API Change Log

Endpoint	Methods	Action	Notes
/CertificateAuthority/Test	POST	Fixed	EJBCA version 8 is supported.
/Enrollment/Renew	POST	Fixed	EJBCA version 8 is supported.
/Templates/Import	POST	Fixed	EJBCA version 8 is supported.

2.3.5 Hot Fix Release 10.4.2 Notes

June 2023

Note: Keyfactor Command 10.4.2 is a hot fix release with a few fixes following the Keyfactor Command 10.4 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Fix: Certificate profiles from EJBCA could not be imported if they were not the first CA listed in the *Allowed CAs* on the certificate profile.
- Fix: End entity profiles configured in EJBCA with *Any CA* in the *Allowed CAs* were not displaying in the Management Portal CSR and PFX enrollment pages.
- Fix: Upgrading to Keyfactor Command version 10 was very slow in environments with a large number of certificate requests in the database.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.

- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

No API endpoint changes were made in this release.

2.3.6 Hot Fix Release 10.4.1 Notes

June 2023

Note: Keyfactor Command 10.4.1 is a hot fix release with one update following the Keyfactor Command 10.4 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Update: BouncyCastle.Crypto is no longer treated as a shared assembly to allow integrations to be built with newer BouncyCastle classes.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.
- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

No API endpoint changes were made in this release.

2.3.7 Incremental Release 10.4 Notes

May 2023

Note: Keyfactor Command 10.4 is a minor release with incremental fixes and updates following the Keyfactor Command 10 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Changes and Improvements

- Workflow now includes two new types—Certificate Entered Collection and Certificate Left Collection—that are designed to help you monitor the comings and goings of certificates from collections and take actions in the event that a certificate unexpectedly appears or disappears from a collection. You might use one of these workflow types to monitor the Weak Keys collection to be alerted via email when a new certificate is added to the collection after being picked up on an SSL scan. Or you might use one of these workflows to monitor a collection of vital certificates and use a PowerShell or REST request to automatically open a support ticket if one of the certificates goes missing. These workflow types work together with the Keyfactor Command Service to periodically evaluate the collections configured for reporting and then initiate workflows for any certificates that have changed membership in the collections. The automated task runs every 10 minutes by default and is not end-user configurable. By default, a maximum of 1000 certificates can be reported on by any one instance of an automated task. This value is end-user configurable with the Concurrent Workflows, setting (see Table: *Keyfactor Command Jobs Services* in the *Keyfactor Command Server Installation Guide*). Certificate collections that are configured for workflows cannot be edited to prevent triggering a large number of entered/left workflows.
- Certificates with a key type of Dilithium2, Dilithium4, or Dilithium5 may now be imported into Keyfactor Command for management and reporting using the Add Certificate function (see Add Certificate in the *Keyfactor Command Reference Guide*). CA synchronization of certificates with this key type will be supported in a future release.
- Certificates with private keys can now be downloaded in JKS format either in PFX enrollment or certificate search. The JKS option for certificates for private keys is in addition to the PEM and PFX options for download format.

- On certificate download in both PFX enrollment and certificate search, you now have the option to select a chain order for the chain certificates in the resulting output file if you opt to include the certificate chain in the download. The choice is either *End Entity First* (at the beginning of the file) or *Root First*.

Updates and Fixes

- Update: The default timeout on the configuration wizard for Keyfactor Command upgrade job executions has been increased to 30 minutes. See Troubleshooting in the *Keyfactor Command Upgrade Overview* for instructions on customizing the timeout.
- Update: The Keyfactor Universal Orchestrator now includes a configuration setting that allows it to skip checking the revocation status (CRL) of the SSL certificate on the Keyfactor Command server when connecting to Keyfactor Command.
- Update: On a new installation of Keyfactor Command, the **Revoke All** option on the Certificates page—controlled with the *Revoke All Enabled* application setting—will default to disabled. This change will not affect existing implementations of Keyfactor Command.
- Update: The wording on the Revoke All option has been changed to clarify that a revocation is occurring.
- Fix: SSL monitoring scans done with the Universal Orchestrator were failing to report TLS 1.3 timeouts.
- Fix: The maintenance job to remove expired stored private keys that are eligible for deletion was not running as expected on a daily basis to remove the keys.
- Fix: A user could be prompted to save changes to a template when viewing a template without making changes in certain template configurations.
- Fix: The certificate template regular expression ^$ to disallow any values in a field was in a catch 22 state requiring entry of a value in the field because a regular expression was defined for it and requiring no value because of the nature of the specific regular expression, causing the field not to function at all.
- Fix: Delegation was not working as expected for certificate revocation when the certificate authority record in Keyfactor Command was configured to *Delegate Management Operations*.
- Fix: Attempting to create records in Keyfactor Command for two certificate stores with the same name on the same server but of different types produced an error indicating that the second was a duplicate of the first; now stores of different types may successfully be created with the same name.
- Fix: Associating a PAM provider with a certificate store container, placing a certificate store in that certificate store container, and then attempting to set the PAM credentials for that certificate store failed with an error of “The supplied Secured Area is invalid for the selected provider”.
- Fix: Certificates with a SAN type of *DS Object Guid* could not be imported, producing an error of:

```
illegal object in GetInstance: Org.BouncyCastle.Asn1.DLTaggedObject
Parameter name: obj
```

- Fix: Attempting to validate a CA record for an EJBCA CA using the *Test Connection* option would fail if the client authentication certificate configured for the CA had no EKU defined, resulting in an error similar to:

  ```
  There is a problem validating the CA with ID '3' (check the logs for more details):
  Object reference not set to an instance of an object.
  ```

- Fix: Attempting to disapprove an instance of the Keyfactor Bash Orchestrator when the orchestrator had an SSH synchronization schedule configured or an instance of the Keyfactor Universal Orchestrator when the orchestrator had an SSL scanning schedule configured resulted in a 500 error.

Known Issues

- The Audit Log page offers a search comparison value of *Instance Signal* for the audit category but the results grid Category column references this same value as *Workflow Signal*.
- When a one-click renewal is done on a certificate from the Certificate Search page, even though the renewal succeeds, the grid doesn't refresh with the new status.
- The latest version of the Logi reporting engine has functionality which avoids a system timeout issue by periodically pinging the IIS session behind the scenes so that the dashboard doesn't time out when the session has been idle. As a result, the dashboard no longer refreshes after 20 minutes, but invokes this new functionality instead. The settings used to control this depend on the **Session State Timeout** and **Session Auto Keep Alive** attribute settings in IIS. For more information on this see:

  https://devnet.logianalytics.com/hc/en-us/articles/1500009515942-Manage-Session-Timeout

- On an edit, if you change the workflow step type, you must also change the **Unique Name**. Changing the workflow step type without changing the unique name will result in an error similar to the following:

  ```
  System.Collections.Generic.KeyNotFoundException: The given key was not present in
  the dictionary
  ```

  Instead of changing both the workflow step type and unique name, you may be prefer to delete the step and create a new step of the desired type.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints

be disabled to reduce exposure to unauthorized or unintended use.

- The Keyfactor Java Agent will be deprecated in version 11.0 of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 6: API Change Log

Endpoint	Methods	Action	Notes
/Enrollment/CSR	POST	Fixed	Includes SANs entered outside the CSR only when the *Allow CSR SAN Entry* application setting is set to true. SANs entered outside the CSR replace SANs in the CSR rather than appending to SANs from the CSR.
/Workflow/Instances	GET	Fixed	Inclused SANs entered outside the CSR in workflow instance details.
/Workflow/Instances/AssignedToMe	GET	Fixed	Inclused SANs entered outside the CSR in workflow instance details.
/Workflow/Instances/My	GET	Fixed	Inclused SANs entered outside the CSR in workflow instance details.
/Workflow/Instances/{instanceId}	GET	Fixed	Inclused SANs entered outside the CSR in workflow instance details.

2.3.8 Hot Fix Release 10.3.1 Notes

April 2023

Note: Keyfactor Command 10.3.1 is a hot fix release with a few fixes following the Keyfactor Command 10.3 incremental release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Updates and Fixes

- Update: Keyfactor Command disallows installation on Windows Server 2016. This is necessary because Windows Server 2016 does not support PFX files being generated with AES encryption.
- Fix: Expiration reports were failing to generate when the user running the report only had permission to view one certificate collection.
- Fix: Saving a report schedule was failing when the user saving the report only had permission to view one certificate collection.
- Fix: The *Certificate Count Grouped by Single Metadata Field* report was failing when the first metadata field in the report dropdown was left at the default.

Deprecation

- The Classic API will be deprecated in Keyfactor Command version 11.0. All existing uses of the Classic API should be migrated to use Keyfactor API prior to upgrading to Keyfactor Command version 11. If these applications cannot be updated to the newer endpoints then the **Allow Deprecate API Calls** setting must be set to *False* (see Application Setting: API tab in the *Keyfactor Command Reference Guide*). Otherwise, Keyfactor recommends that these endpoints be disabled to reduce exposure to unauthorized or unintended use.
- The Keyfactor Java Agent will be deprecated in a future version of Keyfactor Command. Customers are encouraged to begin planning a migration to the Keyfactor Universal Orchestrator with the Remote File custom extension publicly available at:

  https://github.com/Keyfactor/remote-file-orchestrator

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 7: API Change Log

Endpoint	Methods	Action	Notes
/Reports/{id}/Schedules	POST	Fixed	Reports can be scheduled when the user scheduling the report only has permission to view one certificate collection.

2.3.9 Incremental Release 10.3 Notes

March 2023

Note: Keyfactor Command 10.3 is a minor release with incremental fixes and updates following the Keyfactor Command 10 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Changes and Improvements

- The new **CertStoreContainer** certificate search field shows certificates in a certificate store that is included in the certificate store container specified by the search criteria.
- The Keyfactor Bash Orchestrator added additional support for using an SSSD user store (e.g. Active Directory) on requests to create logons and distribute key information, allowing keys to be managed for domain users. Domain users can be managed with or without preexisting home directories.
- Added the ability to use any symbols when creating a new SSH logon. This is required in order to facilitate creating a logon for an AD user using SSSD.
- The Universal Orchestrator now communicates with IIS certificate stores over TCP port 445 rather then using WinRM and default ports 5985/5986.
- The BASH Orchestrator now returns improved warning messages on the Job History page. See SSH-Bash Orchestrator Job History Warning Resolution.

Updates and Fixes

- Update: The Keyfactor Bash Orchestrator now adds the command *restorecon* to the list of commands the orchestrator service account is allowed to execute via sudo on servers running

SELinux.

- Update: The Keyfactor Bash Orchestrator now trims Windows line breaks from JSON payloads on send and receive and ignores any data in the authorized_keys file that is not a key (e.g. a comment).
- Update: An application setting—*Enable Legacy Encryption*—has been added to enable/disable the use of legacy encryption methods in PFX enrollment. When the value is set to true, the historical algorithm set (3DES/SHA1/RC2) is used for PFX enrollments. When the value is set to false, the newer algorithm set provided by Windows (AES256/SHA256/AES256) is used instead. The default is *false*.
- Update: A script has been added to allow the Keyfactor CA Policy Module to be upgraded from versions prior to 10.0 and retain existing configuration.
- Fix: EJBCA certificates with a leading zero in the serial number could not be revoked; an attempt to do so generated an error.
- Fix: EJBCA CA Config will give a notification if the certificate you selected doesn't meet requirements, and indicate exactly what the requirements are and what your certificate is lacking.
- Fix: The GET /SSL API endpoint was returning duplicate records.
- Fix: The DELETE /Workflow/Defintions/{id} API endpoint was returning an error if the workflow contained steps.
- Fix: Expiration alert tests displayed a blank dialog if the alert was configured with no recipients.
- Fix: The Keyfactor Bash Orchestrator install failed when the service account was provided an extremely long password.

Known Issues

- The dashboard will throw a secure key error if you let the dashboard sit idle for around 20 minutes. The temporary work-around is to refresh the page. It will be investigated in 11 for a possible fix.
- Because a "+" (plus sign) in a URL can represent either a space or a "+", Keyfactor Command has chosen to read "+" as a space. For CRL URLs that require a "+" (plus sign), rather than a space, replace plus signs in your CRL's URL with "%2B". Only replace the plus signs you don't wish to be treated as a space.
- A user without *Global Certificates - Read* and *Global Certificates – Import* will see a permission error dialog when attempting to view an enrollment workflow instance that has **completed**. The only impact of this error is that it will result in the certificate's information not being parsed in the *Instance Review dialog*. Users should not need these permissions to view their completed workflow instances, and so should not be seeing this error. This will be fixed in the next Keyfactor Command release. The raw data is still present. As a workaround, if a user wants to see the parsed data for that certificate, they would have to use the **KeyfactorId** (found on the workflow instance) in the certificate search page using the **CertId**.

API Endpoint Change Log

No API endpoint changes were made in this release.

2.3.10 Incremental Release 10.2 Notes

January 2023

Note: Keyfactor Command 10.2 is a minor release with incremental fixes and updates following the Keyfactor Command 10 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Changes and Improvements

- **Keyfactor Command (formerly, Timer) Service now runs in an HA environment.**

  The Keyfactor Command (formerly, Timer) service can be installed on every server that Keyfactor Command is installed on. This will allow the service to check out jobs via a locking mechanism which will enforce that any jobs are running on only one service at a time. There is a new *CMSTimerService.exe.config* timeout setting for the service locking mechanism `<add key="Keyfactor.TimerJobs.LockTimeout" value="5000" />` which is the lock timeout. It's the number of ms Keyfactor Command will wait to acquire a lock. By default Keyfactor Command will attempt to get a lock for 5 secs and if unsuccessful, an error will be thrown.

- **Workflow Definitions can be Created via Copy**

  A new option is available on the workflow definitions page that allows you to create a new workflow definition by copying an existing workflow definition. When you create a new workflow definition by copying an existing one, the word "copy" will be appended to the end of the definition name and the workflow key (template) will be cleared. Other data from the copied workflow will be retained.

- **Workflow Step Type Windows Enrollment Gateway - Populate from AD**

  A new workflow step type has been added to support enrollment requests from the Keyfactor Windows Enrollment Gateway using client-side templates configured with the subject as *Build from this Active Directory information*. This workflow step type allows the requests to be completed in Keyfactor Command using an EJBCA template that is not configured to build the subject from Active Directory using the Active Directory information (subject, SANs, and/or SID) supplied in the request from the client.

Updates and Fixes

- Update: The maximum number of characters allowed in a certificate store path has been increased from 256 to 722.

- Update: Users now receive a warning if they attempt to use the Back button in a certificate template after making changes without saving.
- Update: Workflow steps of type Email and Require Approval now go to a failed state if an error occurs in sending an email.
- Fix: An issue encountered with upgrading larger databases in v10.1 is fixed in the current v10.2 release which addressed this specific portion of the database upgrade, and should allow upgrade without this issue.
- Fix: Agent Application Settings: An agent will not attempt to retry a job when this setting is set to `0`.
- Fix: Certificate stores of a type that required a server but did not require authentication to access that server could not be saved using the “No Value” options for the server username and password.
- Fix: A base-64-encoded PEM certificate added to a PEM certificate store using the Certificates -> Add Certificate feature was not being correctly formatted for the store.
- Fix: If multiple template enrollment fields were added at the same time before saving, only the most recently added one was saved.
- Fix: The PKI Status for Collection report drill-downs did not include unknown certificates when the *Include Unknown* box was checked. The *Include Unknown* box also worked inconsistently.
- Fix: Custom orchestrators with a status of Disapproved changed to a status of New when their capabilities were changed. Only orchestrators with a status of Active should change to a status of new when their capabilities are changed.
- Fix: Certificate templates with a key size of Ed448 were imported and assigned a key type of 456.
- Fix: On an attempt to edit the parameters of a built-in report with a parameter of type RelativeDate, an error message appeared indicating “A saved parameter with type ‘RelativeDate’ is invalid with a value of ‘false’” and the user was not allowed to edit the parameters.
- Fix: Chain not being passed in Management Add Job.
- Fix: Certificates cannot be queried by KeyfactorRequestId.

Known Issues

- CSR enrollment fails against a standalone CA. This will be fixed in a future incremental release. Customers using CSR enrollment and standalone CAs should wait to upgrade.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 8: API Change Log

Endpoint	Methods	Action	Notes
/Security/My	GET	Add	Returns all the security roles and global permissions for the requesting user.
/Enrollment/CSR	POST	Update	The workflow instance ID has been added to the response.
/Enrollment/CSR	POST	Update	A new PrivateKey input field has been added to support private key retention on CSR enrollment.
/Enrollment/PFX	POST	Update	The workflow instance ID has been added to the response.
/Certificates/Analyze	POST	Update	The endpoint requires Global Certificates-Read or Certificates-Import permissions.

2.3.11 Incremental Release 10.1 Notes

November 2022

Note: Keyfactor Command 10.1 is a minor release with incremental fixes and updates following the Keyfactor Command 10 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 10.0, please review the Major Release 10.0 Notes on page 42.

Tip: Keyfactor recommends that you check the Keyfactor GitHub Site (https://keyfactor.github.io/integrations-catalog/) with each release that you install to check if you will need to download the updated orchestrators to work with that version of Keyfactor Command.

Changes and Improvements

- **Keyfactor Universal Orchestrator Supports gMSA**

  The Keyfactor Universal Orchestrator now supports running its service as a group managed service account (gMSA).

- **SSL Discovery and Monitoring Jobs have Reset Scan Option**

  A new Reset Scan option has been added for SSL discovery and monitoring jobs that allows to you recover from an SSL job that appears to be stuck or crashed.

Updates and Fixes

- Update: All Keyfactor Command (timer) service jobs have consistent start and stop log messages in both the file and Windows Event Viewer.
- Update: A PAM provider can be used directly by the Keyfactor Universal Orchestrator, such that the server does not retrieve, and does not have access to, the credential.
- Update: Ed448 and Ed25519 keys are now supported for certificate enrollment, policy, import and search.
- Update: Improved support for the Keyfactor Command (timer) service—including a job locking mechanism—in High-Availability implementations.
- Fix: GET /SSL is returning duplicate info in some instances with endpoints sharing a common chain.
- Fix: Certificate store Discovery jobs could not be executed.
- Fix: AnyGateway was declaring all requests as new instead of renew or reissue.
- Fix: The SMTP Sender Account was not populated during the installation and configuration process.
- Fix: SSL discovery scan job errors for entries with a null display name.

Policy Module Updates

- Migrated the Policy Modules to .NET Core 6.
- Updated the Policy Module to create a Windows Event Log entry when the current license is within 60 days of expiration.
- Updated the Policy Module installer to include the EnterpriseLite, SubjectFormat and SCEPRequester modules.
- Updated the Policy Handler Configuration so that changes no longer require the ADCS service to be restarted.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 9: API Change Log

Endpoint	Methods	Action	Notes
/Templates	PUT, GET	Update	Ed448 and Ed25519 keys are now supported for certificate enrollment, policy, import and search.
/Templates/{id}	GET	Update	Ed448 and Ed25519 keys are now supported for certificate enrollment, policy, import and search.

Endpoint	Methods	Action	Notes
/Templates/Settings	PUT, GET	Update	Ed448 and Ed25519 keys are now supported for certificate enrollment, policy, import and search.

2.4 Major Release 9.0 Notes

August 2021

Release Highlights

We're thrilled to announce Keyfactor Command 9.0, which includes several new features and updates to improve the user experience, deployment flexibility, and risk awareness.

Highlights from the Keyfactor Command 9.0 release are listed here. More details are available in the New Features, and Updates and Improvements sections further down.

⚠ **Important:** There have been several UI updates to the navigation menu, drop-downs, and application settings. Thoroughly review these changes in the New Features section.

UI Enhancements

- **What problem does it solve?**

  The Keyfactor Command interface should be easy to navigate and use.

- **How does it work?**

  As we continue to improve the Keyfactor Command interface, we've added updates to the navigation menu, application settings, and dialogues, as well as an updated color scheme.

- **What's the benefit?**

  Ease of Use: The Keyfactor Command interface is more intuitive for new and experienced users alike.

New Risk Header

- **What problem does it solve?**

  PKI administrators and application owners want to easily identify risks and upcoming expirations for the certificates they have access to.

- **How does it work?**

  A new fixed header above the dashboard displays expiring, weak, and revoked certificates for an at-a-glance view of risks.

- **What's the benefit?**

Risk Mitigation: Enables administrators to quickly identify the state of their certificates.

New Universal Orchestrator

- **What problem does it solve?**

  The current Windows Orchestrator is only able to run on Windows systems.

- **How does it work?**

  The new Keyfactor Universal Orchestrator runs on .NET Core 3.1, which allows it to be installed on servers/instances running either Linux or Windows.

- **What's the benefit?**

  Flexibility: Enables customers to deploy orchestrators in cross-platform environments.

New Remote CA Gateway

- **What problem does it solve?**

  Certain customers are unable to use Keyfactor PKI as-a-Service due to security or regulatory requirements, but they'd still like to leverage a SaaS-based solution for certificate management.

- **How does it work?**

  The new Remote CA Gateway securely connects on-premise private PKI – Microsoft ADCS or PrimeKey EJBCA – to the Keyfactor Cloud. This allows customers to leverage Keyfactor Command as a Service (SaaS) while keeping their PKI within their datacenter.

- **What's the benefit?**

  Cloud: On-premise customers now have more options to deploy Keyfactor in a SaaS model – while keeping their PKI in-house, if required.

Support for TLS 1.3

- **What problem does it solve?**

  Before Keyfactor Command 9.0, Keyfactor Command did not support SSL/TLS scanning on endpoints using TLS 1.3.

- **How does it work?**

  The Keyfactor Universal Orchestrator supports SSL/TLS scanning on endpoints using TLS 1.3.

- **What's the benefit?**

  Increased Visibility: Organizations will have improved visibility over certificates.

Template-Level Metadata

- **What problem does it solve?**

  Before Keyfactor Command 9.0, certificate metadata could only be applied system wide.

- **How does it work?**

  Now administrators can apply metadata on a per-template basis, which will override system-wide settings for that specific template.

- **What's the benefit?**

  Control: This gives administrators more granular control for metadata in certificate enrollment.

Ecosystem Updates

While separate from the Keyfactor Command 9.0 release, we've recently introduced several new integrations in GitHub to support more certificate authorities, applications, and services.

These include:

- Google Cloud CA Service: A new AnyCA Gateway implementation supports discovery and automation of certificates issued by Certificate Authority Service (CAS).
- Google Cloud IoT Core: The IoT Issued Alert Handler publishes device certificates to various cloud providers, including Google Cloud, Azure, and AWS.
- GoDaddy: The GoDaddy CA Gateway enables enrollment, renewal, re-issuance, and revocation of certificates via Keyfactor Command.
- Sectigo Certificate Manager: The Sectigo CA Gateway enables full lifecycle management of certificate issued by Sectigo via Keyfactor Command.
- Kubernetes: A proxy signs certificate-signing requests (CSRs) through Keyfactor via the Kubernetes CSR signer API.
- Azure Key Vault: Allows customers to inventory and manage certificates within their Azure Key Vault instances.

More information and developer resources can be found in the Keyfactor GitHub.

New Features

UI Enhancements

Tip: We encourage existing Keyfactor Command customers to watch the Keyfactor Command 9.0 UI Walkthrough demo and read through the detailed UI changes listed below before upgrading to Keyfactor Command 9.

Keyfactor Command 9.0 includes significant updates to the UI, as well as several changes to the main navigation menu and drop-downs with a focus on improved usability. Please continue reading to review and understand these changes.

Previously, the navigation menu looked like the example below:

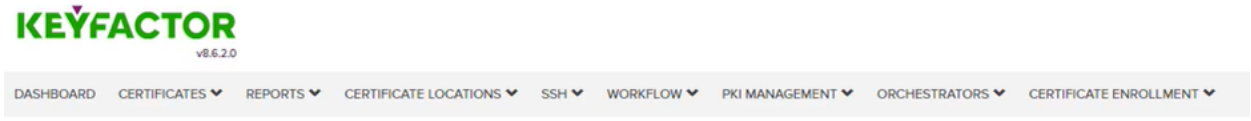

Figure 4: Example Navigation Menu Before Upgrade to 9.0

In Keyfactor Command 9.0, the navigation menu is more concise and user-centric:

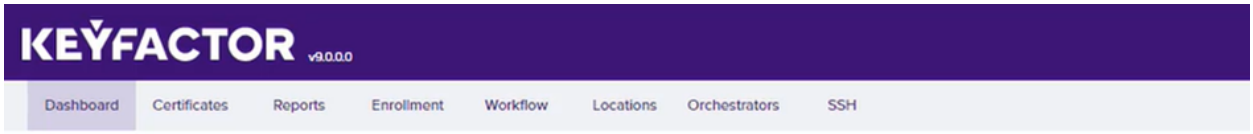

Figure 5: Example Navigation Menu After Upgrade to 9.0

Certificates drop-down

- Add Certificate: The Add Certificate selection is now located in the Certificates tab. Previously, it was accessed via the Certificate Locations tab.

Enrollment drop-down

- Certificate Requests: This option is now found in the new Enrollment tab, rather than the Workflow tab.

Workflow drop-down

- Revocation Monitoring: This option is now located in the Workflow tab. Previously, it was located in the PKI Management tab.
- Expiration: This selection was previously named Expiration Alerts.
- Pending Request: This selection was previously named Pending Request Alerts.
- Issued Request: This selection was previously named Issued Request Alerts.
- Denied Request: This selection was previously named Denied Request Alerts.
- Key Rotation: This selection was previously named Key Rotation Alerts.

Locations drop-down

- Certificate Stores: You will now access the Certificate Stores selection from the new Locations tab. Previously, it was accessed via the Certificate Locations tab.
- Certificate Authorities and Certificate Templates: These menu options are now found in the new Locations. Previously, they were located in the PKI Management tab.
- SSL Discovery: This selection is now located in the Locations tab. It was previously located in the Certificate Locations drop-down.

System Settings menu

- Certificate Store Types: You will now access the Certificate Store Types from the System Settings at the top-right of the screen. It was previously under Certificate Locations.

Certificate Search

- There is a new “ends with” operator. For example:

  ```
  CN -endswith "keyexample.com"
  ```

- A new advanced search option has been added of %ME-AN%. This does a search for account name without domain. For example, the following search in certificate search:

  ```
  NetBIOSRequester -contains "%ME-AN%"
  ```

  Would return certificates requested by the current user as KEYEXAMPLE\jsmith and KEYOTHER\jsmith (assuming the current user is logged in with username jsmith in some domain).

New Risk Header

A *Risk Header* has been added to the Dashboard, which displays relevant information for certificates the user has permissions to. This includes a count of all active certificates, upcoming expirations, expired and revoked certificates, and weak keys (as seen below).

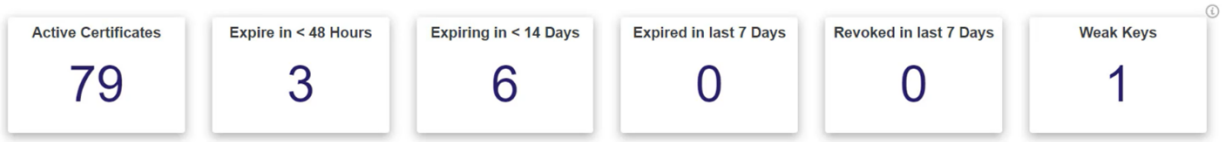

Figure 6: New Risk Header

Note: The new Risk Header is intended to provide an at-a-glance view of key metrics. Unlike items within the dashboard below it, the header cannot be moved or customized.

New Universal Orchestrator

Now available in Keyfactor Command 9.0, the new Keyfactor Universal Orchestrator can perform many of the same functions as the legacy Windows Orchestrator, such as IIS, SSL, FTP and CA management (we will continue to expand its functionality). However, unlike the legacy Windows Orchestrator, the new Keyfactor Universal Orchestrator is able to run on both Windows and Linux servers.

The purpose of orchestrators is to perform SSL scans, manage certificate stores (both Java Key Stores and Windows Certificate Stores), run custom certificate management jobs, inventory CAs, and collect logs to be viewed in the Keyfactor Command Console.

Please review the Deprecation on page 87 section for more information about the eventual deprecation of the legacy Windows Orchestrator. Refer to the *Keyfactor Orchestrators Installation and Configuration Guide* for more information on the new Keyfactor Universal Orchestrator.

New Remote CA Gateway

Before Keyfactor Command 9.0, customers had the option to deploy Keyfactor Command on-premise or hosted in the cloud with a fully managed private PKI as a Service (PKIaaS). Now customers have the additional option to keep their PKI on-premise while leveraging Keyfactor Command in the cloud.

The Keyfactor Remote CA Gateway is the connection point between the new Keyfactor Command as-a-Service deployment model (aka Certificate Lifecycle Automation as a Service or CLAaaS) and a customer's on-premise PKI behind their firewall.

The Remote CA Gateway synchronizes in real-time to provide full visibility and governance over the inventory, enrollment, issuance, revocation and renewal of certificates from your on premise CA, requiring just a single, secure API connection on port 443 back to the Keyfactor Command Cloud.

Template-level Metadata

Certificate metadata fields can now be defined on a per-template basis. Before Keyfactor Command 9.0, metadata fields could only be defined as a system-wide setting.

This allows administrators to apply required, hidden or optional settings to a metadata field on a per-template basis so that only certain metadata fields will appear on certain templates.

System-wide settings for metadata fields can be overridden, so customers can choose which fields are displayed, during enrollment for a certificate, based on the template the user selects when enrolling.

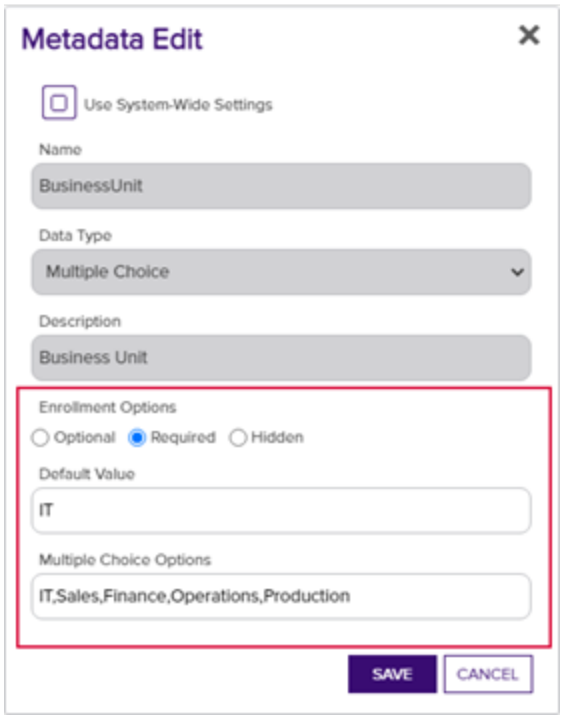

Figure 7: Template Level Metadata

Documentation Structure Updates

Next and Previous buttons have been added to the button row at the top of each page that allow you to navigate through the pages in the documentation in order.

The mini table of contents has been updated to only display by default on pages that contain subpages. This TOC displays—with links—any pages that appear below the current page in the document structure. The TOC button can be used to close and reopen the mini table of contents. The mini table of contents will not display on pages where no subpages are present.

The TOC button now appears when the documents are used in a small browser session (e.g. on a tablet).

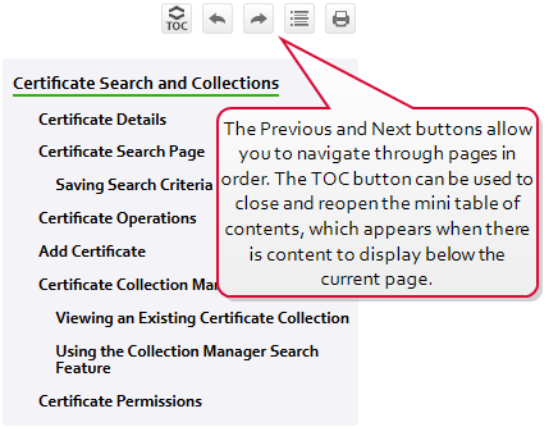

Figure 8: Navigate Forward and Backwards Through Pages

Updates and Improvements

- **Discovery**

  SSL/TLS scanning has been updated to support discovery and monitoring of certificates at endpoints that serve certificates via TLS 1.3. The scan works with the TLS_AES_128_GCM_SHA256 cipher suite. TLS 1.3 connections will also work with SNI.

- **API**

  More API endpoints have been added to do things such as manage security roles, configure certificate store jobs, and manage orchestrators. Please see the *Keyfactor API Reference Guide* for more details. You can access this and the API Endpoint Utility from the portal via the Help icon.

  Additionally, the need for an API application key and secret has been removed. We now control certificate enrollment on the template level within the portal.

- **Logging**

  The log file default locations have moved from C:\CMS\Logs to C:\Keyfactor\Logs. In addition, the NLog.config files have moved from the C:\Program Files\Common Files location to application subfolders of the installation directory, which is C:\Program Files\Keyfactor\Keyfactor Platform by default. Instead of one large CMS_Log file, there are logs for each individual applications.

  See *Editing NLog* in the *Keyfactor Command Reference Guide* for more information.

  Tip: The API is used in conjunction with the applications and both the API log and the relevant other log (e.g. portal) should be consulted when troubleshooting.

- **Administration**

  - There is now an option in the Application Settings to require users to agree to Subscriber Terms to enroll for a certificate. This setting also allows administrators to provide a link to those terms.

- CRL Stale Monitoring has been replaced with the ability for customers to define their own definition of "Stale" by generating alerts—and log entries—off the date that the CRL expires, rather than looking at the Next Publish date.

  The main reason for that is that there is, by definition, a race condition between when the new CRL gets created (exactly at the Next Publish time), and when it is copied to the CRL distribution points. Basing alerts off CRL expiration allows customers to tune timeframes based on the way they handle their CRLs.

- **Automation**

  A new constraint has been added to only allow the PowerShell event handlers to run scripts that are located in the path specified in the *Extension Handler Path* in the application settings. By default, this is "C:\Program Files\Keyfactor\Keyfactor Platform\ExtensionLibrary\". Customers should move scripts to this location or a subdirectory of it and test alerts before going into production. See *Adding PowerShell Handlers to Alerts* in *Keyfactor Command Reference Guide* for more information.

- **Certificates**
  - A new field for *Import Date* has been added to the certificate details page to log when the certificate was imported into the Keyfactor Command database.
  - Certificate Validation now shows the tests that are run when you click on a certificate and the results of those tests.
  - SSL/TLS network name is now displayed on the certificate details dialog.
  - Denied certificate requests now show the denial reason.
  - The CSR generation page has been updated to show the Extended Key Usage of the selected template.

- **Certificates**

  Denied certificate requests are now labeled as *Denied/Failed* to align with public CA terminology.

- **Enrollment**

  Email address subject alternative name option has been added to PFX enrollment.

- **Infrastructure**

  Application pool and service accounts are no longer configured with the db_owner role in SQL, but use a new custom role instead.

- **Orchestrator**

  The certificate thumbprint has been added to the failed job message to help identify which certificate was unable to be deployed to an endpoint.

- **Certificate Authorities**

  A new uniqueness constraint has been added to the CertificateAuthorities table. As a result, Keyfactor Command now checks that no CAs share the same logical name and host name combination.

- **Reporting**

- Added the ability to add a custom logo to scheduled reports.
- A new report has been added called *Expiration Report by Days* that allows for a number of days to be specified to return a table of the certificates expiring in that timeframe.
- A column for Reverse DNS has been added to the *Certificates Found at TLS/SSL Endpoints* report.

- **Templates**

  RFC 2818 enforcement has moved from the CA to the template level since different templates have different requirements. Standalone CAs still have the RFC 2818 setting on the CA level.

- **Certificates**
  - Fixed an issue where container level permissions were being ignored during enrollment preventing users from being able to add a certificate to a certificate store in that container.
  - Fixed an issue where regular expressions were being applied to empty values when they should not have been.

- **Dashboard**

  Resolved an issue where the dashboard CRL widget failed to load when configured with a high number of CRLs.

- **Email**

  An issue is fixed where the emails sent from the SSL/TLS scans sometimes reported incorrect totals.

Upgrade Prerequisites

- **Keyfactor Orchestrators**

  We encourage customers to use the new Keyfactor Universal Orchestrator moving forward, which requires .NET Core version 3.1. For existing deployments, .NET version 4.7.2 is required for systems running the legacy Windows Orchestrator.

- **SQL Server 2016**

  Support for SQL Server 2016 has been removed in Keyfactor Command 9.0. Customers should upgrade to SQL Server 2016 Cumulative Update 2 or higher before upgrading to Keyfactor Command 9.0.

- **Database Compatibility**

  Customers will also need to ensure the database compatibility is updated to support 2016 or higher. For more information on updating the compatibility level, please see System Requirements in the *Keyfactor Command Server Installation Guide*.

Upgrade Tasks

Pre-Installation

- If you are using the CA Policy module v7.0 on the same server that the Keyfactor Command Management Portal is installed on, you'll need to upgrade the module to v7.1 before running the Keyfactor Command 9.0 upgrade.

- Upgrade to SQL Server 2016 CU12 or higher and adjust the database compatibility level if needed (see above).

Post-Installation

After the upgrade is complete, some settings will need to be reconfigured due to changes in the way the Keyfactor Command Console handles tasks in Keyfactor Command 9.0:

- RFC 2818 enforcement has moved from the CA to the template level since different templates have different requirements. Standalone CAs still have the RFC 2818 setting on the CA level.
- Configure template-level metadata (if desired).
- Move all Event Handler scripts to the ExtensionLibrary folder under the Keyfactor program installation directory.
- Scripted alert handlers will fail to run if not in the path (or a subdirectory of it) specified by the *Extension Handler Path* application setting. By default, this is "C:\Program Files\Keyfactor\Keyfactor Platform\ExtensionLibrary\". Customers should move the scripts to this location and test them before moving to production.
- Update any monitoring or other processes that reference the log files to point to the new log file location.

Tip: We encourage customers to contact their customer success manager to discuss the new features and functionality in Keyfactor Command 9, and to schedule an upgrade.

Deprecation

- **API Applications**

  There is no longer the need to configure an API Application in the portal to allow for API enrollment for a certificate with a particular template. Template enrollment permissions are now controlled within the portal on the template level.

- **Classic API**

  The API calls that were previously in the Classic API (CMSAPI) have now been migrated to the Keyfactor API. Customers should use the Keyfactor API going forward and plan to migrate off the CMSAPI in the near future. Support for the CMSAPI will continue for the near future to allow customers time to migrate.

- **Expiration Renewals**

  Existing expiration renewals with Event Handlers will need to have the URLs updated to point to the Keyfactor API instead of the CMSAPI.

- **Windows Orchestrator**

  We will continue to support the Windows Orchestrator. However, all new integrations and extensions will be delivered via the new Keyfactor Universal Orchestrator. We recommend customers use the Keyfactor Universal Orchestrator moving forward as new integrations become available.

- **Verbosity in API Call**s

In a future version of Keyfactor Command, the API will return all data regardless of the verbosity level. For backwards compatibility where performance is concerned, verbosity will be honored when loading certificate location data in the certificate query but has been replaced with new flags to include this data for future requests.

- **Active Directory**

  In future releases, the ability to use the Active Directory (AD) password on PFX enrollment will be deprecated as we upgrade to allow authentication methods other than AD.

Known Issues/Limitations

Administration

- Daylight Savings Time (DST) is now shown as the time zone locale for clients using Keyfactor Command, rather than the UTC offset, which is the Microsoft CA default. This causes issues during DST to appear off by an hour, in time zones that do not have DST.
- Microsoft IIS settings to change authentication must be made manually to support the *Use Active Directory Password* application setting for the Keyfactor Command Management Portal.
- When using Basic Authentication, the authentication in Microsoft IIS may need to be configured manually for the KeyfactorAnalysis site.
- Authentication between the KeyfactorPortal, KeyfactorAPI, and KeyfactorAnalysis sites needs to be configured with the same authentication type, SSL, and host name.
- On the template RegEx settings, if you unselect use system-wide and do not enter a new RegEx the system-wide RegEx will still apply. To fix this, enter .* in the RegEx field to accept all values.
- When creating a new certificate store type, the *Depends On Other* option may not be available when creating the parameter. The workaround is to save the certificate store type and then use Edit to update the parameter.

Certificates

- Editing certificate details on a collection for a CA, while an initial sync is running on the CA, will cause inaccurate numbers to display in the Edit All window.
- If a CA is not scheduled to sync under Locations, it will not appear in lists to select for things like inclusion in Dashboards and Reports.
- Syncing an Issuing CA before syncing its parents in the chain causes Keyfactor Command to show the wrong requester for the chain certificates.

Keyfactor Command cannot support a CA in the local forest, with the same NetBIOS name as a CA in a trusted forest.

Infrastructure

- Running large SSL scans can impact Keyfactor Command application performance, if the Windows Agent/Orchestrator performing the scan is installed on the same server as the Keyfactor Command portal.

- If you receive an error when opening the portal that “the underlying connection was closed” please be sure you have the latest Windows Updates installed.

Reporting

- In Windows, drive mapping is done on a per-user basis. If you would like scheduled reports to be saved to a mapped drive, the timer service account needs to have that mapping created for them beforehand.
- Exporting a report to Microsoft Excel can fail with a 401 error in Microsoft Edge. Chrome or Firefox can successfully export to Excel. This problem is being worked on by the reporting engine vendor (Logi Analytics).
- Users configured for Logi Analytics reporting cannot have double quotes in the password field.

API

- The GET/Certificates API endpoint has a known issue where if a collection ID is not supplied the request fails. This will be fixed in an incremental release. The workaround in the meantime is to provide a collection ID of zero.

UI

- Occasionally, the “Please Wait” message will hang. Control + F5 will fix this.

Orchestrator

- There is an issue where the Universal Orchestrator is missing a task category in the Windows Event Log and instead reporting a task category of “(16)”. This will be fixed in a future release.
- The new Keyfactor Universal Orchestrator provides much of the same functionality as the legacy Windows Orchestrator (see table below).

Table 10: Keyfactor Universal Orchestrator vs Windows Orchestrator Capabilities

Capabilities	Windows Orchestrator	Universal Orchestrator
IIS Management	✔	✔
CA Synchronization	✔	✔
SSL/TLS Discovery	✔	✔
FTP	✔	✔
F5 (SOAP/REST)	✔	

Capabilities	Windows Orchestrator	Universal Orchestrator
AWS	✔	
NetScaler	✔	
Fetch Logs (new)		✔

New capabilities will be added to the Keyfactor Universal Orchestrator in a future release as we phase out use of the existing Windows Orchestrator over time.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 11: API Change Log

Endpoint	Method	Action	Notes
/Agents/Approve	POST	Add	
/Agents/Disapprove	POST	Add	
/CertificateCollections	PUT	Add	
/CertificateCollections/Copy	POST	Add	
/Certificates/{id}/History	GET	Add	
/Certificates/{id}/Security	GET	Add	
/Certificates/{id}/Validate	GET	Add	
/Certificates/Locations/{id}	GET	Add	
/Certificates/Metadata/Compare	GET	Add	
/Certificates/Metadata/All	PUT	Add	
/Certificates/RevokeAll	POST	Add	
/CertificateStoreContainers	GET	Add	
/CertificateStoreContainers/{id}	GET	Add	
/CertificateStores/Certificates/Add	POST	Add	

Endpoint	Method	Action	Notes
/CertificateStores/Certificates/Remove	POST	Add	
/Enrollment/CSR/Context/My	GET	Add	
/Enrollment/PFX/Context/My	GET	Add	
/JobTypes/Custom	GET, POST, PUT	Add	
/JobTypes/Custom/{id}	GET, DELETE	Add	
/OrchestratorJobs/Custom	POST	Add	
/OrchestratorJobs/JobHistory	GET	Add	
/OrchestratorJobs/JobStatus/Data	GET	Add	
/Reports	GET, PUT	Add	
/Reports/{id}	GET	Add	
/Reports/{id}/Parameters	GET, PUT	Add	
/Reports/{id}/Schedules	GET, POST, PUT	Add	
/Reports/Custom	GET, POST, PUT	Add	
/Reports/Custom/{id}	GET, DELETE	Add	
/Reports/Schedules/{id}	GET, DELETE	Add	
/Security/Identities	GET, POST	Add	
/Security/Identities/{id}	DELETE	Add	
/Security/Identities/Lookup	GET	Add	
/Security/Roles	GET, POST, PUT	Add	
/Security/Roles/{id}	GET, DELETE	Add	
/SSH/Keys/Unmanaged	DELETE	Add	
/SSH/ServiceAccounts	DELETE	Add	
/SSH/Users/Access	POST	Add	
/SSL/Networks/{id}/Scan	POST	Add	

2.4.1 Incremental Release 9.10 Notes

June 2022

Note: Keyfactor Command 9.10 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- **Enrollment**

  The enrollment options in Keyfactor Command now support enrolling for SubCA type certificates.
- **Expiration Alert Renewal Handler**
  - Fixed an issue where the expiration alert renewal handler would generate an error if the alert contained more than one email recipient.
  - Fixed an issue where the expiration alert renewal handler would not run on databases that had been upgraded from versions of Keyfactor Command prior to 5.
- **PAM Secret Storage**

  Fixed an issue where PAM parameters of type secret (often passwords) weren't being loaded in Keyfactor Command correctly when returned from the PAM provider.

Note: No changes were made to the Keyfactor Universal Orchestrator in this release, as such the 9.5 version of the Universal Orchestrator should still be considered the latest and is fully compatible with the Keyfactor Command 9.10 release.

API Endpoint Change Log

No API endpoint changes were made in this release.

2.4.2 Incremental Release 9.9 Notes

May 2022

Note: Keyfactor Command 9.9 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

New Features

Metadata Access on View Inventory Dialog

- **What problem does it solve?**

  The View Inventory dialog for certificate stores previously displayed each certificate found in the certificate store but did not include the Keyfactor Command metadata field values configured for the certificates.

- **How does it work?**

  The View Inventory dialog on the Certificate Stores page now includes a Metadata section to allow you to view the metadata fields configured in Keyfactor Command for each certificate found in the certificate store.

- **What's the benefit?**

  Streamlining: You no longer need to look up the metadata fields for the certificates separately.

Updates and Improvements

- **GET /Agents Keyfactor API Endpoint**

  The GET /Agents Keyfactor API endpoint now includes a query parser to allow searching by AgentId. For example:.

  ```
  AgentId -eq "d2f0d545-c3b3-4ea3-bc0a-0232865e24c3"
  ```

- **Logging**

  Changes have been made to the way that Keyfactor Command logs are initialized to support logging from multiple source libraries including Quartz.

- **Alerts Do Not Resume After a Database Connection Failure**

  Fixed an issue in which expiration alerts and pending, issued, and denied certificate alerts that failed due to a database connection problem would not restart on resolution of the database connection issues until the Keyfactor Command service was restarted.

- **Revoke All of Entirely Revoked or Expired Certificates Fails**

  Fixed an issue in which attempting to revoke all for a group of certificates that contains only certificates that are revoked already and/or expired results in an error message.

- **SSH Server Groups Incompatible with Domain Names Containing Hyphens**

  Fixed an issue in which SSH server groups could not be created in environments where the Keyfactor Command domain contains a hyphen because the SSH server group owner field would not support a hyphen in the domain name.

- **Certificate Signing Requests Can Produce an Error on Decoding**

Fixed an issue in which CSR decoder used in CSR enrollment can produce an error on decoding the CSR under select circumstances. These can include SCEP requests with no SANs and CSRs with no extensions.

- **Keyfactor API GET Requests with a Sort Produce a 500 Error**

  Fixed an issue in which Keyfactor API GET endpoints that support query sorting in the URL would produce a 500 error if the sort field was not provided correctly (e.g. the fieldname was entered with a space or was a valid fieldname but not one that was supported for sorting).

Note: No changes were made to the Keyfactor Universal Orchestrator in this release, as such the 9.5 version of the Universal Orchestrator should still be considered the latest and is fully compatible with the Keyfactor Command 9.9 release.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 12: API Change Log

Endpoint	Methods	Action	Notes
/Reports/<any>	GET	Fix	Spaces within the sortField no longer results in an exception.
/Reports/{id}/Schedules	GET	Fix	An invalid sortField no longer results in an exception.
/Agents	GET	Update	New query parser to support the AgentId GUID.

2.4.3 Incremental Release 9.8 Notes

April 2022

Note: Keyfactor Command 9.8 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- **PFX Generation**

  Consolidated PFX generation code so that the PFX files are generated identically from the enrollment and download components.

- **SCEP Intune Integration**

  Keyfactor's Simple Certificate Enrollment Protocol (SCEP) component has been updated to utilize the latest Intune API: Microsoft Authentication Library (MSAL) and Azure AD Graph API.

- **Pending Certificate Request SAN**

  Fixed an issue in which pending certificate requests containing a User Principal Name (UPN) in the Subject Alternative Name (SAN) would be prefixed with '[0]', and IPv6 addresses were not displayed.

- **vSCEP Challenge Error**

  Fixed an issue in which attempting to obtain a Validated SCEP (vSCEP) challenge resulted in an assembly loading error.

- **Denied Alert Email SAN**

  Fixed an issue in which Denied Certificate Alert email did not contain the certificate Subject Alternative Names (SANs).

- **Expiration Alert Logging**

  Fixed an issue in which excessive and superfluous log messages were generated during Expiration Alert processing.

Note: No changes were made to the Keyfactor Universal Orchestrator in this release, as such the 9.5 version of the Universal Orchestrator should still be considered the latest and is fully compatible with the Keyfactor Command 9.8 release.

API Endpoint Change Log

No API endpoint changes were made in this release.

2.4.4 Incremental Release 9.7 Notes

March 2022

Note: Keyfactor Command 9.7 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- **JavaScript Caching**

  Updated pages not to cache static files, including JavaScript.

Note: After upgrading to 9.7, the cache will still need to be cleared one final time so that the latest version of the pages get loaded with the updated cache setting.

- **API CA Auto-selection**

  The Keyfactor API will auto-select an enrollment certificate authority if one is not explicitly provided.
- **Certificate Stores**

  Fixed an issue in which a user could assign a certificate store to a container without explicit permissions to that certificate store.
- **Certificate Stores**

  Fixed an issue in which database upgrades fail on Azure SQL for newly created databases.
- **Certificate Stores—Scheduling**

  Fixed an issue in which jobs could appear to be scheduled for a certificate store with no available agent.
- **Security Configuration**

  Fixed an issue in which the security roles management page could not be loaded after deletion of an associated Active Directory (AD) group.
- **Metadata String & Integer Fields**

  Corrected an issue where default values could not be set for metadata fields of type string or integer.
- **Certificate Store Deployment**

  Fixed an issue where a certificate cannot be deployed to a certificate store when deploying using a property instead of a certificate store type or Id.

Note: No changes were made to the Keyfactor Universal Orchestrator in this release, as such the 9.5 version of the Universal Orchestrator should still be considered the latest and is fully compatible with the Keyfactor Command 9.7 release.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 13: API Change Log

Endpoint	Methods	Action	Notes
/KeyfactorAPI/License	GET	Add	

2.4.5 Incremental Release 9.6 Notes

February 2022

Note: Keyfactor Command 9.6 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- **Configuration Wizard**

  Fixed an issue in which the SQL Server login for the application pool account was not created by the configuration wizard.
- **Azure Database Creation**

  Fixed an issue in which database upgrades fail on Azure SQL for newly created databases.
- **Certificate Store—Scheduling**

  Fixed an issue in which jobs rescheduled for *immediate* would not execute.
- **Command Line Configuration Wizard**

  Fixed an issue in which the console configuration wizard cannot populate Azure SQL databases.
- **Custom Orchestrator Job Blueprint**

  Corrected an issue where a duplicate custom job schedule was created when applying the same blueprint to orchestrator.
- **Expiration Report by Days**

  Corrected an issue where the Expiration Report by Days would crash on DD/MM/YYYY formatted dates.
- **Certificate Renewal in Single Store**

  Fixed an issue where a single certificate stored at multiple aliases within the same certificate store was not renewed successfully.
- **CRL Alert Emails**

  Corrected an issue in which a CRL alert email would be sent even if a new CRL was available.

Note: No changes were made to the Keyfactor Universal Orchestrator in this release, as such the 9.5 version of the Universal Orchestrator should still be considered the latest and is fully compatible with the Keyfactor Command 9.6 release.

API Endpoint Change Log

No API endpoint changes were made in this release.

2.4.6 Incremental Release 9.5 Notes

January 2022

Note: Keyfactor Command 9.5 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- **Agents and Orchestrators**

  Several enhancements have been made to the orchestrators:

  - The alias column size has increased to allow for longer alias names.
  - A new setting allows the IIS stores to be accessed using WinRM over SSL (port 5986).
  - The last thumbprint used for client certificate authentication by orchestrators is now tracked and can be returned using the GET /Agents API method.
  - The UI now allows you to see why an orchestrator could not register for a session rather than having to look in the logs.
  - A new API endpoint has been added to request or require that one or more orchestrators enroll for a new client authentication certificate on the orchestrator's next session registration (POST /Agents/SetAuthCertificateReenrollment).
  - A new API endpoint has been added to reset an orchestrator (POST /Agents/{id}/Reset). Updates include removing orchestrator jobs, deleting associated certificate stores, setting the orchestrator status to new, and clearing thumbprint data as below.
  - The orchestrator reset function in the UI and API now clears the orchestrator client authentication certificate thumbprint data to allow the orchestrator to be reconfigured with a new certificate.

- **Management Portal—Reports**

  The "Expiring in less than two weeks" text in the *PKI Status for Collection* report has been updated to change the color scheme to be more readable (white text on a maroon background).

- **API**

  Fixed an issue with the Enrollment/PFX API call not working without specifying a CA. The JobTypes/Custom API call now returns the Job Retry Count.

- **Certificates—Metadata**

  Fixed an issue so that hidden metadata now shows when using *Edit All*.

- **Certificate Stores—Scheduling**

  Fixed an issue to now prompt the user to enter schedule values for *Exactly once* and for *Daily* schedules.

- **Certificate Store—Inventory**

  Fixed an issue when viewing the inventory of certificate store that has an alias without a certificate.
- **Installation—Modify/Remove**

  Corrected an issue where the MSI would freeze if trying to modify or uninstall an installation that had been done without any components selected to be installed.
- **Orchestrators and Agents—Custom Job Retry**

  Corrected an issue where custom jobs would not retry if the job complete handlers failed.
- **Alerting—Email Address Format**

  Fixed an issue where the email address validation was not allowing some valid subdomains.
- **Registration Handler—Enrollment**

  The registration handler now receives the certificate chain for enrollments performed via the enrollment callback.
- **Management Portal Reports**

  Updates to Management Portal reports to handle upgrade scenarios and other user interface fixes.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 14: API Change Log

Endpoint	Methods	Action	Notes
/Enrollment/PFX	POST	Update	No longer requires a certificate authority name to be provided.

2.4.7 Incremental Release 9.4 Notes

December 2021

Note: Keyfactor Command 9.4 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- Log4j CVE Vulnerability

  Keyfactor has conducted an assessment of the recently-announced CVE for the log4j library (https://github.com/advisories/GHSA-jfh8-c2jp-5v3q). We have identified that the vast majority of the Keyfactor suite of products are NOT affected. This includes EJBCA, SignServer, the Keyfactor Command platform, Keyfactor Control, and Code Assure.

  The only component that does make use of the log4j library is the Java Agent for Keyfactor Command; for clarity, all other Keyfactor agents and gateways are NOT affected.

  Details

  According to the CVE, exploit of the vulnerability requires compelling log4j to log user-controlled input. In the case of the Java agent, there are mitigating factors, such as:

  - The Java agent has an "outbound-only" connection pattern and does not accept inbound network connections of any kind.
  - Users of the Java agent who could control such input are typically Keyfactor administrators.
  - The limited nature of things the Java agent is expected to log.

  From Log4j – Apache Log4j Security Vulnerabilities:

  - Mitigation: This behavior can be mitigated by setting either the system property log4j2.formatMsgNoLookups or the environment variable LOG4J_FORMAT_MSG_NO_LOOKUPS to true.

  Patch Implementation—The 8.7.2 version of the Java Agent to utilize the patched version of Log4j, and mitigate the vulnerability.

- **Orchestrator Certs**

  Ability for an orchestrator to use a TLS client authentication certificate to map to a Windows identity in IIS and to use a different TLS certificate provided in an HTTP header to identify the orchestrator to Keyfactor Command.

- **External Validation Certificate Requests**

  Certificate requests returning a status of EXTERNAL_VALIDATION are not treated as failures and will be sync'd with appropriate metadata when the certificate is available.

- **Certificate Detail Data Efficiency**

  The certificate details are obtained from the server when needed, and not as part of the initial certificate query. This greatly increases the efficiency and performance of the page.

- **Query Optimization for Large Scale Environments**

  Multiple optimizations have been made to improve management portal query performance, scalability, and stability in large scale environments.

- **Pending Certificates API Endpoint**

Metadata for certificate requests in a pending state is now available for retrieval via the /Workflow/Certificates/Pending API endpoints (GET /Workflow/Certificates/{id} and GET /Workflow/Certificates/Pending).

- **SSL Scanning Chunk Sizes**

  Distinct SSL scanning chunk size application settings are now available for discovery and monitoring to allow for greater control over performance tuning.

- **Dashboard Risk Header Clarifications**

  The dashboard Risk Header now contains verbiage to clarify that no filtering exists for renewed certificates in expired query counts.

  In addition, the dashboard Risk Header contains verbiage noting that the certificate counts are global and not limited to only those to which the current user has access.

- **Custom Job Blueprint Duplication**

  An issue was fixed so that a copy operation on a blueprint successfully copies custom jobs.

- **Certificate Count by Template Report**

  An issue was fixed so to properly retain the selected default certificate authority.

- **SSL Quiet Hours Daylight Savings**

  Updates were made to the SSL Quiet Hours to better handle schedules involving Daylight Savings Times.

- **SSL Monitoring Emails**

  SSL Monitoring emails now send the complete and correct data when multiple orchestrators are in simultaneous use.

- **Certificate Detail Before/Not After Dates**

  Certificate details now display the time in addition to the date for Before and Not After dates.

- **SSL Scanning Certificate History**

  A fix was implemented to properly display the history of certificates imported into the system via SSL scanning.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 15: API Change Log

Endpoint	Methods	Action	Notes
/Workflow/Certificates/Pending	GET	Update	Now returns the associated metadata.

2.4.8 Incremental Release 9.3 Notes

November 2021

Note: Keyfactor Command 9.3 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

Updates and Improvements

- **Certificate Search**

  The certificate search functionality has been optimized to increase speed and efficiency, especially with higher numbers of certificates and associated certificate locations. This means certificate searches done in the management portal for large data sets that include certificates found in certificate stores (e.g. 250,000+ certificates each in 5 or more certificate stores) now complete more quickly.
- **Failed Certificate Management Jobs**

  Certificate management jobs that have failed no longer continue to run.
- **PKI Status Report Time Zone**

  Corrected the format of time zones in the PKI Status for Collection Report.
- **Database Encryption Configuration**

  The Configuration Wizard now verifies the selected database encryption certificate has an associated valid private key.
- **SSL Scanning**

  Updates made to the SSL scanning process to be more efficient and eliminate potential process-locking scenarios.
- **Management Portal User Interface**

  Various Management Portal user interface fixes.
- **Management Portal Reports**

  Updates to Management Portal reports to handle upgrade scenarios and other user interface fixes.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 16: API Change Log

Endpoint	Methods	Action	Notes
/JobTypes/Custom	POST	Fix	No longer requires default field values.

2.4.9 Incremental Release 9.2 Notes

October 2021

Note: Keyfactor Command 9.2 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

New Features

UI Support for PAM CA Password Entry

- **What problem does it solve?**

  The API previously supported the entry of certificate authority passwords to be stored within a Privileged Access Management (PAM) instance, but the UI did not implement this functionality.

- **How does it work?**

  The certificate authority editor dialog allows for entry of a password to be stored in a PAM instance.

- **What's the benefit?**

  Flexibility: Allows for multiple ways to securely store and manage certificate authority passwords

Custom Orchestrator Bulk Scheduling

- **What problem does it solve?**

  Custom orchestrator jobs can currently only be scheduled individually.

- **How does it work?**

  An API endpoint (POST OrchestratorJobs/Custom/Bulk) has been created to implement bulk schedules. The job identifiers along with the desired schedule can be provided in a single call.

- **What's the benefit?**

  Ease-of-Use: Enables administrators to easily schedule large batches of custom orchestrator jobs.

Updates and Improvements

- **CA Management with PAM**

  When configuring the *Use Explicit Credentials* option on a CA, you can now choose a PAM provider as the storage location for the credential password or the Keyfactor secrets table.

- **Logi Analytics License**

  A new license for Logi Analytics is required as the previous version is expiring. The 9.2 release includes the license update. Please see Updating Logi Analytics License on the next page for more information.

- **CSR Parsing Containing Spaces**

  CSRs containing spaces can now be parsed successfully during enrollment.

- **Robust SSL Certificate Parsing Error Handling**

  Certificates that fail to be parsed during SSL scanning are now logged but do not cause the entire scan to immediately fail.

- **Robust Alert Failure Error Handling**

  A failure processing an alert no longer prevents processing of subsequent alerts.

- **Hidden Metadata Enrollment Fields**

  Metadata fields which are hidden during the enrollment process are now displayed properly in the resulting certificate details.

- **Collection-based Reports Failing**

  Reports based on collections containing Revocation, Certificate State or Common Name no longer fail.

- **Incorrect CSR Enrollment CA**

  The proper forest certificate authority is used for enrollment when using the API to enroll via CSR.

- **Denied Alerts Template**

  The Denied Certificate Request alerts are once again properly scoped to the selected template. This was a regression from a previous release.

- **Java & C Agent Inventory Error**

  An error was corrected in which an error was thrown if no entry updates were returned during inventory processing.

- **Orchestrator/Agent Re-Enrollment Error**

  Fixed an issue in which an object reference error was thrown during re-enrollment operations.

- **Orchestrator Ceases Processing after Batch Submission**

  Corrected an issue in which the orchestrators would cease processing after submission of a large batch of SSL results.

Updating Logi Analytics License

Logi is a 3rd party BI tool which is used by Keyfactor Command for its dashboard and report features. The license required for Logi is integrated into Keyfactor Command and resides within the product's Logi folder. The license's current term is 3 years with a 7-day grace period after expiration. During that grace period, an alert will appear, and a new license should be used to remediate the issue. Here are two examples:

- License close to expiration:

Figure 9: Keyfactor Logi License Expiration Alert

Dashboard:

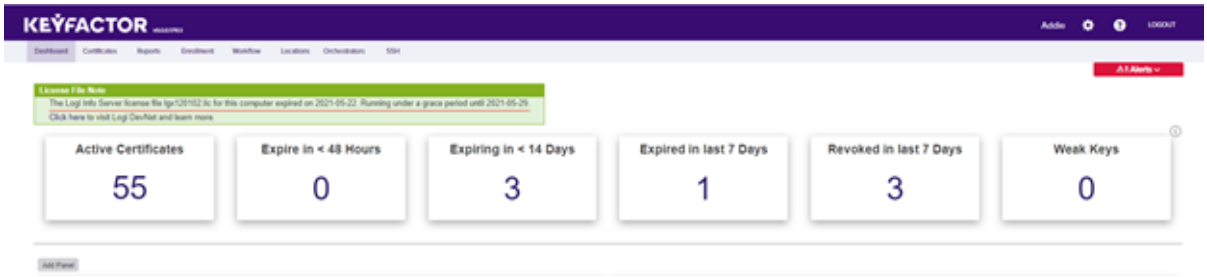

Figure 10: Keyfactor Logi License Expiration Alert on the Dashboard

Report:

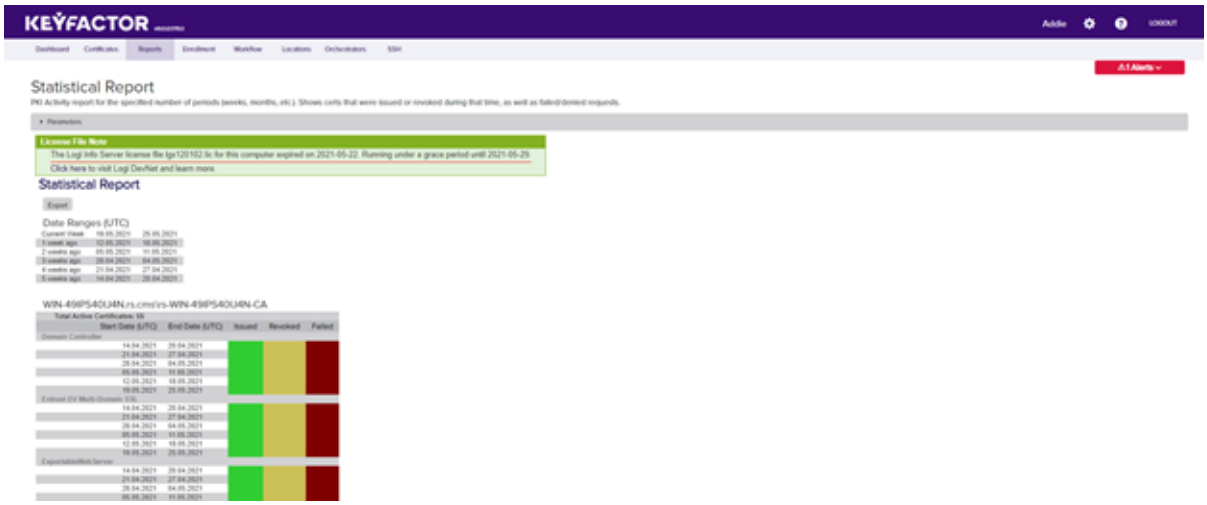

Figure 11: Keyfactor Logi License Expiration Alert on Report

- Expired license:

  The Dashboard and Reporting capability is not available with an error message displayed like the one below.

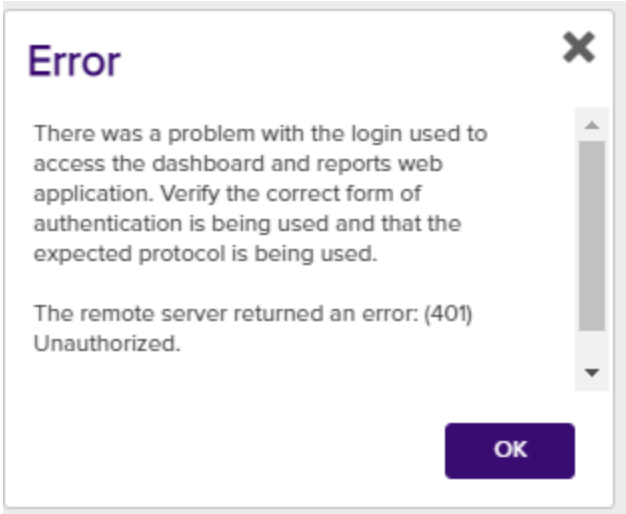

Figure 12: Keyfactor Expired Logi Error Message

Solution

The updated license for Logi is included in release 9.2 and will be installed automatically as part of the upgrade to or fresh installation of this version. If you are not installing Keyfactor Command v9.2, replace the license manually as follows:

1. On your Keyfactor Command server, navigate to the Logi folder in your Keyfactor Command instance. By default, this is:

   C:\Program Files\Keyfactor\KeyfactorPlatform\Logi

   If you are on an earlier version of Keyfactor Command your license file will by default be found in the following directory:

   C:\Program Files\Certified Security Solutions\Certificate Management System\Logi]

2. The license file ends with an extension of *.lic*. Replace the license file with a valid one provided to you by Keyfactor. The license filename cannot be changed and should remain as "lgx120102.lic".

If the license has already expired, once it is replaced with a valid one and the browser is refreshed, the product will work as expected. The alert will no longer appear.

If you upgrade to a version of Keyfactor Command prior to v9.2 after replacing the license file, you will need to manually add the new license file again.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 17: API Change Log

Endpoint	Methods	Action	Notes
/Certificates	GET	Fix	No longer fails if a collection id is not provided.
/OrchestratorJobs/JobHistory	GET	Fix	Request no longer fails for 'Dynamic' job types.
/Reports/Schedules/{id}	DELETE	Fix	Response code is now 200 when the user role does not have *Modify – Report* permission.

2.4.10 Incremental Release 9.1 Notes

September 2021

Note: Keyfactor Command 9.1 is a minor release with incremental fixes and updates following the Keyfactor Command 9 major release. For more details on new features, improvements, and known limitations in Keyfactor Command 9.0, please review the Major Release 9.0 Notes on page 77.

New Features

Custom Certificate Store Parameters

- **What problem does it solve?**

  Provide the ability to associate custom parameters with certificate stores and certificate store entries to associate useful information.

- **How does it work?**

  The certificate store type dialog now provides tabs for entry parameters, in addition to custom fields. These parameters and custom fields can be defined for input during enrollment, storage and management of certificate store inventory. For more information, see *Certificate Store Type Operations: Adding or Editing a Certificate Store Type Entry Parameters Tab* in the *Keyfactor Command Reference Guide*.

- **What's the benefit?**

  Flexibility: Allows for further customization around certificate stores which can be dictated by customizable data.

Certificate Store Inventory

- **What problem does it solve?**

  The previous version of certificate store inventory leveraged the certificate search functionality. While this worked, it was not always well-suited for the viewing of certificate store inventory.

- **How does it work?**

  Clicking on the *View Inventory* button with a certificate store selected will now load a dialog with the inventory of the store.

- **What's the benefit?**

  Ease-of-Use: Enables administrators to efficiently review certificate store inventory.

Certificate Store Type Parameters

- **What problem does it solve?**

  The previous certificate store type parameters were defined via a comma-separated list and were not strongly typed.

- **How does it work?**

  A formalized list is available to define parameters explicitly, including type (String, Boolean, Multiple Choice, Secret).

- **What's the benefit?**

  Flexibility: Enables more powerful definition of certificate stores and data-validity checking.

Certificate Store Parameter Reporting

- **What problem does it solve?**

  The current on-boarding of certificate stores requires manual data entry of custom fields and parameters.

- **How does it work?**

  The Keyfactor Command orchestrator framework provides for orchestrators to report certificate store entry parameters.

- **What's the benefit?**

  Flexibility: Enables customers to more easily track new certificate stores and changes to them made out-of-band from Keyfactor Command.

Keyfactor Command Configuration Wizard

The Keyfactor Command server configuration wizard now supports entry of group managed service accounts (gMSA) in the Administrative Users field on the Keyfactor Portal tab.

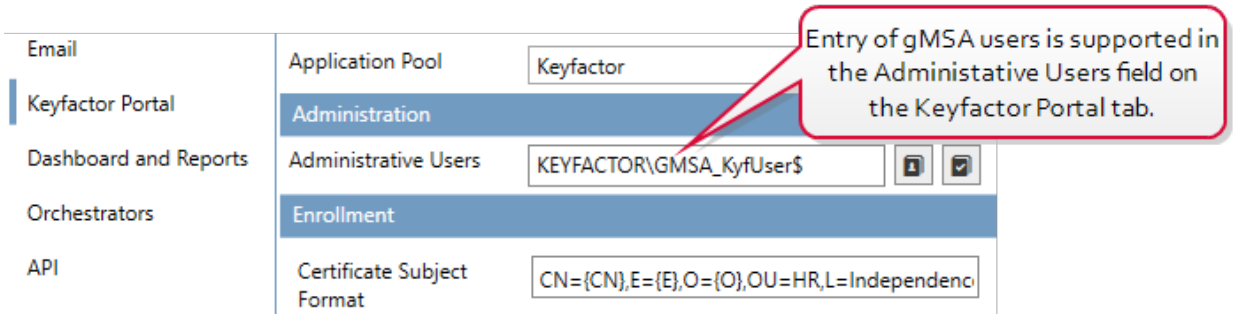

Figure 13: Entry of gMSA Users in the Administrative Users Field

Note: Entry of gMSA users is not supported in the fields that require entry of a password in the configuration wizard (e.g. the service account on the Service tab) at this time. GMSA users cannot be selected using the people picker.

Updates and Improvements

- **Job Completion**

  Job completion handler is now provided the certificate identifier upon renewal so that the handler can perform any related tasks.
- **API Endpoint Deprecation**

  The CertificateCollections/{id}/Permissions endpoint due to an update slated for the Keyfactor Command v10 release and the fact that the endpoint is not updating permissions properly.
- **Permissions Message**

  An incorrect error message was displayed to users without sufficient permissions to a certificate collection.
- **Certificate Store Deletion**

  Fixed an issue in which a Certificate Store cannot be deleted if there is a job staged against it.
- **Pending Alerts**

  Pending alerts were being sent on certificate issuance regardless of the associated template.
- **Certificate Inventory**

  Corrected a permissions problem in which users with only read permissions on a Certificate Store were unable to view inventory of that certificate store.

Known Issues

- **CSR Enrollment**
  In cases where there are duplicate template names in multiple forests, CSR enrollment can sometimes go to the wrong CA. This will be fixed in a future incremental release. Customers with environments with duplicate templates should wait to upgrade.

API Endpoint Change Log

The following changes were made to the API endpoints. Please review these carefully if you have implemented any integration using these endpoints.

Table 18: API Change Log

Endpoint	Methods	Action	Notes
/CertificateStores/{id}/Inventory	GET	Add	
/Enrollment/PFX/Replace	POST	Fix	SuccessfulStores collection now only includes Ids of stores that were successfully processed.
/Enrollment/PFX/Deploy	POST	Update	Now allows for multiple stores of the same type with different parameters.
/CertStoreTypes	POST/PUT	Update	EntryParameters can now be set via these methods.
/CertificateStores/Certificates/Add	POST	Update	Now allows for multiple stores of the same type with different parameters.
/CertificateStores/Certificates/Remove	POST	Update	Now allows for multiple stores of the same type with different parameters.
/CertificateCollections/{id}/Permissions	GET	Deprecate	

2.5 Keyfactor Command v11 Compatibility Matrix

Keyfactor Command version 11's compatibility with the various supported Keyfactor gateways, agents and orchestrators is shown in the table below (see also Keyfactor Command Compatibility Matrix Legend 1).

Table 19: Compatibility Matrix for Keyfactor Command v11

Product	Keyfactor Command Compatibility
Universal Orchestrator	**Version** / **11.0** 11.0.0 ✓ 10.4.1 ■ 10.4.0 ■ 10.2.0 ■ 10.0.1 ■ 9.4.0 ■ 9.2.0 ■ 9.0.2 ■
Windows Orchestrator	**Version** / **11.0** 8.7.2 ✗
EJBCA Native Support	**Version** / **11.0** 8.0.0.0 ✓ 7.12.0.1 ✗ 7.12.0.0 ✗ 7.11.0.0 ✗ 7.10.1.0 ✗ 7.10.0.1 ✗ 7.9.1.0 ✓

Product	Keyfactor Command Compatibility
SSH	Version / 11.0 2.0.0 ✓ 1.0.1 ■
Java Agent	Version / 11.0 8.7.2 ✓
AnyCAGateway REST	Version / 11.0 23.1 ✓
AnyGateway	20.9 / 11.0 23.3.0 ✳ ✓ 22.1.1 ✳ ✓ 22.1.0 ✳ ✓ 21.10.x ✳ ✗ 21.5.x ✳ ✗ 21.3.x ✳ ✗ 20.9 ✳ ✗ 20.7 ✳ ✗
Windows Enrollment Gateway	Version / 11.0 23.3 ✓ 23.1.0 ✓

Product	Keyfactor Command Compatibility
SQL Server	<table><tr><th>Version</th><th>11.0</th></tr><tr><td>2016</td><td>✗</td></tr><tr><td>2017</td><td>✓</td></tr><tr><td>2019</td><td>✓</td></tr><tr><td>2022</td><td>✓</td></tr></table>
Windows Server	<table><tr><th>Version</th><th>11.0</th></tr><tr><td>2016</td><td>✗</td></tr><tr><td>2019</td><td>✓</td></tr><tr><td>2022</td><td>✓</td></tr></table>

2.6 Keyfactor Command v10 Compatibility Matrix

Keyfactor Command version 10's compatibility with the various supported Keyfactor gateways, agents and orchestrators is shown in the table below (with Keyfactor Command Compatibility Matrix Legend 1).

Table 20: Compatibility Matrix for Keyfactor Command v10

Product	Keyfactor Command Compatibility
Universal Orchestrator	<table><tr><th>Version</th><th>10.4</th><th>10.3</th><th>10.2</th><th>10.1</th><th>10.0</th></tr><tr><td>10.4.1</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td></tr><tr><td>10.4.0</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td></tr><tr><td>10.2.0</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td></tr><tr><td>10.0.1</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td><td>✓</td></tr></table>

Product	Keyfactor Command Compatibility

Version	10.4	10.3	10.2	10.1	10.0
9.4.0	✓	✓	✓	✓	✓
9.2.0	✓	✓	✓	✓	✓
9.0.2	✓	✓	✓	✓	✓

Windows Orchestrator

Version	10.4	10.3	10.2	10.1	10.0
8.7.2	✓	✓	✓	✓	✓

EJBCA Native Support

Version	10.4	10.3	10.2	10.1	10.0
8.0.0.0	✗	✗	✗	✗	✗
7.12.0.1	✓	✓	✓	✓	✓
7.12.0.0	✓	✓	✓	✓	✓
7.11.0.0	✓	✓	✓	✓	✓
7.10.1.0	✓	✓	✓	✓	✓
7.10.0.1	✗	✗	✗	✗	✗
7.9.1.0	✓	✓	✓	✓	✓

SSH

Version	10.4	10.3	10.2	10.1	10.0
2.0.0	✓	✓	■	■	■
1.0.1	■	■	✓	✓	✓

Java Agent

Version	10.4	10.3	10.2	10.1	10.0
8.7.2	✓	✓	✓	✓	✓

Product	Keyfactor Command Compatibility

AnyCAGateway REST

Version	10.4	10.3	10.2	10.1	10.0
23.1	✗	✗	✗	✗	✗

AnyGateway

20.9	10.4	10.3	10.2	10.1	10.0
23.3.0 ✳	✓	✓	✓	✓	✓
22.1.1 ✳	✓	✓	✓	✓	✓
22.1.0 ✳	✓	✓	✓	✓	✓
21.10.x ✳	✗	✗	✗	✗	✗
21.5.x ✳	✗	✗	✗	✗	✗
21.3.x ✳	✗	✗	✗	✗	✗
20.9 ✳	✗	✗	✗	✗	✗
20.7 ✳	✗	✗	✗	✗	✗

Windows Enrollment Gateway

Version	10.4	10.3	10.2	10.1	10.0
23.3	✓	✓	✓	✗	✗
23.1.0	✓	✓	✓	✗	✗

SQL Server

Version	10.4	10.3	10.2	10.1	10.0
2016	✗	✗	✗	✗	✗
2017	✓	✓	✓	✓	✓
2019	✓	✓	✓	✓	✓
2022	✓	✓	✓	✓	✓

Product	Keyfactor Command Compatibility
Windows Server	

Version	10.4	10.3	10.2	10.1	10.0
2016	✗	✗	■	■	■
2019	✓	✓	✓	✓	✓
2022	✓	✓	✓	✓	✓

2.7 Keyfactor Command v9 Compatibility Matrix

Keyfactor Command version 9's compatibility with the various supported Keyfactor gateways, agents and orchestrators is shown in the table below (with Keyfactor Command Compatibility Matrix Legend 1).

Table 21: Compatibility Matrix for Keyfactor Command v9

Product	Keyfactor Command Compatibility									
Universal Orchestrator	**Version**	**9.10**	**9.9**	**9.8**	**9.7**	**9.6**	**9.5**	**9.4**	**9.3**	**9.2**	**9.1-9.0**
	10.4.1	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	10.4.0	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	10.2.0	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	10.0.1	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
	9.4.0	✓	✓	✓	✓	✓	✓	✓	🟨	🟨	🟨
	9.3.0	✓	✓	✓	✓	✓	✓	✓	✓	🟨	🟨
	9.2.0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	9.0.2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Product	Keyfactor Command Compatibility

Windows Orchestrator

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
8.7.2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

EJBCA Native Support

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
8.0.0.0	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
7.12.0.1	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
7.12.0.0	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
7.11.0.0	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
7.10.1.0	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
7.10.0.1	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
7.9.1.0	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Product	Keyfactor Command Compatibility

SSH

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
2.0.0	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
1.0.1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Java Agent

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
8.7.2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

AnyCAGateway REST

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
23.1	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

AnyGateway

20.9	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
23.3.0 ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
22.1.1 ✳	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨

Product	Keyfactor Command Compatibility

20.9	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
22.1.0 ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
21.10.x ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
21.5.x ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
21.3.x ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
20.9 ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
20.7 ✳	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Windows Enrollment Gateway

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
23.3	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
23.1.0	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Product	Keyfactor Command Compatibility
SQL Server	

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
2016	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2017	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2019	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2022	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Windows Server

Version	9.10	9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2	9.1-9.0
2016	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2019	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
2022	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

3.0 Glossary

A

AIA

The authority information access (AIA) is included in a certificate--if configured--and identifies a location from which the chain certificates for that certificate may be retrieved.

AnyAgent

The AnyAgent, one of Keyfactor's suite of orchestrators, is used to allow management of certificates regardless of source or location by allowing customers to implement custom agent functionality via an API.

AnyGateway

The Keyfactor AnyGateway is a generic third party CA gateway framework that allows existing CA gateways and custom CA connections to share the same overall product framework.

API

A set of functions to allow creation of applications. Keyfactor offers the Keyfactor API, which allows third-party software to integrate with the advanced certificate enrollment and management features of Keyfactor Command.

Argument

A parameter or argument is a value that is passed into a function in an application.

Authority Information Access

The authority information access (AIA) is included in a certificate--if configured--and identifies a location from which the chain certificates for that certificate may be retrieved.

B

Bash Orchestrator

The Bash Orchestrator, one of Keyfactor's suite of orchestrators, is used to discover and manage SSH keys across an enterprise.

Blueprint

A snapshot of the certificate stores and scheduled jobs on one orchestrator, which can be used to create matching certificate stores and jobs on another orchestrator with just a few clicks.

C

CA

A certificate authority (CA) is an entity that issues digital certificates. Within Keyfactor Command, a CA may be a Microsoft CA or a Keyfactor gateway to a cloud-based or remote CA.

Certificate Authority

A certificate authority (CA) is an entity that issues digital certificates. Within Keyfactor Command, a CA may be a Microsoft CA or a Keyfactor gateway to a cloud-based or remote CA.

Certificate Revocation List

A Certificate Revocation List (CRL) is a list of digital certificates that have been revoked by the issuing Certificate Authority (CA) before their scheduled expiration date and should no longer be trusted.

Certificate Signing Request

A CSR or certificate signing request is a block of encoded text that is submitted to a CA when enrolling for a certificate. When you generate a CSR within Keyfactor

Command, the matching private key for it is stored in Keyfactor Command in encrypted format and will be married with the certificate once returned from the CA.

CN

A common name (CN) is the component of a distinguished name (DN) that represents the primary name of the object. The value varies depending on the type of object. For a user object, this would be the user's name (e.g. CN=John Smith). For SSL certificates, the CN is typically the fully qualified domain name (FQDN) of the host where the SSL certificate will reside (e.g. servername.keyexample.com or www.keyexample.com).

Collection

The certificate search function allows you to query the Keyfactor Command database for certificates from any available source based on any criteria of the certificates and save the results as a collection that will be availble in other places in the Management Portal (e.g. expiration alerts and certain reports).

Common Name

A common name (CN) is the component of a distinguished name (DN) that represents the primary name of the object. The value varies depending on the type of object. For a user object, this would be the user's name (e.g. CN=John Smith). For SSL certificates, the CN is typically the fully qualified domain name (FQDN) of the host where the SSL certificate will reside (e.g. servername.keyexample.com or www.keyexample.com).

Configuration Tenant

A grouping of CAs. The Microsoft concept of forests is not used in EJBCA so to accommodate the new EJBCA functionality, and to avoid confusion, the term forest needed to be renamed. The new name is configuration tenant. For EJBCA, there would be one configuration tenant per EJBCA server install. For Microsoft, there would be one per forest. Note that configuration tenants cannot be mixed, so Microsoft and EJBCA cannot exist on the same configuration tenant.

CRL

A Certificate Revocation List (CRL) is a list of digital certificates that have been revoked by the issuing Certificate Authority (CA) before their scheduled expiration date and should no longer be trusted.

CSR

A CSR or certificate signing request is a block of encoded text that is submitted to a CA when enrolling for a certificate. When you generate a CSR within Keyfactor Command, the matching private key for it is stored in Keyfactor Command in encrypted format and will be married with the certificate once returned from the CA.

D

DER

A DER format certificate file is a DER-encoded binary certificate. It contains a single certificate and does not support storage of private keys. It sometimes has an extension of .der but is often seen with .cer or .crt.

Distinquished Name

A distinguished name (DN) is the name that uniquely identifies an object in a directory. In the context of Keyfactor Command, this directory is generally Active Directory. A DN is made up of attribute=value pairs,

separated by commas. Any of the attributes defined in the directory schema can be used to make up a DN.

DN

A distinguished name (DN) is the name that uniquely identifies an object in a directory. In the context of Keyfactor Command, this directory is generally Active Directory. A DN is made up of attribute=value pairs, separated by commas. Any of the attributes defined in the directory schema can be used to make up a DN.

DNS

The Domain Name System is a service that translates names into IP addresses.

E

ECC

Elliptical curve cryptography (ECC) is a public key encryption technique based on elliptic curve theory that can be used to create faster, smaller, and more efficient cryptographic keys. ECC generates keys through the properties of the elliptic curve equation instead of the traditional method of generation as the product of very large prime numbers.

Endpoint

An endpoint is a URL that enables the API to gain access to resources on a server.

Enrollment

Certificate enrollment refers to the process by which a user requests a digital certificate. The user must submit the request to a certificate authority (CA).

EOBO

A user with an enrollment agent certificate can enroll for a certificate on behalf of another user. This is often used when provisioning technology such as smart cards.

F

Forest

An Active Directory forest (AD forest) is the top most logical container in an Active Directory configuration that contains domains, and objects such as users and computers.

G

Gateway Connector

The Keyfactor Gateway Connector is installed in the customer forest to provide a connection between the on-premise CA and the Azure-hosted, Keyfactor managed Hosted Configuration Portal to provide support for synchronization, enrollment and management of certificates through the Azure-hosted instance of Keyfactor Command for the on-premise CA. It is supported on both Windows and Linux.

H

Host Name

The unique identifier that serves as name of a computer. It is sometimes presented as a fully qualified domain name (e.g. servername.keyexample.com) and sometimes just as a short name (e.g. servername).

Hosted Config Portal

The Keyfactor Hosted Configuration Portal is used to configure connections between on-premise instances of the Keyfactor

Gateway Connector and and on-premise CAs to make them available to Azure-hosted instance of Keyfactor Command.The portal is Azure-hosted and managed by Keyfactor.

Hosted Configuration Portal

The Keyfactor Hosted Configuration Portal is used to configure connections between on-premise instances of the Keyfactor Gateway Connector and and on-premise CAs to make them available to Azure-hosted instance of Keyfactor Command.The portal is Azure-hosted and managed by Keyfactor.

Hostname

The unique identifier that serves as name of a computer. It is sometimes presented as a fully qualified domain name (e.g. servername.keyexample.com) and sometimes just as a short name (e.g. servername).

J

Java Agent

The Java Agent, one of Keyfactor's suite of orchestrators, is used to perform discovery of Java keystores and PEM certificate stores, to inventory discovered stores, and to push certificates out to stores as needed.

Java Keystore

A Java KeyStore (JKS) is a file containing security certificates with matching private keys. They are often used by Java-based applications for authentication and encryption.

JKS

A Java KeyStore (JKS) is a file containing security certificates with matching private keys. They are often used by Java-based applications for authentication and encryption.

K

Key Length

The key size or key length is the number of bits in a key used by a cryptographic algorithm.

Key Pair

In asymmetric cryptography, public keys are used together in a key pair with a private key. The private key is retained by the key's creator while the public key is widely distributed to any user or target needing to interact with the holder of the private key.

Key Size

The key size or key length is the number of bits in a key used by a cryptographic algorithm.

Key Type

The key type identifies the type of key to create when creating a symmetric or asymmetric key. It references the signing algorithm and often key size (e.g. AES-256, RSA-2048, Ed25519).

Keyfactor CA Management Gateway

The Keyfactor CA Management Gateway is made up of the Keyfactor Gateway Connector, installed in the customer forest to provide a connection to the local CA, and the Azure-hosted and Keyfactor managed Hosted Configuration Portal. The solution is used to provide a connection between a customer's on-premise CA and an Azure-hosted instance of Keyfactor Command for synchronization, enrollment, and management of certificates.

Keyfactor Gateway Connector

The Keyfactor Gateway Connector is installed in the customer forest to provide a connection between the on-premise CA and the Azure-hosted, Keyfactor managed Hosted Configuration Portal to provide support for synchronization, enrollment and management of certificates through the Azure-hosted instance of Keyfactor Command for the on-premise CA. It is supported on both Windows and Linux.

Keyfactor Universal Orchestrator

The Keyfactor Universal Orchestrator, one of Keyfactor's suite of orchestrators, is used to interact with servers and devices for certificate management, run SSL discovery and management tasks, and manage synchronization of certificate authorities in remote forests. With the addition of custom extensions, it can provide certificate management capabilities on a variety of platforms and devices (e.g. Amazon Web Services (AWS) resources, Citrix\NetScaler devices, F5 devices, IIS stores, JKS keystores, PEM stores, and PKCS#12 stores) and execute tasks outside the standard list of certificate management functions. It runs on either Windows or Linux servers or Linux containers.

Keystore

A Java KeyStore (JKS) is a file containing security certificates with matching private keys. They are often used by Java-based applications for authentication and encryption.

L

Logical Name

The logical name of a CA is the common name given to the CA at the time it is created. For Microsoft CAs, this name can be seen at the top of the Certificate Authority MMC snap-in. It is part of the FQDN\Logical Name string that is used to refer to CAs when using command-line tools and in some Keyfactor Command configuration settings (e.g. ca2.keyexample.com\Corp Issuing CA Two).

M

MAC Agent

The MAC Agent, one of Keyfactor's suite of orchestrators, is used to manage certificates on any keychains on the Mac on which the Keyfactor MAC Agent is installed.

Metadata

Metadata provides information about a piece of data. It is used to summarize basic information about data, which can make working with the data easier. In the context of Keyfactor Command, the certificate metadata feature allows you to create custom metadata fields that allow you to tag certificates with tracking information about certificates.

O

Object Identifier

Object identifiers or OIDs are a standardized system for identifying any object, concept, or "thing" with a globally unambiguous persistent name.

OID

Object identifiers or OIDs are a standardized system for identifying any object, concept, or "thing" with a globally unambiguous persistent name.

Orchestrator

Keyfactor orchestrators perform a variety of functions, including managing certificate stores and SSH key stores.

P

P12

A PFX file (personal information exchange format), also known as a PKCS#12 archive, is a single, password-protected certificate archive that contains both the public and matching private key and, optionally, the certificate chain. It is a common format for Windows servers.

P7B

A PKCS #7 format certificate file is a base64-encoded certificate. Since it's presented in ASCII, you can open it in any text editor. PKCS #7 certificates always begin and end with entries that look something like ---- BEGIN CERTIFICATE---- and ----END CERTIFICATE----. Unlike PEM files, PKCS #7 files can contain only a certificate and its certifiate chain but NOT its private key. Extensions of .p7b or .p7c are usually seen on certificate files of this format.

P7C

A PKCS #7 format certificate file is a base64-encoded certificate. Since it's presented in ASCII, you can open it in any text editor. PKCS #7 certificates always begin and end with entries that look something like ---- BEGIN CERTIFICATE---- and ----END CERTIFICATE----. Unlike PEM files, PKCS #7 files can contain only a certificate and its certifiate chain but NOT its private key. Extensions of .p7b or .p7c are usually seen on certificate files of this format.

Parameter

A parameter or argument is a value that is passed into a function in an application.

PEM

A PEM format certificate file is a base64-encoded certificate. Since it's presented in ASCII, you can open it in any text editor. PEM certificates always begin and end with entries like ---- BEGIN CERTIFICATE---- and ----END CERTIFICATE----. PEM certificates can contain a single certificate or a full certifiate chain and may contain a private key. Usually, extensions of .cer and .crt are certificate files with no private key, .key is a separate private key file, and .pem is both a certificate and private key.

PFX

A PFX file (personal information exchange format), also known as a PKCS#12 archive, is a single, password-protected certificate archive that contains both the public and matching private key and, optionally, the certificate chain. It is a common format for Windows servers.

PKCS #7

A PKCS #7 format certificate file is a base64-encoded certificate. Since it's presented in ASCII, you can open it in any text editor. PKCS #7 certificates always begin and end with entries that look something like ---- BEGIN CERTIFICATE---- and ----END CERTIFICATE----. Unlike PEM files, PKCS #7 files can contain only a certificate and its certifiate chain but NOT its private key. Extensions of .p7b or .p7c are usually seen on certificate files of this format.

PKCS#12

A PFX file (personal information exchange format), also known as a PKCS#12 archive,

is a single, password-protected certificate archive that contains both the public and matching private key and, optionally, the certificate chain. It is a common format for Windows servers.

PKI

A public key infrastructure (PKI) is a set of roles, policies, and procedures needed to create, manage, distribute, use, store and revoke digital certificates and manage public-key encryption.

Private Key

Private keys are used in cryptography (symmetric and asymmetric) to encrypt or sign content. In asymmetric cryptography, they are used together in a key pair with a public key. The private or secret key is retained by the key's creator, making it highly secure.

Public Key

In asymmetric cryptography, public keys are used together in a key pair with a private key. The private key is retained by the key's creator while the public key is widely distributed to any user or target needing to interact with the holder of the private key.

Public Key Infrastructure

A public key infrastructure (PKI) is a set of roles, policies, and procedures needed to create, manage, distribute, use, store and revoke digital certificates and manage public-key encryption.

R

Rogue Key

A rogue key, in the context of Keyfactor Command, is an SSH public key that appears in an authorized_keys file on a server managed by the SSH orchestrator without authorization.

Root of Trust

A root of trust (RoT) is a source within a cryptographic system that can always be trusted. It is typically a hardened hardware module. HSMs (hardware security modules) and TPMs (trusted platform modules) are examples of RoTs.

RoT

A root of trust (RoT) is a source within a cryptographic system that can always be trusted. It is typically a hardened hardware module. HSMs (hardware security modules) and TPMs (trusted platform modules) are examples of RoTs.

RPC

Remote procedure call (RPC) allows one program to call a function from a program located on another computer on a network without specifying network details. In the context of Keyfactor Command, RPC errors often indicate Kerberos authentication or delegation issues.

rsyslog

Rsyslog is an open-source software utility used on UNIX and Unix-like computer systems for forwarding log messages in an IP network.

S

SAN

The subject alternative name (SAN) is an extension to the X.509 specification that allows you to specify additional values when enrolling for a digital certificate. A variety of SAN formats are supported, with DNS name being the most common.

server name indication

Server name indication (SNI) is an extension to TLS that provides for including the hostname of the target server in the initial handshake request to allow the server to respond with the correct SSL certificate or allow a proxy to forward the request to the appropriate target.

SMTP

Short for simple mail transfer protocol, SMTP is a protocol for sending email messages between servers.

SNI

Server name indication (SNI) is an extension to TLS that provides for including the hostname of the target server in the initial handshake request to allow the server to respond with the correct SSL certificate or allow a proxy to forward the request to the appropriate target.

SSH

The SSH (secure shell) protocol provides for secure connections between computers. It provides several options for authentication, including public key, and protects the communications with strong encryption.

SSL

TLS (Transport Layer Security) and its predecessor SSL (Secure Sockets Layer) are protocols for establishing authenticated and encrypted links between networked computers.

Subject Alternative Name

The subject alternative name (SAN) is an extension to the X.509 specification that allows you to specify additional values when enrolling for a digital certificate. A variety of SAN formats are supported, with DNS name being the most common.

T

Template

A certificate template defines the policies and rules that a CA uses when a request for a certificate is received.

TLS

TLS (Transport Layer Security) and its predecessor SSL (Secure Sockets Layer) are protocols for establishing authenticated and encrypted links between networked computers.

Trusted CA

A certificate authority in the forest in which Keyfactor Command is installed or in a forest in a two-way trust with the forest in which Keyfactor Command is installed.

U

Untrusted CA

A certificate authority in a forest in a one-way trust with the forest in which Keyfactor Command is installed or in a forest that is untrusted by the forest in which Keyfactor Command is installed. Non-domain-joined standalone CAs also fall into this category.

W

Web API

A set of functions to allow creation of applications. Keyfactor offers the Keyfactor API, which allows third-party software to integrate with the advanced certificate enrollment and management features of Keyfactor Command.

Windows Orchestrator

The Windows Orchestrator, one of Keyfactor's suite of orchestrators, is used to manage synchronization of certificate authorities in remote forests, run SSL discovery and management tasks, and interact with Windows servers as well as F5 devices, NetScaler devices, Amazon Web Services (AWS) resources, and FTP capable devices, for certificate management. In addition, the AnyAgent capability of the Windows Orchestrator allows it to be extended to create custom certificate store types and management capabilities regardless of source platform or location.

Workflow

A workflow is a series of steps necessary to complete a process. In the context of Keyfactor Command, it refers to the workflow builder, which allows you automate event-driven tasks when a certificate is requested or revoked.

X

x.509

In cryptography, X.509 is a standard defining the format of public key certificates. An X.509 certificate contains a public key and an identity (e.g. a host name or an organization or individual name), and is either signed by a certificate authority or self-signed. When a certificate is signed by a trusted certificate authority it can be used to establish trusted secure communications with the owner of the corresponding private key. It can also be used to verify digitally signed documents and emails.

4.0 Copyright Notice