nCIPHER

www.ncipher.com

MIRANTIS CERTIFIED

Docker Enterprise from Mirantis

Integration Guide

Version: 1.3

Date: Friday, September 25, 2020

Copyright 2020 nCipher Security Limited. All rights reserved.

Copyright in this document is the property of nCipher Security Limited. It is not to be reproduced, modified, adapted, published, translated in any material form (including storage in any medium by electronic means whether or not transiently or incidentally) in whole or in part nor disclosed to any third party without the prior written permission of nCipher Security Limited neither shall it be used otherwise than for the purpose for which it is supplied.

Words and logos marked with ® or ™ are trademarks of nCipher Security Limited or its affiliates in the EU and other countries.

Mac and OS X are trademarks of Apple Inc., registered in the U.S. and other countries.

Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Information in this document is subject to change without notice.

nCipher Security Limited makes no warranty of any kind with regard to this information, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. nCipher Security Limited shall not be liable for errors contained herein or for incidental or consequential damages concerned with the furnishing, performance or use of this material.

Where translations have been made in this document English is the canonical language.

Contents

1 Introduction

This guide describes the steps to integrate the nShield Container Option Pack (nSCOP) with the Docker Enterprise from Mirantis container platform. The nSCOP provides application developers, within a container-based environment, the ability to access the cryptographic functionality of an nShield Hardware Security Module (HSM).

The nSCOP is installed on top of your existing Security World Software installation, allowing you to continue using your existing Security World and keys.

This integration has been tested with Docker Enterprise from Mirantis.

1.1 Product configurations

This integration comprised the following components:

Product	Version
nSCOP	1.1
Docker Enterprise	3.1
Docker Engine - Enterprise	19.03.8
OS	CentOS 7.8
Security World	12.60.3
nShield HSM	Connect XC/+
VMware	ESXi 6.7.0 on a Dell PowerEdge R740

1.2 Requirements

1.2.1 Before starting the integration process

Familiarize yourself with:

- nShield Container Option Pack User Guide
- nShield Connect User Guide
- Docker Enterprise from Mirantis documentation and setup process

2 Procedures

2.1 nShield software prerequisites

The nShield software versions used in this integration:

nSCOP	Security World Client	nShield Connect Image	Firmware
1.1	12.60.3	12.60.2	12.50.11, 12.50.8

The nShield Container Option Pack requires nShield Security World Software and Docker Engine - Enterprise to be installed prior to the use of the nShield Container Option Pack scripts. For Instructions on how to setup an nShield Connect, a Remote File System (RFS) for the nShield Connect, a client computer, and installation instructions for nShield Security World, see the *nShield Connect User Guide*.

These instructions assume nSCOP is installed. They also refer to scripts used in nSCOP to create and configure the hardserver and application containers. For more information, see the *nShield Container Option Pack User Guide*.

To access and use cryptographic keys from within a Security World you will need to load or create a Security World on the nShield Connect and map the key management data folder (**kmdata**) from your container host machine into the running application containers. For more information, see the *nShield Connect User Guide*.

2.2 Using a Docker repository

In this integration, it is assumed the repository is shared on Docker Hub. The following steps push the image to the Docker Hub repository.

The repository needs to be set to private. Software is under export control and cannot be distributed through third parties.

1. Log in to your Docker Hub account:

   ```
   > docker login docker.io -u <docker username>
   ```

   You are prompted to enter your password.

2. Tag the Docker image. This tag name is used to push the image to Docker Hub and to run the container.

```
> docker images
> docker tag <IMAGE ID> <accountname>/<repo name>:<tagname>
> docker images
```

3. Push the Docker image to Docker Hub:

```
> docker push <accountname>/<repo name>:<tagname>
```

2.3 The hardserver container

The hardserver container, **nshield-hwsp**, controls communication between the configured nShield Connect(s) and application containers. Only one hardserver container is required per deployment, regardless of the number of nShield Connects or application containers.

2.3.1 Create nshield-hwsp

1. Create the directory to be used to mount the Security World ISO:

```
> mkdir /opt/nfast/SecWorld-12.60.3
```

2. Mount the Security World ISO:

```
> sudo mount -o loop SecWorld_Lin64-12.60.3.iso /opt/nfast/SecWorld-12.60.3
mount: /dev/loop0 is write-protected, mounting read-only
```

3. Create the new hardserver Docker image using **make-nshield-hwsp**. The image ID is retrieved when you push the image to Docker Hub.

```
> ./make-nshield-hwsp /opt/nfast/SecWorld-12.60.3
[...]
Successfully tagged nshield-hwsp:12.60.3
```

The default base image for nShield hardserver containers is **ubuntu:bionic**. For more information regarding support for other base images, see the *nShield Container Option Pack User Guide*. The default tag reflects the version of nShield Security World software that the container was built from. If you want to use a different base image, or specify a different tag, use the **--from** and **-tag** options. See **make-nshield-hwsp --help** for more information.

By default the **nfast** user and group in the container will match those on the host machine; you should create them if they do not exist on the host, or if this is a bad fit for deployment the **--uid** and **-gid** options should be used to set them instead.

2.3.2 Configure nshield-hwsp

Create and edit the hardserver container configuration using **make-nshield-hws-config**:

1. Create the hardserver container configuration:

```
> ./make-nshield-hwsp-config --output /opt/nfast/config <nShield HSM IP Address>
```

2. Edit the config file to confirm the nShield Connect HSM information:

```
> cat /opt/nfast/config

syntax-version=1

[nethsm_imports]
local_module=1
remote_esn=1111-2222-3333
remote_ip=<nShield HSM IP Address>
remote_port=9004
keyhash=000102030405060708090a0b0c0d0e0f10111213
privileged=0
```

Key hash values are retrieved from remote HSMs without any trust; the generated configuration file should be compared against values recorded from the front panel, or another trusted path. You can alternatively create the config file based on the template above by simply entering the **remote_esn**, **remote_ip**, **remote_port**, and **keyhash** values in the **nethsm_imports** section.

3. Give **nfast** user permissions to the config file:

```
> sudo chown -R nfast:nfast /opt/nfast/config
```

2.3.3 Run nshield-hwsp

1. Create a volume to be mounted:

```
> sudo mkdir -m755 -p /opt/nfast/sockets.hwsp
```

2. Give nfast user permissions to **sockets.hwsp**:

```
> sudo chown -R nfast:nfast /opt/nfast/sockets.hwsp
```

3. Tag the Docker image. This tag name is used to push the image to Docker Hub and to run the

container. The **<IMAGE ID>** corresponds to the hardserver image created earlier (see "Create nshield-hwsp" on page 6). To see a list of all images use the command **docker image**.

```
> docker images
> docker tag <IMAGE ID> <accountname>/<repo name>:<tagname>
> docker images
```

4. Push the Docker image to Docker Hub:

The repository needs to be set to private. Software is under export control and cannot be distributed through third parties.

```
> docker push <accountname>/<repo name>:<tagname>
```

5. Run the hardserver container:

```
> docker run -d -v /opt/nfast/config:/opt/nfast/kmdata/config/config:ro -v /opt/nfast/sockets.hwsp:/opt/nfast/sockets
<accountname>/<repo name>:<tagname>

<Container id is displayed>
```

Note that you can use the following command to check if the connection to the HSM is configured:

```
> docker logs <Container id>
```

2.4 Application Container

An nShield application container is a container with the nShield Security World software installed.

2.4.1 Create the Application Container

As detailed in the nShield Container Option Pack User Guide you can create an application based container using **make-nshield-application**. The application base container is created in a host Linux system. The only required argument is the path to a mounted Security World ISO.

1. Create the application container:

```
> ./make-nshield-application SecWorld-12.60.3
[...]
Successfully tagged nshield-ubi7:12.60.3
```

2.4.2 Run the Application Container using Docker Hub

1. Tag the Docker image. This tag name is used to push the image to Docker Hub and to run the container. The **<IMAGE ID>** corresponds to the Docker image, with the nShield support software installed, that you tagged previously (see "Run nshield-hwsp" on page 7).

```
> docker images
> docker tag <IMAGE ID> <accountname>/<repo name>:<tagname2>
> docker images
```

2. Push the Docker image to Docker Hub:

The repository needs to be set to private. Software is under export control and cannot be distributed through third parties.

```
> docker push <accountname>/<repo name>:<tagname2>
```

3. Assuming you have your Security World and module file in **/opt/nfast/kmdata/local** and the Docker sockets volume created for the hardserver container from above, you can then launch the container from the image specified by **tagname2**:

```
> docker run -it -v /opt/nfast/kmdata:/opt/nfast/kmdata:ro -v /opt/nfast/sockets.hwsp:/opt/nfast/sockets
<accountname>/<repo name>:<tagname2>
```

Please note that this command will create a Read Only volume. Use the **rw** option to create a writable volume for operations that need to store data in the **kmdata** directory, for example **generatekey**.

4. From within the running container, **enquiry** can be run to test connection to the nShield Connect HSM:

```
[root@5fb084cafd8f /]# /opt/nfast/bin/enquiry

Server:
enquiry reply flags none
enquiry reply level Six
serial number 530E-02E0-D947
mode operational
version 12.60.3
speed index 478
rec. queue 110..208
```

```
level one flags Hardware HasTokens SupportsCommandState
[...]
```

5. Additional commands such as **nfkminfo** and **generatekey** demonstrate that the Security World information in **kmdata** is available to the containers as a result of the mount volume command:

```
[root@5fb084cafd8f /]# /opt/nfast/bin/nfkminfo

World
generation 2
state 0x37a70008 Initialised Usable Recovery PINRecovery !ExistingClient RTC NVRAM FTO
AlwaysUseStrongPrimes !DisablePKCS1Paddin g !PpStrengthCheck !AuditLogging SEEDebug
n_modules 1
hknso d20c12ed89c7a1c47b8f49cf285126307c8a6a99
hkm e73aa10db7e8a680526d7d04ce1154631bc73fd8 (type Rijndael)
hkmwk c2be99fe1c77f1b75d48e2fd2df8dffc0c969bcb
hkre e3c0fe535a1c167503b75d3e75d03e7dd3845d66
hkra 154b7b13901818f1db5a1ea5a0c821ca282ebf91
hkmc a5b3dfdb403389753fc32cc751d368f7417c2e93
hkp 51f23a982ccdf2c7fd8b0d1f6cc1703b6a11e3a6
hkrtc 2ddbd544efd10caecbdc4eff3417c09b1f2c8fac
hknv c6728c8488fdcada2e82f6ee93a5a98e3c87520c
hkdsee e98d831884b171d656fbff92e143eee93b1aefb9
hkfto c653ea05f1fc81ce71756bd8221423d1858c0f49
hkmnull 0100000000000000000000000000000000000000

[...]



[root@5fb084cafd8f /]# /opt/nfast/bin/generatekey -b -m1 simple type=rsa size=2048 ident=testrsa
key generation parameters:
operation Operation to perform generate
application Application simple
verify Verify security of key yes
type Key type rsa
size Key size 2048
pubexp Public exponent for RSA key (hex)
ident Key identifier testrsa
plainname Key name
nvram Blob in NVRAM (needs ACS) no
Key successfully generated.
Path to key: /opt/nfast/kmdata/local/key_simple_testrsa
```

Contact Us

Web site:	https://www.ncipher.com
Support:	https://help.ncipher.com
Email Support:	support@ncipher.com
Online documentation:	Available from the Support site listed above.

You can also contact our Support teams by telephone, using the following numbers:

Europe, Middle East, and Africa

United Kingdom:	+44 1223 622444 One Station Square Cambridge CB1 2GA UK

Americas

Toll Free:	+1 833 425 1990
Fort Lauderdale:	+1 954 953 5229 Sawgrass Commerce Center – A Suite 130, 13800 NW 14 Street Sunrise FL 33323 USA

Asia Pacific

Australia:	+61 8 9126 9070 World Trade Centre Northbank Wharf Siddeley St Melbourne VIC 3005 Australia
Japan:	+81 50 3196 4994
Hong Kong:	+852 3008 3188 31/F, Hysan Place 500 Hennessy Road Causeway Bay Hong Kong

www.ncipher.com

About nCipher Security

nCipher Security, an Entrust Datacard company, is a leader in the general-purpose hardware security module (HSM) market, empowering world-leading organizations by delivering trust, integrity and control to their business critical information and applications. Today's fast-moving digital environment enhances customer satisfaction, gives competitive advantage and improves operational efficiency – it also multiplies the security risks. Our cryptographic solutions secure emerging technologies such as cloud, IoT, blockchain, and digital payments and help meet new compliance mandates. We do this using our same proven technology that global organizations depend on today to protect against threats to their sensitive data, network communications and enterprise infrastructure. We deliver trust for your business critical applications, ensure the integrity of your data and put you in complete control – today, tomorrow, always.
www.ncipher.com

Search: nCipher Security

TRUST. INTEGRITY. CONTROL