Encryption Consulting

MyCodeSigner

ENCRYPTION
CONSULTING
www.encryptionconsulting.com

nCipher Integration Guide

Version 1.0

Date: MAY 14, 2020

N FINITY

STRATEGIC TECHNOLOGY
PARTNER PROGRAM

Contents

Introduction

The Encryption Consulting MyCodeSigner and nCipher nShield HSM integration provides customers the confidence that keys use to sign their code are stored securely.

Encryption Consulting MyCodeSigner provides a secure and flexible solution to your code-signing needs for signing Windows, Linux, Macintosh, Docker and Android/iOS apps. The framework can be extended to protect any other code or document as requested.

1. Download Windows SDK containing SignTool.
2. Import, run and login to MyCodeSigner virtual machine.
3. Create a build server

Integration configurations

The integration between the nShield HSM and the Encryption Consulting MyCodeSigner has been tested for the following combinations:

Example: Add a row in the table for each combination that the nFinity Partner tested:

Operating system	Encryption Consulting MyCodeSigner	nCipher nShield HSM	nShield firmware version	nShield Security World version	Certification support
Windows 2016 Server	Version number	Connect XC	12.50.11	12.60.3	FIPS 140-2
Linux	Version number	Connect XC	12.50.11	12.60.3	FIPS 140-2

Supported cryptographic algorithms

If relevant, for example because it is a selling point, list whether the cryptographic algorithms supported in the nCipher products are supported in the integrated product.

Supported nCipher features

If relevant, for example because not all features are supported in the integrated product or because the support of a particular feature is a selling point, include per-feature support. Example: if an nCipher API or Option Pack, Softcards, CodeSafe SEE, etc. are supported in a particular configuration.

Requirements and prerequisites

Describe licensing prerequisites for: nFinity Partner Product, nCipher product(s), any third-party products such as operating systems. Describe any prerequisites to installing and configuring: nFinity Partner Product with nCipher product(s). Example: A certain operating system patch is required for one of the products to run; administrator privileges or internet connectivity is required to perform one of the installation tasks. Examples can be found in guides published at https://www.ncipher.com/resources/integration-guides.

Related documents and support

Add links to documents on: nFinity Partner Product; nCipher hardware, software. Examples:
** Installation Guide and User Guide for the nShield HSM.*
** User Guide for the nFinity Partner Product.*

Add a link to the nFinity Partner support pages: www.encryptionconsulting.com/contact-us/

Security World Configuration - Windows

- Run command prompt as administrator.
- In command prompt go to setup directory; install the nCipher setup in command prompt (setup.msi).
- Once installation completes, add nCipher setup path in the environment variable
- To test access, use the following command:

  ```
  > anonkneti  --port  9044  [Assigned HSM IP Address]
  ```

- The command should return two numbers, a Serial Number (ESN) and KNETI HASH.

  ```
  [ESN] [KNETI HASH]
  ```

- Copy the world and module files from the table above into the Security World Client's **kmdata/local** folder.
  Windows -C:\ProgramData\nCipher\Key Management Data\local
- Rename the module file by copying the ESN from the output of the anonkneti and append the ESN following an underscore to the name of the module file. Ex. module_090E-03E0-D947
- To complete enrolment, use the following command:

  ```
  > nethsmenroll--port 9044 [Assigned HSM IP Address]
  ```

- To test HSM connectivity, use the following diagnostic command(s):

  ```
  > enquiry
  ```

  Note: enquiry will confirm that the module is connected and provide a list of available nShield features.
  Or

  ```
  > nfkminfo
  ```

  Note: nkfminfo will display information about the Security World.
- To test benchmarking and verify the consistency of the Security World, use the following command(s):

  ```
  > perfcheck -m1 signing:219
  ```

  Note: perfcheck performs a test of the module.
  Or

```
> nfkmcheck
```

Note: nfkmcheck will check the consistency of the Security World data.

Security World Configuration - LINUX

- Completely uninstall and remove gpg, use the following command

```
> sudo apt-get remove --auto-remove gnupg
```

- Copy the iso file SecWorld_Lin64-12.60.3.iso
- Open a terminal on the file path
- Run the below command to mount the iso file.

```
> sudo mount -o loop -t iso9660 SecWorld_Lin64-12.60.3.iso /mnt
```

Note: Output:/dev/loop0 is write-protected, mounting read-only

- Go to the folder called /mnt
- Give permission using command

```
> sudo chmod -R 777 opt
```

- Run the below commands to untar all the compress files.

```
> tar xvf /mnt/linux/amd64/ctd.tar.gz
> tar xvf /mnt/linux/amd64/ctls.tar.gz
> tar xvf /mnt/linux/amd64/devref.tar.gz
> tar xvf /mnt/linux/amd64/hwsp.tar.gz
> tar xvf /mnt/linux/amd64/javasp.tar.gz
> tar xvf /mnt/linux/amd64/jd.tar.gz
> tar xvf /mnt/linux/amd64/ncsnmp.tar.gz
> tar xvf /mnt/linux/amd64/raserv.tar.gz
```

- Go to the folder /opt/nfast/sbin.Install the client by running command.

```
> sudo ./install
```

- Check the connectivity by running the below command in /opt/nfast/bin folder

```
> ./anonkneti --port 9044 213.121.187.217
```

Note: Output: 090E-03E0-D947 e3ccceb081699312254fccb70607dadc8cf220e5

- Run the below commands to untar all the compress file
- Rename the module file (received from client) with the **ESN** number. The world file should look like module_090E-03E0-D947.
- Copy the world file and module to /opt/nfast/kmdata/local folder
- Do configuration by the following command

```
> ./nethsmenroll --port 9044 [ Assigned HSM IP Address]
```

Note: Output: Remote module returned ESN: 090E-03E0-D 947HKNETI: e3ccceb081699312254fccb70607dadc8cf220e5

```
Is the above correct? (yes/no): yes
OK configuring hard server's nethsm imports.
```

- Check the output of the command.

```
>./perfcheck -m1 signing:219
```

Note: Output: perfcheck: using modules 1 perfcheck: decided on a total maxq of 151

- Copy the folder gnupg, hwcrhk and openssl(given) to the /opt/nfast/toolkits.
- Install rpm using command.

```
> sudo apt install rpm
```

- Edit the gnupg-1.4.20.hwcrhk.diff file in /opt/nfast/toolkits/gnupg folder. Find the line "char fnbuf[100], lbuf[1024]".Change lbuf[1024] to lbuf[10000].
- Download gnupg-1.4.23 from the official gnupg website for linux os.
- Go to the gnupg-1.4.23 folder.
- Do patch by the following command.

```
> patch -p1 < /opt/nfast/toolkits/gnupg/gnupg-1.4.20.hwcrhk.diff
```

- Do configuration for gpg by the following command.

```
> ./configure CPPFLAGS="-I/opt/nfast/toolkits/gnupg"
LDFLAGS="-ldl"
```

- Install gpg using below commands.

```
> make
> sudo make install
```

- Add the below lines in /etc/profile file at the end.

```
export NFAST_BIN=/opt/nfast/bin
export PATH=$NFAST_BIN:$PATH
export LD_LIBRARY_PATH=/opt/nfast/toolkits/hwcrhk
```

- Add the below variables in /etc/environment file

```
GNUPG_HWCRHK="/opt/nfast/toolkits/hwcrhk/libnfhwcrhk.so"
GNUPG_RSA_19XX_IMPORT="1"
```

- Restart the system
- Go to the root directory using command.

```
> ~
> nano .rmmacros
```

Create a file called .rmmacros file with the below content.

```
%_gpg_name SampleKey
%__gpg /usr/local/bin/gpg
%_gpg_path /root/.gnupg/
%_signature gpg
```

- Copy the pem2keytxt (given in email) file to /opt/nfast/bin folder.
- Give root level permission by the command.

```
> sudo chmod -R 777 pem2keytxt
```

- Install curl by using the command.

```
> sudo apt install curl
```

- Install nodeJs by using command.

```
> curl -sL https://deb.nodesource.com/setup_12.x | sudo -E bash -
> sudo apt-get install -y nodejs
```

- Copy the rpm_signing_backend.zip file. Unzip it. Then run npm install in the rpm_signing_backend folder terminal.
- Run the command to check whether server is running.

```
>node server.js
```

- To install pm2 by using command.

```
> npm install pm2 –g
```

- To run the node server using pm2 by the command.

```
>pm2 start server.js
```

nCipherKM JCA/JCE CSP – JAR Signer

The nCipherKM JCA/JCE CSP (Cryptographic Service Provider) allows Java applications and services to access the secure cryptographic operations and key management provided by nCipher hardware. This provider is used with the standard JCE (Java Cryptographic Extension) programming interface.

The following version of JAVA is required for use with nShield Security World Client:

- Java8 (or Java 1.8x).

Add the Java executable to your system path.

Install nCipherKM JCA/JCE CSP:

- The nShield Java package includes nShield Java jars and Keysafe.

Always use the latest version of Java supported by nCipher which is compatible with your application requirements. If you are maintaining older Java versions for legacy reasons, and need compatibility with current nCipher software, contact nCipher support.

Depending on the base OS being used, installation of Java may have specific dependencies and require other pre-installed packages.

Download jdk 8 from the Oracle website https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html as per your operating system.

Installing the nCipherKM JCA/JCE CSP

- To install the nCipher Security World Client reference the nCipher User Guide – nShield Connect – 12.60 - Windows.pdf.
- Copy the nCipherKM.jar file from "C:\Program Files\nCipher\nfast\java\classes" to "C:\Program Files\Java\jre1.8.0_231\lib\ext".
- Add "C:\Program Files\Java\jdk1.8.0_231\bin" in PATH variable (Environment).
- Download JCE policy from website "https://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html".
- Once Unlimited JCE Policy Downloaded. Extract the files local_policy.jar and US_export_policy.jar.
- Copy the files local_policy.jar and US_export_policy.jar to C:\Program Files\Java\jre1.8.0_231\lib\security.
- Copy the files local_policy.jar and US_export_policy.jar to C:\Program Files\Java\jdk1.8.0_231\jre\lib\security.
- Edit the C:\Program Files\Java\jdk1.8.0_231\jre\lib\security\java.security file using notepad. Add thesecurity.provider.1=com.ncipher.provider.km.nCipherKM in the security provider list.

  Note:

  Output:

  ```
  security.provider.1=com.ncipher.provider.km.nCipherKM
  security.provider.2=sun.security.provider.Sun
  security.provider.3=sun.security.rsa.SunRsaSign
  security.provider.4=sun.security.ec.SunEC
  security.provider.5=com.sun.net.ssl.internal.ssl.Provider
  security.provider.6=com.sun.crypto.provider.SunJCE
  security.provider.7=sun.security.jgss.SunProvider
  security.provider.8=com.sun.security.sasl.Provider
  security.provider.9=org.jcp.xml.dsig.internal.dom.XMLDSigRI
  security.provider.10=sun.security.smartcardio.SunPCSC
  security.provider.11=sun.security.mscapi.SunMSCAPI
  ```

- Edit the C:\Program Files\Java\jre1.8.0_231\lib\security\java.security file using notepad.

  Add the security.provider.1=com.ncipher.provider.km.nCipherKM in the security provider list.

  Note:

  Output:

```
security.provider.1=com.ncipher.provider.km.nCipherKM
security.provider.2=sun.security.provider.Sun
security.provider.3=sun.security.rsa.SunRsaSign
security.provider.4=sun.security.ec.SunEC
security.provider.5=com.sun.net.ssl.internal.ssl.Provider
security.provider.6=com.sun.crypto.provider.SunJCE
security.provider.7=sun.security.jgss.SunProvider
security.provider.8=com.sun.security.sasl.Provider
security.provider.9=org.jcp.xml.dsig.internal.dom.XMLDSigRI
security.provider.10=sun.security.smartcardio.SunPCSC
security.provider.11=sun.security.mscapi.SunMSCAPI
```

Testing the nCipherKM JCA/JCE CSP installation

After installation, test the nCipherKM JCA/JCE CSP is working properly by running the following command.

- For Java 8:

```
> java com.ncipher.provider.InstallationTest
```

Note:

Output:

```
Installed providers:

1: nCipherKM

2: SUN

3: SunRsaSign

4: SunJSSE

5: SunJCE

6: SunJGSS

7: SunSASL

Unlimited strength jurisdiction files are installed.

The nCipher provider is correctly installed.

nCipher JCE services:

Alg.Alias.Cipher.1.2.840.113549.1.1.1
Alg.Alias.Cipher.1.2.840.113549.3.4

Alg.Alias.Cipher.AES

Alg.Alias.Cipher.DES3

...............................

security.provider.1=com.ncipher.provider.km.nCipherKM
```

Contact Information

To request technical support for **Encryption Consulting** reference:

130 N Preston Rd,
Prosper, TX 75078, USA
Direct Phone: +1- 469-815-4136
info@encryptionconsulting.com
https://www.encryptionconsulting.com/contact-us/

To request technical support for **nCipher nShield** Products reference:

https://www.ncipher.com/services/support/contact-support for technical support contact details.