Red Hat Certificate System 9

Release Notes

Highlighted features and updates related to Red Hat Certificate System 9 (9.0 - 9.4)

Last Updated: 2018-10-25

Red Hat Certificate System 9 Release Notes

Highlighted features and updates related to Red Hat Certificate System 9 (9.0 - 9.4)

Marc Muehlfeld
Red Hat Customer Content Services
mmuehlfeld@redhat.com

Petr Bokoč
Red Hat Customer Content Services

Filip Hanzelka
Red Hat Customer Content Services

Tomáš Čapek
Red Hat Customer Content Services

Legal Notice

Abstract

These release notes contain important information related to Red Hat Certificate System 9.4 that may not be currently available in the Product Documentation Guides. New features, system requirements, installation notes, known problems, resources, and other current issues are addressed here. You should read these Release Notes in their entirety before deploying Red Hat Certificate System 9.4.

Table of Contents

CHAPTER 1. RED HAT CERTIFICATE SYSTEM 9.4

This section describes changes in Red Hat Certificate System 9.4.

1.1. SUPPORTED PLATFORMS

This section describes the different server platforms, hardware, tokens, and software supported by Red Hat Certificate System 9.4.

1.1.1. Server Support

Running the Certificate Authority (CA), Key Recovery Authority (KRA), Online Certificate Status Protocol (OCSP), Token Key Service (TKS), and Token Processing System (TPS) subsystems of Certificate System 9.4 is supported on Red Hat Enterprise Linux 7.6 and later. The supported Directory Server version is 10.3 and later.

NOTE

Certificate System 9.4 is supported running on a Red Hat Enterprise Linux virtual guest on a certified hypervisor. For details, see the Which hypervisors are certified to run Red Hat Enterprise Linux? solution article.

1.1.2. Client Support

The Enterprise Security Client (ESC) is supported on:

- Red Hat Enterprise Linux 7.
- The latest versions of Red Hat Enterprise Linux 5 and 6.

  Although these platforms do not support Red Hat Certificate System 9.4, those clients can be used with the Token Management System (TMS) system in Red Hat Certificate System 9.4.

1.1.3. Supported Web Browsers

Certificate System 9.4 supports the following browsers:

Table 1.1. Supported Web Browsers by Platform

Platform	Agent Services	End User Pages
Red Hat Enterprise Linux	Firefox 60 and later [a]	Firefox 60 and later [a]
Windows 7	Firefox 60 and later [a]	Firefox 60 and later Internet Explorer 10 [b]

Platform	Agent Services	End User Pages

[a] This Firefox version no longer supports the **`crypto`** web object used to generate and archive keys from the browser. As a result, expect limited functionality in this area.

[b] Internet Explorer 11 is currently not supported by Red Hat Certificate System 9 because the enrollment code for this web browser depends upon Visual Basic Script, which has been deprecated in Internet Explorer 11.

NOTE

The only fully-supported browser for the HTML-based instance configuration is Mozilla Firefox.

1.1.4. Supported Smart Cards

The Enterprise Security Client (ESC) supports Global Platform 2.01-compliant smart cards and JavaCard 2.1 or higher.

The Certificate System subsystems have been tested using the following tokens:

- Gemalto TOP IM FIPS CY2 64K token (SCP01)
- Giesecke & Devrient (G&D) SmartCafe Expert 7.0 (SCP03)
- SafeNet Assured Technologies SC-650 (SCP01)

The only card manager applet supported with Certificate System is the **CoolKey** applet, which is part of the pki-tps package in Red Hat Certificate System.

1.1.5. Supported Hardware Security Modules

The following table lists Hardware Security Modules (HSM) supported by Red Hat Certificate System:

HSM	Firmware	Appliance Software	Client Software
Thales nCipher nShield Connect 6000	2.61.2	CipherTools-linux64-dev-12.30.00	CipherTools-linux64-dev-12.30.00
Gemalto SafeNet Luna SA 1700 / 7000 (limited) (Limited support [a])	6.24.0	6.2.0-15	libcryptoki-6.2.x86_64

[a] For details about supported features, see Section 1.1.5.1, “Gemalto SafeNet Luna SA 1700 / 7000 (limited)”.

1.1.5.1. Gemalto SafeNet Luna SA 1700 / 7000 (limited)

This section provides information on supported features when using the Gemalto SafeNet Luna SA 1700 / 7000 HSM.

Gemalto SafeNet Luna SA only supports PKI private key extraction in its CKE - Key Export model, and only in non-FIPS mode. The Luna SA Cloning model and the CKE model in FIPS mode do not support PKI private key extraction. Then Luna SA CKE – Key Export Model is in FIPS mode, PKI private keys cannot be extracted.

CL - Cloning Model

- Cloning of symmetric keys and objects: Possible to other Luna SAs/G5 or Luna Backup HSM
- Cloning of asymmetric (private) keys and objects: Possible to other Luna SAs/G5 or Luna Backup HSM
- Replication of symmetric keys and objects: All symmetric keys and objects are replicated when configured in an HA group
- Replication of asymmetric keys and objects: All asymmetric keys and objects are replicated when configured in an HA group
- Wrapping private (asymmetric) keys off the HSM: Not possible

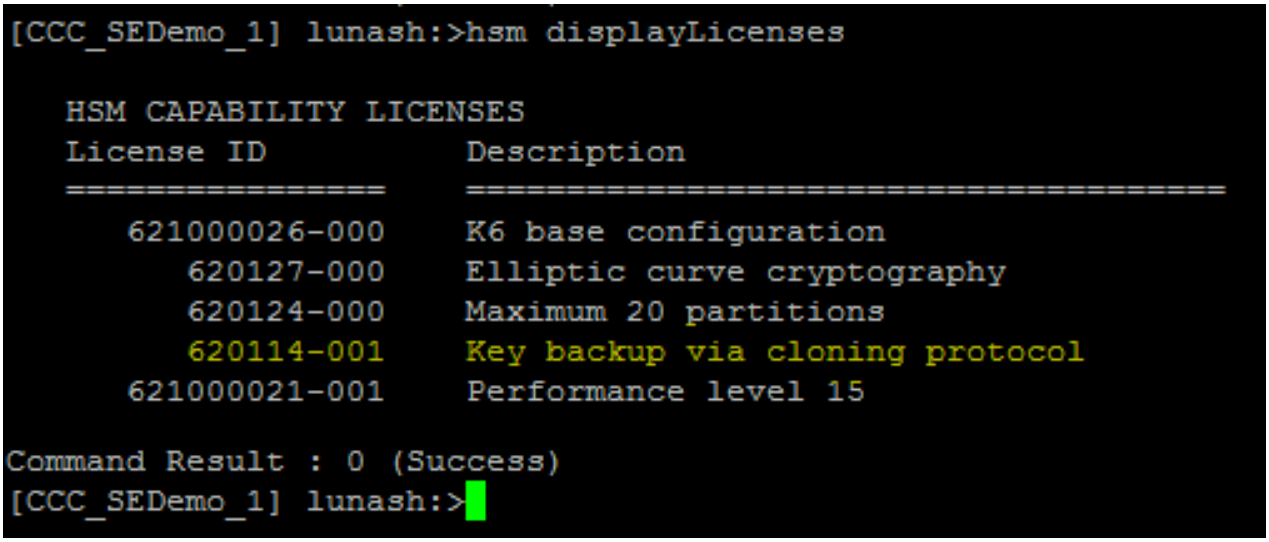


Figure 1.1. Example of a Cloning Model

CKE - Key Export Model

- Cloning of symmetric keys and objects: Possible to other Luna SAs/G5 or Luna Backup HSM
- Cloning of asymmetric (private) keys and objects: Not possible
- Replication of symmetric keys and objects: All symmetric keys and objects are replicated when configured in an HA group
- Replication of asymmetric keys and objects: Not possible
- Wrapping private (asymmetric) keys off the HSM: Possible

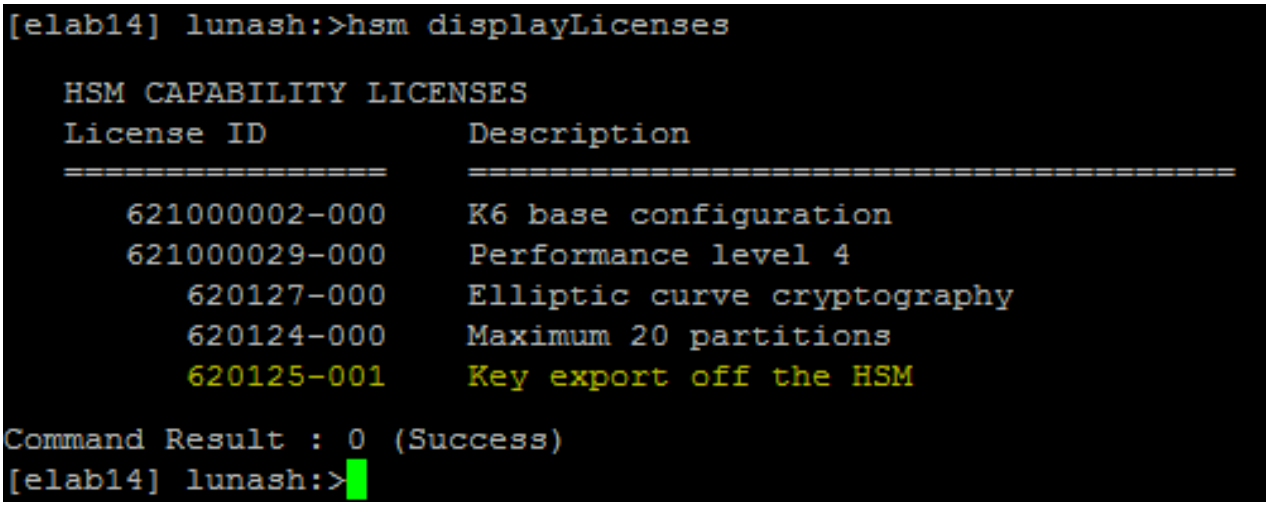


Figure 1.2. Example of a Key Export Model

1.2. HIGHLIGHTED UPDATES AND NEW FEATURES

This section documents new features and important updates in Red Hat Certificate System 9.4:

Certificate System packages rebased to version 10.5.9

The pki-core, redhat-pki, redhat-pki-theme, and pki-console packages have been upgraded to upstream version 10.5.9, which provides a number of bug fixes and enhancements over the previous version.

Highlighted Updates and New Features in the pki-core Package

Features in Red Hat Certificate System, that are included in the pki-core package, are documented in *Red Hat Enterprise Linux 7.6 Release Notes*:

- Certificate System now supports additional strong ciphers by default
- The CRMFPopClient utility supports CRMF requests without key archival
- Certificate System now adds the SAN extension to server certificates
- Certificate System automatically applies ECC profiles when setting up root CA with ECC certificates

1.3. BUG FIXES

This part describes bugs fixed in Red Hat Certificate System 9.4 that have a significant impact on users:

Server-side key generation succeeds with only one identity type certificate

Previously, if a user attempted to deploy a custom token enrollment profile with server-side key generation enabled and requested only one identity type certificate, the server-side key generation request failed. With this update, Certificate System successfully executes the server-side key generation and, as a result, enrolling a token in the mentioned scenario succeeds.

Bug Fixes in the pki-core Package

Bug fixes in Red Hat Certificate System, that are included in the pki-core package, are documented in *Red Hat Enterprise Linux 7.6 Release Notes*:

- The **pkiconsole** utility no longer accepts ACLs with an empty expression
- CMC CRMF requests using ECC keys work correctly
- The **client-cert-request** utility no longer fails to create CSRs for ECC certificates
- Installing Certificate System subsystems with ECC keys no longer fail
- The Certificate System installation no longer fails on hosts with multiple IP addresses
- The **nuxwdog** service starts correctly when a sub-CA is installed

1.4. DEPRECATED FUNCTIONALITY

This section describes deprecated functionality in Certificate System 9.4:

SCP01 support in Certificate System has been deprecated
In the next major release of Red Hat Certificate System, support for Secure Channel Protocol 01 (SCP01) will be removed. Red Hat recommends using smart cards which support SCP03.

CHAPTER 2. RED HAT CERTIFICATE SYSTEM 9.3

This section describes changes in Red Hat Certificate System 9.3.

2.1. SUPPORTED PLATFORMS

This section describes the different server platforms, hardware, tokens, and software supported by Red Hat Certificate System 9.3.

2.1.1. Server Support

Running the Certificate Authority (CA), Key Recovery Authority (KRA), Online Certificate Status Protocol (OCSP), Token Key Service (TKS), and Token Processing System (TPS) subsystems of Certificate System 9.3 is supported on Red Hat Enterprise Linux 7.5 and later. The supported Directory Server version is 10.2 and later.

NOTE

Certificate System 9.3 is supported running on a Red Hat Enterprise Linux virtual guest on a certified hypervisor. For details, see the Which hypervisors are certified to run Red Hat Enterprise Linux? solution article.

2.1.2. Client Support

The Enterprise Security Client (ESC) is supported on:

- Red Hat Enterprise Linux 7.
- The latest versions of Red Hat Enterprise Linux 5 and 6.

  Although these platforms do not support Red Hat Certificate System 9.3, those clients can be used with the Token Management System (TMS) system in Red Hat Certificate System 9.3.

2.1.3. Supported Web Browsers

Certificate System 9.3 supports the following browsers:

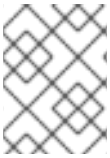
Table 2.1. Supported Web Browsers by Platform

Platform	Agent Services	End User Pages
Red Hat Enterprise Linux	Firefox 52 and later [a]	Firefox 52 and later [a]
Windows 7	Firefox 52 and later [a]	Firefox 52 and later Internet Explorer 10 [b]

Platform	Agent Services	End User Pages

[a] This Firefox version no longer supports the **`crypto`** web object used to generate and archive keys from the browser. As a result, expect limited functionality in this area.

[b] Internet Explorer 11 is currently not supported by Red Hat Certificate System 9 because the enrollment code for this web browser depends upon Visual Basic Script, which has been deprecated in Internet Explorer 11.



NOTE

The only fully-supported browser for the HTML-based instance configuration is Mozilla Firefox.

2.1.4. Supported Smart Cards

The Enterprise Security Client (ESC) supports Global Platform 2.01-compliant smart cards and JavaCard 2.1 or higher.

The Certificate System subsystems have been tested using the following tokens:

- Gemalto TOP IM FIPS CY2 64K token (SCP01)
- Giesecke & Devrient (G&D) SmartCafe Expert 6.0 (SCP03)
- SafeNet SC-650 (SCP01)

The only card manager applet supported with Certificate System is the **CoolKey** applet, which is part of the pki-tps package in Red Hat Certificate System.

2.1.5. Supported Hardware Security Modules

The following table lists Hardware Security Modules (HSM) supported by Red Hat Certificate System:

HSM	Firmware	Appliance Software	Client Software
Thales nCipher nShield Connect 6000	2.61.2	CipherTools-linux64-dev-12.30.00	CipherTools-linux64-dev-12.30.00
Gemalto SafeNet Luna SA 1700 / 7000 (limited) (Limited support [a])	6.24.0	6.2.0-15	libcryptoki-6.2.x86_64

[a] For details about supported features, see Section 2.1.5.1, “Gemalto SafeNet Luna SA 1700 / 7000 (limited)”.

2.1.5.1. Gemalto SafeNet Luna SA 1700 / 7000 (limited)

This section provides information on supported features when using the Gemalto SafeNet Luna SA 1700 / 7000 HSM.

Gemalto SafeNet Luna SA only supports PKI private key extraction in its CKE - Key Export model, and only in non-FIPS mode. The Luna SA Cloning model and the CKE model in FIPS mode do not support PKI private key extraction. Then Luna SA CKE – Key Export Model is in FIPS mode, PKI private keys cannot be extracted.

CL - Cloning Model

- Cloning of symmetric keys and objects: Possible to other Luna SAs/G5 or Luna Backup HSM
- Cloning of asymmetric (private) keys and objects: Possible to other Luna SAs/G5 or Luna Backup HSM
- Replication of symmetric keys and objects: All symmetric keys and objects are replicated when configured in an HA group
- Replication of asymmetric keys and objects: All asymmetric keys and objects are replicated when configured in an HA group
- Wrapping private (asymmetric) keys off the HSM: Not possible

```
[CCC_SEDemo_1] lunash:>hsm displayLicenses

   HSM CAPABILITY LICENSES
   License ID          Description
   ================    ========================================
      621000026-000    K6 base configuration
         620127-000    Elliptic curve cryptography
         620124-000    Maximum 20 partitions
         620114-001    Key backup via cloning protocol
      621000021-001    Performance level 15

Command Result : 0 (Success)
[CCC_SEDemo_1] lunash:>
```

Figure 2.1. Example of a Cloning Model

CKE - Key Export Model

- Cloning of symmetric keys and objects: Possible to other Luna SAs/G5 or Luna Backup HSM
- Cloning of asymmetric (private) keys and objects: Not possible
- Replication of symmetric keys and objects: All symmetric keys and objects are replicated when configured in an HA group
- Replication of asymmetric keys and objects: Not possible
- Wrapping private (asymmetric) keys off the HSM: Possible

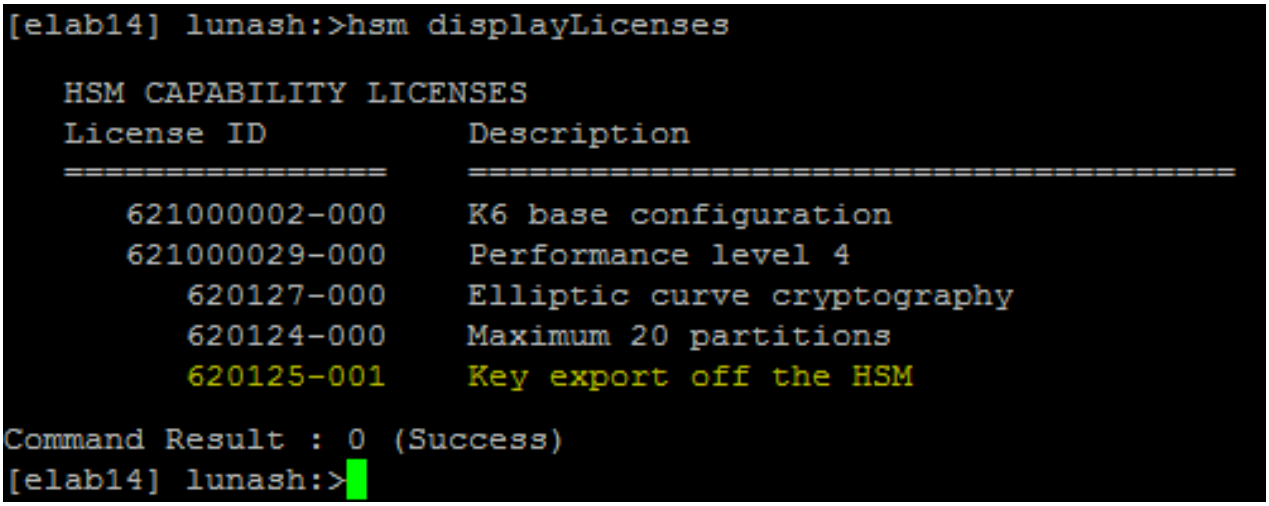


Figure 2.2. Example of a Key Export Model

2.2. HIGHLIGHTED UPDATES AND NEW FEATURES

This section documents new features and important updates in Red Hat Certificate System 9.3:

Certificate System packages rebased to version 10.5.1

The pki-core, redhat-pki, redhat-pki-theme, and pki-console packages have been upgraded to upstream version 10.5.1, which provides a number of bug fixes and enhancements over the previous version. Notably, this update addresses the requirements for the Common Criteria Protection Profile for Certification Authorities Version 2.1.

Certificate System is now RFC 5272-compliant

With this enhancement, Certificate System now complies with RFC 5272 - Certificate Management over CMS (CMC).

Therefore, several CMC features, such as the following, have been added and enhanced:

- The identity proof by signing with another certificate owned by the same entity to support enrollment, renewal, and revocation
- The **`IdentityProof V2`** control with Shared Secret for both enrollment and revocation
- The identification control to support Shared Secret
- The **`EncryptedPOP`** and **`DecryptedPOP`** controls for non-signing certificates
- The **`POPLinkWitnessV2`** control
- The TLS client authentication enforcement for user-signed CMC requests
- The **`CMCStatusInfoV2`** response

Additionally, the **`CMCRequest`** and **`CMCResponse`** utilities have been updated to support these new features, and the **`CMCSharedToken`** utility has been introduced to support the CMC Shared Secret feature.

Highlighted Updates and New Features in the pki-core Package

Features in Red Hat Certificate System, that are included in the pki-core package, are documented in *Red Hat Enterprise Linux 7.5 Release Notes*:

- Certificate System supports installing CA, KRA, and OCSP subsystems with CMC
- Certificate System CAs can now process CMC renewal requests signed by a previously issued signing certificate
- Certificate System now supports CMC-based system certificate requests
- The **`pki`** command-line interface automatically creates a default NSS database
- Certificate System can now create PKCS #12 files using PBES2 with PBKDF2 key derivation
- Certificate System disables weak 3DES ciphers by default
- Certificate System supports creating instances running as a different user
- Certificate System now supports configurable hashing algorithms for the SKI extension
- Certificate System now uses the Mozilla NSS secure random number generator
- The Certificate System CA subsystem's OCSP provider now includes the ***`nextUpdate`*** field in responses
- The Certificate System profile configuration update method now correctly handles backslashes

2.3. BUG FIXES

This part describes bugs fixed in Red Hat Certificate System 9.3 that have a significant impact on users:

In-place update of the TPS subsystem now adds the `externalRegISEtoken` profile to the `CS.cfg` file

A previous update of Certificate System added the **`externalRegISEtoken`** Token Processing System (TPS) profile to the default **`/usr/share/pki/tps/conf/CS.cfg`** TPS configuration file. However, the new configuration was not added to the **`/var/lib/pki/pki-instance_name/tps/conf/CS.cfg`** file when an administrator performed an in-place upgrade of an older TPS system. Consequently, upgraded TPS systems did not use the new configuration. This update fixes the problem. As a result, the token profile is now added automatically to the TPS configuration when performing an in-place upgrade.

Bug Fixes in the pki-core Package

Bug fixes in Red Hat Certificate System, that are included in the pki-core package, are documented in *Red Hat Enterprise Linux 7.5 Release Notes*:

- Certificate System no longer fails to import PKCS #12 files
- Certificate System CAs no longer display an error when handing subject DNs without a CN component
- The Certificate System CA key replication now works correctly
- Certificate System now validates the banner file

- The TPS user interface now displays the token type and origin fields
- The **pki-server-upgrade** utility no longer fails if target files are missing
- The TPS subsystem no longer fails when performing a symmetric key changeover on a HSM
- Certificate System no longer incorrectly logs **ROLE_ASSUME** audit events
- Signed audit log verification now works correctly
- Updated attributes in **CERT_STATUS_CHANGE_REQUEST_PROCESSED** audit log event
- CA certificates without SKI extension no longer causes issuance failures
- Certificate System correctly logs the user name in CMC request audit events
- The **pkidestroy** utility now fully removes instances that are started by the **pki-tomcatd-nuxwdog** service
- Certificate System issued certificates with an expiration date later than the expiration date of the CA certificate
- In-place update of Certificate System's TPS subsystem now adds the **externalRegISEtoken** profile to the **CS.cfg** file
- A race condition has been fixed in the Certificate System clone installation process
- Certificate System now uses strong ciphers by default
- The **pkispawn** utility no longer displays incorrect errors
- The Certificate System deployment archive file no longer contains passwords in plain text

2.4. KNOWN ISSUES

This part describes known problems and, if applicable, workarounds in Red Hat Certificate System 9.3:

Known Issues in the pki-core Package

Known Issues in Red Hat Certificate System, that are included in the pki-core package, are documented in *Red Hat Enterprise Linux 7.5 Release Notes*:

- **KRATool** fails to migrate data in certain situations

CHAPTER 3. RED HAT CERTIFICATE SYSTEM 9.2

The following sections detail changes for Red Hat Certificate System 9.2.

3.1. SUPPORTED PLATFORMS

This section covers the different server platforms, hardware, tokens, and software supported by Red Hat Certificate System 9.2.

3.1.1. Server and Client Support

The Certificate System 9.2 subsystems (CA, KRA, OCSP, TKS, and TPS) are supported on the Red Hat Enterprise Linux 7.4 and later platforms.

The Enterprise Security Client (ESC), which manages smart cards for end users, is also supported on Red Hat Enterprise Linux 7.

The ESC is also supported on latest versions of Red Hat Enterprise Linux 5 and 6. Although these platforms do not support Red Hat Certificate System 9.2, those clients can be used against the TMS system in Red Hat Certificate System 9.2.

3.1.2. Supported Web Browsers

The services pages for the subsystems require a web browser that supports SSL/TLS. It is strongly recommended that users such as agents or administrators use Mozilla Firefox to access the agent services pages. Regular users should use Mozilla Firefox .

NOTE

The only browser that is fully-supported for the HTML-based instance configuration is Mozilla Firefox.

Table 3.1. Supported Web Browsers by Platform

Platform	Agent Services	End User Pages
Red Hat Enterprise Linux	Firefox 38 and later	Firefox 38 and later
Windows 7	Firefox 40 and later	Firefox 40 and later Internet Explorer 10
Windows Server 2012	Firefox 40 and later	Firefox 40 and later

WARNING

Firefox versions 33, 35 and later, on all platforms, no longer support the `crypto` web object used to generate and archive keys from the browser. As a result, expect limited functionality in this area.

NOTE

Internet Explorer 11 is not currently supported by Red Hat Certificate System 9 because the enrollment code for this web browser depends upon VBScript, which has been deprecated in Internet Explorer 11.

3.1.3. Supported Smart Cards

The Enterprise Security Client supports Global Platform 2.01-compliant smart cards and JavaCard 2.1 or higher.

The Certificate System subsystems have been tested using the following tokens:

- Gemalto TOP IM FIPS CY2 64K token, both as a smart card and GemPCKey USB form factor key
- SafeNet Assured Technologies Smart Card 650 (SC650), with support for both SCP01
- G&D Smart Cafe 6.0 for SCP03

Note that all versions of SC650 require the Omnikey 3121 reader. Legacy smart cards can be used with the SCM SCR331 CCID reader.

The only card manager applet supported with Certificate System is the **CoolKey** applet, which is part of the pki-tps package in Red Hat Certificate System.

3.1.4. Supported HSM

Red Hat Certificate System 9.2 has been tested to support two hardware security modules (HSM): nCipher NShield connect 6000, and Gemalto SafeNet Luna SA 1700.

HSM	Firmware	Appliance Software	Client Software
nCipher nShield connect 6000	0.4.11cam2	CipherTools-linux64-dev-11.70.00	CipherTools-linux64-dev-11.70.00
Gemalto SafeNet Luna SA 1700	6.22.0	6.0.0-41	libcryptoki-5.4.1-2.x86_64

3.2. HIGHLIGHTED UPDATES AND NEW FEATURES

Red Hat Certificate System 9.2 has introduced the following new features and important updates:

IMPORTANT

Note that this document only contains release notes for features which are not available in the base Red Hat Enterprise Linux 7.4 release. Many of the new features in Red Hat Certificate System are in the pki-core, and those are documented in Red Hat Enterprise Linux 7.4 Release Notes.

New audit events have been added for SSL/TLS session events on Red Hat Certificate System servers

Red Hat Certificate System now supports several new audit log events related to **SSL** and **TLS** session events, namely successful and unsuccessful connection establishments and connection terminations.

The new log events are:

- ACCESS_SESSION_ESTABLISH_SUCCESS for successful connections
- ACCESS_SESSION_ESTABLISH_FAILURE for failed connections
- ACCESS_SESSION_TERMINATED for terminated connections

These new events are logged in the server audit log file by default. Use the **CS.cfg** file to further configure these settings. (BZ#1404080)

Red Hat Certificate System can now display a custom banner at the start of a secure connection

New configuration options have been added to Red Hat Certificate System to allow a customizable banner to be displayed at the beginning of a secure connection. This allows organizations to display messages such as advisory notices and warning messages regarding unauthorized use. The message will be displayed each time a PKI client (the PKI command line, web user interface, or PKI Console) connects to the server using a **SSL** or **TLS** connection. The connecting user will be prompted to confirm they read the banner before resuming normal client operation.

To enable this functionality, create a file at **/etc/pki/pki-tomcat/banner.txt** and place the message you want to display into this file. Make sure the file is encoded as **UTF-8** and readable by the **pkiuser** user account. To remove the banner, delete the aforementioned file. No server restart is required to add, change, or remove the banner. (BZ#1404085)

New tools to retrieve audit logs from Red Hat Certificate System server

New tools for retrieving audit logs have been added to Red Hat Certificate System in order to allow auditors to retrieve audit logs locally for inspection and verification.

To list existing audit log files, use the following command:

```
pki <subsystem>-audit-file-find
```

To retrieve a specific audit log file, use the following command:

```
pki <subsystem>-audit-file-retrieve <filename>
```

After retrieving audit logs you require, use standard tools such as **grep** to search for specific log entries, and the **AuditVerify** tool to verify their authenticity. For more information on these tools, see their respective man pages. (BZ#1417307)

New session timeout parameter for `PKI Console`

A new parameter, **keepAliveTimeout**, has been added to Certificate System's server configuration file. This parameter controls the session timeout period for **PKI Console**. **PKI Console** will be automatically disconnected from the server after it has been idle for a time period specified in this parameter; the Console will then display an error message and terminate.

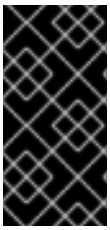
The timeout is configured in the **server.xml** file, and accepts an integer which specifies the timeout period in miliseconds. The default value is **300000**, which is 5 minutes. (BZ#1446877)

Certificate System now supports SCP03-enabled tokens

With this enhancement, Certificate System now supports the secure channel protocol 03 (SCP03) enabled Giesecke & Devrient (G&D) Smart Cafe 6 and Smart Cafe 7 tokens in Token Management System (TMS). This allows TMS users to perform token operations, such as token formatting and enrollment upon smart cards that respond to SCP03, which provides extra security using the advanced encryption standard (AES) during token token operations. (BZ#1274086)

3.3. BUG FIXES

Red Hat Certificate System 9.2 has introduced the following important bug fixes:



IMPORTANT

Note that this document only contains release notes for features which are not available in the base Red Hat Enterprise Linux 7.4 release. Many of the bug fixes in Red Hat Certificate System are in the pki-core, and those are documented in Red Hat Enterprise Linux 7.4 Release Notes.

Token memory is now cleaned after the deletion of keys and certificates

Previously, the Token Processing System (TPS) left old data in a token's Coolkey applet in certain situations when re-enrolling the token with new certificates and keys. This bug is now fixed, and only the data associated with certificates, which is actually on the token, is preserved after a successful re-enrollment. (BZ#1405655)

CHAPTER 4. RED HAT CERTIFICATE SYSTEM 9.1

The following sections detail changes for Red Hat Certificate System 9.1.

4.1. SUPPORTED PLATFORMS

This section covers the different server platforms, hardware, tokens, and software supported by Red Hat Certificate System 9.2.

4.1.1. Server and Client Support

The Certificate System 9.1 subsystems (CA, KRA, OCSP, TKS, and TPS) are supported on the Red Hat Enterprise Linux 7.3 and later platforms.

The Enterprise Security Client (ESC), which manages smart cards for end users, is also supported on the Red Hat Enterprise Linux 7.3 and later platforms.

The ESC is also supported on latest versions of Red Hat Enterprise Linux 5 and 6. Although these platforms do not support Red Hat Certificate System 9.1, those clients can be used against the TMS system in Red Hat Certificate System 9.1.

4.1.2. Supported Web Browsers

The services pages for the subsystems require a web browser that supports SSL/TLS. It is strongly recommended that users such as agents or administrators use Mozilla Firefox to access the agent services pages. Regular users should use Mozilla Firefox .

NOTE

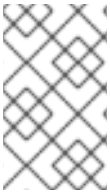
The only browser that is fully-supported for the HTML-based instance configuration is Mozilla Firefox.

Table 4.1. Supported Web Browsers by Platform

Platform	Agent Services	End User Pages
Red Hat Enterprise Linux	Firefox 38 and later	Firefox 38 and later
Windows 7	Firefox 40 and later	Firefox 40 and later Internet Explorer 10
Windows Server 2012	Firefox 40 and later	Firefox 40 and later

WARNING

Firefox versions 33, 35 and later, on all platforms, no longer support the `crypto` web object used to generate and archive keys from the browser. As a result, expect limited functionality in this area.



NOTE

Internet Explorer 11 is not currently supported by Red Hat Certificate System 9 because the enrollment code for this web browser depends upon VBScript, which has been deprecated in Internet Explorer 11.

4.1.3. Supported Smart Cards

The Enterprise Security Client supports Global Platform 2.01-compliant smart cards and JavaCard 2.1 or higher.

The Certificate System subsystems have been tested using the following tokens:

- Gemalto TOP IM FIPS CY2 64K token, both as a smart card and GemPCKey USB form factor key
- SafeNet Assured Technologies Smart Card 650 (SC650), with support for both SCP01 and SCP02

Note that all versions of SC650 require the Omnikey 3121 reader. Legacy smart cards can be used with the SCM SCR331 CCID reader.

The only card manager applet supported with Certificate System is the **CoolKey** applet, which is part of the pki-tps package in Red Hat Certificate System.

4.1.4. Supported HSM

Red Hat Certificate System 9.1 has been tested to support two hardware security modules (HSM): nCipher NShield connect 6000, and Gemalto SafeNet Luna SA 1700.

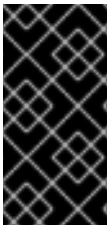
HSM	Firmware	Appliance Software	Client Software
nCipher nShield connect 6000	0.4.11cam2	CipherTools-linux64-dev-11.70.00	CipherTools-linux64-dev-11.70.00
Gemalto SafeNet Luna SA 1700	6.22.0	6.0.0-41	libcryptoki-5.4.1-2.x86_64

4.2. NOTE ON TOKEN PROCESSING SYSTEM UPGRADES

Because the Token Processing System (TPS) subsystem in Red Hat Certificate System 9.0 was released as a tech preview, upgrade to Red Hat Certificate System 9.1 or later is not supported. The TPS subsystem needs to be uninstalled from Red Hat Certificate System 9.0, and then reinstalled on Red Hat Certificate System 9.1. See the Red Hat Certificate System Planning, Installation and Deployment Guide for installation and uninstallation instructions.

4.3. HIGHLIGHTED UPDATES AND NEW FEATURES

Red Hat Certificate System 9.1 has introduced the following new features and important updates:



IMPORTANT

Note that this document only contains release notes for features which are not available in the base Red Hat Enterprise Linux 7.3 release. Many of the new features in Red Hat Certificate System are in the pki-core, and those are documented in Red Hat Enterprise Linux 7.3 Release Notes.

New Java-based Token Processing System

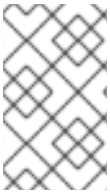
Red Hat Certificate System 9.1 replaces the Apache HTTPD-based Token Processing System (TPS) with a Java Tomcat-based TPS. The new Java-based TPS retains feature parity with the existing C-based implementation and provides a new user interface for better user experience.

NOTE

This feature was offered as a Technology Preview in the previous release of Red Hat Certificate System. This release changes the feature status to fully supported.

Global Platform 2.1.1 in the Token Processing System

The latest version of Global Platform has been included and supported in the version of TPS that comes with Red Hat Certificate System 9. TPS is now able to provision cards that support newer versions of Global Platform and the latest cryptographic operations. In particular, the **gp211** applet has been introduced that provides support for Secure Channel Protocol 02 (SCP02). SCP02 has been tested with SafeNet Assured Technologies Smart Card 650.



NOTE

This feature was offered as a Technology Preview in the previous release of Red Hat Certificate System. This release changes the feature status to fully supported.

Certificate System now supports setting SSL ciphers for individual installation

Previously, if an existing **Certificate Server** had a customized cipher set that did not overlap with the default ciphers used during the installation, a new instance could not be installed to work with existing instances. With this update, Certificate System enables you to customize the **SSL** cipher using a two-step installation, which avoids this problem.

To set the ciphers during a Certificate System instance installation:

1. Prepare a deployment configuration file that includes the **pki_skip_configuration=True** option.

2. Pass the deployment configuration file to the **pkispawn** command to start the initial part of the installation.

3. Set the ciphers in the **sslRangeCiphers** option in the **/var/lib/pki/*instance*/conf/server.xml** file. Replace *instance* with the instance name.

4. Replace the **pki_skip_configuration=True** option set in the first step with **pki_skip_installation=True** in the deployment configuration file.

5. Run the same **pkispawn** command to complete the installation.

Man pages updates

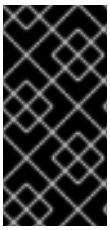
Man pages for many tools provided by Red Hat Certificate System 9 have been added, rewritten or significantly updated in this release. Important usage information that was previously published in the *Red Hat Certificate System 9 Command-Line Tools Guide* is now in man pages, ensuring access to this information on any system where Certificate System is installed, even without internet access. At the same time, the Command-Line Tools Guide is deprecated for Red Hat Certificate System 9.1 and will not be published on the Red Hat Customer Portal.

Certificate System now uses a specific JDK and version and no longer supports alternatives

Red Hat Certificate System 9.1 no longer relies on the system java selectable using the **/usr/sbin/alternatives** mechanism. Instead, Red Hat Certificate System 9.1 always uses its own specified JDK and version. For Red Hat Certificate System 9.1, this JDK is java-1.8.0-openjdk, and the version is 1:1.8.0.

4.4. BUG FIXES

Red Hat Certificate System 9.1 has introduced the following important bug fixes:



IMPORTANT

Note that this document only contains release notes for features which are not available in the base Red Hat Enterprise Linux 7.3 release. Many of the bug fixes in Red Hat Certificate System are in the pki-core, and those are documented in Red Hat Enterprise Linux 7.3 Release Notes.

Installing the redhat-pki package no longer requires the Server-optional repository

Previously, the redhat-pki package depended on the jss-javadoc package, which was only available from the **Server-optional** repository. Therefore, installation failed on systems where this repository was not enabled.

With this update, the redhat-pki package no longer depends on jss-javadoc, and therefore the **Server-optional** repository is not required in order to install redhat-pki. The jss-javadoc package is still only provided by **Server-optional**, which must be enabled using **Subscription Manager** before you can install the package.

4.5. KNOWN ISSUES

Red Hat Certificate System 9.1 is affected by the following known issues:

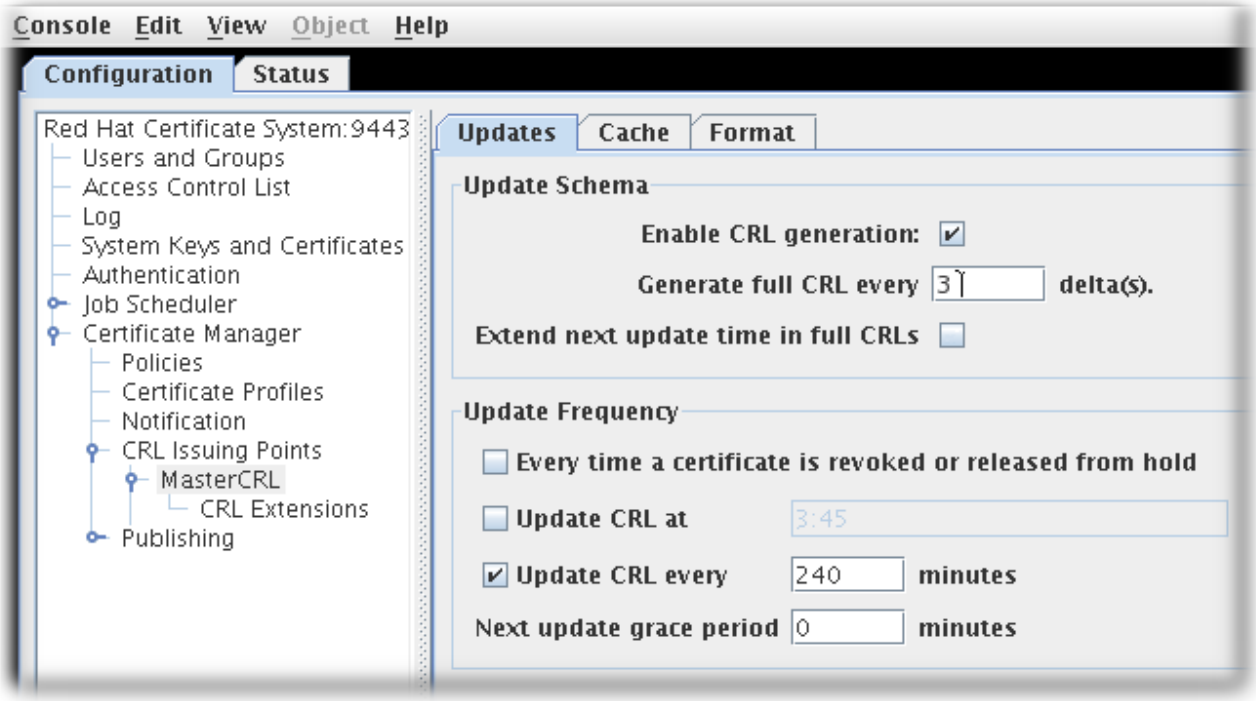
IMPORTANT

Note that this document only contains release notes for features which are not available in the base Red Hat Enterprise Linux 7.3 release. Some known issues in Red Hat Certificate System are in the pki-core, and those are documented in Red Hat Enterprise Linux 7.3 Release Notes

CRL cannot be configured in console to update after every revocation or release from hold unless grace period is set

Currently, the Certificate Revocation List (CRL) update cannot be configured solely based on certificate revocation events. When setting full and delta CRL schedules, the **`Update CRL every time a certificate is revoked or released from hold`** option also requires you to fill out the two **`grace period`** settings.

Thus, in order to select this option you need to first select the **`Update CRL every`** option and enter a number for the **`Next update grace period # minutes`** box.



Firefox can no longer enroll a signing and an archival certificate at the same time

The **`caDualCert.cfg`** profile previously used the Mozilla cryptographic object to create two requests, one for a signing certificate and the other for an encryption certificate, with private key archival specified for the encryption certificate. Since Mozilla has removed the **`generateCRMFRequest()`** object, Red Hat Certificate System can no longer support this type of enrollment within the browser.

The following procedure specifies how to generate the same two certificates using the **`pki`** command line interface (CLI) tools. It describes manual user signing and encryption certificates enrollment.

1. Enroll for the signing-only certificate:

    a. Create the certificate signing request (CSR) using **`certutil`**:

```
certutil -R -k rsa -g 2048 -s "CN=John Smith,O=Example
Corp,L=Mountain View,ST=California,C=US" -d ./  -a -o cert.cer
```

b. Send the request to the Certificate Authority (CA) using the **caSigningUserCert** profile:

```
pki  ca cert-request-submit --csr-file ./cert.cer --profile
caSigningUserCert --subject "CN=John Smith,O=Example
Corp,L=Mountain View,ST=California,C=US"
```

c. The final certificate can be retrieved using the standard end entity (EE) graphical user interface of the product:

2. Enroll for the encryption-only certificate:

a. Obtain the transport certificate from the Key Recovery Authority (KRA):

```
pki -C ""  -U 'https://localhost:8443/ca'  cert-show 0x07 --
encoded --output transport.pem
```

b. One can make sure that certificate **0x07** is actually the transport certificate for the KRA by consulting the end entity interface of the product. If the certificate happens to have a different ID, use that one instead of **0x07**.

c. Use the **CRMFPopClient** command to create the CSR for the encryption certificate that will have the private key archived to the KRA.

Here we use the **caEncUserCert** profile to obtain this certificate:

```
CRMFPopClient -d . -p password "secret123" -o csr -a rsa -l 2048
-n "UID=username" -f caEncUserCert -b transport.pem
```

d. Get the enrollment template for the **caEncUserCert** profile:

```
pki -v -C "secret123" -U https://localhost:8443/ca cert-request-
profile-show caEncUserCert --output encuser.xml
```

e. Sanitize the line endings in the **csr** file you just created:

```
dos2unix csr
```

f. Fill in the enrollment template as follows:

```
cert_request_type = crmf
cert_request = <copied certificate request blob from the file
csr>
sn_cn = <your cn value>
```

g. Submit the final request to the CA:

```
pki -v -C "secret123" -U https://localhost.localdomain:8443/ca
cert-request-submit encuser.xml
```

The terminal should print out success or failure of the enrollment.

If the enrollment is successful, the agent can approve this request, issuing the certificate. Note that the approval will trigger an archival of the encryption key to the KRA.

The EE interface of the product can be used to obtain the new encryption certificate.

caUserCert profile request using Internet Explorer 10 results in an Invalid request error

Currently, when trying to submit request using **caUserCert** profile using Internet Explorer 10 from Windows 7, the request results in the "Invalid request" error. The following procedure specifies how to work around this problem.

1. Go to the **Internet Options/Advanced/Security** section and uncheck the **TLS 1.2** box to connect to the SSL port.

2. Go to the **End Entity** page which allows downloading and importing the CA certificate chain. Add the CA certificate to the **Trusted CAs list**.

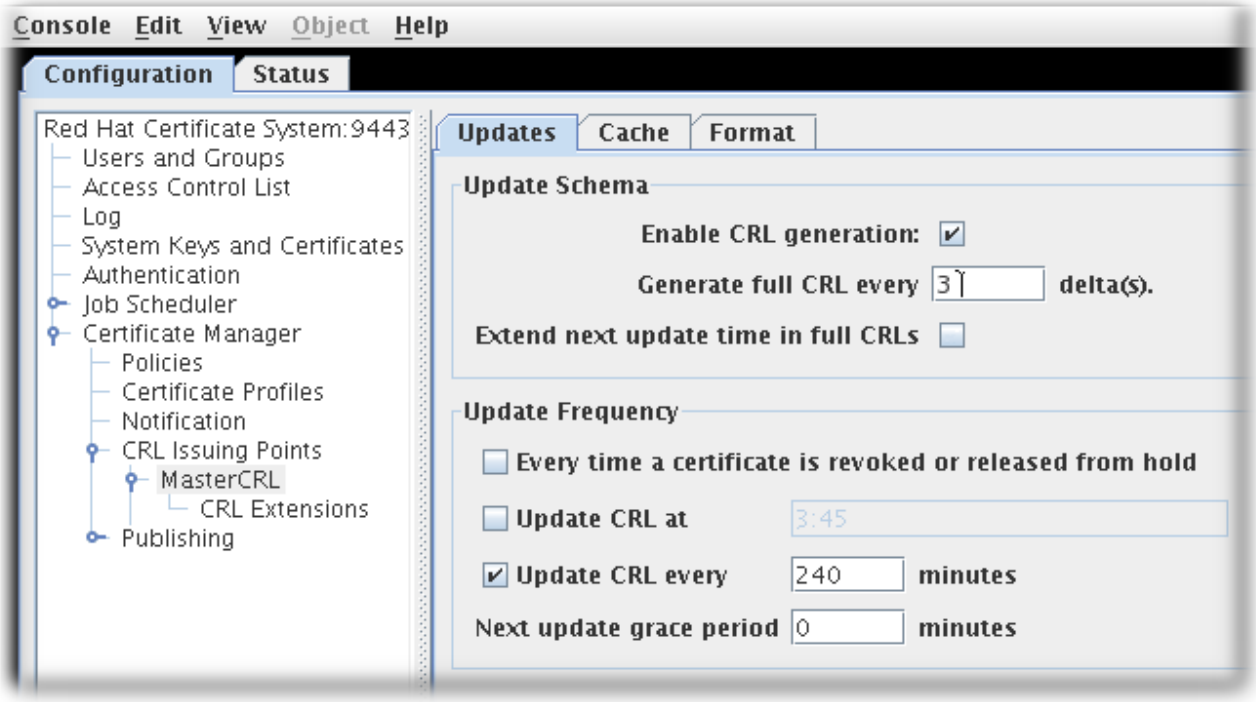
3. Go to **Internet Options** and enter the **Security** tab. Add the SSL url to the **Trusted Sites** list. Set the security slider to medium high or, alternatively, choose medium or below if trying to troubleshoot problems.

4. Go to the **Compatibility View Settings** settings by clicking the **Tools** dropdown menu on the right and add the site to the list. Alternatively, enable the view for intranet sites or all sites.

5. Go to the usual dual use profile enrollment page. The browser will probably issue a warning that a cryptographic operation is about to occur. Accept that by clickign OK. At this point the display should have a drop down list next to the key size list that contains the Communication Service Providers (CSPs). If this list is not empty, attempt an enrollment.

The console does not enable a separate ticking of the Update CRL every time a certificate is revoked or released from hold option

Currently, when setting full and delta CRL schedules, the **Update CRL every time a certificate is revoked or released from hold** option also requires you to fill out the two **grace period** settings. Thus, in order to select the **Update CRL every time a certificate is revoked or released from hold** option you need to first select the **Update CRL every** option and enter a number for the **Next update grace period # minutes** box.



Additional steps needed when uninstalling TPS

Due to a known issue, you must perform some additional steps when uninstalling the Token Processing System which were not necessary in earlier versions. See the **pkidestroy(8)** man page for instructions.

CHAPTER 5. RED HAT CERTIFICATE SYSTEM 9.0

Red Hat Certificate System 9.0 is a major release of the product, new, contemporary features have been added, and existing features have been made more robust and flexible.

5.1. SUPPORTED PLATFORMS

This section covers the different server platforms, hardware, tokens, and software supported by Red Hat Certificate System 9.0.

5.1.1. Server and Client Support

The Red Hat Certificate System subsystems (CA, KRA, OCSP, TKS, and TPS) are supported on the Red Hat Enterprise Linux 7.1 and later (64-bit) platforms.

The Enterprise Security Client (ESC), which manages smart cards for end users, is also supported on the Red Hat Enterprise Linux 7.1 and later (64-bit) platforms.

The ESC is also supported on latest versions of Red Hat Enterprise Linux 5 and 6. Although these platforms do not support Red Hat Certificate System 9, those clients can be used against the TMS system in Red Hat Certificate System 9.

5.1.2. Supported Web Browsers

The services pages for the subsystems require a web browser that supports SSL/TLS. It is strongly recommended that users such as agents or administrators use Mozilla Firefox to access the agent services pages. Regular users should use Mozilla Firefox.

NOTE

The only browser that is fully-supported for the HTML-based instance configuration is Mozilla Firefox.

Table 5.1. Supported Web Browsers by Platform

Platform	Agent Services	End User Pages
Red Hat Enterprise Linux	Firefox 38 and later	Firefox 38 and later
Windows 7	Firefox 40 and later	Firefox 40 and later Internet Explorer 10
Windows Server 2012	Firefox 40 and later	Firefox 40 and later

WARNING

Firefox versions 33, 35 and later, on all platforms, no longer support the **`crypto`** web object used to generate and archive keys from the browser. As a result, expect limited functionality in this area.

NOTE

Internet Explorer 11 is not currently supported by Red Hat Certificate System 9 because the enrollment code for this web browser depends upon VBScript, which has been deprecated in Internet Explorer 11.

5.1.3. Supported Smart Cards

The Enterprise Security Client supports Global Platform 2.01-compliant smart cards and JavaCard 2.1 or higher.

The Red Hat Certificate System subsystems have been tested using the following tokens:

- Gemalto TOP IM FIPS CY2 64K token, both as a smart card and GemPCKey USB form factor key
- SafeNet Assured Technologies Smart Card 650 (SC650), with support for both SCP01 and SCP02

Note that all versions of SC650 require the Omnikey 3121 reader. Legacy smart cards can be used with the SCM SCR331 CCID reader.

The only card manager applet supported with Red Hat Certificate System is the CoolKey applet, which is part of the pki-tps package in Red Hat Certificate System.

5.1.4. Supported HSM

Red Hat Certificate System 9 has been tested to support two hardware security modules (HSM): nCipher NShield connect 6000, and Gemalto SafeNet Luna SA 1700.

HSM	Firmware	Appliance Software	Client Software
nCipher nShield connect 6000	0.4.11cam2	CipherTools-linux64-dev-11.70.00	CipherTools-linux64-dev-11.70.00
Gemalto SafeNet Luna SA 1700	6.22.0	6.0.0-41	libcryptoki-5.4.1-2.x86_64

5.2. INSTALLING RED HAT CERTIFICATE SYSTEM SUBSYSTEMS

The following sections contain information on the prerequisites and procedures for installing Red Hat Certificate System subsystems, including basic information that you need to begin installing the packages. Basic installation process is as follows:

1. Install a Red Hat Enterprise Linux 7.1 server with an active network connection.

NOTE

While not required, for most real-world deployments, the recommended approach is to install the Red Hat Directory Server and the Red Hat Certificate System on separate machines with Red Hat Enterprise Linux 7.1 installed. If separate machines are used in the deployment, unless otherwise noted, execute the following command sequences on both machines.

2. Subscribe the system using Red Hat Subscription Manager and attach the subscription providing Red Hat Certificate System:

```
# subscription-manager register
      # subscription-manager list --available --all
```

Make note of the pool ID providing Red Hat Certificate System. In addition, all certificate subsystems also require access to Red Hat Directory Server:

```
# subscription-manager attach --pool=POOL_ID_CERT_SYSTEM
      # subscription-manager attach --pool=POOL_ID_DIR_SERVER
```

For the machine where the certificate system resides, also make sure to attach the subscription for the Red Hat Enterprise Linux Server, and then enable the Red Hat Enterprise Linux Optional repository:

```
# subscription-manager attach --
pool=POOL_ID_Red_Hat_Enterprise_Linux_Server
      # subscription-manager repos --enable rhel-7-server-optional-
rpms
```

3. Enable the certificate system and directory server repositories.
   - Enable the certificate server repository on the machine where it will reside:

     ```
     # subscription-manager repos --enable=rhel-7-server-rhcs-9-rpms
     ```

   - Enable the directory server repository on the machine where it will reside:

     ```
     # subscription-manager repos --enable=rhel-7-server-rhds-10-rpms
     ```

4. Before continuing, make sure that the latest updates have been applied to each Red Hat Enterprise Linux 7.1 system you use:

```
# yum update
```

5. On the machine where it will reside, install the directory server packages:

```
# yum install redhat-ds
```

6. Ensure that a real domain name is specified in each **/etc/resolv.conf** file and that a host name is set within each **/etc/hosts** file.

7. On the machine where it will reside, run the directory server installation script, selecting the defaults or customizing as required:

```
# /usr/sbin/setup-ds-admin.pl
```

8. On the machine where it will reside, install the certificate system packages:

```
# yum install redhat-pki
```

9. On the machine where the certificate server resides, run the **pkispawn** script to create and configure the subsystem instances. At least one CA subsystem must be installed and fully configured before any other type of subsystem can be configured. For details, see the pkispawn man page.

10. To access the agent interface of various Red Hat Certificate System subsystems, use a properly configured local or remote Mozilla Firefox web browser.

Installing and configuring Red Hat Certificate System subsystems is described in more detail in the Planning, Installation, and Deployment Guide.

5.2.1. Verifying JDK version

Red Hat Certificate System supports and automatically installs OpenJDK 1.7.0.

If you require another version, the OpenJDK can be installed by using **yum** or by downloading the packages directly from http://openjdk.java.net/install/. For example:

```
# yum install java-1.7.0-openjdk
```

After installing the JDK, run **/usr/sbin/alternatives** as root to ensure that the proper JDK is available and selected in order to use Red Hat Certificate System 9:

```
# /usr/sbin/alternatives --config java
      There are 3 programs which provide 'java'.
      Selection    Command
      -----------------------------------------------
      1           /usr/lib/jvm/jre-1.4.2-gcj/bin/java
      + 2           /usr/lib/jvm/jre-1.7.0-openjdk/bin/java
      *  3           /usr/lib/jvm/jre-1.6.0-sun.x86_64/bin/java
```

Use the **/usr/sbin/alternatives** command to configure the appropriate selection if it has not already been selected.

5.2.2. Installing through `yum`

To install the subsystems on Red Hat Enterprise Linux 7.1, run a command like the following for each subsystem:

```
# yum install pki-subsystem
```

subsystem can be any of the Red Hat Certificate System subsystems:

- **ca** for the Certificate Manager.
- **kra** for the Key Recovery Authority.
- **ocsp** for the Online Certificate Status Protocol Responder.
- **tks** for the Token Key System.
- **tps** for the Token Processing System.
- **console** for the Java console.

To install all Red Hat Certificate System 9 certificate server PKI packages, enter:

```
# yum install redhat-pki
```

5.2.3. Installing from an ISO image

Red Hat Certificate System 9 can be downloaded from Content Delivery Network as an ISO image. This ISO image contains an **RPMS/** directory which can be used as a local**yum** repository.

For the machine where the certificate system resides, make sure to attach the subscription for the Red Hat Enterprise Linux Server, and then enable the Red Hat Enterprise Linux Optional repository:

```
# subscription-manager attach --
pool=POOL_ID_Red_Hat_Enterprise_Linux_Server
      # subscription-manager repos --enable rhel-7-server-optional-rpms
```

Place the **RPMS/** directory on a web server and then configure**yum** to use that location as a repository. After that, install Red Hat Certificate System as described in Section 5.2.2, “Installing through **yum**”.

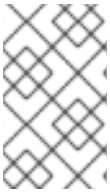
Red Hat Directory Server can also be obtained as an ISO image. See the Directory Server documentation for details.

5.3. HIGHLIGHTED UPDATES AND NEW FEATURES

Red Hat Certificate System 9.0 on Red Hat Enterprise Linux 7.1 requires packages from the `Optional` repository
When the Red Hat Certificate System 9.0 layered product is deployed on Red Hat Enterprise Linux 7.1, it requires access to packages that only exist in the Red Hat Enterprise Linux **Optional** repository. These are the required packages:

```
resteasy-base-client >= 3.0.6-1 is needed by pki-base-10.2.4-2.el7.noarch
    resteasy-base-jackson-provider >= 3.0.6-1 is needed by pki-base-
```

```
10.2.4-2.el7.noarch
    libsvrcore.so.0()(64bit) is needed by pki-tps-10.2.4-2.el7.x86_64
    jss-javadoc >= 4.2.6-35 is needed by redhat-pki-10.2.4-1.el7.noarch
    nuxwdog-client-java >= 1.0.1-11 is needed by pki-server-10.2.4-
2.el7.noarch
```



NOTE

Note that as of Red Hat Enterprise Linux 7.2, these packages will be added among common dependencies, thus eliminating the requirement to use the **Optional** repository.

A New `pki` Command-line Utility

Red Hat Certificate System 9 introduces a new **pki** command-line utility that provides an interface to access PKI services on a PKI server. The main purpose of the utility is to:

- allow commonly used CA and KRA functionality to be usable from the command line for end users and for simple scripting and automation purposes.
- allow use of the new REST API operations from the command line.

For more information about the **pki** utility, see the pki man page.

Simplified Installation and Deployment

Several new features for simplified installation and deployment have been introduced in Red Hat Certificate System 9.0 to provide the following functions:

- Simplify silent installation by using INI-like configuration files instead of command-line arguments
- Instance creation and configuration can be performed in a single automated operation
- Multiple subsystems can be deployed in a single Tomcat instance.

For more information about the improvements to installation and deployments, see the pkispawn man page.

Technology Preview: Global Platform 2.1.1 in TPS

NOTE

Note that this feature is offered as a technology preview, provides early access to upcoming product functionality, and is not yet fully supported under subscription agreements.

The latest version of Global Platform has been included and supported in the version of TPS that comes with Red Hat Certificate System 9. TPS is now able to provision cards that support newer versions of Global Platform and the latest cryptographic operations. In particular, the **gp211** applet has been introduced that provides support for Secure Channel Protocol 02 (SCP02). SCP02 has been tested with Assured Technologies SafeNet Smart Card 650.

REST Web Service APIs

Red Hat Certificate System 9 provides a new set of REST APIs to access various web services of the Certificate System. It also provides Java and Python client libraries to allow easier integration with other applications.

Technology Preview: New Java-based Token Processing System

NOTE

Note that this feature is offered as a technology preview, provides early access to upcoming product functionality, and is not yet fully supported under subscription agreements.

Red Hat Certificate System 9 replaces the Apache HTTPD-based TPS with a Java Tomcat-based TPS. The new Java-based TPS retains feature parity with the existing C-based implementation and provides a new user interface for better user experience.

KRA Enhancements

Previously, the Key Recovery Authority (KRA) only archived private (asymmetric) encryption keys when enrolling certificates using certain profiles in the CA. In Red Hat Certificate System 9, KRA has been extended to archive other types of secrets, such as passphrases or symmetric keys. These keys can be archived and retrieved by agents contacting the new KRA REST interfaces directly.

This capability allows KRA to function as a secure and audited vault for all kinds of secrets. In fact, KRA serves as the secure back-end store for the Vault feature in Red Hat Identity Management.

In addition, KRA's ability to generate and archive asymmetric keys to support server-side key generation for TMS workflows has been extended to allow the generation of symmetric key. This feature has also been exposed to the KRA REST interface.

Support for KRA Transport Key Rotation

Employing transport key rotation in a large enterprise environment with cloned certificate system instances may be impractical as it required shutdowns for the transition. Red Hat Certificate System 9 introduces a KRA transport key rotation feature that allows for seamless transition between CA/KRA subsystem instances using a current and a new transport key. This feature allows KRA transport keys to be periodically rotated for enhanced security by allowing both old and new transport keys to operate during the time of the transition; individual subsystem instances take turns being configured while other clones continue to serve with no downtime.

External Authorization LDAP Server

Red Hat Certificate System 9 introduces an "External Authorization" mechanism to work in conjunction with the directory-based authentication during enrollments. When any of the directory-based authentications is defined, new parameters pertaining to the group evaluation of the users can also be defined. This feature enhances the authentication methods with authorization so that if required, certain profile enrollment can be restricted to users of certain group(s) defined in the external authentication/authorization LDAP server.

Adding SAN to a Server Certificate during Installation

Previously, administrators had no control over the Subject Alternative (SAN) Extension that is used for system SSL certificates. In this release, a new feature has been added to allow the administrators to specify a SAN extension in the **`pkispawn`** configuration.

Common Criteria Evaluation

Red Hat Certificate System 9 has not yet been evaluated for Common Criteria.

The PKI Configuration Has Been Removed from the GUI-based Installation Wizard

Previously, Certificate System provided a web interface for the public key infrastructure (PKI) configuration. Due to unclear support of features associated with the GUI in Firefox, the PKI configuration has been removed from Red Hat Certificate System 9.0. To install and configure PKI instances, use the **pkispawn** utility.

5.4. KNOWN ISSUES

These are known issues in the 9.0 release of Red Hat Certificate System. When available, workarounds are included.

BZ#1041414

Due to a bug, Certificate System sets an incorrect CA profile ID when you install a TPS. To work around the problem, manually set the ***op.enroll.delegateISEtoken.keyGen.encryption.ca.profileId*** parameter in the **/var/lib/pki/*instance_name*/tps/conf/CS.cfg** file to **caTokenUserDelegateAuthKeyEnrollment**:

```
op.enroll.delegateISEtoken.keyGen.encryption.ca.profileId=caTokenUserDel
egateAuthKeyEnrollment
```

BZ#1256901

When certain HSMs are used while **TLS_ECDHE_RSA_*** ciphers are enabled, subsystems experience communication problems. The issue occurs in the following scenarios:

- When a CA has been installed and a second subsystem is being installed and tries to contact the CA as a security domain, thus preventing the installation from succeeding.
- While performing a certificate enrollment on the CA, when archival is required, the CA encounters the same communication problem with the KRA. This scenario can only occur if the offending ciphers were temporarily disabled for the installation.

To work around this problem, keep the **TLS_ECDHE_RSA_*** ciphers turned off if possible. Note that while the Perfect Forward Secrecy provides added security by using the **TLS_ECDHE_RSA_*** ciphers, each SSL session takes about three times longer to establish. Also, the default **TLS_RSA_*** ciphers are adequate for the Certificate System operations.

issue tracked upstream

Red Hat Certificate System 9 provides SCEP enrollment using RSA transport certificates only. If ECC certificates are required to be issued using SCEP, the administrator should set up an RSA system certificate to be used for transporting purposes, instead of the CA signing certificate.

BZ#1202527

If a CUID provided by a client is not properly converted in format, the **tokenType** and **keySet** mapping resolver framework sometimes fails to evaluate properly the mapping filter for the CUID range (**tokenCUID.start / tokenCUID.end**).

BZ#1256984

Currently, the External Registration Recovery does not calculate the size of each key to recover individually and only works properly for 1024-bit keys by default. For example, at attempt to recover a certificate with a 2048-bit private key will fail.

To work around this problem, add the following setting to the **externalRegAddToToken** profile in the **CS.cfg** file:

```
op.enroll.externalRegAddToToken.keyGen.encryption.keySize=2048
```

This configuration will work if all the requiredcertificates to add have keys of the same size.

BZ#1255963

When using the latest TPS applet version that supports **scp01** smart cards, format operations fail on the SafeNet 330 Java (330J) smart card.

Note that the TPS server is currently offered as a Technology Preview and is not yet fully supported under subscription agreements.

BZ#1202526

Token terminations force revocation of all certificates on the token and previously, there was little ability to customize that process. Red Hat Certificate System adds granular control over operations performed on certificates. However, in order to work properly, this feature requires the following list of parameters to be added to the TPS **CS.cfg** file for all token types:

```
op.enroll.tokenType.keyGen.keyType.recovery.terminated.revokeCert

op.enroll.tokenType.keyGen.keyType.recovery.terminated.revokeCert.reason
  op.enroll.tokenType.keyGen.keyType.recovery.terminated.scheme
```

The above parameters ensure that the **terminated** and **keyCompromise** states can be configured to have different revocation reasons.

```
op.enroll.tokenType.keyGen.keyType.recovery.endState.holdRevocationUntil
LastCredential = true/false
```

The above set of new parameters for each token allows to set behavior so that if a certificate is shared by multiple tokens, then that certificate is not revoked until the last token containing that certificate is terminated or lost. If the certificate is finally revoked on the last token, then the status of all the other tokens is set to **revoked** as well.

```
op.enroll.tokenType.keyGen.keyType.recovery.state.revokeExpiredCerts =
true/false
```

The above set of new parameters for each token type allows to set behavior so that expired certificates do not get revoked.

BZ#1255192

In **Certificate Manager - Certificate Profiles**, if you select a Profile instance that is disabled and click **Edit/View** to get the **Certificate Profile Rule Editor** window, changes to the inputs are not applied as they should.

BZ#1253502

When a **caDirUserCert** certificate is issued, the job notification email is not sent when the job notification for issued certificates is enabled.

BZ#1252952

When using a SCP02 token with the **gp211** Coolkey applet, which is currently offered as a technology preview, attempting a re-enroll operation results in a failure near the end of the process.

To work around this problem, perform a format operation before re-enrolling.

BZ#1254804

CRMF key generation request types are no longer supported in Firefox 33, 35 or later. As a consequence, it is no longer possible to perform browser-based enrollments, particularly in the key archival functionality. Note that limited support for simple keygen-based enrollments now exists for the profiles not performing key archival.

To work around this problem, perform enrollments through the **pki** CLI utility. In Red Hat Certificate System 9, the **client-cert-request** command supports both PKCS #10 and CRMF certificate requests. To generate and submit a CRMF certificate request with key archival, first download the transport certificate:

```
# pki cert-find --name "<KRA Transport certificate's subject common
name>"
   # pki cert-show serial_number --output transport.pem
```

Then, submit the certificate request:

```
# pki -c password client-init
   # pki -c password client-cert-request subject_DN --profile caDualCert
--type crmf --transport transport.pem
```

BZ#1257670

When a CA with KRA is installed and an archival attempt through archival-enabled CA profile is made, if the connection between CA and KRA was attempted with the user **pkidbuser** instead of the subsystem user, the archival attempt fails.

To work around this problem, add the **pkidbuser** user to the Trusted Managers group.

BZ#1244965

The synchronous key recovery mechanism has been deprecated in Red Hat Certificate System 9. Red Hat recommends to use asynchronous key recovery instead.

BZ#1250741

When cloning a CA and the master CA having **serialCloneTransferNumber=0** set, the **pkispawn** utility currently does not return a proper error message as it should.

BZ#1255431

An incompatibility with the authentication plug-in interface protocol has resulted in the **UdnPwdDirAuth** plug-in not working properly in Red Hat Certificate System 9.0, so that this plug-in cannot be placed in any profile at this time.

BZ#1250734

When cloning a CA, if the serial number range is less than the value of **serialCloneTransferNumber**, the **pkispawn** utility terminates with an exception rather than returning a proper error message.

BZ#1252621

The **pkispawn** utility uses the following default ports as defined in **/etc/pki/default.cfg**:

```
pki_https_port=8443
    pki_http_port=8080
```

While **pkispawn** allows for highly flexible customizations, any attempt to override the default port values using ports that have been pre-allocated for other uses may cause installation or configuration to fail with error messages similar to the following:

```
pkispawn    : DEBUG    ....... Error Type: Exception
     pkispawn    : DEBUG    ....... Error Message: port 9180 has
invalid selinux context pki_ca_port_t
```

The availability of a given port can be checked by running the following commands:

```
# semanage port -l | grep 9180
      pki_ca_port_t      tcp      829, 9180, 9701, 9443-9447
      # semanage port -l | grep 18443
      (if the port is unused, nothing will be displayed)
```

NOTE

Red Hat Certificate System 9 primarily uses the **http_port_t** SELinux context, even though the default HTTP port 8080 uses **http_cache_port_t**. The following ports and their SELinux contexts were added to the system policy for previous versions of Red Hat Certificate System, and as such, cannot be used for Red Hat Certificate System 9:

```
# semanage port -l | grep pki
        pki_ca_port_t        tcp      829, 9180, 9701, 9443-
9447
        pki_kra_port_t       tcp      10180, 10701, 10443-
10446
        pki_ocsp_port_t      tcp      11180, 11701, 11443-
11446
        pki_ra_port_t        tcp      12888-12889
        pki_tks_port_t       tcp      13180, 13701, 13443-
13446
        pki_tps_port_t       tcp      7888-7889
```

BZ#1246635

The **pki user-cert-add** command provides an option to import the user certificate directly from CA. However, this option does not work properly if the command is executed over SSL port due to incorrect client library initialization. Consequently, the command fails with the following error message:

```
javax.net.ssl.SSLPeerUnverifiedException: peer not authenticated.
```

To work around this problem, download the certificate from CA into a file using the **pki cert-show** command, then upload the certificate from a file using the **pki user-cert-add** command.

BZ#1247410

Currently, the **ocspResponderURL** configuration parameter does not work when it uses a HTTPS secure port. Consequently, trying to enable OCSP checking from the KRA (Key Recovery Authority) subsystem using CA's secure port causes self tests to fail during KRA restart.

To work around this problem, you can safely use an insecure HTTP port because the response is signed and time-stamped.

BZ#1251581

When an enrollment request is submitted using the End-Entity page using the **Manual User Signing & Encryption Certificates Enrollment** profile, the CA will generate both encryption and signing certificates. However, when the request is submitted using the **CRMFPopClient** or **pki** utilities, the CA only generates the encryption certificate.

To work around this problem, the signing certificate can be requested separately.

BZ#1231261

The **pkispawn** utility has an interactive mode to primarily help users deploy the most straightforward configurations, and become familiar with Red Hat Certificate System. Therefore, **pkispawn** does not currently provide an interactive session for HSM, cloning subsystems, sub-CA, and externally signed CA.

As a temporary relief, the user is properly informed during the interactive **pkispawn** session as to which functionality is not yet supported, to prevent any confusion from other related error messages.

BZ#753311

When restarting the CA, SELinux returns AVC denied error messages.

The CA ultimately restarts properly, so these errors can be ignored.

BZ#699456

If an administrator creates a custom log type, any modifications made to the file or to the log file configuration is not recorded in the audit log. This means that the log file is not secure .

BZ#693412

Using the KRA agent's page to search for pending recovery requests does not return the list of pending requests.

Search for the specific recovery request by using the reference number given when the recovery request was submitted. Searching by the reference number successfully returns the recovery request record. From there, the request can be approved by clicking the **Grant** button.

BZ#678320

Resetting the password on a token with an applet upgrade operation does not work properly. Both the password reset operation and the applet upgrade operation fail.

To work around this problem, disable applet upgrade in the PIN reset profile.

BZ#673182

ECC keys are not supported for signing audit logs. Neither the servers nor the **AuditVerify** utility support ECC keys for signed audit log files.

To work around this problem, use RSA keys for signing audit logs.

BZ#664594

After a key recovery request is approved and complete, the recovery request page should display a list of which KRA agents approved the recovery. Instead, the **Recovery Approving Agents** field remains blank.

The recovering page used by agents to approve the request is updated with the list of approving agents. That page can be referenced.

BZ#616532

When attempting to recover keys, if you search for pending requests based on the key identifier and click the **Recover** button, it returns an error that it had a problem processing the request. The form used to submit the search request sends a malformed request, which results in an invalid X.509 certificate error.

To work around this problem, search for the recovery certificate by pasting in the full certificate blob in the search criteria form.

BZ#512029

If the same HSM partition is used to multiple Red Hat Certificate System subsystem instances, the instance names cannot be used more than once, even if the instances are on different hosts. If the user tries to configure a new instance with the same name (including the default options) as an existing instance, then configuration process stops at the key generation step with an error that the certificate subject name already exists.

To work around this problem, when using an HSM, always specify a distinct **pki_*XYZ*_subject_dn** individually.

BZ#226823

An error in the **<Connector>** entry in the **server.xml** file causes the server to start and listen on that connector port, but does not provide any services. This problem occurs if the system is configured to use an HSM, not the internal token, and can be recognized by the following JSS configuration error returned by the Tomcat server:

```
Failed to create jss service: java.lang.SecurityException: Unable to
initialize security library
```

BZ#454559

Attempting to connect to the Online Certificate Status Manager using the **wget** utility or HTTP POST to send OCSP requests times out.

To work around this problem, use the **OCSPClient** utility to send status requests.

APPENDIX A. REVISION HISTORY

Note that revision numbers relate to the edition of this manual, not to version numbers of Red Hat Certificate System.

Revision 9.4-0 **Thu Oct 25 2018** **Marc Muehlfeld**
Red Hat Certificate System 9.4 release of the guide

Revision 9.3-1 **Tue Apr 10 2018** **Marc Muehlfeld**
Red Hat Certificate System 9.3 release of the guide

Revision 9.2-1 **Tue Dec 12 2017** **Petr Bokoč**
Asynchronous update.

Revision 9.2-0 **Tue Aug 01 2017** **Petr Bokoč**
Red Hat Certificate System 9.2 release of the guide

Revision 9.1-1 **Thu Mar 09 2017** **Petr Bokoč**
Asynchronous update

Revision 9.1-0 **Tue Nov 01 2016** **Petr Bokoč**
Red Hat Certificate System 9.1 release of the guide

Revision 9.0-1 **Fri Sep 04 2015** **Tomáš Čapek**
Minor factual update

Revision 9.0-0 **Fri Aug 28 2015** **Tomáš Čapek**
Version for Red Hat Certificate System 9 release