Axway Validation Authority

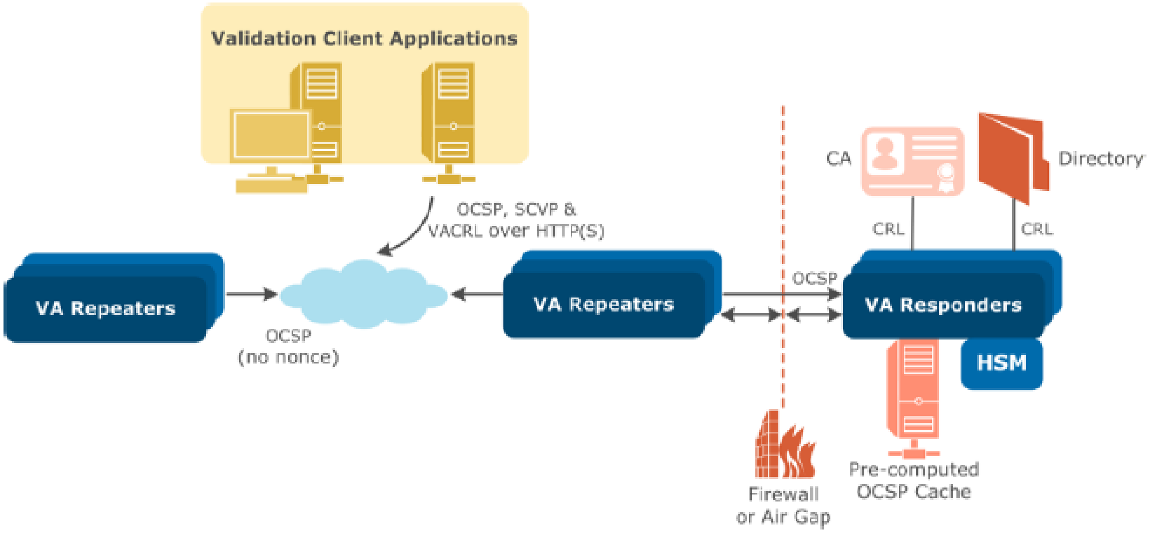
nShield® HSM Integration Guide

2024-04-03

Table of Contents

Chapter 1. Introduction

The Axway Validation Authority (VA) Server is an Online Certificate Status Protocol (OCSP) server for distribution of certificate revocation information for certificates issued by any certification authority (CA). The VA Server provides integrity and validity for online transactions by validating, in real-time, digital certificates issued by a CA. The Entrust nShield Hardware Security Module (HSM) integrates with the Axway VA responder server through the nShield PKCS #11 cryptographic API to securely generate and store the OCSP response signing keys. The following image shows such an integration:



1.1. Requirements

The Axway VA installation requires either Microsoft Windows Server or Red Hat Enterprise Linux as the base operating system. Conceptually, a CentOS platform will work the same way. Obtain the installation package for Windows or Linux from Axway Support.

Reference the *Axway Validation Authority Administrators Guide* for product specific requirements.

Before starting this integration, review:

- The documentation for the nShield Connect HSM.
- The documentation and configuration process for Axway VA.

Before using nShield products:

- When creating a Security World, identify custodians of the administrator card set (ACS).
- Obtain enough blank smart cards to create the ACS.
- Define the Security World parameters. For details of the security implications of the choices, see the *nShield Security Manual*.

> Entrust recommends that you allow only unprivileged connections unless you are performing administrative tasks.

1.2. Licensing

Configuring Axway VA requires importing a license text file into the Axway VA administration web UI. Obtain the license file to configure Axway VA.

1.3. Product configurations

Entrust tested nShield HSM integration with Axway VA in the following configurations:

Product	Version
Axway Validataion Authority	v5.2 BN31823 UP202206
Windows	Windows Server 2022
Red Hat Enterprise Linux	release 8.8
HSM Hardware	Connect XC and nShield 5C

1.4. Supported features

Entrust tested nShield HSM integration with the following features:

Softcard	Module	OCS	nSaaS
Yes	Yes	Yes	Not Tested

1.5. Supported nShield hardware and software versions

Entrust tested with the following nShield hardware and software versions:

nShield Hardware	nShield HSM Firmware	Security World Software	FIPS
Connect XC	12.50.11	13.3.2	140 Level 2
Connect XC	12.72.1	13.3.2	140 Level 3
nShield 5c	13.2.2	13.3.2	140 Level 3 (FIPS Pending)

1.6. Supported nShield functionality

Feature	Support
Key Generation	Yes
Key Management	Yes
Key Import	No
Key Recovery	Yes
FIPS 140 Level 3 mode support for Connect XC	Yes
FIPS 140 Level 3 mode support for nShield 5c	Yes (FIPS Pending)
Common Criteria mode support	N/A
1-of-N Operator Card Set	Yes
K-of-N Operator Card Set	Yes
Softcards	Yes
Module-only keys	Yes
Load Sharing	Yes
Failover	Yes

Chapter 2. Procedures

An overview of the integration procedures is as follows:

1. Install and configure the nShield HSM.
2. Select the key protection method.
3. Initial VA server setup and configuration.
4. Perform basic integration tests.

2.1. Install and configure the nShield HSM

This guide does not cover the basic installation and configuration of the nShield HSM or the nShield Security World client software. For instructions, see the *Installation Guide* for your HSM.

> When creating the Operator Card Set (OCS) or Softcards for the Security World, the passphrases must match the VA server password that will be set in the initial VA server configuration process. The VA server password must be *at least* 8 characters in length and include one uppercase letter, one lowercase letter, one number, and one special character. The same is true of the OCS/Softcard passphrase.

Add the following lines to the `cknfastrc` configuration file of the Security World. The file is in the `%NFAST_HOME%` directory, which is `C:\Program Files\nCipher\nfast` on Windows and `/opt/nfast` on Linux.

- Module protection:

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=none
CKNFAST_FAKE_ACCELERATOR_LOGIN=1
CKNFAST_LOADSHARING=1
```

- Softcard protection:

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=none
CKNFAST_LOADSHARING=1
CKNFAST_NO_ACCELERATOR_SLOTS=1
```

  For FIPS 140 Level 3 you need an additional variable:

```
CKNFAST_TOKEN_HASH=27f10e7fa846a13cba324856e75902b26ee998bb
```

You can get the hash value of the Softcard being used by running the following command:

```
$ nfkminfo -s

SoftCard summary - 1 softcards:
Operator logical token hash               name
27f10e7fa846a13cba324856e75902b26ee998bb  axwaysoftcard
```

- OCS protection with a K/N quorum where K=1:

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=none
CKNFAST_LOADSHARING=1
CKNFAST_NO_ACCELERATOR_SLOTS=1
```

- OCS protection with a K/N quorum where K>1 (assuming that you have created the `nfast-nfkm-tokensfile` file):

```
CKNFAST_OVERRIDE_SECURITY_ASSURANCES=none
CKNFAST_LOADSHARING=1
CKNFAST_NO_ACCELERATOR_SLOTS=1
NFAST_NFKM_TOKENSFILE=C:\ProgramData\nCipher\nfast-nfkm-tokensfile
```

On Windows, `C:\ProgramData\nCipher\nfast-nfkm-tokensfile` is an example location for creating the `preload` file.

On Linux, an example location is `NFAST_NFKM_TOKENSFILE=/opt/nfast/kmdata/nfast-nfkm-tokensfile`.

2.2. Select the key protection method

If more than one key protection mechanism is available (for example: OCS and Softcard; OCS and module protection; or two Softcards), modify the `C:\ProgramData\nCipher\Key Management Data\local` directory on Windows to contain the *minimum* required key protection mechanism files. On Linux, this directory is `/opt/nfast/kmdata/local`. When using a key protection method, make sure that only the files pertaining to that specific method are present in the `local` directory.

- If using module protection:
  1. Remove all OCS and Softcard files from `local`.
  2. For FIPS 140 Level 3:
     a. Make sure that an ACS card is inserted into an available slot of the HSM.

   b. The ACS card will provide FIPS-authorization in place of the OCS card for this application. This will not work since the associated OCS card file must be removed.

- If using Softcard protection:
  1. For FIPS 140 Level 2:
     a. Remove all OCS files from `local`.
  2. For FIPS 140 Level 3:
     a. You must have an OCS card created and inserted to provide FIPS-authentication.
     b. Do not remove the OCS card files from `local`.
     c. Ensure the cknfastrc file has `CKNFAST_TOKEN_HASH` set to the softcard hash value.
- If using OCS protection:
  1. Remove all Softcard files from `local`.
  2. Insert the OCS quorum to provide FIPS-authorization.

For more information about the environment variables used in `cknfastrc`, see:

- The *nShield Cryptographic API Guide*.
- The PKCS #11 library environment variables section of the *User Guide* for the HSM.

2.3. Initial VA server setup and configuration

Axway VA consists of:

- A VA Host Server acting as either a Repeater or Responder operating on Windows Server or Red Hat Enterprise Linux.
- A web-based Administration Server that provides centralized management of the validation processing components.

Client applications can query the VA Server utilizing open standard protocols including the Online Certificate Status Protocol (OCSP) or the Server-based Certificate Validation Protocol (SCVP), allowing clients to delegate the entire certificate validation operation including path construction and intermediate CA validation to the VA Server.

This section describes how to set up and configure a Responder. Before setting up the Axway VA Server (Responder), complete the following:

- Obtain a Responder product license from Axway and make it available on the host platform.
- Obtain a root certificate from a CA and make it available in on the host platform.
- Obtain an associated Certificate Revocation List (CRL) for the CA and make it available on the host platform.

To install the Axway VA (Responder) server:

1. Before installing Axway VA, install the nShield Security World software. This is especially important for Linux installation. Make sure the user running the the VA server is part of the `nfast` group on Linux.
2. See the *Axway VA Administrators Guide* for steps on installing the server on Windows and Linux.
3. After installing Axway VA, browse to the Axway VA Web Administration and log in using the credentials specified during installation.

   If using OCS protected keys with a K/N quorum where K>1, use `preload` to load the OCS K/N quorum. Enter the OCS passphrase when prompted.

   The `nfast-nfkm-tokensfile` file must exist already and the `NFAST_NFKM_TOKENSFILE` variable must point to it. See Install and configure the nShield HSM.

   ```
   % preload -m<module-number> -c <ocs-cardset-name> -f <path-to-nfast-nfkm-tokensfile> pause
   ```

   If you restart the server, you will have to run the `preload` command again.

4. Select the **Enter License** tab.
5. Paste the license certificate from Axway into the license text box and select **Submit License**.
6. On the **Axway Validation Authority License** page, confirm the license details and select **Next Step**.
7. On the **Import Configuration File** page, select **Skip**.
8. On the **Install Custom Extensions** page, select **NO**, then select **Submit**.
9. On the **Server Password** page, enter and confirm the new VA server password.

   The password must match the OCS or Softcard passphrase and is required to have at least:

   - 8 characters in length
   - one alphabetic character

  - one digit
  - one special character
  - one upper case character
  - one lower case character

> ℹ If the server password already matches the OCS or Softcard passphrase and meets these minimum requirements, skip this step. Select **Create/Import Key Pair** to go to the next step.

10. Select **Submit** when finished.
11. On the **SUCCESS!** page, select **Next Step**.
12. On the **Key Type Selection** page, under **Mandatory**, select **Default OCSP/SCVP Response Signing** and select **Submit Key Type**.
13. On the **Key Generation/Import Mechanism: Default OCSP/SCVP Response Signing** page, select **Hardware Key Generation/Import using Entrust**.

    This option is only available on the Linux distribution of the VA Server. If that is not an option, select the following:

    - **Hardware Key Generation/Import on custom PKCS11 provider**
    - **Vendor**: Entrust
    - **PKCS#11 Library Path**:
      - Windows: `C:\Program Files\nCipher\nfast\toolkits\pkcs11\cknfast.dll`
      - Linux: `/opt/nfast/toolkits/pkcs11/libcknfast.so`
14. Select **Submit Key Generation Technique**.
15. Select **Generate new private key**.
16. Select **Submit Key Generation or Import**.
17. On the **Generate Hardware key and Certificate: Default OCSP/SCVP Response Signing** page:
    a. Under **PKCS11 Token Information**:
      - For **User PIN**, enter the VA server password, which is also the OCS or Softcard passphrase. If using module protection, enter the VA server password.
      - For **Friendly Key Name**, enter a name to identify the key.
      - For **Key Expiration in days**, enter **0** for non-expiring keys or enter another number for the key lifetime.

        - For **Slot ID**, select **Auto Sense** or the decimal number representing the PKCS11 slot.

          For all key protection mechanisms (loadsharing enabled in `cknfastrc`), this decimal number will begin with **7614066**.

        - For **Key Algorithm**, select **RSA**.
        - For **Key Length**, select **2048**.
        - For **Hash Algorithm**, select **SHA256** (or any other algorithm except **SHA1**).

    b. Under **Certificate Information**:

        - For **Type**, select **Self-signed Certificate**.

          Alternatively, select **Certificate Request** if you want to have an external CA sign the certificate.

        - For **Certificate Validity (days)**, enter the certificate validity period (default = 365 days).
        - Select **Simple DN Entry** and enter the certificate parameters (country, city, and so on).

    c. Under **Certificate Options**, select **Key Use: Sign/Signature Verification**.

       Leave all other options clear.

18. Review the PKCS11 token and certificate parameters and select **Submit** when finished.
19. If the key and certificate were successfully generated, a **SUCCESS!** message appears, followed by **Self signed certificate for Default OCSP/SCVP Response Signing was created successfully. Click here to view certificate information**.

    If there is an error and PKCS#11 debugging is enabled in `cknfastrc`, check the contents of the debug file at the path specified to troubleshoot key generation.

    To generate a private key for OCSP responses, the following files are created:

    - `OCSP_RESP_SIGN_<DateTimeStamp>_GMT.crt` (Self-signed OCSP Responder certificate)
    - `OCSP_RESP_SIGN_<DateTimeStamp>_GMT.req` (PKCS#10 request)
    - `vacs<DateTimeStamp>`

These files are located as follows:

- Windows: `C:\ProgramData\Axway\VA\entserv\.vacsbak`
- Linux: `/var/lib/va/entserv/.vacsbak`

20. Open a command prompt and run the following command to verify the key is listed under the key protection method intended:

```
% nfkminfo -l
Keys protected by cardsets:
  key_pkcs11_ucf581378f4a81d3ba312fcd19859247049bf18161-53ec9f29251f88e948217499c4736daf16027193 `<Friendly
Key Name> RSAPrv'
```

21. Back on the VA server web UI, select **Click here to view certificate information**.

    A dialog appears and displays the OCSP response signing certificate that was generated.

22. Select **Next Step**.

    The generated certificate is used to digitally sign OCSP and SCVP responses from the Validation Authority server. OCSP requests are essentially queries to the VA server asking for the status of a certificate (good, revoked, or unknown) for a specific Certificate Authority.

    The private key used to sign the responses from the VA server is stored and protected within the HSM.

    The next step is to configure CA certificates for which the VA server will provide OCSP responses.

23. On the **Manage Certificate Store** page:

    a. Under **Mandatory Stores**, select **CA Certificates [OCSP Protocol]**.

    b. Select **Submit**.

24. On the **Certificate Import Method** page:

    a. Select **Local File**.

    b. Select **Submit Certificate Import Method**.

25. On the **Import Certificate File** page:

    a. Select **Choose File** and select the root CA certificate for which you want the VA server to provide OCSP responses.

    b. Select **Submit Certificate File**.

26. On the **Select Certificates** page:

   a. Confirm the details for the imported CA certificate.
   b. Select **Submit Certificates** if the displayed information is correct.

27. On the **Configure VA Certificate Store** page:
   a. Make sure the root CA certificate is listed.
   b. Select **Add** to add more root CA certificates (repeat above process) or select **Next Step** to continue.
28. On the **Configure CRL Imports** page:
   a. Select the appropriate method for retrieving a CRL associated with the CA.
   b. Select **Add CRL Source**.

   ℹ Integration testing was done using an HTTPS CRL source. The next few steps reflect this selection.

29. On the **Configure CRL Import (HTTP/FTP/FILE)** page:

   - Under **CRL Source**:
     - Set **Protocol** to the appropriate protocol for retrieving the CRL source information. For this guide, **HTTPS** was used.
     - **CRL Source URL**: Enter the URL for the CRL source. See the *Axway Validation Authority Administrators Guide* for the appropriate syntax for the selected protocol.
     - **CRL Encoding**: Select the appropriate encoding from the dropdown.
     - Configure other parameters as needed.
   - Leave the **Import Schedule** and **Connection Settings** to their defaults and select **Add Source**.

30. Return to the **Configure CRL Imports** page and repeat the steps above to add additional CRL sources.
31. Select **Next Step** to continue.
32. On the **Configure Server URLs** page, enter the following:

   On Windows, the server hostname and port are automatically added. On Linux:

   - For **Hostname**, enter the hostname or IP address set during installation.
   - For **Port**, select **8080** or another port above 1024.
   - Select **Add**.

   ℹ This is the URL and port the VA server will listen on for

> OCSP requests/queries.

  - Select **Submit**.

33. On the **SUCCESS!** page, select **Next Step**.
34. On the **VA Responder Server Configuration Parameters** page:
    a. Configure as appropriate for your environment. See the *Axway Validation Authority Administrators Guide*.
    b. Select **Submit Configuration Parameters**.
35. On the **SUCCESS!** page, select **Next Step**.
36. On the **Server Start/Stop** page:
    a. Enter the VA server password, which is also the OCS or Softcard passphrase.
    b. Select **Start Server**.
37. The server status changes from **OFF** to **ON**. The VA responder server is now operational.
38. Confirm the Responder is importing CRLs from the configured CRL URL address by accessing the server log to view publisher-specific events. You can view CRLs published on the Responder by navigating to **CRLs** > **CRLs & OCSP Databases**.

2.4. Perform basic integration tests

The following sections will test the Axway VA nShield HSM integration.

2.4.1. Verify OCSP response signing key

To verify the OCSP signing key was generated on the HSM, run the following commands. Replace `<pkcs11-key-hash>` with the hash at the end of the `key_pkcs11_<pkcs11-key-hash>` file that is generated in the `local` directory. For example:

```
% nfkminfo -l
Keys protected by cardsets:
  key_pkcs11_<pkcs11-key-hash> `OCSKeyOCSPCert RSAPrv'

% nfkmverify -v -m<module-number> pkcs11 <pkcs11-key-hash>
** [Security world] **
Ciphersuite: DLf3072s256mAEScSP800131Ar1
...
    ---

** [Application key pkcs11 <pkcs11-key-hash>] **
```

```
[Named `OCSKeyOCSPCert RSAPrv']
Useable by HOST applications
Cardset protected: 1/2 PERSISTENT [0s `axwayva_ocs']
Cardset hash f581378f4a81d3ba312fcd19859247049bf18161
(Currently in Module #1 Slot #2: Card #2)
...
    Verification successful, confirm details above.  1 key verified.
```

2.4.2. Verify OCSP signing certificate

To verify the OCSP signing certificate that is presented to clients, use either the `vatest` tool provided by Axway or the `curl` and `openssl` tools if they are on your system.

- To use Axway's `vatest` tool:
  - For Windows:

    ```
    % C:\Program Files\axway\va\tools\vatest getconfig -url http://127.0.0.1:80
    ```

  - For Linux:

    On Linux Red Hat 8, the `vatest` tool fails to run with the following error:

    ```
    /opt/axway/va/tools/vatest: error while loading shared libraries: libxerces-c-3.2.so: cannot open
    shared object file: No such file or directory
    ```

    Just use the `curl` and `openssl` method below.

- To use `curl` and `openssl`:

  ```
  % curl "http://127.0.0.1:80/getvaconfig?mirroring" | openssl x509 -out ocspcerts.pem
  ```

Running either command generates a certificate file `ocspcerts.pem` in the directory from which the command was run.

Open the `ocspcerts.pem` file to see the OCSP signing certificate in Base64 format.

2.4.3. Test OCSP server functionality

To test the VA server's OCSP response capability to requests on the status of various certificates:

1. Open a command prompt.
2. Use the `openssl OCSP` client to make a request to the VA server for the status of

a certificate. For example:

```
% openssl ocsp -text -host 127.0.0.1:80 -issuer "<full-path-to-root-CA-cert>" -VAfile
"C:\ProgramData\Axway\VA\entserv\.vacsbak\OCSP_RESP_SIGN_*_GMT.crt" -serial <cert-serial-number>
```

In this example:

- Replace `<full-path-to-root-CA-cert>` with the path to the root CA certificate.
- Replace `<cert-serial-number>` with the serial number of the certificate whose status you want to check.
- On Linux:
  - For the port for the `host`, use **8080**.
  - For the path for the `VAfile`, use `/var/lib/va/entserv/.vacsbak/`.

An example response for a valid certificate that is *not* on the CRL:

```
OCSP Request Data:
    Version: 1 (0x0)
    Requestor List:
        Certificate ID:
          Hash Algorithm: sha1
          Issuer Name Hash: 6F07DF4E8868D5BF4B09F7C6EB8EEDA505EE03E7
          Issuer Key Hash: 6C8A94A277B180721D817A16AAF2DCCE66EE45C0
          Serial Number: 00
    Request Extensions:
        OCSP Nonce:
            04101896F2C786AA2AEA9A8489CFD987D601
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: C = us, CN = interops.com
    Produced At: Jul  7 16:03:25 2023 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: 6F07DF4E8868D5BF4B09F7C6EB8EEDA505EE03E7
      Issuer Key Hash: 6C8A94A277B180721D817A16AAF2DCCE66EE45C0
      Serial Number: 00
    Cert Status: good
    This Update: Jun 21 15:10:41 2023 GMT
    Next Update: Jul  7 22:03:25 2023 GMT

    Response Extensions:
        OCSP Nonce:
            04101896F2C786AA2AEA9A8489CFD987D601
    Signature Algorithm: sha256WithRSAEncryption
         09:9f:aa:43:a4:b2:24:33:05:2d:1e:33:7d:01:a9:4d:db:d4:
         42:e2:58:0f:f6:16:58:d4:e4:f9:7d:17:f3:5d:c0:b7:60:d7:
         44:4a:4b:64:93:7f:de:9a:b8:e9:eb:2f:e6:e0:8d:7a:e7:6f:
         d8:91:19:f3:ed:1a:07:cb:88:e8:b4:07:d5:5f:b5:14:61:65:
         17:3a:63:95:58:75:66:a2:fb:7e:c6:97:23:76:28:61:c2:b7:
         24:cf:e1:69:32:4c:1e:71:e1:cd:59:4d:8c:53:22:19:44:be:
         07:c2:b4:1c:e8:9e:25:ee:29:22:4a:f5:ec:10:a3:16:87:6d:
         90:45:cf:2d:fe:5e:e9:75:c3:e4:66:db:d7:6e:59:4f:0c:72:
         4f:7e:5a:c1:79:a3:7c:61:80:76:92:6d:dc:ac:ca:7d:ef:97:
```

```
         46:a7:ec:0f:37:ed:e0:23:7a:b2:e8:e2:4e:29:aa:fb:57:a2:
         11:de:8d:9f:50:5b:46:8d:68:63:50:38:40:99:00:a4:ae:7a:
         f2:79:54:a3:0b:31:5a:6a:6e:cf:0b:85:55:43:85:ff:9a:14:
         b9:b1:16:f0:26:7e:c1:61:63:30:e0:af:ea:64:87:a4:a3:f7:
         c9:02:f8:b8:9f:f0:f2:3b:dc:f5:d8:c1:10:19:2e:27:99:6a:
         ec:e3:0c:3f
Response verify OK
0x0: good
        This Update: Jun 21 15:10:41 2023 GMT
        Next Update: Jul  7 22:03:25 2023 GMT
```

Chapter 3. Additional resources and related products

3.1. nShield Connect

3.2. nShield as a Service

3.3. Entrust digital security solutions

3.4. nShield product documentation