Black•Vault HSM

RedHat Certificate System

Integration Guide

Revision 1.0

9565 Soquel Drive, Aptos, CA 95003

+1 831.688.1021

+1 877.ENGAGE4

https://www.engageblack.com

https://www.engageinc.com

sales@engageblack.com

Disclaimer and Warranty

Engage Black is a business unit of Engage Communication.

Engage Communication makes no representations or warranties with respect to the contents of this document and specifically disclaims any implied warranties of merchantability of fitness for any particular purpose. Information in this document is subject to change without notice and does not represent a commitment on the part of Engage Communication, Inc. Product specifications are subject to change without notice. Engage Communication assumes no responsibility for any inaccuracies in this document or for any obligation to update the information in this document.

Contents

1 Introduction

The BlackVault Hardware Security Module (HSM) integrates with the RedHat Certificate System to enable you to secure operations, root certificates, and keys. The RedHat Certificate System has a NIAP certified release[1] and should be backed by a FIPS certified HSM, like the BlackVault HSM, for complete security. The benefits of using the BlackVault HSM with the RedHat Certificate System include,

- Secure storage of private keys
- Secure execution of cryptographic operations
- FIPS 140-2 level 3 validated hardware

2 Environment

2.1 Software and versions:

This guide was tested using a RedHat virtual machine running in Windows VMWare Workstation 15 Pro.

- Windows 10 Pro 1903, 64-bit
- VMWare Workstation 15 Pro (15.5.1 build-15018445)
- RedHat Enterprise Linux Server 7.6
- RedHat Certificate System 9.4
- BlackVault utilities & firmware: 7.0.33.2.8

3 Installation

3.1 RedHat Enterprise Linux (RHEL)

If FIPS is needed, pass the `fips=1` kernel option to the kernel command-line.[2] After installation of your system, set the preferred OS version using,[3]

```
$ subscription-manager release --set 7.6
```

[1] Current NIAP compliance list link
[2] RedHat FIPS documentation link
[3] RedHat preferred OS documentation link

For client communication, the firewall must be disabled on certain ports. The ports that must be disabled for the Certificate System are: 8080, 8443, 8009, and 8005;[4] we must also disable ports for the Directory Server: 389, 636, and 9830.[5] Make sure the firewall service, `firewalld` is running and then open the ports using,

```
$ systemctl status firewalld

$ systemctl start firewalld

$ systemctl enable firewalld

$ firewall-cmd --permanent --add-port={<PORT 1>/tcp,<PORT 2>/tcp,<OTHER PORTS>}
```

Next, enable the required RedHat repositories,

```
$ subscription-manager register --auto-attach

$ subscription-manager list --available --all

$ subscription-manager attach --pool=<RedHat Certificate System Pool ID>

$ subscription-manager attach --pool=<RedHat Directory Server Pool ID>

$ subscription-manager repos --enable rhel-7-server-hrcmsys-9-rpms

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

Install the required packages,

```
$ yum update

$ yum install redhat-ds openldap-clients redhat-pki
```

Correct `/etc/hosts` to include the correct host name and FQDN. Finally, reboot the system.

3.2 Directory Server `setup-ds.pl`

Now, create a Directory Server (DS) instance. To do this, create a DS configuration file (i.e. ds-setup.conf) and run `setup-ds.pl`,[6]

```
$ setup-ds.pl --file=ds-setup.conf
```

[4] RedHat Certificate System documentation link
[5] RedHat Directory Server documentation link
[6] See A.2 for example DS configuration.

3.3 CA subsystem `pkispawn`

Preparing the system for installation,

- Upgrade BlackVault firmware and install utilities (>= 7.0.33.2.8)

  ```
  $ yum install bvhsm-7.0.33.2.8-1.x86_64.rpm
  ```

- Correct `pkcs.dat` in `/usr/local/BlackVault/Configuration` so that it contains the correct IP.
- Copy the `pkcs.dat` file to another location (i.e. `$HOME/pkcs.dat`).[7]
- Add, `BV_PKCS_PATH=<PATH TO PKCS DAT FILE>`, to `/etc/environment` and as an environment variable in Tomcat (See A.1 for syntax),

  ```
  $ systemctl edit pki-tomcatd@.service
  ```

- If the Black Vault HSM has not been initialized, initialize now.
  - See the Black Vault User Guide for HSM initialization instructions.
- Create a `pkispawn` configuration file.[8]

Clear any keys on the BlackVault and then run,

```
$ sudo pkispawn -f <PKI CONF> -s CA
```

- Make sure to save the `pkispawn` output (example output in appendix, B.1)

3.4 Post-installation

The following steps are provided to get the certificates into Firefox so we can have the correct permissions in the Admin console and a verified SSL connection. To install the Admin certificate,

- Copy the Admin certificate and change the group and owner (path is found in `pkispawn` output from earlier)

  ```
  $ sudo cp <ADMINISTRATOR CERT PATH>/ca_admin_cert.p12 .
  $ sudo chown $USER ca_admin_cert.p12
  $ sudo chgrp $USER ca_admin_cert.p12
  ```

- Alternatively, export the Admin certificate using `pk12util`

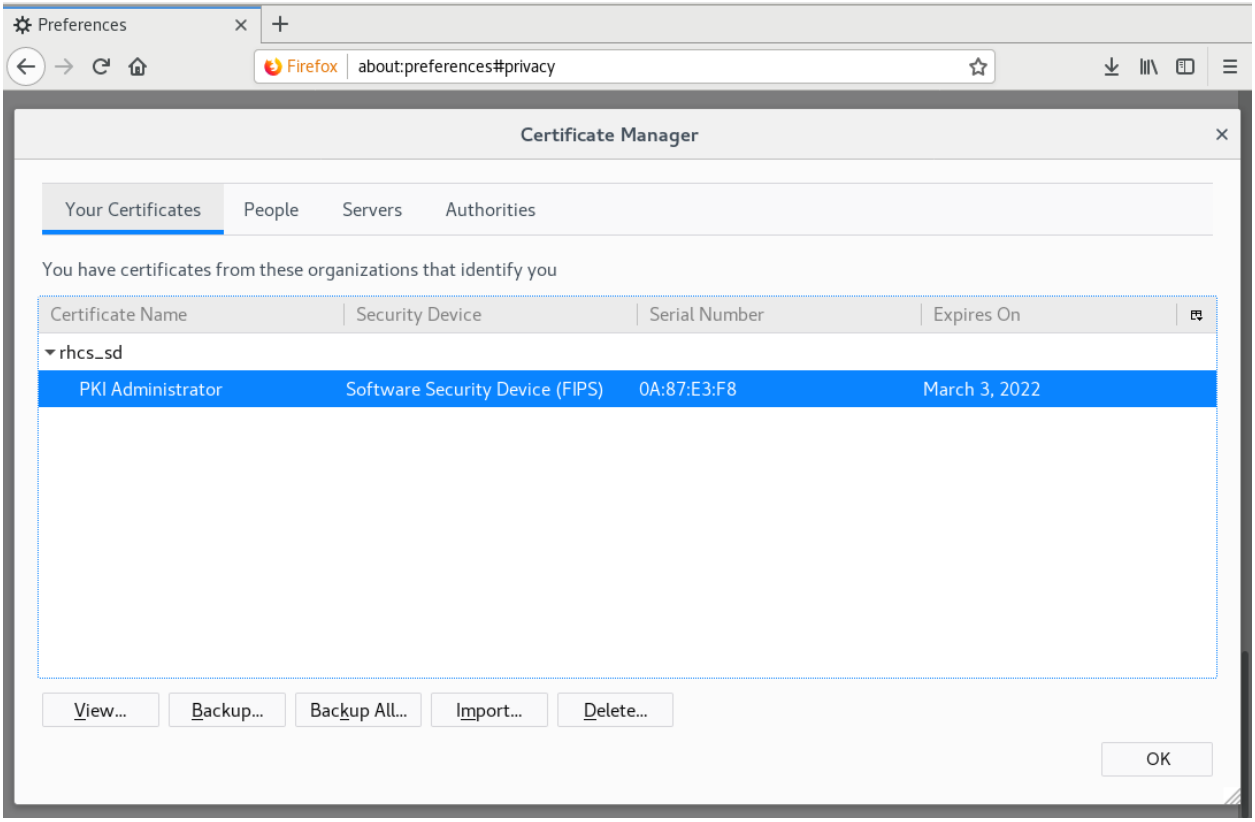
  ```
  $ sudo pk12util -d /root/.dogtag/<PKI INSTANCE>/ca/alias -o ca_admin_cert.p12 -n ‘‘<CERT
    NAME>’’.
  ```

[7] Optional, however, if the utilities are upgraded, the `pkcs.dat` in `/usr/local/BlackVault/Configuration` will be reset.
[8] See A.3 for example `pkispawn` configuration

  - `chown` and `chgrp` is still needed

- Import the bundle into Firefox (*Preferences* → *Privacy & Security* → *View Certificates* → *Your Certificates* → *Import*).

The imported admin certificate should look like the following,



Installing the CA certificate,

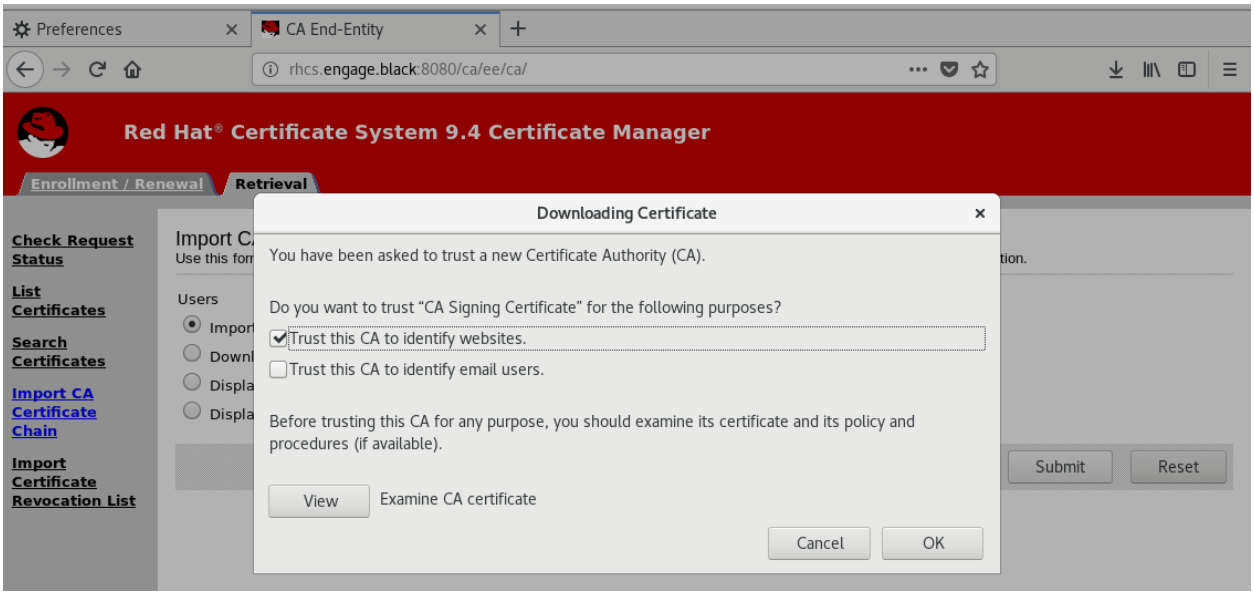
- Run,[9]

```
$ sudo pkidaemon status <PKI INSTANCE>
```

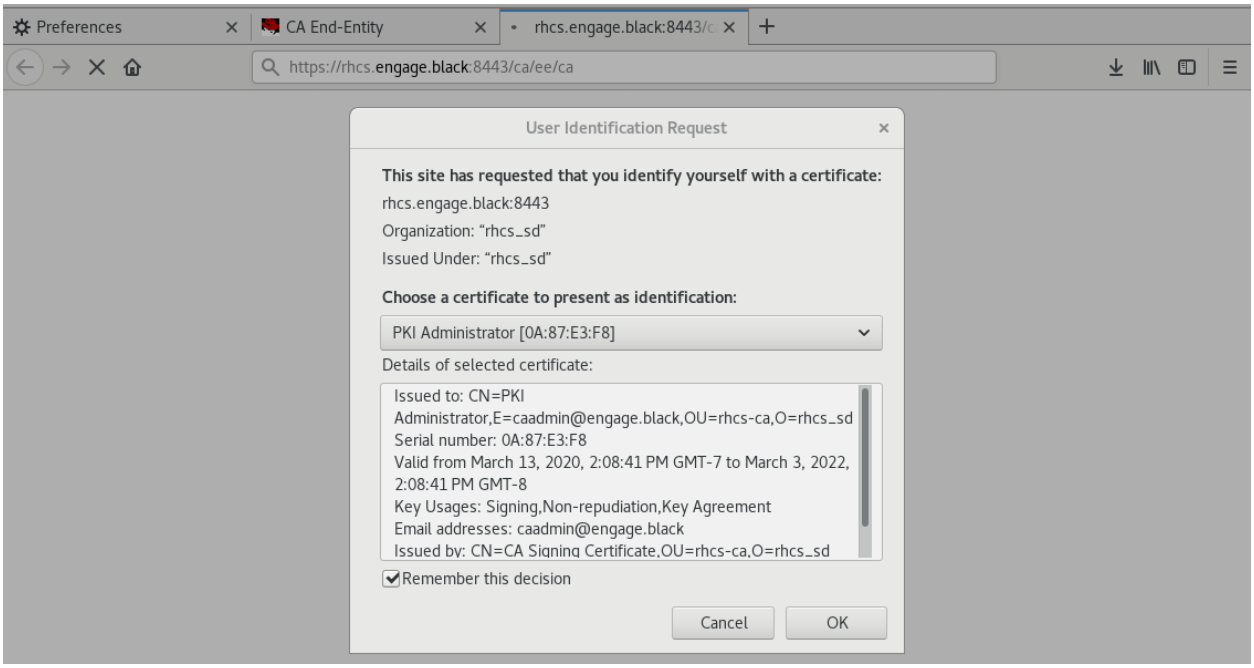
- Follow the *Unsecure URL* and navigate to the *Retrieval* tab
- Select *Import CA Certificate Chain* → *Import the CA certificate chain into your browser*
- Check mark the *Trust this CA to identify websites* option
  - View and verify the certificate
- Press *OK*

[9]The output of this command will be used later as well. Example output in appendix B.2

The CA import window should look like the following,



Note that the first time you open a secure webpage, you will be asked to use the admin certificate. For example,



4 Basic usage

This section is an introduction to some of the basic operations of the RHCS.

4.1 Creating a certificate

Creating a CSR,

- Go back to the `pkidaemon` output and follow the link to the *Secure EE URL*
- Select any of the "Certificate Profiles"
  - For this example, we will use, *Manual User Dual-Use Certificate Enrollment*
    - The only required field is the *UID* field
  - Select *Submit* and write down the *Request ID*
  - Note that if the Certificate Profile uses CRMF, the CSR may fail to be created

Alternatively, you can create a PEM formatted CSR using OpenSSL. For example,

```
$ openssl req -newkey rsa:2048
```

Issuing a certificate,

- Return to the `pkidaemon` output and follow the link to the *Secure Agent URL*
  - In the *List Requests* menu, select *Find*
  - Select the CSR that you issued previously, scroll to the bottom and select *submit*

Checking the issued certificate,

- Return to the *Secure EE* webpage and select the Retrieval tab
- Under *Check Request Status* enter the Request ID and select *submit*
  - The certificate status should now say "complete" and there should be a link to the issued certificate

4.2 OCSP

Here we are using the OCSP responder built into the CA subsystem. Get the issuing certificate and find the X509 extensions[10]. There should be an OCSP link under the *Authority Info Access* extension.

```
$ openssl ocsp -issuer <ISSUING CERT> -cert <ISSUED CERT> -url <OCSP URL> -text -CAfile
  <CA CERTIFICATE CHAIN>
```

- In this case, `<CA CERTIFICATE CHAIN>`, is the same as, `<ISSUING CERT>`

[10] Print PEM formatted certificates using: `openssl x509 -text -in <PEM CERTIFICATE>`

A Example configurations

A.1 Tomcat configuration

```
[Service]
Environment="BV_PKCS_PATH=<PATH TO PKCS DAT FILE>"
```

A.2 `setup-ds.pl` configuration

```
[General]
FullMachineName=rhcs.engage
SuiteSpotUserID=dirsrv
SuiteSpotGroup=dirsrv

[slapd]
ServerIdentifier=rhcs
ServerPort=389
Suffix=dc=red,dc=local
RootDN=cn=Directory Manager
RootDNPwd=Pa55word!
```

A.3 `pkispawn` configuration

```
[DEFAULT]
pki_admin_nickname=caadmin
pki_admin_name=ca-admin
pki_admin_password=Pa55word!
pki_admin_uid=caadmin
pki_client_database_password=Pa55word!
pki_client_pkcs12_password=Pa55word!
pki_ds_secure_connection=false
pki_ds_secure_connection_ca_nickname=DS Certificate
pki_ds_secure_connection_ca_pem_file=/tmp/ds.crt
pki_ds_password=Pa55word!
pki_ds_ldaps_port=636
pki_ds_ldap_port=389
pki_ds_bind_dn=cn=Directory Manager
pki_ds_hostname=rhel-ca-1
pki_admin_key_algorithm=SHA384withEC
pki_admin_key_size=nistp384
pki_admin_key_type=ecc
pki_sslserver_key_algorithm=SHA384withEC
pki_sslserver_key_size=nistp384
pki_sslserver_key_type=ecc
pki_subsystem_key_algorithm=SHA384withEC
pki_subsystem_key_size=nistp384
pki_subsystem_key_type=ecc
```

```
pki_audit_signing_key_algorithm=SHA384withEC
pki_audit_signing_key_size=nistp384
pki_audit_signing_key_type=ecc
pki_audit_signing_signing_algorithm=SHA384withEC
pki_instance_name=rhcs-ca
pki_security_domain_name=rhcs_sd
pki_security_domain_user=sdadmin
pki_security_domain_password=Pa55word!
pki_host=rhcs.engage
pki_hsm_enable=true
pki_hsm_libfile=/usr/lib64/libbvpkcs.so
pki_hsm_modulename=BlackVault
pki_token_name=BlackVaultUser
pki_token_password=Pa55word!
pki_audit_signing_token=BlackVaultUser
pki_sslserver_token=BlackVaultUser
pki_subsystem_token=BlackVaultUser

[CA]
pki_random_serial_numbers_enable=true
pki_ca_signing_key_algorithm=SHA384withEC
pki_ca_signing_key_size=nistp384
pki_ca_signing_key_type=ecc
pki_ca_signing_signing_algorithm=SHA384withEC
pki_ocsp_signing_key_algorithm=SHA384withEC
pki_ocsp_signing_key_size=nistp384
pki_ocsp_signing_key_type=ecc
pki_ocsp_signing_signing_algorithm=SHA384withEC
pki_source_admincert_profile=/usr/share/pki/ca/conf/eccAdminCert.profile
pki_source_subsystemcert_profile=/usr/share/pki/ca/conf/eccSubsystemCert.profile
pki_source_servercert_profile=/usr/share/pki/ca/conf/eccServerCert.profile
pki_client_database_purge=false
pki_ca_signing_nickname=RHCS-signing
pki_ocsp_signing_nickname=RHCS-ocsp
pki_audit_signing_nickname=RHCS-audit
pki_sslserver_nickname=RHCS-webserver
pki_subsystem_nickname=RHCS-subsystem
pki_ca_signing_token=BlackVaultUser
pki_ocsp_signing_token=BlackVaultUser
```

B Example output

B.1 `pkispawn` output

```
$ sudo pkispawn -f pki-config -s CA
Log file: /var/log/pki/pki-ca-spawn.20200312153359.log
Loading deployment configuration from pki-config.
Installing CA into /var/lib/pki/rhcs-ca.
Storing deployment configuration into /etc/sysconfig/pki/tomcat/rhcs-ca/ca/deployment.cfg.
```

```
Module "BlackVault" added to database.
Notice: Trust flag u is set automatically if the private key is present.
certutil: could not change trust on certificate: SEC_ERROR_TOKEN_NOT_LOGGED_IN: The
operation failed because the PKCS#11 token is not logged in.
pki.nssdb   : WARNING  certutil returned non-zero exit code (bug #1393668)

    ==========================================================================
                               INSTALLATION SUMMARY
    ==========================================================================

      Administrator's username:             caadmin
      Administrator's PKCS #12 file:
            /root/.dogtag/rhcs-ca/ca_admin_cert.p12

      Administrator's certificate nickname:
            PKI Administrator for engage
      Administrator's certificate database:
            /root/.dogtag/rhcs-ca/ca/alias

      This CA subsystem of the 'rhcs-ca' instance
      has FIPS mode enabled on this operating system.

      REMINDER:  Don't forget to update the appropriate FIPS
                 algorithms in server.xml in the 'rhcs-ca' instance.

      To check the status of the subsystem:
            systemctl status pki-tomcatd@rhcs-ca.service

      To restart the subsystem:
            systemctl restart pki-tomcatd@rhcs-ca.service

      The URL for the subsystem is:
            https://rhcs.engage:8443/ca

      PKI instances will be enabled upon system boot

    ==========================================================================
```

B.2 pkidaemon output

```
$ sudo pkidaemon status rhcs-ca
Status for rhcs-ca: rhcs-ca is running ..

    [CA Status Definitions]
    Unsecure URL        = http://rhcs.engage:8080/ca/ee/ca
    Secure Agent URL    = https://rhcs.engage:8443/ca/agent/ca
    Secure EE URL       = https://rhcs.engage:8443/ca/ee/ca
    Secure Admin URL    = https://rhcs.engage:8443/ca/services
    PKI Console Command = pkiconsole https://rhcs.engage:8443/ca
    Tomcat Port         = 8005 (for shutdown)
```

```
    [CA Configuration Definitions]
    PKI Instance Name:   rhcs-ca

    PKI Subsystem Type:  Root CA (Security Domain)

    Registered PKI Security Domain Information:
    ============================================================================
    Name:  rhcs_sd
    URL:   https://rhcs.engage:8443
    ============================================================================
```

B.3 `setup-ds.pl` output (truncated)

```
$ sudo setup-ds.pl --file=ds-setup.config

==========================================================================
This program will set up the 389 Directory Server.

It is recommended that you have "root" privilege to set up the software.
Tips for using this  program:
  - Press "Enter" to choose the default and go to the next screen
  - Type "Control-B" or the word "back" then "Enter" to go back to the previous screen
  - Type "Control-C" to cancel the setup program
```

...Truncated output...

```
Your new DS instance 'rhcs' was successfully created.
Exiting . . .
Log file is '/tmp/setup8ee0cu.log'
```