Time Stamp Option Pack

# TSOP v7.20.0 Administrator Guide

10 April 2024

© Entrust Corporation. All rights reserved.

# Table of Contents


1. Introduction . . . 1
2. Product overview . . . 3
    2.1. Roles and responsibilities . . . 3
        2.1.1. Security Officer . . . 3
        2.1.2. Network Manager . . . 3
    2.2. Features . . . 3
        2.2.1. nShield PCIe module . . . 3
        2.2.2. Public Key Infrastructure (PKI) technology . . . 3
        2.2.3. Certificates for signing and authentication . . . 4
        2.2.4. Support for leap second events . . . 4
        2.2.5. Compliance with IETF standards . . . 4
        2.2.6. Time-stamp protocol support . . . 4
    2.3. Getting an auditing service provider . . . 5
    2.4. TCP/IP and UDP requirements . . . 5
    2.5. Understanding Security Worlds . . . 6
    2.6. Security . . . 6
    2.7. Administrator and Operator Card Sets . . . 6
    2.8. FIPS 140-2 compliance . . . 7
    2.9. FIPS 140-2 level 3 compliance . . . 7
    2.10. Using Operator Card Sets to protect keys . . . 8
        2.10.1. Using card sets for extra security . . . 8
        2.10.2. Using pass phrases . . . 8
        2.10.3. Using persistent Operator Card Sets . . . 8
        2.10.4. Resilience . . . 9
        2.10.5. Backup and recovery . . . 9
        2.10.6. Replacing the ACS . . . 9
        2.10.7. Risk management . . . 10
3. Prerequisites . . . 11
    3.1. Assumptions . . . 11
    3.2. Supported Platforms . . . 11
    3.3. TCP/IP and UDP port access . . . 11
    3.4. TSOP installation prerequisites . . . 11
4. Installing the Time Stamp Option Pack . . . 13
    4.1. Enabling features . . . 13
    4.2. Configuring the TSS and creating a Security World . . . 14
    4.3. Accessing the TSS Web Interface . . . 14
5. Creating and managing Security Worlds . . . 16

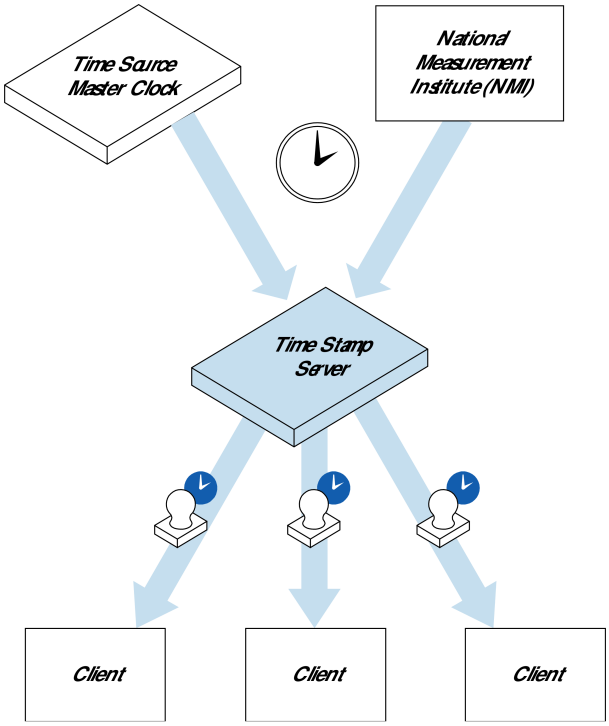
5.1. Creating a Security World . . . 16
5.2. Replacing a Security World . . . 16
5.2.1. Operator cards . . . 17
5.3. Joining an existing Security World . . . 17
5.4. Security World: SEE delegation . . . 17
5.5. Viewing the Security World status . . . 17
6. Configuring the TSS on the network . . . 18
6.1. Getting a static IP address . . . 18
6.2. Configuring the firewall . . . 18
6.2.1. Adding a TLS certificate . . . 18
7. Configuring the TSS . . . 20
7.1. Time Stamping Authority (TSA) keys . . . 20
7.1.1. About multiple TSAs . . . 20
7.1.2. How multiple TSAs work . . . 20
7.2. Adding the CA certificates of an upper clock . . . 21
7.3. Adding an upper clock . . . 22
7.4. Creating a new TSA . . . 22
7.5. Configuring a TSA . . . 23
7.6. Initiating and fulfilling TSA certificate requests . . . 25
7.6.1. Creating Operator Card Sets . . . 25
7.6.2. Initiating a TSA certificate request . . . 27
7.6.3. Importing the TSA certificate chain . . . 29
7.6.4. Fulfilling a TSA certificate request . . . 30
7.7. Loading a TSA . . . 31
7.8. Registering a TSA . . . 32
7.9. Checking the operational status . . . 32
8. Administering and using the TSS . . . 35
8.1. Restoring a TSA key . . . 35
8.2. Removing an upper clock . . . 36
8.3. Enabling or disabling the clock . . . 36
8.4. Viewing card set lists . . . 37
8.5. Viewing or changing card sets . . . 37
8.6. Viewing TSA certificate information . . . 37
8.7. Viewing Time Attribute Certificate (TAC) information . . . 38
8.8. Viewing time-stamps issued . . . 39
8.9. Viewing the uptime . . . 40
8.10. Viewing or resetting Administrator and Board logs . . . 40
8.11. Recording TACs and time-stamps . . . 41
8.12. Viewing user login statistics . . . 41

8.13. Viewing the log archive . . . 42
8.14. Adding a user . . . 42
8.15. Modifying or deleting users . . . 43
8.16. Modifying user information . . . 44
8.17. Restarting the service . . . 44
9. Upgrading an existing deployment . . . 45
9.1. Upgrading TSOP . . . 45
9.2. Upgrading Security World Software . . . 45
9.3. Upgrading Firmware . . . 45
10. TSS FAQs and solutions . . . 47
10.1. Digital certificates . . . 47
10.2. Standards . . . 48
10.3. Synchronization . . . 48
10.4. TSS . . . 48
11. Local Audit / NTP service . . . 50
12. Time Stamp Tokens (TST) TAC encoding and binding . . . 51
12.1. CertificateChoices1 with ESSCertID/ESSCertIDv2 (compatibility mode) . . . 51
12.2. CertificateChoices2 with ESSCertID/ESSCertIDv2 (RFC3369 & 3852) . . . 51
12.3. Signer Attribute (RFC3126 & ETSI) . . . 51
13. TST options . . . 52
13.1. Use ESSCertIDv2 (RFC5816) . . . 52
13.2. Exclude TAC from certificate list . . . 52
13.3. Use rsaEncryption OID Fix . . . 52
14. Error messages and alerts . . . 54
15. Security Guidance . . . 59

# 1. Introduction

The Entrust nShield Time Stamp Server (TSS) provides secure and auditable time signing for electronic business transactions and documents. The TSS serves as a reliable source for time signing, and enables you to:

- Provide authoritative proof of when an event has occurred
- Ensure that time-stamps are secure, authentic, and auditable.




TSS trust model

The Universal Co-ordinated Time (UTC) is the world standard for time. The TSS obtains UTC from one of the following sources:

- Time Source Master Clock (TSMC): TSMC serves as an intermediate node for the acqui sition and distribution of Coordinated Universal Time (UTC). or more information, see the *Time Source Master Clock Administrator Guide*.
- National Measurement Institute (NMI): An NMI acts as a source of UTC for its country. An NMI supplies time to a hierarchy of lower clocks such as the TSS, which make UTC available to applications.

The TSMC or the NTP server (from the NMI) calibrates the TSS at regularly scheduled inter- vals, depending on the accuracy required. During this process, all transactions are digitally signed and logged. All communication is by means of an authenticated, secure network con nection.

After calibrating and auditing the TSS, the TSMC or the NTP server issue a signed Time Attribute Certificate (TAC) for:

- Authorizing the operation of the TSS
- Certifying the calibration and traceability of the TSS. The TSS can then issue time-stamps for any Extended Public Key Infrastructure (PKIX) compliant time sign request.

A time-stamp includes:

- Details of the actual time it was issued.
- A hash of the digital information being time signed.
- A time certification or calibration pointer. The certification provides the necessary information to confirm that the time signing is accurate, valid, and traceable back to an official time authority.

The time-stamps issued by TSS conform to the IETF Time-Stamp and Time-Stamp Token protocols.

# 2. Product overview

## 2.1. Roles and responsibilities

The TSS supports the following types of users:

- Security Officer: to handle the key and certificate management
- Network Manager: to manage the day-to-day operations.

### 2.1.1. Security Officer

The Security Officer is responsible for completing all tasks related to the installation of the TSS, including:

- Contracting with an auditing service provider
- Preparing the TSS environment
- Configuring the TSS
- Creating and certifying TSA keys
- Certifying upper clocks.

### 2.1.2. Network Manager

The Network Manager is responsible for:

- Enabling or disabling time-stamping and auditing
- Configuring audit settings.

## 2.2. Features

### 2.2.1. nShield PCIe module

The TSS uses an integrated nShield PCIe module with Secure Execution Engine (SEE). All cryptographic functions such as processing, time-stamping, and clock operations are performed within the module. The module also meets the internationally recognized FIPS 140-2 at level 3 standard.

### 2.2.2. Public Key Infrastructure (PKI) technology

The TSS uses the PKI technology to ensure authenticity, integrity, and traceability of information. The PKI technology validates not only the time used for issuing a time-stamp, but also the actual time-stamp issued by the TSS. In PKI-based time signings, the actual document or file to be time signed does not leave your computer. When you request a time-stamp, a message digest of the file is created using a one-way hash function. The original file cannot be modified without invalidating this hash. The PKI technology provides the following benefits:

- Privacy: only you can see the message
- Integrity: you receive what was actually sent
- Non-repudiation: the sender cannot deny they sent the message
- Authenticity: you can be sure of the identity of the person who sent the message.

### 2.2.3. Certificates for signing and authentication

The TSS uses X.509 public key certificates to sign the time-stamps and audit records. A public key certificate is issued by a Certification Authority (CA), and is used for authenticating the identity of a person, or business entity, or system device.

### 2.2.4. Support for leap second events

The TSS offers complete support for leap second events.

### 2.2.5. Compliance with IETF standards

Internet Engineering Task Force (IETF) is an international organization that sets standards for internet protocols in their Request for Comment (RFC) papers. The TSS implements time-signing as specified by the IETF PKIX Time-Stamp RFC (RFC3161).

### 2.2.6. Time-stamp protocol support

The TSS supports the following time-stamp protocols:

- RFC3161 Socket Base Protocol on TCP/318
- RFC3161 Time-stamp Protocol by HTTP at the following URL: `http://<tss-host-name>/TSS/HttpTspServer`
- Authenticode Time-stamp Protocol by HTTP at the following URL: `http://<tss-host-name>/TSS/AuthenticodeTS`

In these URLs, *<tss-hostname>* represents the IP address or domain name of your TSS.

By appending the `tsa=` query string to the RFC3161 or Authenticode URLs above, it is possible to access a specific TSA. This query string can accept either a TSA's number (e.g.`tsa=tsaid_<tsa number>`) or the TSA's name. For example:

- TSA number: `http://<tss-hostname>/TSS/HttpTspServer?tsa=tsaid_1`
- TSA name: `http://<tss-hostname>/TSS/AuthenticodeTS?tsa=exampleTSA`

If specifying the TSA number, `tsaid` must be in lower case. Otherwise, the TSA name is case sensitive.

## 2.3. Getting an auditing service provider

Before you begin, ensure that you are contracted for auditing services with an auditing service provider. For information about service providers, contact Entrust nShield Support, https://nshieldsupport.entrust.com.

After you have signed your agreement with your auditing service provider, your provider will supply you with, at a minimum, the entire certificate chain for all upper clocks and the IP address for all upper clocks you will be using.

## 2.4. TCP/IP and UDP requirements

Your service provider will require the TCP/IP and UDP information regarding the TSS. The TSS listens, by default, on the following TCP/IP or UDP ports for time-stamp, Datum Secure Network Time Protocol (DS/NTP), and administrative functions:

| Listens on: | For: |
|---|---|
| UDP/user-configured port | DS/NTP audits when configured to use an upper clock. If using an upper clock, you must configure a unique UDP port number for each TSA. See Configuring a TSA for the instructions. |
| TCP/318 | RFC3161 Socket-Based Time-Stamp protocol (TSP). |
| TCP/80 and TCP/443 | HTTP and HTTPS connections. HTTP and HTTPS are used for the administrative user interface. HTTP also supports:<br>• RFC3161 TSP over HTTP (`http://<tss-hostname>/TSS/HttpTspServer`).<br>• Authenticode TSP over HTTP (`http://<tss-hostname>/TSS/AuthenticodeTS`). |

## 2.5. Understanding Security Worlds

The Security World is a paradigm or construct that provides secure life-cycle management for cryptographic keys. Key management involves procedures and protocols used throughout the life cycle of cryptographic keys. These procedures and protocols include the genera tion, distribution, use, storage, destruction, and optional archiving and disaster recovery of cryptographic keys. The Security World infrastructure enables you to perform and control all these activities under your chosen security policy. A Security World consists of the following components:

- At least one nShield module.
- An Administrator Card Set (ACS) requires access to Security World configuration and recovery operations.
- Optionally, one or more Operator Card Sets (OCS) for controlling access to application keys.
- Some cryptographic key and certificate data. The data is encrypted using the Security World key and stored on the hard disk.

The Security World key can be restored as it is stored on the hard drive.

## 2.6. Security

The Security World has been designed to ensure that keys remain secure throughout their life cycle. The Security World uses multiple interlocking keys, and because of this, each key is always protected by another key, even during recovery operations. The keys in the TSS are never available in the plain text format. When a Security World is created, a cryptographic key is generated. This key protects the Time Stamping Authority (TSA) keys and card sets that are later created and used in that Security World.

## 2.7. Administrator and Operator Card Sets

A Security World uses smart cards for both administering and operating the TSS. The Administrator Card Set (ACS) is used to control access to recovery functions. One or more OCS are used to protect access to signing operations for the TSAs in the TSS. You can create any number of OCS within a Security World.

In FIPS 140-2 level 3 Security Worlds, smart cards are required to authorize some operations, including the creation of keys and card sets.

All card sets are distinct. An individual smart card can only belong to either the ACS or to a

specific OCS. Each user can access the keys protected by the Security World and the keys protected by their OCS. They cannot access keys that are protected by any other OCS.

The smart cards that are used in an OCS use the Security World key for a challenge-response operation with the TSS. This means that Operator Cards can only be used in a TSS belonging to the same Security World.

Each card set must consist of (*K* of *N*) smart cards, where:

- *N* is the total number of smart cards in the card set.
- *K* is the required number of cards (or quorum) required to authorize an action.

The value for *K* must be less than *N*. We recommend that you do not set *K* equal to *N*, because an error on one card will render the card set unusable. If this happens with your ACS, you must replace your Security World and generate new keys.

A Common Criteria CMTS Security World requires a 2-of-2 ACS.

## 2.8. FIPS 140-2 compliance

FIPS 140-2 level 2 is the default setting for the Security World. However, when you create a Security World, you can choose whether you want compliance with the roles and services section of the FIPS 140-2 standard at level 2 or level 3. This choice influences the algorithms, key sizes, and curves available.

A Security World that complies with the roles and services section of FIPS 140-2 level 2 does not require any authorization to create an OCS or an application key.

## 2.9. FIPS 140-2 level 3 compliance

This option is included for those customers who have a regulatory requirement for compliance with FIPS 140-2 level 3.

A Security World that complies with FIPS 140-2 level 3 requires authorization from any smart card that is part of the Security World's ACS or an OCS, before you can create or erase an OCS. If you create a Security World that complies with FIPS 140-2 level 3, the TSS initializes in FIPS mode. This option ensures that the TSS complies with the roles and services, key management, and self-test sections of the FIPS 140-2 level 3 standard, as described in its validation certificate. For more information about FIPS validation visit: http://csrc.nist.gov/groups/STM/cmvp/standards.html

## 2.10. Using Operator Card Sets to protect keys

If you want to restrict key access to a particular user, you can create a set of smart cards known as an OCS. An OCS belongs to a specific Security World. An OCS can be read, erased, or formatted only by a TSS that is within the Security World of the OCS. An OCS stores a number of symmetric keys that are used to protect the working TSA keys. Each card in an OCS stores only a fragment of the OCS keys. These keys can only be re-created if you have access to a sufficient number of fragments (*K*). Because cards sometimes fail or are lost, the number of fragments required to re-create the key (*K*) must be less than the total number of fragments (*N*).

The Security World key can be restored as it is stored on the hard drive.

### 2.10.1. Using card sets for extra security

If you want to create a card set for extra security, you need to make *K* large and *N* less than twice *K* (for example: 3 of 5, or 5 of 9). This practice ensures that if you have a set of *K* cards that can be used to re-create the key, you can be certain that there is no other such set in existence.

### 2.10.2. Using pass phrases

You can optionally assign a pass phrase for each Operator Card. The pass phrases are independent. You can assign pass phrases only to a few cards in a card set. To change a pass phrase of a card, you require the card, the existing pass phrase, and a TSS that belongs to the Security World.

A pass phrase can contain a maximum of 255 characters and only those that are accepted by a HTML form.

### 2.10.3. Using persistent Operator Card Sets

When you create an OCS, you can decide whether or not to make the OCS persistent. However, the property that you choose cannot be modified later.

If you create a standard (non persistent) OCS, the TSA keys protected by the card can only be used while the card—or the last card loaded from a card set—is in the TSS smart card reader. The keys protected by this card are removed from the memory of the TSS as soon as the card is removed from the smart card reader. Although this feature provides added security, it means that only one user can load keys at any time.

If you create a persistent OCS, the keys protected by a card persist after the card has been removed. The TSS maintains strict separation between the TSA keys loaded by each user, and each user can access only those keys that are protected by their OCS.

TSA keys protected by a persistent card set are automatically removed from the TSS:

- After the time limit specified during the card set creation
- When the **Clear** button on the nShield PCIe module is pressed
- When the TSS is restarted.

A Security World can contain a mix of persistent and non-persistent card sets.

### 2.10.4. Resilience

The Security World ensures that in the event of a disaster, you can recover the TSS without compromising the security of the system, providing that you back up the system regularly.

### 2.10.5. Backup and recovery

Ensure that you back up the following on a regular basis:

- The `%NFAST_KMDATA%` directory
- The `%NFAST_HOME%\dse200\UserFiles` directory. This directory contains the log files and the configuration files for your TSS.

If an attacker obtains the backup data, they cannot use it without the encryption keys stored in your nShield PCIe module and the Administrator cards for that Security World.

When you create a Security World, it automatically creates recovery data for the Security World key. This data is encrypted and the cryptographic keys that protect this data are stored on the ACS. The keys are split among the cards in the ACS using the *K*-of-*N* mechanism.

The ACS protects several keys that are used for different operations. The cards in the ACS are only used for recovery operations and adding additional modules to a Security World. At all other times, these cards should each be stored separately, in secure locations.

In FIPS 140-2 Level 3 Security Worlds, the ACS or an OCS is also used for controlling the creation of keys and OCSs.

### 2.10.6. Replacing the ACS

If you lose one of the smart cards from the ACS, or if the card fails, you must immediately create a replacement set using the `racs` command-line utility. The TSS does not store recovery data for the ACS. It relies on there being at least enough smart cards to reach the number of cards in the quorum *K* from the original *N* cards created. Therefore, it can re-create all the keys on the TSS even if the information from one of the cards is missing.

Although replacing the ACS deletes the copy of the recovery data on the host, the old ACS can still be used with the old host data, which may be on backup tapes and other hosts. To protect against this risk, you must immediately erase the old Administrator Cards after you create a new ACS.

### 2.10.7. Risk management

Although a Security World can manage keys and control which user has access to which keys, it cannot prevent a trusted user from using a key fraudulently. For example, a Security World can only determine whether a user is authorized to use a TSA key. It cannot determine whether the message being time-stamped with that key is accurate. To protect the security of your system, ensure that you:

- Keep your smart cards safe.
- Always obtain smart cards from Entrust.
- Never use your TSS and smart cards with an untrusted smart card reader.
- Keep your pass phrase secure.

If you suspect that the security of a key or the Security World has been compromised, you must replace that key or Security World with a newly generated one.

# 3. Prerequisites

## 3.1. Assumptions

You are installing, configuring, and administering the Entrust nShield Time Stamp Server™ (TSS), part code DSE200. You are familiar with operating and maintaining networks.

## 3.2. Supported Platforms

TSS can be installed on the following platforms:

- Microsoft Windows Server 2012 R2 x64
- Microsoft Windows Server 2016 x64
- Microsoft Windows Server 2019 x64

## 3.3. TCP/IP and UDP port access

Before you begin, ensure that you have access to the following ports:

- TCP/80 (HTTP)
- TCP/443 (HTTPS)
- TCP/318 (TSP, Time-stamp Protocol)

During installation of the Entrust nShield Time Stamp Option Pack (TSOP), the default HTTP and HTTPS ports can be updated as required.

If you change the default HTTP or HTTPS ports, ensure that you specify ports in URLs for the TSS Web interface see *Accessing the TSS Web Interface*, before continuing with the configuration processes described in *Configuring the TSS*.

## 3.4. TSOP installation prerequisites

Before installing TSOP:

- Confirm that the following items are available:
  - The TSOP installation media
  - An SEE Activation feature card

    - An Elliptic Curve algorithms feature card
    - The Security World Software installation media
    - An nShield PCIe module (with smart card reader and a set of smart cards).
- Install the nShield PCIe module and nShield Security World Software as described in the *Installation Guide* for your module. TSOP supports the following module:
    - NC4433E-500: nShield 500+ F3, SEE Ready

Do not install TSOP on systems fitted with any other nShield hardware or with more than one module.

Installation will abort if an nShield Security World Software installation cannot be found.

- TSOP requires an installation of a Windows 32-bit Standard Edition Java Runtime Environment (JRE) version 1.8. This must be installed before installing TSOP.

The software install wizard determines whether you have a suitable JRE installed. The install process will abort if no suitable JRE can be found.

- A Network Time Protocol (NTP) distribution or Time Stamp Master Clock (TSMC) is required to calibrate and audit the TSS. See the *TSOP Administrator Guide* for further information.
- Install the latest Microsoft security updates. See: https://www.microsoft.com/security.

# 4. Installing the Time Stamp Option Pack

During the TSOP installation, the hardserver (nFast Server and any associated service dependencies) will be restarted.

To install TSOP on a supported Microsoft Windows operating system:

1. Log in with the Administrator role or as a user with local administrator rights.
2. Place the TSOP support software disk in the CD/DVD drive. If Autorun is enabled, the installer `setup.exe` runs automatically, detects the version of Windows and launches the appropriate installation program. If Autorun is not enabled, launch the installer manually.
3. Click Next to continue.

   The installer displays the license agreement.
4. Accept the license agreement.

   The install process will automatically detect the location of the Security World Software installation and will install alongside this.
5. When prompted, enter the port settings for the HTTP and HTTPS protocols.
6. Follow the installer instructions until the installation process is complete.

After you have installed the software, enable the SEE feature and, if desired, the Elliptic Curve algorithms feature, as described in the following section.

## 4.1. Enabling features

After TSOP is installed, enable the SEE Activation (Restricted) feature. This feature enables the TSS to perform specific tasks using the SEE.

If you are intending to use ECDSA-based keys, it will be necessary to enable the Elliptic Curve algorithms feature.

Entrust provides you with smart cards that contain the Feature Enabling Certificates for the SEE Activation (Restricted) feature and the Elliptic Curve algorithms feature.

To enable a feature using the provided smart card:

1. Insert the Feature Enabling smart card into a smart-card reader connected to the TSS.
2. Start the Feature Enable Tool by running the following command:

```
%NFAST_HOME%\bin\fet.exe
```

If you start the Feature Enable Tool without a Feature Enabling smart card from Entrust, the tool displays various options for reading a Feature Enabling certificate (FEM)

3. Choose the option to read the FEM certificate from a smart card, and follow the onscreen instructions.

   After the feature is enabled, the system returns a success message.

If you do not enable the SEE Activation (Restricted) feature, the TSS cannot load the SEE machine. As a result, the Operation Status page of the TSS web interface returns the error message **SEE_Load MachineFailure**.

If you do not enable the Elliptic Curve algorithms feature, it will not be possible to use ECDSA-based keys.

See the *nShield Solo User Guide* for more information on the Feature Enable Tool.

## 4.2. Configuring the TSS and creating a Security World

After you have enabled the appropriate features, complete the setup process by configuring the TSS and creating a Security World.

In order to use an existing Security World, the Security World will need to have been created with the `SEEDebugForAll` feature enabled. In addition, SEE delegation will need to be performed. For instructions, see the *TSOP Administrator Guide*

## 4.3. Accessing the TSS Web Interface

A modern web browser, such as Microsoft Internet Explorer, Microsoft Edge, Mozilla Firefox or Google Chrome, can be used to access the TSS Web interface via:

```
https://localhost/TSS/index.jsp
```

Or:

```
https://<tss-hostname>/TSS/index.jsp
```

*<tss-hostname>* represents the IP address or domain name server (DNS) of your TSS.

If you changed the default HTTP or HTTPS ports (see TCP/IP and UDP port access), make sure that you specify ports in URLs for the TSS web interface, before continuing with the configuration process described in the *TSOP Administrator Guide*.

The default test TLS certificate for the web user interface must be replaced (see *Adding a TLS certificate* in the *TSOP Administrator Guide*).

At the Administrator Login, enter the appropriate user ID and password and then click **Login**:

| Field | Security Officer | Network Manager |
|---|---|---|
| Name | superuser | admin |
| Password | superuser | administrator |

For security purposes, we recommend that you change the default user names and passwords as soon as possible. See Modifying or deleting users for more information.

# 5. Creating and managing Security Worlds

This section explains how to create and manage Security Worlds.

## 5.1. Creating a Security World

Before you begin, ensure you have enough smart cards to create the ACS needed in the Security World.

When you create a Security World, you generate the cryptographic key that protects the TSA keys and OCSs that are later created and used. To create a new Security World, see the *Creating a Security World using new-world* section of the *nShield Solo User Guide for Windows*.

When creating a Security World, it is necessary to enable the `SEEDebugForAll` feature by specifying `dseeall`.

Currently, TSS only supports creation of TSA keys. However, future versions of the TSS may support the creation of keys for which the key recovery option is desirable. We therefore recommend that you select Yes. If the Security World supports key recovery, it is always possible to create a key with recovery disabled, but if the Security World does not support key recovery, then you cannot create a key with recovery enabled without reinitializing your Security World and discarding all your existing keys.

Once the Security World has been created, SEE delegation will need to be performed, see Security World: SEE delegation.

If you discover at any time that one or more of the cards in your Security World's ACS has been damaged or lost, use the command-line utility `racs.exe` to create a new set immediately. See the *nShield Solo User Guide* for more information about replacing a ACS. If further cards are damaged or lost, you may not be able to re-create your Security World.

## 5.2. Replacing a Security World

You can, if necessary, replace a Security World. Replacing an existing Security World in this way does not delete the Security World's host and recovery data. It renames the existing local directory within `%NFAST_KMDATA%` directory in which these reside as `local<nn>` (where *<nn>* is an integer, 0 or greater, depending on how many Security Worlds have been previously saved).

### 5.2.1. Operator cards

Any Operator Cards created in a previous Security World cannot be used in a new Security World.

If you are replacing a Security World, you must erase all Operator Cards created in the previous Security World before you create the new Security World.

However, if you want to discard a Security World, we recommend that you erase all your Operator Cards first or create a backup of the `%NFAST_KMDATA%` directory.

## 5.3. Joining an existing Security World

To add a new module into your Security World, or to restore an existing module after a firmware upgrade, it is necessary to reprogram the module. See the section *Adding an HSM to a Security World with new-world* of the *nShield Solo User Guide* for more information.

In order to use an existing Security World, the Security World will need to have been created with the `SEEDebugForAll` feature enabled. In addition, SEE delegation will need to be performed, for instructions see Security World: SEE delegation.

## 5.4. Security World: SEE delegation

The **Security World > SEE Delegation** option enables you to give the TSS SEE machine the permanent ability to set the real-time clock (RTC), allocate nonvolatile memory (NVRAM), and to originate keys. When the SEE delegation operation is complete, the delegation certificate signatures are stored in the file `%NFAST_KMDATA%\local\dsedelegation`.

If you lose the `dsedelegation` file, use the **Security World > SEE Delegation** option to re-create the appropriate privileges.

## 5.5. Viewing the Security World status

To view the status information of your Security World and its ACS, navigate to **About** and examine the **nfkminfo** output. See the *nfkminfo: information utility* section of the *nShield Solo User Guide* for more information.

# 6. Configuring the TSS on the network

The following sections explain how to configure your TSS on the network.

## 6.1. Getting a static IP address

Your auditing service provider must be able to connect to your UDP port. This means that your TSS must have a static IP address. Request the following information from your IS department:

- An IP address
- A subnet mask
- A gateway
- DNS servers and, if available, an SMTP server.

If you want e-mail alerts, you need an SMTP server with which your TSS can communicate.

## 6.2. Configuring the firewall

Check with your IS department about firewalls. Configure them now to allow your TSS to receive communications from your auditing service provider.

### 6.2.1. Adding a TLS certificate

A TLS certificate is required in order to securely communicate with the web user interface. This is stored within a Java keystore (JKS).

The default test TLS certificate must be replaced before TSS deployment. This is located in `%NFAST_HOME%\dse200\UserFiles\KeyStore\tomcat.keystore`.

Due to the sensitivity of its files, ensure that the `%NFAST_HOME%\dse200\UserFiles` directory has suitable access permissions e.g. that it is readable only by members of the local Administrators built-in group (SID S-1-5-32-544).

The following steps provide an example on how to replace the default test TLS certificate:

The official Apache Tomcat documentation provides information on how to install a certificate and configure SSL/TLS.

Refer to the official `keytool` documentation for further information on `keytool` (e.g. how to manage the keystore and display data).

1. Create a local self-signed certificate as follows:

```
keytool.exe -genkeypair -alias tomcat -keyalg EC -keystore tomcat.keystore
```

2. Create the certificate signing request (`certreq.csr`):

```
keytool.exe -certreq -alias tomcat -file certreq.csr -keystore tomcat.keystore
```

3. Once the certificate request has been signed by a CA, import the CA certificate chain (`ca.cer`):

```
keytool.exe -import -alias root -keystore tomcat.keystore  -trustcacerts -file ca.cer
```

   Confirm that you trust this certificate.

4. Import the signed certificate (`tomcat.cer`):

```
keytool.exe -import -alias tomcat -keystore tomcat.keystore -file tomcat.cer
```

5. Make sure `%NFAST_HOME%\dse200\UserFiles\certificate-config.xml` is updated to reflect the location of the keystore and the associated keystore and key passwords:

```
certificateKeystoreFile=<path and filename of keystore>

certificateKeystorePassword=<keystore password>

certificateKeyPassword=<key password>
```

6. Restart the DSE200 service.

# 7. Configuring the TSS

## 7.1. Time Stamping Authority (TSA) keys

A Time Stamping Authority (TSA) key is used for signing time-stamps. You can either create a single TSA and use the same signature key for all time-stamps, or create multiple TSAs depending on your requirements. TSA keys are created by your organization's Security Officer.

### 7.1.1. About multiple TSAs

TSS supports the creation of multiple TSAs. You can create multiple TSAs for:

**Departments**

Each department in your organization might need a different TSA.

**Customers**

If you are a service provider, you might want to operate a TSS for several customers. In such cases, you can create a separate TSA for each customer so that the TSA certificate name is related to the customer.

**Policies**

You might require different TSAs for different policies within your organization. For example, you might have policies that require different signature algorithms.

The number of TSAs that you can create depends on the NVRAM available in your nShield PCIe module.

### 7.1.2. How multiple TSAs work

#### 7.1.2.1. RFC3161 time-stamps

The TSS uses Policy Object Identifiers (OIDs) and hash algorithms (SHA-1, SHA-256, SHA-384, and SHA-512) to determine which TSA should be used to issue the time-stamp. However, the policy OIDs need not be unique for each TSA. When you create multiple TSAs, you can assign one of them as the default TSA. You can configure each TSA to support a specific policy OID and a list of hash algorithms. When a client requests a time-stamp, the TSS checks the policy OID and the hash algorithm on the request and one of the following occurs:

- If the request does not include an OID, it is sent to the default TSA or the first TSA that supports the hash algorithm.
- If the request includes an OID, it is sent to the first TSA that supports the OID and the hash algorithm.
- If the request includes an OID and none of the TSAs support the OID and the hash algo rithm, the TSS returns an error.

A TSA on TSS must receive a DS/NTP audit before it can issue time-stamps. When there are multiple TSAs, each TSA is assigned to a unique port. Each TSA listens to and receives the DS/NTP audit on the specified port.

#### 7.1.2.2. Authenticode time-stamps

Authenticode time-stamp requests do not include policy OIDs. If you intend to support Authenticode time-stamp requests, you must create a TSA that uses an Authenticode time-stamp mode key. When the client requests an Authenticode time-stamp, the TSS sends the request to this TSA.

## 7.2. Adding the CA certificates of an upper clock

The Upper Clock CA Cert Store holds the root CA certificate and intermediate CA certifi- cates of upper clocks that are authorized to audit the TSS. To add the certificate you have received from your auditing service provider:

1. Log in with the Security Officer role.
2. Navigate to **Certificate Management > Upper Clock Cert Store**.
   The **UC Certificate Store dialog** opens.
3. Click **Add**.

   Browse to locate the certificate, and click **Add**.

   To enable an upper clock to audit a TSA you must:

   - Add the root and intermediate CA certificates to the **Upper Clock CA Cert Store**
   - Add the clock to the TSA. See Configuring a TSA for more information.

4. Log out of the TSS.

At this stage, your organization's Network Manager can log in and configure settings so

that your service provider can audit your time settings.

## 7.3. Adding an upper clock

This section explains how to add an upper clock and configure its settings so that it can be audited by your service provider. If you have multiple TSAs, you can configure the audit set tings of each TSA. Each TSA can have a different upper clock.

1. Log in with the Network Manager role.
2. Navigate to **TSA Management > Configuration**.

   The **TSA Configuration** dialog opens, listing all the TSAs that exist in the TSS.

3. Select the TSA to which you would like to add the upper clock, then click **Configure**.

   The **TSA Configuration** dialog opens.

4. Click **Add.**
5. Enter the following clock information:

| Field | Enter / Select |
|---|---|
| Upper Clock IP | The IP address to your upper clock. The default port number is 318.<br> Your audit service provider must be able to access this IP address. |
| Upper Clock Port | The default port for your upper clock. |
| Audit Request Retry Delay | How long in seconds before your service provider should wait before try- ing to audit your clock. |
| Audit Request Max Retry | The maximum number of audit attempts in the case of communication fail ure. |

6. Click **Add.**

   The new upper clock is added to your list.

7. Back in the **Audit Configuration** dialog, click **Update** to save the information you have entered.

   For information on how to remove upper clocks, see Removing an upper clock.

## 7.4. Creating a new TSA

A TSA key is used for signing time-stamps. You can either create a single TSA to issue a single type of signature across all departments in your organization, or create multiple TSAs based on your requirements.

1. Log in with the Security Officer role.
2. Navigate to **TSA Management > Configuration**.

   The **TSA Configuration** dialog opens.

3. Click **Add**.

   The **Create a New TSA** dialog opens.

4. Enter a name for the TSA.
5. Enter a TSA policy identifier.

   The policy OID is used to determine which TSA should sign the time-stamp request. When a time-stamp request is received, the TSS checks the policy OID and sends it to the TSA that supports the OID. It is possible to change the policy OID later. See Configuring a TSA for more information.

6. Click **Add**.
7. Click **OK** to confirm the details of the TSA.

   The TSA is created and the status appears as **Uncertified**. You can now configure the settings of the TSA. The TSA is not functional until you initiate and fulfill the certificate request.

## 7.5. Configuring a TSA

After you create a TSA, you can configure its settings.

The Configuration option enables you to:

- Change time-stamp policy OIDs
- Select the acceptable time-stamp hashes
- Configure the port on which the TSA listens on for DS/NTP audits
- Select the DS/NTP audit source
- Add the root CA certificate of the upper clock that must audit the TSA.

To configure the settings:

1. Log in with the Security Officer role.

2. Navigate to **TSA Management > Configuration**.

   The **TSA Configuration** dialog opens, listing all the TSAs that exist in the TSS.

3. Select the TSA that you would like to configure, then click **Configure**.

   The **TSA Configuration** dialog opens.

4. Enter the name of the TSA in the TSA Name field.
5. Optionally, to change the policy OID in the TSA, enter a new OID in the Acceptable Pol icy field.
6. Select the time-stamp hashes that the TSA must support.
7. Select a TST TAC encoding and binding option.

   These options determine in which part of the time-stamp token the TAC is stored. The ETSI option adds two additional signed attributes that are required by RFC3126. For more information about these options, see Time Stamp Tokens (TST) TAC encoding and binding.

8. Select a TST option.

   For more information about TST options, see TST options.

9. Enter a DS/NTP port number on which the TSA will receive the audit.

   This port number has to be unique for each TSA.

10. Select an DS/NTP audit source:

    **Local Audit**

    If you choose this option, a Windows Administrator must log into the System con- sole and restart the NTP Service. Also, a reference to an NTP server (on the local network) should be added to the NTP configuration file. The Local Audit setting pro vides time that can be traced only to the TSS host PC clock. If you select this option, we recommend that you use a good security policy with regard to the phys ical security of the TSS and the network connection to the NTP server. See Local Audit / NTP service.

    **Upper Clock**

    If you choose this option, select the upper clocks that will audit the TSA. Option- ally, select the Auto-trust New Upper Clock Certificates check box — when new upper clocks are added to the Upper Clock CA Cert Store, the TSA automatically starts trusting them.

    All TSAs share a single Upper Clock CA Cert Store, which holds

> all the root and intermediate CA certificates of the upper clocks that are authorized to audit the TSS. When you add an upper clock to a TSA, you are essentially choosing (from the upper clock CA cert store) the ones that will audit the TSA.

11. Click **Update**. The changes take effect immediately.

## 7.6. Initiating and fulfilling TSA certificate requests

The following sections describe the steps involved in initiating and fulfilling certificate requests.

Create an OCS before you create a TSA Key, otherwise you cannot use the TSA backup/restore feature and you will not have a backup of the key.

### 7.6.1. Creating Operator Card Sets

This section explains how to create an OCS for authorizing access to TSA keys. Creating an OCS is optional.

Card sets belong to the Security World in which they are created. When you create an OCS, the smart cards in that set can only be read by the nShield PCIe modules belonging to the same Security World. The cards cannot be read, erased, or reformatted by a module from a different Security World.

If the OCS you create is to be used to protect TSA keys that have the Disable Unattended Start-up feature enabled (see Initiating a TSA certificate request), it is important that you understand the effect of the persistence and time-out options.

By default, the TSS creates non-persistent card sets, which means that keys protected by this card set become unavailable when the last card is removed. Thus, a TSA key can be fulfilled by the TSA only while a card remains in the slot. When the card is removed, the TSS essentially becomes non-operational. Moreover, if you define a time-out when creating the OCS, the TSA key is subject to this time-out.

However, if the TSA Key does not have the Disable Unattended Start-up feature enabled, time-stamp requests can be fulfilled even after the card has been removed from the card reader, and the availability of the TSA key is not subject to a time-out.

To create an OCS:

1. Log in with the Security Officer role.
2. Navigate to **Card Set Management > Create Card Set**.

   The **Create Operator Card Set** dialog opens.

3. Enter the following information to create an OCS you can use when creating a TSA key you want to be able to backup and restore:

| Field | Description |
|---|---|
| Operator Card Set name | Enter a name for the OCS. |
| Total number of cards in set (*N*) | Enter the total number of cards (*N*) that you want to have in the OCS. This number (*N*) must be less than or equal to 64. |
| Number of cards required for access | Enter the number of cards (*K*) needed to restore the key. This number (*K*) must be less than or equal to the total number of cards (*N*). |
| Card set is persistent? | The default value for this option is 'No'.<br><br>This means that any key protected by the OCS becomes unavailable when the last card is removed from the card reader. If you select 'Yes', the key will still be valid even after the last card has been removed. |
| Card set has a timeout? | Optionally select to enable a time-out value in seconds for the OCS. The time-out is the length of time that a card from the set can remain effective when inserted in the card reader. After the time-out is reached, the card must be re-inserted before it can be used. |
| Timeout in seconds | Enter the number of seconds to elapse before the timeout is enforced. The maximum value is 31622400 seconds. |

4. Click **Next**.

   The TSS prompts you to insert the cards that will form the OCS.

5. For each card, follow the onscreen instructions to either set a pass phrase for the card or to create a card without a pass phrase. Each card can have a different pass phrase, and any card's pass phrase can be changed later. See Using pass phrases for more information on pass phrases.

   When creating a card set, the TSS recognizes a card that belongs to the set before the card set is complete. If you accidentally insert a card to be rewritten, the system returns a warning.

   On completion the TSS displays a message indicating that the OCS has been successfully created.

## 7.6.2. Initiating a TSA certificate request

A TSA is not functional until you initiate and fulfill the certificate request.

1. Log in with the Security Officer role.
2. Navigate to **TSA Management > Certification Status.**

   The **TSA Certification Status** dialog opens, listing all the TSAs that have been created.

3. Select the TSA that you would like to work with, then click **Initiate** to initiate a new TSA certificate request.

   The **Select Cardset for TSA Key Backup** dialog opens. If there are no Operator Card Sets installed on the TSS, the **TSA Key Backup** dialog does not open. Instead the **TSA Certificate Request** dialog opens, see step 6 below. The TSA keys you generate without an OCS will not have TSA Key Backup enabled.

4. TSA keys cannot be restored if the OCS is lost. However, to restore keys from a disk crash or data corruption, make regular backups of the TSA key backup files and other Security World data. See Restoring a TSA key for the instructions to restore a TSA key.
5. To enable TSA Key Backup for the TSA Key you are about to generate, select the OCS you want to use.

   If you choose the **Do not enable TSA Key backup for this key** option, you cannot enable key backup for this key later.

6. Click **Next**.

   The **Loading OCS:***OCSName* dialog opens. This dialog displays general information about the OCS followed by loading state information and a **Next** button. Each time you click **Next**, you are redirected to an updated display of this dialog.

   The **Loading OCS:***OCSName* dialog requests cards and pass phrases until you have presented the number of cards required to load the OCS. When the OCS is loaded, the **Loading OCS:***OCSName* dialog displays the message Operator Card Set *OCS Name* loaded.

7. Click **Next**.

   The **TSA Certificate Request** dialog opens.

8. Enter the following information:

| Field | Enter |
|---|---|
| Common Name | A name for the TSA certificate |

| Field | Enter |
|---|---|
| Organization | The name of your organization |
| Organizational Unit | The name of your organizational unit (optional) |
| Serial Number | A number that uniquely identifies the certificate (optional) |
| Organizational Unit | Any additional information you want to include in the certificate name (optional) |
| Organization Identifier | A string to identify the organization (optional) |
| Locality | The name of your locality (optional) |
| State | The name of your state (optional) |
| Country | The name of your country |

A message appears, asking you to confirm the details that you have entered.

9. Click **OK** to confirm the information that you have entered.
10. Click **Next**.

    The **TSA Certificate Request Parameters** dialog opens.

11. Enter the following information:

| Field / Option | Description |
|---|---|
| Key Type | Select one of the following key types:<br>• DSA<br>• ECDSA<br>• RSA |
| Signature Algorithm | Select an algorithm that is acceptable to your CA. |
| Key Length/Curve | Select a key length/curve. |
| Time-stamp Mode | Select the time-stamp mode. The available values are:<br>• *RFC 3161*<br>RFC 3161 is the Internet X.509 Public Key Infrastructure Time-Stamp Protocol.<br>• *Authenticode*<br>Authenticode, from Microsoft, allows developers to include information about themselves and their code with their programs through the use of digital signatures. |

| Field / Option | Description |
|---|---|
| Distinguished Name | Review the information used for the certificate's name. |
| Disable unattended startup for this key! | Select this option to disable the ability for your TSS to startup unattended. By default, a TSS can start up and become completely operational without user intervention. However, in some environments, this is not appropriate. Disabling this option allows the TSS to operate under such policies. The ability to disable unattended start-up of the TSS can be granted to a TSA Key only when it is created. This feature depends on the OCS protection that is part of the new TSA Key Backup and Restore feature. Therefore this option is only available on a TSA Key that is being created with the Backup feature enabled.<br> This option is only available if you selected to enable TSA key backup for the TSA key.<br> If you select this option, you must load the TSA key for the unit to operate. |
| Generate a self-signed certificate for this key | Select this option to generate a self-signed certificate. This is useful for testing purposes. |

12. Click **Submit Request**.

    The **TSA PKCS10 Certificate Request** dialog opens.

13. Click **Download**.

    A browser window opens:
    On Google Chrome and Mozilla Firefox, the certificate request is directly displayed in the TSS Web interface after the **Download** button is clicked.
    On Microsoft Internet Explorer and Microsoft Edge, the certificate request is downloaded at `TSAcertreq_tsaid_n.pem` where *n* is the TSA number.

14. Send the TSA certificate request file to your CA. If the CA approves your request, then the CA returns a TSA certificate to you.
15. Save the TSA certificate in a secure location. You use this certificate in Fulfilling a TSA certificate request.

    The TSA certificates remain in the pending state until they are fulfilled. To view the list of all TSA certificates that are not fulfilled, navigate to **TSA Management > Certification Status**. The Key Status of any certificate that has not been fulfilled appears as **UncertifiedPending.**

## 7.6.3. Importing the TSA certificate chain

After the CA approves your certificate request, you must import the CA's certificate chain into your TSA certificate store. The TSA Certificate Store contains one or more CA certificate chains that can be used to validate your TSA certificate during fulfillment. A CA certificate chain typically includes a root certificate and an intermediate issuing CA certificate. A simple CA certificate chain might include a single root certificate, which is used as an issuing CA certificate. A more complex CA certificate chain might include a root certificate, intermediate CA certificate, and an issuing CA certificate. You must add the certificates to the TSA certificate store in the following order:

- Root CA certificate
- Certificates signed by the root CA certificate in the order in which they are signed
- Issuing CA certificate.

To import the CA certificate chain into the TSA certificate store:

1. Navigate to **Certificate Management > TSA Cert Store**.

   The **TSA Certificate Store** dialog opens.

2. Click **Add**.

   The **Add TSA Certificate** dialog opens.

3. Do one of the following:
   - Browse and locate the certificate, then click **Load File**.
   - Copy and paste the contents of the certificate in the text box.
4. Click **Add**.
5. Repeat the steps until you have finished adding all the certificates in the CA certificate chain, including the issuing CA certificate.

## 7.6.4. Fulfilling a TSA certificate request

After you receive your TSA certificate from your CA, you can fulfill it. Before proceeding with this step, ensure that you have imported the complete CA certificate chain including the root, intermediate, and issuing CA certificates.

1. Navigate to **TSA Management > Certification Status.**

   The **TSA Certification Status** dialog opens, listing all the TSAs that have been created.

   The Key Status of a pending certificate appears as **Uncertified-Pending**.

2. Select the TSA certificate request that you would like to fulfill, then click **Fulfill**.

   The `Fulfill TSA Certificate` dialog opens.

3. Browse to the TSA certificate file you received from your CA, or copy it in base-64 format into the box.

   If your CA has sent the certificate in Binary DER, ask for one in `.pem` or `base-64` format.

4. Click **Accept**.

   If you have chosen to enable key backup for this key, the TSS creates a unique backup file for the TSS on the system hard drive when the certificate request is fulfilled.

   Backup files are stored under `%NFAST_KMDATA%\local\key_dsetsa_tsakey(n)`, where `'n'` is the number allotted to the TSA based on the order in which it is created. For example, the TSS assigns the filename `key_dsetsa_tsakey(1)` for the first TSA that you create. Back up the contents of the `%NFAST_KMDATA%` directory to an appropriate storage device.

## 7.7. Loading a TSA

Whenever the TSS software is restarted with a TSA Certificate with the Disable Unattended Startup feature enabled, the Key Status is **Certified_NotLoaded** (or **CertifiedPending_NotLoaded**). Before the TSS can be audited to start issuing time-stamps, you must load the TSA Key.

1. Log in with the Security Officer role.
2. Navigate to **TSA Management > Operational Status**.

   The **Operational Status** dialog opens, listing all the TSAs that exist in the TSS.

3. Select the TSA that you would like to load, then click **Details**.

   The **Operational Status** dialog opens. The last line of this dialog is the Key Status. If the Key Status includes the **_NotLoaded** label, the dialog also includes a **Load** button next to the indicated Key Status.

4. Click **Load**.

   The **Loading OCS:***OCSName* dialog is displayed. This dialog displays general information about the OCS followed by loading state information and a **Next** button. Each time you click **Next**, you are redirected to an updated display of this dialog.

The **Loading OCS:***OCSName* dialog requests cards and pass phrases until you have presented a quorum and the OCS is loaded. When the OCS is loaded, the **Loading OCS:***OCSName* dialog displays the message Operator Card Set *OCSName* loaded.

5. Click **Next**.

   The TSA Key is loaded, and you are directed back to the **Operational Status** dialog. The key state is now **Certified** or **CertifiedPending**.

   The TSA Key remains loaded as long as the application continues to run and the protecting OCS allows the key to be loaded. If the protecting OCS is not persistent and the last card is removed from the card reader, the TSA key is unloaded. If the OCS has a time-out enabled, the TSA Key is unloaded when the time-out period expires. In such cases, you must reload the TSA Key by following the instructions in this section.

## 7.8. Registering a TSA

The TSA Registration section of the Certificate Management menu focuses on a key part of the security used to register a TSA certificate with an auditing service provider.

1. Log in with the Security Officer role.
2. Navigate to **Certificate Management > TSA Registration**.

   The **TSA Registration** dialog opens, displaying the list of TSA certificates that exist in the TSS.

3. Select the TSA that you would like to register, then click **Details**.

   The **TSA Registration** dialog opens, displaying the TSA certificate and the TSS certificate. This information guarantees that the TSA certificate is in the TSS. Also, the audit service provider can know for certain that the TSA certificate is controlled by the TSS.

4. Click **E-mail** to send the `TSARegistration.pem` file (a binary file containing no useful plain-text information) to the audit services provider.
5. Add the appropriate information, then click **Send Registration**.

## 7.9. Checking the operational status

If you log in as a Security Officer, you will only be able to view the details.

The final phase of getting started with your TSS is for the Network Manager to check opera

tional status.

1. Log in with the Network Manager role.
2. Navigate to **TSA Management > Operational Status**.

   The **Operational Status** dialog opens, listing all the TSAs that exist in the TSS.

3. Select the TSA that you would like to review, then click **Details**.

   The operational status for the selected TSA is displayed:

| Parameter | Description |
|---|---|
| Clock Status | A green light indicates that the clock is running. |
| Audit Status | A **green** light indicates that the auditing software is running, and that the clock can be audited. |
| Time-stamping Status | A green light indicates that the TSS is ready to time-stamp documents. An **amber** light indicates that the TSS needs a new TAC before it can continue. A **red** light indicates that time-stamping has been disabled. |
| Time Attribute Certificate | The status of the current TAC:<br>**Received**<br>a valid, operational TAC has been received and can be used for time-stamping.<br>**AntiTAC_Rcvd**<br>a non-operational TAC has been received, which has halted time-stamping until the next audit (which is likely to provide an operational TAC).<br>**Expired**<br>the current time is later than the "Valid To:" time specified in the current TAC.<br>**Invalid**<br>the current TAC has been invalidated, for instance, by rebooting the TSS or by disabling the clock.<br>For more information, see Viewing Time Attribute Certificate (TAC) information. |

| Parameter | Description |
| --- | --- |
| Key | The status of the TSA Key.<br>**Certified**<br>the TSA is operational.<br>**CertifiedPending**<br>the TSA has an operational certificate and also has an outstanding certificate request.<br>**Uncertified**<br>the TSA does not have any certificate and does not have an outstanding certificate request.<br>**UncertifiedPending**<br>the TSA does not have any certificate but has an outstanding certificate request.<br>**Expired**<br>either the TSA is past its validity period, or the current time in the TSS board clock is set to a time outside the range defined by the TSA certificate. |

4. All objects should be **green**. If one or more are not, click the object's **Enable** button.

# 8. Administering and using the TSS

This section includes information on the daily administration and usage tasks.

## 8.1. Restoring a TSA key

The **Restore** option is enabled only if the TSA key status is **Uncertified**. If the key is in any other state:

- Delete the TSA certificate from the TSA Certificate Store
- Cancel any pending TSA certification requests
- Re-create the TSA certificate so that the **Restore** option is enabled.

Before attempting to restore a TSA key, ensure that the TSA Key Backup file `key_dsetsa_tsakey(n)` is present in the system hard drive's `%NFAST_KMDATA%` directory. If necessary, copy the back-up file you made at the time of creation to this directory. Ensure that the `%NFAST_KMDATA%\local\key_dsetsa_tsakey(n)` file is in place.

1. Log in with the Security Officer role.
2. Navigate to **TSA Management > Certification Status**.

   The TSA Certification dialog opens, listing all the TSAs that exist in the TSS.

3. Select the TSA that you would like to restore, then click **Restore**.

   If the `%NFAST_KMDATA%\local\key_dsetsa_tsakey(n)` file is not found, the TSS returns the following message:

   ```
   The restore blob for the dsetsa,tsakey(n) could not be found.
   Unable to restore the TSA Certification.
   ```

   The **Loading OCS:***OCSName* dialog opens. This dialog displays general information about the OCS followed by loading state information and a Next button. Each time you click **Next**, you are redirected to an updated display of this dialog. (This is the same dialog that was used to enable the TSA Key for backup.) The **Loading OCS:***OCSName* dialog requests cards and pass phrases until you have presented a quorum and the OCS is loaded.

   If you cannot present the required number (*K*) of Operator Cards (for example, if you have lost too many cards from the OCS), the TSA backup facility does not work.

4. When the OCS is loaded, the TSS returns the following message:

```
Operator Card Set OCSName loaded.
```

5. Click **Next**.

   The TSA key and certificate are restored, and you are directed to the **TSA Certificate Info** dialog, which displays information about the restored certification.

## 8.2. Removing an upper clock

The **TSA Configuration** dialog is where you manage audit settings and add or remove upper clocks. See Adding an upper clock>> for information on how to add a new upper clock.

1. Log in with the Network Manager role.
2. Navigate to **TSA Management > Configuration**.

   The **TSA Configuration** dialog opens, listing all the TSAs that have been created.

3. Select the required TSA, and click **Configure.**
4. In the **TSA Configuration** dialog, select the clock you want to remove, then click **Remove.**
5. When prompted, click **OK** to confirm that you want to remove the clock.

## 8.3. Enabling or disabling the clock

Use this option only if you cannot import valid certificates or if the logs are showing that the DS/NTP transaction has failed due to the TSS claiming that valid certificates have expired or are not yet valid.

1. Log in with the Network Manager role.
2. Navigate to **TSA Management > Clock Management.**
3. To enable or disable the clock, click **Enable** or **Disable** as appropriate, then click **Set Clock** to enable the TSS to synchronize the time to the host operating system clock. Ensure that the host clock is accurate.

   To accurately set the time, the clock server and the time zone must be correct. Get the time close enough so that an audit can confirm or fix the time.

## 8.4. Viewing card set lists

It is often necessary to obtain information from card sets. For security reasons, card sets usually do not include identification marks.

1. Log in with the Security Officer role.
2. Navigate to **Card Set Management > List Card Sets**.

   The following details are displayed.

| Object | Description |
|---|---|
| Name | This is the name the card set was given when it was created. |
| K of N | This shows the number of Operator Cards that you want to require in order to re-create a key (*K*) and the number of the total number of cards (*N*). |
| Persistent | This shows whether or not a card set is persistent. By default, the TSS creates non-persistent card sets, which means that keys protected by this card set become unavailable when the last card is removed. |
| Timeout | The time-out is the length of time that an Operator Card from the set can remain effective when inserted in the card reader. After the time-out is reached, the card must be re-inserted before it can be used. You cannot set a time-out value greater than a year (that is, 31622400 seconds). |

## 8.5. Viewing or changing card sets

The **View/Change** menu option enables you to examine cards inserted into a TSS from the same Security World as the TSS on which they were created. To change a card pass phrase, you require both the card and the old pass phrase.

1. Log in as a Security Officer.
2. Navigate to **Card Set Management > View/Change**.

   The **Card Utilities** dialog opens.

3. Click **Set Pass Phrase** to change the pass phrase of a card that has been inserted.

   For information on pass phrases, see Using pass phrases.

4. Optionally, click **Erase** to erase all the data of a card that has been inserted.

## 8.6. Viewing TSA certificate information

Use this menu option to view details of a TSA certificate.

1. Log in with the Security Officer role.
2. Navigate to **TSA Management > Certification Status**. The **TSA Certification Status** dialog opens, displaying all the TSAs that exist in the TSS.
3. Select the TSA you would like to work with, then click **Cert Info**. The **TSA Certificate Information** dialog displays the following information for the selected TSA:

| Field | Description |
|---|---|
| TSA Key Status | A green, amber, or red light here informs you of the key status. |
| Export | You can export (download) the certificate in base64 format by clicking here. |
| Version | Refers to the version of ANSI X.509 that defines the certificate syntax. The TSS uses the V3 version, standard in the industry. |
| Serial Number | A unique number assigned by the Certificate Authority that issues the certificate. It is simply a way to uniquely identify a specific certificate issued by a particular CA. |
| Signature algorithm | The signature algorithm used by the CA. |
| Issued by | The issuer of the certificate. |
| Issued to | The entity to whom the certificate is issued. |
| Valid from | Beginning date that the certificate is valid. |
| Valid to | Date that ends the certificate's validity. All dates in the TSS are displayed in the format **yyyy/mm/dd** and all times are displayed as **hh:mm:ss**. |
| Public key type | Displays the certificate's public key type. |
| Public key | This field is a long string of characters. Click on **Public Key** to see the key in its usual block format in the field here. Click any of the tags, and a pop-up window opens with the information presented in block format. |

## 8.7. Viewing Time Attribute Certificate (TAC) information

When the TSS is successfully audited, it gets a new TAC. The new TAC overlaps the previous TAC under which your TSS was issuing time-stamps. If you have an operational TAC and an audit fails because of time drift, the audit produces a non-operational TAC that overlaps the previous TAC. In such a case, the TSS is no longer able to issue time-stamps until a

successful audit is completed.

1. Log in with either the Network Manager or the Security Officer role.
2. Navigate to **TSA Management > Operational Status**.

   The **TSA Operational Status** dialog opens, displaying all the TSAs that exist in the TSS.

3. Select the required TSA, then click **Details**.

   The Operational Status for the selected TSA is displayed.

4. Click **TAC Info**.

   The following information is displayed:

| Field | Description |
|---|---|
| TAC Status | A green, amber, or red light informs you of the status. |
| Initiate Audit | When you click this button, an audit request is sent to the Upper Clock indicated in your network setup, as described in Adding an upper clock. The display will not automatically refresh when the audit is received. Click the menu item again to refresh. |
| Delay | The go around time, or round-trip communication delay between the TSS and the audit clock. |
| Offset | Tells you how much your clock differs from the clock that audited it (an NMI-based Trusted Master Clock, using UTC). |
| Max Delay | The maximum allowable network delay to receive a successful audit. |
| Max Offset | The maximum time offset allowed to receive a successful audit. This is measured in seconds. |
| Valid From | Tells you the date the TAC became valid. All dates in the TSS are displayed in the format **yyyy/mm/dd**, and all times as **hh/mm/ss**. |
| Valid To | Tells you the date the TAC will become invalid. |
| Leap Event | Any scheduled leap second event is noted here. |
| Timing Policy OID | Refers to the timing policy statement of the upper clock which issued this TAC. |

## 8.8. Viewing time-stamps issued

The **Status** menu option enables you to view all the time-stamps that have been issued.

1. Login with the Network Manager role.
2. Navigate to **TSA Management > Time Stamps Issued**.

   The **TSA Time-Stamps Issued** dialog opens, listing all the TSAs that exist in the TSS.

3. Select the TSA that you would like to work with, then click **Details**.

   The **Time-stamps** dialog opens, displaying the following time-stamp statistics:

| Area / button | Description |
|---|---|
| Total | Displays in total and by percentage the time-stamps requested, granted, and rejected. |
| Under Current TAC | Displays time-stamps requested, granted, and rejected under the current operational TAC. |
| Refresh | Click this button to display the latest statistics. |

## 8.9. Viewing the uptime

The **Status** menu enables you to know when the DSE200 service was last restarted.

1. Log in with the Network Manager role.
2. Navigate to **Server Management > Uptime.**

   The details of the TSS, UTC, the current time of the browser, and time zone settings are displayed.

## 8.10. Viewing or resetting Administrator and Board logs

1. Log in with either the Security Officer or the Network Manager role.
2. Navigate to **Logging** and select one of the following:
   - **Administrator Log**: records all TSS activities done within the Security Officer and Network Manager user interfaces.
   - **Board Log**: contains information about the internal state of the time-stamp server (for example, whether time-stamping and logging are enabled or disabled, records of audit requests generated by the expiration of a TAC, receipt of a TAC, and clock calibration).

Depending on the option that you select, either the **Administrator Log** or the **Board Log** dia log opens.

To view the selected log:

1. Click **Show Log** to display the log dialog.
2. Choose the parameters for viewing the log, then click **Display**.

   The log records are displayed with the most recent entries at the bottom. For more on Board log errors and alerts, see Error messages and alerts.

To reset the selected log:

1. Click **Reset** to display the reset log dialog.
2. Click **Start** to rotate the log.

## 8.11. Recording TACs and time-stamps

If you wish to record the TACs that are received and the time-stamps that are issued, set the environment variable `TSS_LOGTIMESTAMPS=1`. If you set this environment variable:

- Each TAC is recorded in an individual file in the `%NFAST_HOME%\dse200\User-Files\tac_log\` directory. There are no configuration options.
- All the issued time-stamps are saved in the `%NFAST_HOME%\dse200\UserFiles\tst_log\` directory.

This feature is optional and is disabled by default.

## 8.12. Viewing user login statistics

The **User Login Statistics** menu option enables you to view the login statistics by user type. Server statistics displays the consolidated login details for all user types.

1. Login with either the Network Manager or Security Officer role.
2. Navigate to **Logging > User Login Info**.

   The **User Login Statistics** dialog opens, displaying login information by user and for the server as a whole.

3. To view login statistics by user type:

   a. Select a user type from the **Statistics For** drop-down menu.

   b. Click the **Details** button next to the **Statistics For** drop-down menu.

   The login statistics for the selected user type are displayed.

4. To view the consolidated login information for all user types, click the **Details** button next to **Server Statistics**.

All dates in the TSS are displayed in the format **yyyy/mm/dd**, and all times as **hh/mm/ss**.

## 8.13. Viewing the log archive

The archive stores the old Admin and Board logs.

1. Login with either the Network Manager or Security Officer role.
2. Navigate to **Logging > Archive**.

   The **Log Archive** dialog opens, displaying links to the **Admin** and **Board** logs.

3. Click the required link.
4. In the **Log Archive** dialog, select the required options and click **Display**.

   The log records are displayed, with the most recent entries at the bottom.

## 8.14. Adding a user

User roles are defined as follows:

- Security Officer: authorized to perform key and certificate management. The Security Officer manages keys, certificates, and the log file, but is not responsible for the day-to-day management or operational tasks.
- Network Manager: authorized to manage time-stamping and auditing. The Network Manager is responsible for managing and operating the TSS on a day-to-day basis. The Network Manager can also enable or disable time-stamping and auditing.

Any users you add must be assigned one of these roles. There may be more than one Security Officer and more than one Network Manager.

1. Log in with the Security Officer role.
2. Navigate to **User Management > Add User**. The **Add User** dialog opens.
3. Enter the following information:

| Field | Enter / Select |
|---|---|
| User Name | The user's first name or nickname. |

| Field | Enter / Select |
|---|---|
| User Role | The user's role: Security Officer or Network Manager. |
| Real Name | The user's real name. |
| E-mail | The user's e-mail address. This is just for display purposes, and can be the same e-mail address as in Notify E-Mail, described below. |
| Notify E-mail | The e-mail address to which notices from the board log are sent. |
| Notify E-Mail From | The e-mail address from which the user receives Notify e-mails. |
| Notify E-Mail Subject | If left blank, the default value (**%<N DSE200>%*E Notification**) is used, where N - expands to the hostname of the TSS and E expands to "Error", "Alert", or "Error/Alert" based on the log messages. |
| Notify SMTP | The SMTP server handling such notices. This is the e-mail server (example: `smtp.<servername>.com`) that the TSS uses to send e-mail notifications. |
| Password | A password for the user. The maximum length of a password is 128 characters. |
| Verify password | The password again to confirm it. |

4. Click **Add User**.

   If the **Modify User** screen opens (see next section), then the new user has been added.

## 8.15. Modifying or deleting users

1. Log in with the Security Officer role.
2. Navigate to **User Management > Modify/Delete Users**.
3. Enter their identifying data, then click the appropriate button to change its status: **Update**, **Delete**, or **Change Password**.

   A user account can be locked or unlocked via the **Account Locked** checkbox.

4. Click **OK** to complete the action to change the field.
5. When prompted, enter and re-enter the new password.

Use the **Prev** and **Next** buttons to go to the previous or next user's profile to be modified.

## 8.16. Modifying user information

1. Log in with the Network Manager role.
2. Navigate to **User Management > Modify Users**.
3. Change the information as required.
4. Optionally, click **Change Password** to change your password.
5. After you have made the required changes, click **OK** to save the changes.

## 8.17. Restarting the service

You may have to restart the service at times, for example after you have installed a new TLS certificate. As a Windows administrator, restart the DSE200 service via the standard Windows Services facility.

# 9. Upgrading an existing deployment

Before upgrading an existing TSOP installation, we recommend:

- Following the advice contained within the Backup and recovery section of the *TSOP Administrator Guide*.
- Reading the *TSOP Release Notes* in their entirety and acknowledging the implications of performing a firmware upgrade.

## 9.1. Upgrading TSOP

The following instructions apply when upgrading from a TSS 6.20.00 installation or later.

1. Uninstall any previous TSOP installation through Program and Features.
2. Restart the machine.
3. Install TSOP, following the instructions and observing any installation prerequisites. See Installing the Time Stamp Option Pack.

## 9.2. Upgrading Security World Software

1. Stop the DSE200 service via Microsoft's Windows Services.
2. Uninstall the Security World Software, following the *Uninstalling Windows Software* section of the *nShield Solo Installation Guide*.

   Make a note of the location where the existing Security World Soft ware is installed.

3. Restart the machine.
4. Install the Security World Software, following the *Installing the software* section of the *nShield Solo Installation Guide*.

   Ensure that you install in the same location as the previous Security World Software installation.

5. Restart the DSE200 service via Microsoft's Windows Services.

## 9.3. Upgrading Firmware

1. Stop the DSE200 service via Microsoft's Windows Services.

2. Follow the *Upgrading firmware* section of the *nShield Solo User Guide for Windows*.
3. Restart the DSE200 service via Microsoft's Windows Services.
4. It is then necessary to:
    a. Join an existing Security World, see the *Joining an existing Security World* section of the *TSOP Administrator Guide*.
    b. Delete the following files from `%NFAST_KMDATA%\local`: `distore`, `tsastore`, `localaudit` and `key_dseapp_tssinteg`.
    c. Perform SEE delegation, see the *Security World: SEE delegation* section of the *TSOP Administrator Guide*.
    d. Recreate the existing TSA configuration, restoring any TSA keys which were protected by an operator card set, see *Restoring a TSA key* section of the *TSOP Administrator Guide*.

# 10. TSS FAQs and solutions

This section addresses the frequently-asked questions about the TSS and provides solutions to common problems. The following answers to frequently-asked questions are in alphabetical order by topic.

## 10.1. Digital certificates

1. *What are the roles and respective keys and certificates of the TSS?*

   See this table for the answers:

| Cert Type | Role |
|---|---|
| TSA | Sign time-stamps and authenticate DS/NTP communications |
| TSA Root Chain | Certificate fulfillment with CA |
| DI Root Chain | Authentication of devices for DS/NTP |
| HTTPS | Enable `https://` communication for browser-based communication |

2. *In the TSS, where are the keys and certificates?*

   See this table for the answer:

| Product / Cert Type | Private Key |
|---|---|
| TSA | nShield Solo |
| TSA Root Chain | nShield Solo |
| DI Root Chain | nShield Solo |
| HTTPS | Host Drive |

3. *How are these keys and certificates generated?*

   See this table for the answers:

| Product / Cert Type | How Keys Are Generated |
|---|---|
| TSA | Keys and certificate request generated by TSS Security Officer. Certificate acquired from public or private CA. |
| TSA Root Chain | Loaded by TSS Security Officer |
| DI Root Chain | Loaded by TSS Security Officer |
| HTTPS | The default test TLS certificate must be replaced before TSS deployment. |

4. *How are certificates obtained with the TSS?*

   Administrators can use the local, browser-based administration system of the TSS to obtain TSA certificates. In local certification, administrators generate a key pair and a certificate request. The PKCS #10 request can then be submitted to the appropriate CA for certificate fulfillment.

## 10.2. Standards

1. *What Time Stamp Protocol does the TSS support?*

   The TSS supports the IETF Time Stamp Protocol (RFC 3161). Time-stamps are requested by means of either the Transmission Control Protocol (TCP) or Hypertext Transfer Protocol (HTTP), as described by RFC 3161. The extensions to RFC 3161 described by RFC 5816 are also supported.

2. *A European standard (ETSI) says that a recommended time-stamp protocol (RFC 3161) should be the http protocol. Do you support the http protocol for time-stamping?*

   The TSS includes support for the RFC3161 Time-Stamp Protocol through HTTP (Hypertext Transfer Protocol).

## 10.3. Synchronization

1. *Is the time that is used for Windows synchronized to the time-stamp token's time?*

   No, the time is not synchronized. The time for Windows is from the Windows system clock.

## 10.4. TSS

1. *In the customer's installation, can the customer integrate the TSS in the firewall? Can the TSS operate correctly in the firewall? Specifically, can the DS/NTP and DSMP be transferred by NAT (Network Address Translation)?*

   Yes, you can use firewalls/NAT. Customers presently use TSMCs to audit their TSS devices through a firewall and VPN.

2. *When the user employs the TSS with their application server, is a secure connection required between the TSS and the application server?*

   When issuing time-stamps there is no need for a secure connection between the application server and the TSS. The PKIX TSP (time-stamp protocol) includes all the security that should be needed. The data returned from the TSS is signed and includes the origi

nal hash and nonce. Security is part of the reason that the time-stamp request includes only a hash of the original data. Someone watching a session would see only hash infor mation, which is safe. If you need security for other reasons such as identifying the user, billing, and so forth, this security should be implemented on an application server that has a direct unsecured connection to one or more TSSs.

3. *How does the TSS acquire time after a restart?*

   After a restart, the TSS does not have an operational Time Attribute Certificate and can not provide time-stamps. The TSS initially contacts the assigned Upper Clock and requests an audit. The first Upper Clock audit may fail, depending on the clock drift rate, and issue a non-operational TAC. The TSS corrects its clock to this first audit. The TSS then does another audit request which is normally successful and receives an oper ational TAC from the Upper Clock. The Upper Clock then continues to audit the TSS per the configured interval.

# 11. Local Audit / NTP service

If you have selected Local Audit as the method by which the TSS secure clock is to be audited, this requires a running Network Time Protocol (NTP) service which has been config ured to use an NTP server on the local network. Configuration of an NTP service requires a Windows Administrator to log in to the System Console.

The NTP service must be set up for Automatic start-up. You do this using the Windows Services applet.

We advise checking that the default configuration file is compliant with your internal security policy.

The Local Audit setting provides time that is traceable only to the TSS host PC clock. If you use the Local Audit setting, we recommend also using a good security policy with regard to the physical security of the TSS and the network connection to the NTP server.

To enable the NTP Service on the TSS so that you can use Local Audit:

1. In the TSS Web interface, configure one or more TSAs to use Local Audit. (See Configuring a TSA for instructions.)

   If you have any TSA configured to use an Upper Clock, it must not use port 9124 for DS/NTP: this would cause a conflict with the NTP services.

2. Restart the DSE200 Service.
3. Once the NTP service synchronizes the local PC clock to the configured NTP server, the DSE200 service will audit the SEE application using the local PC time. You can look at the `board.log` or the TAC Info page (see Viewing Time Attribute Certificate (TAC) information) to see when this happens. Use `ntpq` to look at the status of the NTP service synchronization.

# 12. Time Stamp Tokens (TST) TAC encoding and binding

Supporting RFC3852 and RFC3126 requires optional encoding formats for RFC3161 TST. This relates to how the TAC is stored in the TST and how the TAC is cryptographically bound to the signature.

## 12.1. CertificateChoices1 with ESSCertID/ESSCertIDv2 (compatibility mode)

This is the current implementation. The TAC is encoded in the CHOICE [1] field in the CertificateChoices and the hash of the TAC is stored in the ESSCertID/ESSCertIDv2.

## 12.2. CertificateChoices2 with ESSCertID/ESSCertIDv2 (RFC3369 & 3852)

This option puts the TAC into the CHOICE [2] field in the CertificateChoices and sets the CMS version of the time-stamp token to 4 (because a V2 attribute certificate is present).

Adobe Acrobat time-stamping support rejects CMS V4 as a bad version number. If you are using Adobe Acrobat time-stamping, we recommend continuing to use an older option that is Acrobat-compatible until a fix from Adobe is made available.

## 12.3. Signer Attribute (RFC3126 & ETSI)

This option puts the entire TAC into a signed attribute. In this case, the hash of the TAC is not included in the ESSCertID/ESSCertIDv2 because it would be redundant and RFC3126 requires it not to be present. This option also adds the SigningTime signed attribute (which is redundant but required by the RFC) and the SignaturePolicyId signed attribute. The policy is NULL because a time-stamp token must already include a PolicyID in the TSTInfo.

# 13. TST options

TST options determine or affect the type of data stored in a TST. This appendix describes the options you can choose when configuring a TSA.

## 13.1. Use ESSCertIDv2 (RFC5816)

Enabled by default, ESSCertIDv2, as defined by RFC5816, cryptographically binds the TAC to the signature (using SHA-256). See Time Stamp Tokens (TST) TAC encoding and binding for more information about ESSCertID/ESSCertIDv2.

## 13.2. Exclude TAC from certificate list

Select this option to disable the inclusion of the TAC in the time-stamp token certificate list. You can use this option to support time-stamp client software that cannot decode the TAC. This option is required if you must support Oracle's `jarsigner`. Oracle `jarsigner` cannot decode the TAC and cannot support time-stamp tokens that include the TAC in the certificate list. When a time-stamp client requests certificates to be included in the time-stamp token, the TSS by default includes the TSA certificate and the TAC. The Exclude TAC from certificate list option enables you to exclude the TAC from the certificate list.

This option only excludes the TAC from the certificate list in the time-stamp token. The TST TAC Encoding and Binding option SignerAttribute (RFC3126 & ETSI) does not encode the TAC in the certificate list. The SignerAttribute option encodes the TAC in a signed attribute. Therefore if you use the SignerAttribute option, the Exclude TAC option has no effect.

## 13.3. Use rsaEncryption OID Fix

The rsaEncryption algorithm identifier is used to identify RSA (PKCS #1 v1.5) signature values regardless of the message digest algorithm employed. CMS implementations that include the RSA (PKCS #1 v1.5) signature algorithm must support the rsaEncryption signature value algorithm identifier. CMS implementations may support RSA (PKCS #1 v1.5) signature value algorithm identifiers that specify both the RSA (PKCS #1 v1.5) signature algorithm and the message digest algorithm. Earlier versions of the TSS always used signature value algorithm identifiers that specified both the RSA (PKCS #1 v1.5) signature algorithm and the message digest algorithm. However, this may not be compatible with some applications that implement the RFC. All applications must support the rsaEncryption algorithm

identifier and may optionally support the hash specific algorithm identifier. This option enables the Security Officer to use the more compatible rsaEncryption option instead of the old TSS behavior.

If you do not enable this option for Oracle `jarsigner`, it fails to validate the signature in the TST.

This option is incompatible with earlier versions (before V5.0) of the TSS SDK.

# 14. Error messages and alerts

This section lists the error messages and alerts you might encounter while using the user interface. The default subject of e-mail notifications is: `%N DSE %E Notification`

In this subject line, `%N` is the host name of the TSS that sent the message, and `%E` is one of "Error", "Alert", or "Error/Alert".

If your TSS does not recognize the certificate from the Upper Clock, the body of the message sent contains lines of the form:

```
ERROR: 05-23-02 22:27.56 > DSNTP: Failure to Validate UC Certificate.
ERROR: 05-23-02 22:27.56 > DSNTP: Failed to process UC's Cert.
```

Such errors can be fixed if the Security Officer adds the CA certificate chain for the Upper Clock certificate to the Upper Clock Cert Store. You can get the CA certificate chain from your audit service provider.

If an audit session fails due to a general communication error, the body of the message sent contains a line of the form:

```
ERROR: 05-02-02 14:01.07 > DSNTP: COMMUNICATION ERROR Failed to read from socket.
```

Such an error is usually the result of a temporary network failure, possibly due to heavy traffic on the Internet or on your local area network. The best thing to do is initiate a new audit from the TSS, or your service provider may initiate a new audit from the Upper Clock. If this is a persistent problem, work with your audit services provider to identify the network issue.

When you restart the TSS, the body of the message sent contains lines of the form:

```
ALERT: 05-24-02 19:10.08 > Logging Service has been Enabled.
ALERT: 05-24-02 19:10.38 > TSA Startup: Version: 1.1, Build 1.0
ALERT: 05-24-02 19:10.38 > Logging Service has been Enabled.
ALERT: 05-24-02 19:10.39 > DSNTP: Sent initiate request to upper clock (172.16.33.1:318)
```

After the first audit and restart, the body of the message sent contains lines of the form:

```
ALERT: 05-24-02 19:11.16 > Non-Operational TAC has been received: offset = -19.792432, ntpTime = 19:11:16 -
May/24/02, expiration = -19:11:16 - May/24/02, leapAction = 0, leapTime = 0.000000, delay = 0.022242
ALERT: 05-24-02 19:11.17 > DSNTP: Sent initiate request to upper clock (172.16.33.1:318)
```

The first audit after a restart usually results in a non-operational TAC because the offset of the clock is large.

After an audit, the body of the message sent contains a line of the form:

```
ALERT: 05-24-02 19:11.35 > Operational TAC has been received: offset = 0.002194, ntpTime = 19:11:35 - May/24/02,
expiration =
19:11:35 - May/31/02, leapAction = 0, leapTime = 0.000000, delay = 0.022087
```

When the Network Manager initiates an audit, the body of the message sent contains a line of the form:

```
ALERT: 05-15-02 20:54.38 > DSNTP: Sent initiate request to upper clock (172.16.33.1:318)
```

When the Network Manager make changes under Network Configuration, the body of the message sent contains lines of the form:

```
ALERT: 04-25-02 14:32.01 > TSA: Upper Clock configuration changed.
172.16.33.1:318
ALERT: 04-25-02 14:32.01 > TSA: Lower Clock Public IP.
172.27.20.4:123
ALERT: 04-25-02 14:32.02 > DSNTP: Sent initiate request to upper clock (172.16.33.1:318)
```

If the Network Manager disables the clock, the body of the message sent contains lines of the form:

```
ALERT: 04-10-02 18:58.06 > Clock Service has been Disabled.
ALERT: 04-10-02 18:58.06 > Operational TAC has been Invalidated!
```

If the Network Manager enables the clock, the body of the message sent contains a line of the form:

```
ALERT: 04-10-02 18:58.32 > Clock Service has been Enabled.
```

If the Network Manager disables the time-stamping, the body of the message sent contains a line of the form:

```
ALERT: 04-10-02 18:56.39 > Time Stamping Service has been Disabled.
```

If the Network Manager enables the time-stamping, the body of the message sent contains a line of the form:

```
ALERT: 04-10-02 18:57.32 > Time Stamping Service has been Enabled.
```

When the Security Officer initiates TSA certificate request, the body of the message sent contains a line of the form:

```
ALERT: 05-23-02 18:38.11 > TSA Certificate generation initiated for DN: C=US\S=Massachusetts\L=Lexington\O=J
Scott John\OU=Finance\CN=CompanyABC
```

When the Security Officer fulfills a TSA certificate request, the body of the message sent contains lines of the form:

```
ALERT: 05-21-02 20:14.57 > Audit Service has been Disabled.
ALERT: 05-21-02 20:14.57 > Time Stamping Service has been Disabled.
ALERT: 05-21-02 20:14.57 > Clock Service has been Disabled.
ALERT: 05-21-02 20:14.57 > TSA Certificate fulfillment successful for DN: C=US\O=Datum, Inc\OU=Datum Trusted Time
StampServer SN:90D00217\CN=<host name>
ALERT: 05-21-02 20:14.57 > Clock Service has been Enabled.
ALERT: 05-21-02 20:14.57 > Time Stamping Service has been Enabled.
ALERT: 05-21-02 20:14.57 > Audit Service has been Enabled.
```

Audit alerts generate e-mails in which the body of the message sent contains lines of the form:

```
ALERT: Audit Service has been Disabled.
ALERT: Audit Service has been Enabled.
```

Certificate alert/errors generate e-mails in which the body of the message sent contains lines of the form:

```
ALERT: "Certificate has EXPIRED (<expiredDN>)"
```

Certificate and key management alerts/errors generate e-mails in which the body of the message sent contains lines of the form:

```
ALERT: TSA Certificate fulfillment successful for DN: <name>
ALERT: The TSA Certificate has expired.
ALERT: The TSA Certificate expires in <n> hours, <n> minutes.
ALERT: The TSA Certificate expires in <n> days, <n> hours.
ALERT: The TSA Certificate expires in <n> weeks, <n> days.
ALERT: Certificate added to TSA store.
DN: <name>
ALERT: Certificate added to DI store.
DN: <name>
ALERT: Certificate removed from TSA store.
DN: <name>
ALERT: Certificate removed from DI store.
DN: <name>
ALERT: TSA Certificate generation initiated for DN: <name>
ALERT: TSA Certificate generation has been canceled.
ERROR: CreateCertRequest: DecodeCRI failed <n>
ERROR: CreateCertRequest: GenerateKeyPair failed <n>
ERROR: CreateCertRequest: Unexpected private key cert mech <n>
ERROR: CreateCertRequest: GetHKM0 failed <n>
ERROR: CreateCertRequest: MakeModuleBlob failed <n>
ERROR: CreateCertRequest: ExportRSAPubKey failed <n>
ERROR: CreateCertRequest: EncodeRSAPublicKey failed <n>
ERROR: CreateCertRequest: ExportDSAPubKey failed <n>
ERROR: CreateCertRequest: EncodeDSAPublicKey failed <n>
```

```
ERROR: CreateCertRequest: EncodeCertificationRequestInfo failed <n>
ERROR: CreateCertRequest: Sign failed <n>
ERROR: CreateCertRequest: BEncCertificationRequest failed
ERROR: FulfillCertRequest: Failed to find stored key!
ERROR: FulfillCertRequest: DecodeCertificate failed <n>
ERROR: FulfillCertRequest: LoadModuleBlob failed <n>
ERROR: FulfillCertRequest: LoadRSAPubKey failed: <n>
ERROR: FulfillCertRequest: Sign failed <n>
ERROR: FulfillCertRequest: Public key mismatch.
ERROR: FulfillCertRequest: LoadDSAPubKey failed: <n>
ERROR: FulfillCertRequest: Sign failed <n>
ERROR: FulfillCertRequest: Public key mismatch.
ERROR: EncryptKeyStore: GetHKM0 failed <n>
ERROR: EncryptKeyStore: Encrypt failed <n>
ERROR: VerifyCertSignature, LoadRSAPubKey failed: <n>
ERROR: VerifyCertSignature, LoadDSAPubKey failed: <n>
ERROR: Unable to encode issuer or subject.
ERROR: EE Certificate failed validity time check!
ERROR: Failed to locate CA cert subject DN!
ERROR: CA Certificate failed validity time check!
ERROR: EE Cert validity time is not constrained by the CA Cent's validity time!
ERROR: EE Certificate signature invalid!
```

Clock alerts generate e-mails in which the body of the message sent contains lines of the form:

```
ALERT: Clock Service has been Enabled.
ALERT: Clock Service has been Disabled.
```

DS/NTP errors generate e-mails in which the body of the message sent contains lines of the form:

```
ERROR: DSNTP: CreateHello, Cryptographic random generation failed, extended error: <n>
ERROR: DSNTP: ProcessUCHello, Hello message is invalid
ERROR: DSNTP: ProcessUCHello, Upper Clock name is too large
ERROR: DSNTP: ProcessUCCertificate, CreateCert returns <n>
ERROR: DSNTP: CreateLCKeyExchange, DH Key Pair generation failed: <n>
ERROR: DSNTP: CreateLCKeyExchange, Export DH pub key failed: <n>
ERROR: DSNTP: CreateLCKeyExchange, Signing function failed: <n>
ERROR: DSNTP: ProcessUCKeyExchange, invalid DSNTP message received
ERROR: DSNTP: ProcessUCKeyExchange, LoadRSAPubKey failed: <n>
ERROR: DSNTP: ProcessUCKeyExchange, LoadDSAPubKey failed: <n>
ERROR: DSNTP: ProcessUCKeyExchange, Unknown key type
ERROR: DSNTP: ProcessUCKeyExchange, DH Key derivation failed: <n>
ERROR: DSNTP: ProcessUCKeyExchange, signature was invalid: <n>
ERROR: DSNTP: AuthenticateFrame, MAC size is invalid
ERROR: DSNTP: AuthenticateFrame, HMAC verification failed, error: <n>
ERROR: DSNTP: ProcessHandshake, handshake length is invalid.
ERROR: DSNTP: ProcessHandshake, not expecting UC Certificate.
ERROR: DSNTP: ProcessHandshake, DSNTP_INVALID_MESSAGE
ERROR: DSNTP: ProcessHandshake, ProcessUCCertificate ERROR
ERROR: DSNTP: Upper clock, CertificateACK message was invalid
ERROR: DSNTP: ProcessHandshake, CreateLCKeyExchange ERROR
ERROR: DSNTP: Upper clock, CertificateACK processing failed
ERROR: DSNTP: Received invalid NTP frame, length < 48 bytes.
ERROR: DSNTP: VerifyTACSignature, signature invalid: <n>
ERROR: DSNTP: VerifyTACSignature, certInfo encode failed
ERROR: DSNTP: TAC Invalid, AttributeCertificate decode failed.
ERROR: DSNTP: Received TAC in unauthenticated state.
ERROR: DSNTP: GetTransportData, HMAC sign failed, <n>
```

Log alerts generate e-mails in which the body of the message sent contains lines of the form:

```
ALERT: Logging Service has been Enabled.
ALERT: Logging Service has been Disabled.
```

Time-stamping alerts/errors generate e-mails in which the body of the message sent contains lines of the form:

```
ALERT: Operational TAC has been Invalidated.
ALERT: Time Stamping Service has been Enabled.
ALERT: Time Stamping Service has been Disabled.
ERROR: IssueTimestamp: GetTime failed <n>
ERROR: IssueTimestamp: Sign failed: <n>
```

# 15. Security Guidance

The key material used by the TSS (e.g. the TSA signing key) is protected by its internal nShield HSM that is initialized into your Security World. However, as with all secure systems, administrators must remain diligent when it comes to who uses the system, see the security-related recommendations below. The network traffic between the TSS and its different communicating entities is protected by different mechanisms:

- The confidentiality and integrity of network traffic between the TSS and TSMC is protected using the DS/NTP protocol, where PKI certificates are used for mutual authentication, before the secure channel is created.
- The confidentiality and integrity of network traffic between the TSS and computer hosting the browser of the Security Office or Network Manager, is protected by the HTTPS protocol, where PKI certificates are used for server authentication.
- There is no protection explicitly provided for the confidentiality or integrity of the network traffic passed between the TSS and the user (who is using an RFC3161 based interface to request a Time Stamp Token (TST) for the TSS). The TST, produced by the TSS, contains implicit integrity protection, provided by the TSA signature (which should be verified by the user, as indicated in RFC3161). The RFC3161 protocol explicitly excludes authentication of the user and does not include provision for authentication of the server (i.e. TSS).
- When Local Audit is selected, the NTP server is assumed to be running on a trusted local network.

Entrust make the following security related recommendations when operating the TSS:

- The network environment of TSS should be suitably protected to maintain its availability (e.g. through use of firewalls, intrusion detection/prevention systems).
- Standard virus prevention and detection measures should be taken on the platform hosting the TSS.
- All available security patches should be applied in a timely manner to the platform host ing the TSS.
- Only authorized system administrators should have access to the TSS and only essential trusted software should be run on the platform hosting the TSS.
- Only authorized and trusted individual should be assigned the roles of Security Officer and Network Manager, and it is assumed that they will not perform any malicious opera tions that would degrade the security of the TSS.
- Make sure log levels are set appropriately for your environment. More verbose log levels may expose information that is useful for auditing who uses TSS, but the log information will also contain which administrative and user operations have been per-

formed. While this may be useful for diagnostics, be aware that this log information could be considered sensitive and should be suitably protected when stored.

- The inactivity timeouts set for HTTPS sessions to both the Security Officer or Network Manager should be set to appropriate values that manage both the potential risk of use by an unauthorized individual and usability.
- Make sure that all certificates created have a chain of trust to a trusted root certificate.
- The user must verify that TSA's Certificates have not been revoked before using the TST (as indicated in RFC3161) and the Security Officer should remove from service any TSA Certificates that have been compromised/revoked.
- TSA certificates and keys should have appropriate key lengths and cryptoperiods, that reflect the sensitivity of the artefacts for which the TST are being created.
- On first usage of the TSS, operators should make sure that they replace both the Security Officer/Network Manager default usernames and passwords with unpredictable values, and for the passwords, values that contain sufficient randomness (as dictated by the security policy of the operator).