nShield Post-Quantum Software Development Kit

PQSDK v1.1.3 Release Notes

04 July 2024

Table of Contents

1. Introduction

These release notes apply to version v1.1.3 of the nShield Post-Quantum Software Development Kit (PQSDK). They contain information specific to this release such as new features, defect fixes, and known issues.

The Release Notes may be updated with issues that have become known after this release has been made available. Please check https://nshieldsupport.entrust.com/hc/en-us/sections/360001115837-Release-Notes for the latest version of this document.

Access to the Support Portal is available to customers under maintenance. Please contact Entrust nShield Technical Support at nshield.support@entrust.com to request an account.

1.1. Versions of these Release Notes

Revision	Date	Description
1.1	2024-07-04	Location of C example files added.
1.0	2024-07-02	Version for v1.1.3.

2. Purpose of this release

v1.1.3 only supports nShield XC devices and Linux. It introduces the following changes:

- Algorithms are described using strings instead of an integer enumeration. This is more consistent with the way `liboqs` works and should be easier for users.
- Leighton-Micali Hash Based Signature (LMS) support has been introduced. This is intended to conform to RFC 8554 but has not yet been externally validated.
- A Python program for creating raw user data has been introduced. This does the same job the Java `UserDataTool` does, but can also create an NVRAM delegation certificate that allows the SEE machine to store LMS keys in NVRAM.
- A dyanmic shared object (DSO) has been introduced. This allows C and C++ code to access the features of PQSDK without Java.

3. Features of nShield Post-Quantum Software Development Kit v1.1.3

3.1. LMS

LMS key pairs are effectively a collection of one-time signature keys. It is important that no one-time signature key ever be reused. To ensure this, PQSDK requries all LMS keys to be stored in the NVRAM of a single nShield HSM. This has two important consequences:

- LMS keys will eventually run out of signatures.
- The SEE machine must be allowed to create NVRAM files. Therefore, a delegation certificate must be created and embedded in the raw user data file. This is done using the new Python user data tool, which will prompt for the `nv` quorum of the administrator card set (ACS).

The tool is found in `/opt/nfast/pqsdk/src/pqsdk/scripts/userdatatool.py`.

```
$ /opt/nfast/python/bin/python /opt/nfast/pqsdk/src/pqsdk/scripts/userdatatool.py -h
Usage: userdatatool.py [options]

Options:
  --version             show program's version number and exit
  -h, --help            show this help message and exit
  -I IDENT, --seeinteg=IDENT
                        SEE integrity key for signing [default=signer]
  -f FILE, --file=FILE  Output or input filename [default: <stdout> or
                        <stdin>]
  -p, --print           Read input data and print details; this overrides the
                        default create userdata behavior.

Create the raw binary userdata on stdout or to a file. Or input data and print
details.
```

3.2. Algorithm strings

Rather than providing an integer to identify the algorithm to be used, PQSDK now requires a string.

The example code in Java has been updated to demonstrate how this works.

Note that the string `LMS` is used to generate LMS keys, but there are additional `lmotsTypeCode` and `lmsTypeCode` parameters that control the features of the generated LMS key. Attempting to generate LMS keys will fail with `AccessDenied` if an old user data file without the NVRAM delegation certificate is used.

Note that in this release, it is not possible to import LMS keys.

3.3. Dynamic Shared Object

In addition to the Java interface, it is now possible to use PQSDK directly from C code. You can find the C examples in `\opt\nfast\pqsdk\src\pqsdk\source\dynlib-test\pqsdk-test.c`.

The `install_pqsdk_see_xc` script builds a shared library in the `/opt/nfast/pqsdk/dynlib` directory.

The following header file should be used by programs which link to the shared library: `/opt/nfast/pqsdk/src/pqsdk/source/dynlib/pqsdk.h`. This header describes the interface, which is analogous to the Java interface.

All functionality is available through both `HostCommands.java` and `pqsdk.h` with the shared object.

In addition, there is a sample C program that can be linked against the shared object to exercise most of the features of PQSDK from the command line. This program can be used to list `pqsdk` keys, generate key pairs, create signatures and more. Most of the direct use of PQSDK is done in the main routine. The rest is infrastructure for parsing command line options and reading keys to import.

4. Compatibility

4.1. Supported hardware

This release is targeted at deployments with any combination of the following nShield HSMs:

- nShield Solo XC (Base, Mid, High)
- nShield Connect XC (Base, Mid, High, Serial Console)

4.2. Supported operating systems

This release has been tested for compatibility with the following operating systems:

- Red Hat Enterprise Linux 8 x64
- Additional mainstream x64 based Linux distributions other than the one listed above may be compatible, however Entrust cannot guarantee this compatibility.

4.3. Supported Security World Versions

The following tables show the supported configuration of firmware and software installations.

4.3.1. nShield Solo

HSM Type	Firmware Version	Security World Version	CodeSafe Version
nShield Solo XC	v12.72.1	v12.70.4	v12.70.4
nShield Solo XC	v12.72.1	v13.4.3	v13.4.3

4.3.2. nShield Connect

HSM Type	Image Version	Firmware Version	Security World Version	CodeSafe Version
nShield Connect XC	v12.80.5	v12.72.1	v12.70.4	v12.70.4
nShield Connect XC	v12.80.5	v12.72.1	v13.4.3	v13.4.3

Other combinations of Firmware, Security World software, and CodeSafe software may work but are not supported at this time.

5. Required licenses

Please use the Feature Enable Tool (FET) to set and view enabled features.

5.1. nShield XC

The Unrestricted SEE (`SEE Activation (EU+10)`) feature must be enabled on your HSM. Restricted SEE (`SEE Activation (Restricted)`) is not currently supported.