

nShield Security World

nShield v13.9.6 Utilities Reference

03 September 2026

© Entrust Corporation. All rights reserved.

Table of Contents

1. Summary of utilities . . . 1
2. anonkneti . . . 7
    2.1. anonkneti examples . . . 8
        2.1.1. Example 1: Run `anonkneti` against an HSM to check availability . . . 8
        2.1.2. Example 2: Run `anonkneti` against localhost to obtain the softkneti hash . . . 8
        2.1.3. Example 3: Compare the IP address of a network-attached HSM from the front panel and the anonkneti response . . . 8
3. appliance-cli . . . 9
    3.1. setrtc . . . 10
4. bondcfg . . . 11
5. bondlink . . . 12
6. bulkerase . . . 13
7. cardpp . . . 14
8. cef-audit-verify . . . 15
9. cfg-mkdefault . . . 16
10. cfg-dynamicslots . . . 17
11. cfg-mkcardlist . . . 18
12. cfg-pushnethsm . . . 19
13. cfg-pushntp . . . 21
    13.1. Example . . . 22
14. cfg-remoteslots . . . 23
    14.1. cfg-remoteslot ACTIONS . . . 23
    14.2. cfg-remoteslot OPTIONS . . . 23
15. cfg-reread . . . 24
16. checkmod . . . 25
17. chkserv . . . 26
18. ckaesgen . . . 27
19. ckariagen . . . 29
20. ckcerttool . . . 30
    20.1. Import a cardset- or softcard-protected certificate . . . 31
    20.2. Import module-only (no passphrase or cardset name) . . . 31
    20.3. Import a trusted public certificate with no corresponding private key . . . 31
21. ckcmac-ctr . . . 32
22. ckcrypt . . . 33
23. ckdes3gen . . . 34
24. ck_ecedwards_gen . . . 35
25. ckecies . . . 36

26. ck_ecmontgomery_gen . . . 37
27. ckhyper . . . 38
28. ckcheckinst . . . 39
28.1. ckcheckinst output examples: Security World validity . . . 39
28.2. ckcheckinst output examples: invalid cards . . . 40
29. cknfkmid . . . 42
30. ckshahmac . . . 43
31. cksigtest . . . 44
32. ckinfo . . . 47
33. ckkeyloop . . . 48
34. cklist . . . 49
35. ckmechinfo . . . 50
36. ckmldsa . . . 51
37. ckrestrictkey . . . 53
38. ckrsagen . . . 54
39. ckslhdsa . . . 56
40. cksotool . . . 58
41. ck-xfer-fix . . . 59
42. config-auditlogging . . . 60
43. cpioc . . . 61
44. cngimport . . . 62
45. cnginstall32, cnginstall . . . 63
46. cnglist32, cnglist . . . 64
47. cngregister . . . 65
48. cngsoak, cngsoak64 . . . 66
49. config-serverstartup . . . 67
50. configure-csp-poolmode, configure-csp-poolmode64 . . . 68
51. createocs . . . 69
51.1. Restrictions on using createocs . . . 70
52. cryptest . . . 72
53. csadmin . . . 73
54. cspcheck, cspcheck64 . . . 74
55. cspimport, cspimport64 . . . 75
56. cspmigrate, cspmigrate64 . . . 76
57. cspnvfix, cspnvfix64 . . . 77
58. csptest, csptest64 . . . 78
59. csputils, csputils64 . . . 79
60. date . . . 80
61. des_kat . . . 81

62. display-pubkey . . . 82
63. dump-marshalled . . . 83
64. edit-world . . . 85
64.1. Prerequisites for using edit-world . . . 85
64.2. edit-world syntax . . . 85
64.3. edit-world NAME=VALUE syntax . . . 85
64.4. edit-world [OPTIONS] . . . 85
64.5. edit-world examples . . . 85
65. elftool . . . 87
66. enquiry . . . 88
66.1. enquiry output info . . . 88
66.2. Flag explanations . . . 90
66.2.1. Level one flags . . . 90
66.2.2. Level two flags . . . 90
66.2.3. Level three flags . . . 91
66.2.4. Level four flags . . . 91
66.2.5. Level six flags . . . 92
67. esn . . . 93
68. factorystate . . . 94
69. fet . . . 95
70. floodtest . . . 96
71. fwcheck . . . 98
72. fwversion . . . 99
73. gateway . . . 100
74. gateway6 . . . 101
75. generatekey . . . 102
76. getrtc . . . 104
77. hakever . . . 105
78. help . . . 106
79. hsc_codesafe . . . 107
79.1. Privileges and access requirements . . . 107
79.2. Options . . . 107
79.3. Positional arguments and interactive execution . . . 107
80. hsc_configurepoolmodule . . . 109
81. hsc_configureslots . . . 110
82. hsc_loadseemachine . . . 111
83. hsc_loadwarrants . . . 112
84. hsc_nethsmexports . . . 113
85. hsc_nethsmimports . . . 114

86. hsc_remotefilesystem . . . 115
87. hsc_serverremotecomms . . . 116
88. hsc_serversettings . . . 117
89. Administration of platform services (nShield 5 HSMs) . . . 118
89.1. hsmadmin . . . 118
89.1.1. hsmadmin factorystate . . . 119
89.1.2. hsmadmin status . . . 120
89.1.3. hsmadmin npkginfo . . . 120
89.1.4. hsmadmin upgrade . . . 121
89.1.5. hsmadmin reset . . . 122
89.1.6. hsmadmin enroll . . . 122
89.1.7. hsmadmin keys . . . 123
89.1.8. hsmadmin logs . . . 127
89.1.9. hsmadmin info . . . 130
89.1.10. hsmadmin esn . . . 131
89.1.11. hsmadmin settime . . . 132
89.1.12. hsmadmin gettime . . . 132
89.1.13. hsmadmin setminvsn . . . 133
89.1.14. hsmadmin getenvstats . . . 134
89.1.15. hsmadmin cs5 . . . 136
89.1.16. hsmadmin select . . . 137
90. hsmdiagnose . . . 139
91. initunit . . . 140
92. killrecov . . . 141
93. km-plode . . . 142
94. kmfile-dump . . . 143
95. kneti . . . 144
96. kptest . . . 145
97. keytst, keytst64 . . . 147
98. loadmache . . . 148
99. loadrom . . . 150
100. loadsee-setup . . . 151
101. logout . . . 153
102. sbin/logrotate-hardserver . . . 154
103. logs . . . 155
104. maintmode . . . 156
105. makecspyuserdata . . . 157
106. migrate-world . . . 158
106.1. Prerequisites for using migrate-world . . . 158

106.2. migrate-world modes . . . 158
106.3. Restrictions on using migrate-keys . . . 160
106.4. migrate-world to migrate keys using custom protection pairs . . . 161
106.5. Troubleshoot migrate-world . . . 162
107. mkaclx . . . 165
108. modstate . . . 167
109. ncdate . . . 168
110. ncperftest . . . 169
111. ncsvcdep . . . 172
112. ncversions . . . 173
113. ncthread-test . . . 174
114. netcfg . . . 175
115. netcfg6 . . . 176
116. netdiagnose . . . 177
117. netenable . . . 178
118. nethsmadmin . . . 179
119. nethsmenroll . . . 182
120. netlink . . . 184
121. new-world . . . 185
121.1. Prerequisites for using new-world . . . 185
121.2. new-world [ACTIONS] . . . 185
121.3. new-world [OPTIONS] . . . 186
121.4. new-world [FEATURE] syntax . . . 188
121.5. new-world [FEATURES] . . . 189
121.6. new-world examples . . . 193
122. nfcp . . . 195
123. nfdiag . . . 197
123.1. Include additional files for Support in the zip output of nfdiag . . . 197
123.2. Content of the text output of nfdiag . . . 198
124. nfkmattest . . . 199
125. nfkmcheck . . . 200
126. nfkminfo . . . 201
126.1. Front panel flags mapped to nfkminfo fields . . . 202
126.1.1. nfkminfo: information utility . . . 202
127. nfkmverify . . . 213
127.1. nfkmverify options . . . 213
127.2. Verify a migrated key . . . 214
128. nfloadmon . . . 215
129. nfls . . . 216

130. nfrm . . . 217
131. nfwarrant . . . 218
132. nopclearfail . . . 219
133. npkgtool . . . 221
134. nshieldaudit . . . 222
135. ntokenenroll . . . 223
136. nvram-backup . . . 224
137. nvram-sw . . . 225
138. openssl . . . 227
139. p11hyper . . . 228
140. passwd . . . 229
141. perfcheck . . . 230
141.1. perfcheck example command lines . . . 230
141.2. perfcheck syntax . . . 230
141.3. perfcheck tests . . . 233
141.4. How perfcheck calculates statistics . . . 236
142. ping . . . 238
143. pollbare . . . 239
144. postload-bsdlib . . . 240
145. postrocs . . . 241
146. ppmk . . . 242
147. preload . . . 244
147.1. Pattern matching in preload commands . . . 246
148. pubkey-find . . . 247
149. push . . . 249
150. raccmd . . . 250
151. racgui . . . 251
152. racs . . . 252
153. randchk . . . 253
154. reboot . . . 254
155. retrievewarrants . . . 255
156. rfs-setup . . . 256
157. rfs-sync . . . 258
158. rfsaddr . . . 260
159. rocs . . . 261
159.1. rocs interactive mode commands . . . 262
160. route . . . 266
161. route6 . . . 267
162. routing . . . 268

163. routing6 . . . 269
164. rserverperm . . . 270
165. rtc . . . 272
166. see-sock-serv, see-stdioe-serv, see-stdioesock-serv, see-stdoe-serv . . . 273
166.1. Error output from SEE machine with SEElib architecture . . . 275
167. setrtc . . . 276
168. sigtest . . . 277
169. slotinfo . . . 279
169.1. slotinfo output . . . 279
170. stattree . . . 281
170.1. Example outputs . . . 281
170.2. Node tags . . . 289
170.3. Statistics IDs . . . 290
170.4. ModuleDriverStats fields . . . 297
171. sworldcheck . . . 298
172. tamperlog . . . 299
173. tct2 . . . 300
173.1. Sign with tct2 . . . 302
173.2. Pack with tct2 . . . 302
173.3. Encrypt with tct2 . . . 303
174. trial . . . 304
175. uptime . . . 306
176. version . . . 307

1. Summary of utilities

- anonkneti
- appliance-cli
- bondcfg
- bondlink
- bulkerase
- cardpp
- cef-audit-verify
- cfg-dynamicslots
- cfg-mkcardlist
- cfg-mkdefault
- cfg-pushnethsm
- cfg-pushntp
- cfg-remoteslots
- cfg-reread
- checkmod
- chkserv
- ck_ecedwards_gen
- ck_ecmontgomery_gen
- ck-xfer-fix
- ckaesgen
- ckariagen
- ckcerttool
- ckcheckinst
- ckcmac-ctr
- ckcrypt
- ckdes3gen
- ckecies
- ckhyper
- `ckimportbackend`
  Use when instructed to do so by Support
- ckinfo
- ckkeyloop

- cklist
- ckmechinfo
- ckmldsa
- cknfkmid
- ckrestrictkey
- ckrsagen
- ckshahmac
- cksigtest
- ckslhdsa
- cksotool
- cngimport
- cnginstall32, cnginstall
- cnglist32, cnglist
- cngregister
- cngsoak, cngsoak64
- config-auditlogging
- config-serverstartup
- configure-csp-poolmode, configure-csp-poolmode64
- cpioc
- createocs
- cryptest
- csadmin
- cspcheck, cspcheck64
- cspimport, cspimport64
- cspmigrate, cspmigrate64
- cspnvfix, cspnvfix64
- csptest, csptest64
- csputils, csputils64
- date
- des_kat
- display-pubkey
- dump-marshalled
- elftool
- enquiry

- esn
- factorystate
- fet
- floodtest
- fwcheck
- fwversion
- gateway
- gateway6
- generatekey
- getrtc
- hakever
- help
- hsc_codesafe
- hsc_configurepoolmodule
- hsc_configureslots
- hsc_loadseemachine
- hsc_loadwarrants
- hsc_nethsmexports
- hsc_nethsmimports
- hsc_remotefilesystem
- hsc_serverremotecomms
- hsc_serversettings
- Administration of platform services (nShield 5 HSMs)
- hsmdiagnose
- initunit
- keytst, keytst64
- killrecov
- km-plode
- kmfile-dump
- kneti
- kptest
- `libcknfast.so`
  See PKCS#11 Developer libraries
- loadmache

- loadrom
- loadsee-setup
- logout
- sbin/logrotate-hardserver
- logs
- maintmode
- makecspyuserdata
- migrate-world
- mkaclx
- modstate
- ncdate
- ncperftest
- `ncpy`
  Internal, only called by other nShield utilities.
- ncsvcdep
- ncthread-test
- ncversions
- netcfg
- netcfg6
- netdiagnose
- netenable
- nethsmadmin
- nethsmenroll
- netlink
- new-world
- nfcp
- nfdiag
- nfkmattest
- nfkmcheck
- nfkminfo
- nfkmverify
- nfloadmon
- nfls
- nfrm

- nfwarrant
- nopclearfail
- nshieldaudit
- ntokenenroll
- nvram-backup
- nvram-sw
- openssl
- p11hyper
- passwd
- perfcheck
- ping
- pollbare
- postrocs
- ppmk
- preload
- pubkey-find
- push
- raccmd
- racgui
- racs
- randchk
- `rdlinene`
  Internal, only called by other nShield utilities.
- reboot
- retrievewarrants
- rfs-setup
- rfs-sync
- rfsaddr
- rocs
- route
- route6
- routing
- routing6
- rserverperm

- rtc
- see-sock-serv, see-stdioe-serv, see-stdioesock-serv, see-stdoe-serv
- setrtc
- sigtest
- slotinfo
- stattree
- sworldcheck
- tamperlog
- tct2
- trial
- uptime
- version
- `with-nfast`
  No longer supported, use preload

2. anonkneti

```
anonkneti [OPTIONS] <ADDRESS>
```

Anonymous kNETI request command that polls an HSM for its connection details. It returns the ESN and HK_{NETI} key hash from the HSM identified by its IP address.

For more information, see Configuring the remote file system (RFS).

Option	Description
`<ip-address>`	If your network is secure and you know the IP address of the HSM, you can obtain the ESN and hash of the K_{NETI} key by running `anonkneti` on the client computer. A manual double-check is recommended for security. For guidance on network security, see the *nShield Security Manual*. *<ip-address>* is the IP address of the HSM, which could be one of the following: • an IPv4 address • an IPv6 address, including a link-local IPv6 address • a hostname The command returns output in the following form: `A285-4F5A-7500 2418ec85c86027eb2d5959fef35edc5e1b3b698f` In this example output, `A285-4F5A-7500` is the ESN and `2418ec85c86027eb2d5959fef35edc5e1b3b698f` is the hash of the K_{NETI} key.
`-p, --port=PORT`	Confirms connectivity to an HSM that you expect to be at *<port-number>*. The output format is the same as without the port number: the ESN and the hash of the K_{NETI} key. `anonkneti -p 9004 <ip-address>` **Default**: `9004`. nethsmenroll uses `-P`, with upper-case `P`, for port numbers. `anonkneti` uses -p, with lower-case `p`.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If module `0` is specified, `anonkneti` displays the hash of the software key generated by the remote server. If you do not specify a module ID, `anonkneti` uses all modules by default.

Option	Description
Help options	
`-h`, `--help`	Displays help for `anonkneti`.
`-u`, `--usage`	Displays a brief usage summary for `anonkneti`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `anonkneti`.

2.1. anonkneti examples

2.1.1. Example 1: Run `anonkneti` against an HSM to check availability

```
anonkneti <ip-address>
```

If `anonkneti` can't reach the HSM, it displays an error: `no route to host/destination unreachable`.

If the remote device is **not** an HSM it will also error.

2.1.2. Example 2: Run `anonkneti` against localhost to obtain the softkneti hash

```
anonkneti -m 0 127.0.0.1
```

`anonkneti` polls the local hardserver for its softkneti hash. You can then provide the softkneti hash to the HSM alongside, or instead of, the IP address when configuring client connections for stronger authentication.

2.1.3. Example 3: Compare the IP address of a network-attached HSM from the front panel and the anonkneti response

```
anonkneti <network-attched-hsm-ip-address>
```

3. appliance-cli

Enables certain *appliance* administration or status commands. All operations require a privileged client (by default, that means an elevated command-prompt on Windows or membership of the nfast group on Linux). Operations run against a remote module additionally require that the module has been enrolled as privileged in the local hardserver.

`appliance-cli` supports remote administration of nShield Connect and nShield 5c devices, in addition to enabling certain administration or status commands to be made available to privileged clients running in guest VMs that have access to nShield SoloXC or nShield 5s local HSMs running on the host machine over nCipher Secure Transport/Impath.



Available commands will depend on your version and HSM type. `cs5` is nShield 5c specific, for more information, see Configure the nShield 5c for client authorization.

Appliance command	Action
`expirehsmlog`	Expire signed nShield 5 HSM syslogs
`exporthsmlog`	Get signed nShield 5 HSM syslogs
`gethsmlog`	Get nShield 5 HSM syslogs
`getsavedhsmlogs`	Get signed nShield 5 HSM syslogs saved on the 5c file system
`removesavedhsmlogs`	Remove signed nShield 5 HSM syslogs saved on the 5c file system
`gethsmenvstats`	Get nShield 5 HSM general statistics
`gethsmstatus`	Get nShield 5 HSM status and versions
`gethsminfo`	Get nShield 5 HSM info
`gethsmoption`	Get nShield 5 HSM option
`sethsmoption`	Set nShield 5 HSM option
`gethsmcs5stats`	Get nShield 5 HSM CodeSafe 5 statistics
`sethsmminvsn`	Set the minimum VSN for nShield 5 HSM
`cs5`	Configure CodeSafe5 for nShield 5
`gethsmlogkey`	Get nShield 5 HSM syslog signing key
`getservcfg`	Get server config file
`getservlog`	Get server logs
`getrtc`	Get the RTC time for nShield 5

Appliance command	Action
`setrtc`	Set the HSM RTC. Only available for nShield 5c version 13.3 or later. See setrtc for more information.
`serialctrl`	Enable or disable the Connect/5c's serial CLI
Module selection	
`-m`, `--module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `appliance-cli` uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `appliance-cli`.
`-u`, `--usage`	Displays a brief usage summary for `appliance-cli`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `appliance-cli`.

3.1. setrtc

You can only change the RTC forward to a time later than the one currently set. You cannot change it to an earlier time. This is to ensure the validity of certificates associated with the SSH keys used for attestation.

If you need to change the current time to an earlier time, for example if a mistake was made when the initial time was set, you will need to factory state the HSM.

4. bondcfg

```
bondcfg [mode=802.3ad] [miimon=100] [primary=eth0]
        [resend_igmp=1] [lacp_rate=slow] [xmit_hash_policy=layer2]
```

If bond interface is not set, the current setting will be displayed.

If bond interface is specified with no args then the current setting will be displayed.

The bond interface's address, `netmask` and `linkspeed` configuration are inherited from `eth0` (`iface=0`) configuration.

Option	Description
`lacp_rate`	(Only in `802.3ad` mode) The rate in which we'll ask our link partner to transmit LACPDU packets in `802.3ad` mode. Possible values are `slow` or `fast`. Default: `slow`.
`miimon`	The MII link monitoring frequency in milliseconds. Range: `0-10000`, default: `100`.
`mode`	Bond mode, one of `802.3ad` and `active-backup`. Default: `802.3ad`.
`primary`	(Only in `active-backup` mode) Primary device, one of `eth0` and `eth1`. Default: `eth0`.
`resend_igmp`	(Only in `active-backup` mode) The number of IGMP membership reports to be issued after a failover event. Range: `0-255`, default: `1`.
`xmit_hash_policy`	(Only in `802.3ad` mode) The transmit hash policy to use for slave selection in `802.3ad` mode. Possible values are `layer2`, `layer2+3` or `encap2+3`. Default: `layer2`.

5. bondlink

```
bondlink [enable/disable]
```

If the bond interface is not set, the current interface status will be displayed.

Option	Description
`action`	The action to take for the interface (`enable` or `disable`).

6. bulkerase

```
bulkerase [-v] [-m MODULE [-m MODULE ...]]
```

Erases multiple smart cards including Administrator Cards, Operator Cards, and FEM activation cards, in the same session.



Do not use the `bulkerase` utility to erase Administrator Cards from the current Security World.

Option	Description
`-v`, `--verbose`	Runs in verbose mode.
Option to address HSMs	
`-m`, `--module=<MODULE>`	Specifies the number of the module to erase cards. May be repeated, default = all.
Help options	
`-h`, `--help`	Displays help for `bulkerase`.
`-u`, `--usage`	Displays a brief usage summary for `bulkerase`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `bulkerase`.

7. cardpp

```
cardpp --examine|--change|--check|--recover [-m MODULE]
```

Changes, verifies, or recovers a passphrase of a card.

Option	Description
`-c`, `--change`	Change the passphrase of a card. Works both on ACS or OCS cards.
`-e`, `--examine`	Reads the card that is inserted in the slot of module `<MODULE>`. Works both on ACS or OCS cards.
`-k`, `--check`	Checks the passphrase. Works both on ACS or OCS cards. `cardpp` polls all available slots. If there is no card inserted, it prompts you to insert one. If the card belongs to this Security World, `cardpp` either tells you if no passphrase is set or prompts you to enter the passphrase and checks to see if it is correct.
`-r`, `--recover`	Allows you to set a new passphrase of a card if passphrase replacement was enabled when the Security World was created. See Changing unknown card passphrase with cardpp and passphrase replacement. Only works on OCS cards and the password replacement for the OCS card requires authorization by the ACS cards.
Module selection	
`-m`, `--module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `cardpp` uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `cardpp`.
`-u`, `--usage`	Displays a brief usage summary for `cardpp`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `cardpp`.

8. cef-audit-verify

```
cef-audit-verify [-h] [-u] [-v] [-e ESN] [-w WARRANT] [-r ROOT] [-o OUTDIR] [LOG]
```

Verifies audit logs produced on HSMs running a firmware version older than 13.5, which produced audit logs in CEF format. Replaces the `NFAST_HOME/python/examples/audit-log-verifier.py` script, which was previously provided for this purpose.

Option	Description
`-e ESN`, `--esn ESN`	The ESN of the logevents to verify.
`-o OUTDIR`, `--outdir OUTDIR`	The path to the output directory. `cef-audit-verify` generates output files in JSON format to describe the content and verification status of the logs.
`-r ROOT`, `--root ROOT`	The key for the root nShield HSM warrant Default: `KWARN-1`
`-w WARRANT`, `--warrant WARRANT`	The path to the warrant file or warrants directory. If you specify a warrant file or directory, the utility verifies up to the nShield HSM warrant root of trust.
`LOG`	Positional argument for you to enter the location of the CEF format audit log file to verify. This is typically either a hardserver log or a syslog log, depending on how audit was configured. If a hardserver log is provided, the utility can automatically distinguish CEF audit records from other hardserver log entries.
Help options	
`-h`, `--help`	Displays help for `cef-audit-verify`.
`-u`, `--usage`	Displays a brief usage summary for `cef-audit-verify`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `cef-audit-verify`.

9. cfg-mkdefault

```
cfg-mkdefault [-r] [-f FILENAME] [-c]
```

Creates a default client configuration file for the hardserver configuration sections. The con figuration file `cfg-mkdefault` creates can only be transferred to network-attached HSMs when it is created with the `--connect-config` option.

Option	Description
`-c, --connect_config`	Creates a config file for a network-attached HSM
`-f, --defaultfile=FILENAME`	Name of the file to write the default settings to. Default: `cardlist.default`
`-r, --resetmasterfile`	Resets the master config file to the default.
Help options	
`-h, --help`	Displays help for `cfg-mkdefault`.
`-u, --usage`	Displays a brief usage summary for `cfg-mkdefault`.
`-v, --version`	Displays the version number of the Security World Software that deploys `cfg-mkdefault`.

10. cfg-dynamicslots

Configures dynamic slots.

Option	Description
`-c, --count=N`	The number of dynamic slots to set. `0` disables dynamic slots.
`-e, --esn=ESN`	ESN of the local module to configure. Defaults to the current module 1.
`-s, --map=S`	Swaps the local reader slot `0` with this slot. `0` disables remapping.
`-U, --usefile=FILENAME`	Use `FILENAME` file as the masterconfigfile.
Help options	
`-h, --help`	Displays help for `cfg-dynamicslots`.
`-u, --usage`	Displays a brief usage summary for `cfg-dynamicslots`.
`-v, --version`	Displays the version number of the Security World Software that deploys `cfg-dynamicslots`.

11. cfg-mkcardlist

```
cfg-mkcardlist
```

Writes out a default `cardlist` file for controlling which smartcards are allowed to be used. To make this the default `cardlist`, set the name as `cardlist` and ensure that the file is in the config file directory.

Option	Description
`-f, --defaultfile=FILENAME`	Name of the file to write the default settings to. Default: `cardlist.default`
Help options	
`-h`, `--help`	Displays help for `cfg-mkcardlist`.
`-u`, `--usage`	Displays a brief usage summary for `cfg-mkcardlist`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `cfg-mkcardlist`.

12. cfg-pushnethsm

```
cfg-pushnethsm [-p PORT -a ADDR -f -k -m MODULE] FILE
```

Copies a specified configuration file from a remote file system to the file system on a speci fied module.

Some changes propagated with `cfg-pushnethsm` need further actions. For example, you have to clear the module after changing the dynamic slot configuration.

- Remote configuration of additional clients.
- About user privileges.
- CodeSafe utilities.

Option	Description

`-a, --address=ADDR`	Network address of the network-attached HSM to push configuration file to, or `""` (quotation marks without any content between) to just validate the file.
	
	`cfg-pushnethsm --address=<module_IP_address> <full_path_to_config_file>`
	
	<module_IP_address> is that of the nShield Connect on which to load the con figuration and *<full_path_to_config_file>* is the path to, and name of the updated configuration file. For example:
	
	`/opt/nfast/bin/cfg-pushnethsm –address 192.168.156.30 /opt/nfast/kmdata/hsm-49D5-C944-F159/config/config.new`
	
	Default: `""`.

`-f, --force`	Pushes the configuration file even if validation check fails.
`-k, --use-kneti`	Uses a local module KNETI to authenticate this client to the network-attached HSM. If no KNETI is specified with this `-k` option, the hardserver's software KNETI is used.
`-m, --module=MODULE`	Uses this module's KNETI. This `-m` option is ignored unless `--use-kneti` is used. Default: `1`.
`-n, --no-rfs-check`	Overrides the RFS check.
`-p, --port=PORT`	Sets the port to use to connect to the network-attached HSM. Default: `9004`
Help options	

Option	Description
`-h`, `--help`	Displays help for `cfg-pushnethsm`.
`-u`, `--usage`	Displays a brief usage summary for `cfg-pushnethsm`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `cfg-pushnethsm`.

13. cfg-pushntp

```
cfg-pushntp -a ADDR [-p PORT -k -m MODULE] -1 ADDR [-2 ADDR -3 ADDR] enable
cfg-pushntp -a ADDR [-p PORT -k -m MODULE] disable
```

Configures time synchronisation on the nShield HSM, using NTP. Enables or disables NTP time synchronization on the specified HSM. When enabling NTP synchronization, the IP addresses of up to 3 NTP servers may be specified.



The new NTP settings will take effect the next time the target HSM is restarted.

Option	Description
`-1, --ntp1=ADDR`	IP address of NTP server. **nShield 5c:** This can be an IPv4 or an IPv6 address.
`-2, --ntp2=ADDR`	IP address of NTP server. **nShield 5c:** This can be an IPv4 or an IPv6 address.
`-3, --ntp2=ADDR`	IP address of NTP server. **nShield 5c:** This can be an IPv4 or an IPv6 address.
`disable`	Disables the NTP service on the HSM.
`enable`	Enables the NTP service on the HSM.
`-a, --address=ADDR`	IP address of nShield HSM to configure NTP on. **nShield 5c:** This can be an IPv4 or an IPv6 address, or it can be a hostname that resolves to the HSM's IP address.
`-k, --use-kneti`	Uses KNETI to authenticate.
`-m, --module=MODULE`	Sets the HSM to use for KNETI authentication. The default is HSM 1. This option can only be used with the `--use-kneti` option.
`-p, --port=PORT`	Sets the port to use to connect to the nShield HSM (default=9004).
Help options	
`-h, --help`	Displays help for `cfg-pushntp`.
`-u, --usage`	Displays a brief usage summary for `cfg-pushntp`.
`-v, --version`	Displays the version number of the Security World Software that deploys `cfg-pushntp`.

13.1. Example

Linux

```
cfg-pushntp --address=192.30.100.150 --ntp1=192.23.24.256 enable
```

Windows

```
cfg-pushntp.exe --address=192.30.100.150 --ntp1=192.23.24.256 enable
```

Returns:

```
The requested NTP configuration changes have been uploaded and will take effect when the target nShield HSM is
restarted.
```

14. cfg-remoteslots

```
cfg-remoteslots [--export|--import|--unimport] [options]
```

Configures Remote Operator slot imports and exports. See Remote Operator.

14.1. cfg-remoteslot ACTIONS

Action	Description
`--export`	Allow remote reading of a slot.
`--import`	Import a slot from a remote machine.
`--unimport`	Unimport a slot previously imported from a remote machine.

14.2. cfg-remoteslot OPTIONS

Option	Description
`-e, --local-esn=ESN`	ESN of the local module. Default: current module `1`

`-p, --port=PORT`
`-p, --remote-ip=IPADDR`
`-p, --remote-esn=ESN`
	Port to connect to on the remote machine.
	IP address of the machine hosting the remote slot.
	ESN of the remote module.

`-s, --slot=SLOTID`	`SlotID` of the slot to be exported or the `SlotID` used to refer to the slot when it is imported.
`-U, --usefile=FILENAME`	Use this file as the master configfile.
Help options	
`-h, --help`	Displays help for `cfg-remoteslots`.
`-u, --usage`	Displays a brief usage summary for `cfg-remoteslots`.
`-v, --version`	Displays the version number of the Security World Software that deploys `cfg-remoteslots`.

15. cfg-reread

```
cfg-reread
```

Loads the hardserver configuration from the configuration file, which means that it reconfigures the hardserver according to the master configuration file.



No changes are overwritten in the `[server_startup]` and `[remote_administration_service_startup]` sections. To apply changes to these sections, you must restart the hardserver or remote administration service, respectively.

Option	Description
Help options	
`-h`, `--help`	Displays help for `cfg-reread`.
`-u`, `--usage`	Displays a brief usage summary for `cfg-reread`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `cfg-reread`.

16. checkmod

```
checkmod FILENAME...
```

Checks modulo exponentiations performed on the module against the test data located in `opt/nfast/testdata` (**Linux**) or `%NFAST_HOME%\testdata` (**Windows**).

Option	Description
Help options	
`-h`, `--help`	Displays help for `checkmod`.
`-u`, `--usage`	Displays a brief usage summary for `checkmod`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `checkmod`.

17. chkserv

```
chkserv
```

Attempts to open a connection to the hardserver.

Option	Description
`-r, --retry`	Retry until success or timeout Default: only 1 try
`-t, --timeout=TIME`	Timeout in (s)econds or (m)inutes Default: `240s`
Help options	
`-h, --help`	Displays help for `chkserv`.
`-u, --usage`	Displays a brief usage summary for `chkserv`.
`-v, --version`	Displays the version number of the Security World Software that deploys `chkserv`.

18. ckaesgen

```
ckaesgen [ -n | -p PIN ] [ -s token-name ] [ template options ]
```

Generates an AES secret key.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
`-s, --slot-name=SLOT`	Use only named SLOT.
Template options	
`-l, --keylength=VALUE_LEN`	Sets key length to `128` or `256`. Default: `128`
`-L, --label=LABEL`	Sets CKA_LABEL. Default: `Example label`
`--sign`	Sets `CKA_SIGN` to `true` (default).
`--nosign`	Sets `CKA_SIGN` to `false`.
`--encrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `true` (default).
`--noencrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `false`.
`--wrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `true` (default).
`--nowrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `false`.
`--derive`	Sets `CKA_DERIVE` to `true` (default).
`--noderive`	Sets `CKA_DERIVE` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--nosensitive`	Sets `CKA_SENSITIVE` to `false`.
`--extractable`	Sets `CKA_EXTRACTABLE` to `true`.
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false` (default).
`--modifiable`	Sets `CKA_MODIFIABLE` to `true` (default).
`--nomodifiable`	Sets `CKA_MODIFIABLE` to `false`.
Help options	
`-h, --help`	Displays help for `ckaesgen`.
`-u, --usage`	Displays a brief usage summary for `ckaesgen`.

Option	Description
`-V`, `--version`	Displays the version number of the Security World Software that deploys `ckaesgen`.

19. ckariagen

```
ckariagen [ -n | -p PIN ] [ -s token-name ] [ template options ]
```

Generates an ARIA secret key.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`.  Exposes PIN, use for testing only.
`-s, --slot-name=SLOT`	Use only named SLOT.
Template options	
`-l, --keylength=VALUE_LEN`	Sets the key length to `128`, `192`, or `256`. Default: `128`
`-L, --label=LABEL`	Sets CKA_LABEL. Default: `Example label`
`--encrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `true` (default).
`--noencrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `false`.
`--wrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `true` (default).
`--nowrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--nosensitive`	Sets `CKA_SENSITIVE` to `false`.
`--extractable`	Sets `CKA_EXTRACTABLE` to `true`.
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false` (default).
`--modifiable`	Sets `CKA_MODIFIABLE` to `true` (default).
`--nomodifiable`	Sets `CKA_MODIFIABLE` to `false`.
Help options	
`-h, --help`	Displays help for `ckariagen`.
`-u, --usage`	Displays a brief usage summary for `ckariagen`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckariagen`.

20. ckcerttool

```
ckcerttool -c CARDNAME -f FILENAME -k KMDATAKEYID [-L NAME] [-i CKA_ID]
ckcerttool -n -f FILENAME -k KMDATAKEYNAME [-L NAME] [-i CKA_ID]
ckcerttool -T -c CARDNAME -f FILENAME [-L NAME] [-i CKA_ID]
```



Do not use PKCS #11 to perform any task that requires an Administrator Card. Use the equivalent nShield utilities instead.

Imports a certificate as a PKCS #11 `CKO_CERTIFICATE` object of type `CKC_X_509`, and optionally, associates it with the corresponding private key.

Option	Description
Required	
`-c, --cardset=CARDNAME`	Name of cardset or softcard to use
`-f, --certfile=FILENAME`	Name of file of certificate (pem format)
`-k, --keyident=KMDATAKEYID`	Provides the NFKM key ident of the corresponding key
`-n, --nopin`	Doesn't call `C_Login`, the object will be a public object.
Optional	
`-i --cka_id=CKA_ID`	Sets `CKA_ID` on the certificate. If `CKA_ID` is set on the private key it must be set to the same value on the certificate, otherwise the import will fail. If `CKA_ID` is not set on the private key, using the `-i` option will set `CKA_ID` on the private key and, if the public key is present, on the public key as well.
`-L, --certname=NAME`	Gives the certificate a name stored as `CKA_LABEL`. Defaults to the value on the private key or "ncipher-cert" if that is not set. If CKA_LABEL is not set on the key private key CKA_LABEL will be set to this value on the private and public key, if present.
`-T, --trusted`	Sets `CKA_TRUSTED` to `true`.
Help options	
`-h, --help`	Displays help for `ckcerttool`.
`-u, --usage`	Displays a brief usage summary for `ckcerttool`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckcerttool`.

20.1. Import a cardset- or softcard-protected certificate

```
ckcerttool -c CARDNAME -f FILENAME -k KMDATAKEYID [-L NAME]
```

20.2. Import module-only (no passphrase or cardset name)

```
ckcerttool -n -f FILENAME -k KMDATAKEYNAME [-L NAME]
```

20.3. Import a trusted public certificate with no corresponding private key

```
ckcerttool -T -c CARDNAME -f FILENAME [-L NAME]
```

21. ckcmac-ctr

```
ckcmac-ctr [-p | -q]
```

Tests AES-CMAC-CTR derive known answer test.

Option	Description
`-n, --nopin`	Doesn't call `C_Login` or list private objects.
`-p, --pin-for-testing=PIN`	Uses PIN for `C_Login`.  This will expose the PIN to other users of your system, use for testing only.
`-q, --quit-on-error`	Quit on first failure.
Help options	
`-h, --help`	Displays help for `ckcmac-ctr`.
`-u, --usage`	Displays a brief usage summary for `ckcmac-ctr`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckcmac-ctr`.

22. ckcrypt

```
ckcrypt [ -p PIN | -n ]
```

Performs some encryption.

Option	Description
`-n, --nopin`	Doesn't call `C_Login` or list private objects.
`-p, --pin-for-testing=PIN`	Uses PIN for `C_Login`.  This will expose the PIN to other users of your system, use for testing only.
Help options	
`-h, --help`	Displays help for `ckcrypt`.
`-u, --usage`	Displays a brief usage summary for `ckcrypt`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckcrypt`.

23. ckdes3gen

```
ckdes3gen [ -n | -p PIN ] [ template options ]
```

Generates an DES3 secret key.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
Template options	
`-L, --label=LABEL`	Sets CKA_LABEL. Default: `Example label`
`--sign`	Sets `CKA_SIGN` and `CKA_VERIFY` to `true` (default).
`--nosign`	Sets `CKA_SIGN` and `CKA_VERIFY` to `false`.
`--encrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `true` (default).
`--noencrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `false`.
`--wrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `true` (default).
`--nowrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `false`.
`--derive`	Sets `CKA_DERIVE` to `true` (default).
`--noderive`	Sets `CKA_DERIVE` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--nosensitive`	Sets `CKA_SENSITIVE` to `false`.
`--extractable`	Sets `CKA_EXTRACTABLE` to `true`.
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false` (default).
Help options	
`-h, --help`	Displays help for `ckdes3gen`.
`-u, --usage`	Displays a brief usage summary for `ckdes3gen`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckdes3gen`.

24. ck_ecedwards_gen

```
ck_ecedwards_gen [-p | -n]
```

Tests elliptic curve Edwards key generation.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object. Forces `--nosensitive --extractable`.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
`-s, --slot-name=SLOT`	Use only named SLOT.
Template options	
`-L, --label=LABEL`	Sets CKA_LABEL. Default: `Example label`
`--wrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `true`.
`--nowrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `false` (default).
`--sign`	Sets `CKA_SIGN` to `true` (default).
`--nosign`	Sets `CKA_SIGN` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--nosensitive`	Sets `CKA_SENSITIVE` to `false`.
`--extractable`	Sets `CKA_EXTRACTABLE` to `true`.
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false` (default).
Help options	
`-h, --help`	Displays help for `ck_ecedwards_gen`.
`-u, --usage`	Displays a brief usage summary for `ck_ecedwards_gen`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ck_ecedwards_gen`.

25. ckecies

```
ckecies [ -p PIN | -n ]
```

Option	Description
`--base-label=LABEL`	Base key label (default: `eciesbase`)
`--display-wrap`	Displays the public half of the wrapping key.
`--generate-base`	Generates a base key.
`--generate-wrap`	Generates a wrapping key.
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
`-s, --slot-name=SLOT`	Use only named SLOT.
`--unwrap`	Unwraps a ciphertext with the wrapping key.
`--unwrapped-label=LABEL`	Unwrapped key label (default: `eciesunwrapped`)
`--variant=VARIANT`	ECIES variant, XOR or (default) CTR.
`--wrap`	Wraps the base key with the wrapping key.
`--wrap-file=PATH`	Filename to read/write wrapped data
`--wrap-label=LABEL`	Wrapping key label (default: `ecieswrap`)
`--wrap-pubkey=PUBKEY`	Wrapping public key
Help options	
`-h, --help`	Displays help for `ckecies`.
`-u, --usage`	Displays a brief usage summary for `ckecies`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckecies`.

26. ck_ecmontgomery_gen

```
ck_ecmontgomery_gen [-p | -n]
```

Tests elliptic curve Montgomery key generation.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object. Forces `--nosensitive --extractable`.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
`-s, --slot-name=SLOT`	Use only named SLOT.
Template options	
`-L, --label=LABEL`	Sets CKA_LABEL. Default: `Example label`
`--wrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `true` (default).
`--nowrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `false`.
`--derive`	Sets `CKA_DERIVE` to `true` (default).
`--noderive`	Sets `CKA_DERIVE` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--nosensitive`	Sets `CKA_SENSITIVE` to `false`.
`--extractable`	Sets `CKA_EXTRACTABLE` to `true`.
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false` (default).
Help options	
`-h, --help`	Displays help for `ck_ecmontgomery_gen`.
`-u, --usage`	Displays a brief usage summary for `ck_ecmontgomery_gen`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ck_ecmontgomery_gen`.

27. ckhyper

28. ckcheckinst

```
ckcheckinst
```

PKCS #11 information utility.



Do not use PKCS #11 to perform any task that requires an Administrator Card. Use the equivalent nShield utilities instead.

For instructions how to verify the installation of the nShield PKCS #11 libraries, see Checking the installation of the nCipher PKCS #11 library.

Option	Description
`-s, --slot=SLOT`	Uses slot SLOT for tests rather than prompting.
`-p, --pin=PIN`	Uses PIN for the slot rather than prompting. [WARNING] This will expose the PIN to other users of your system.
Help options	
`-h, --help`	Displays help for `ckcheckinst`.
`-u, --usage`	Displays a brief usage summary for `ckcheckinst`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckcheckinst`.

28.1. ckcheckinst output examples: Security World validity

If you have an invalid Security World (for example, if all your HSMs are in the initialization state), `ckcheckinst` quits with the following error message:

```
ckcheckinst: C_Initialize failed rv = 00000006
Is the security world initialized? (Use nfkminfo to check)
```

If your Security World is valid, `ckcheckinst` displays information similar to the following:

```
PKCS#11 library interface version 2.40
 flags 0
 manufacturerID "nCipher Corp. Ltd "
 libraryDescription "nCipher PKCS#11 1.#.# "
 implementation version 1.##
 Load sharing and Failover enabled

slot Status Label
===== ====== ===== 0 Fixed token "accelerator "
1 Operator card "card2 "
```

```
2 Operator card "card3 "
Select slot Number to run library test or 'R'etry or to 'E'xit:
```

In this example output:

- `PKCS #11 library interface version 2.40` refers to the version of the PKCS #11 specification supported
- `implementation version 1.##` refers to the version of the nCipher PKCS #11 library
- `Loadsharing and Failover enabled` is shown if load-sharing has been enabled. Alternatively `Pool mode enabled` is shown if Pool mode has been enabled.

Slots that contain a valid Operator Card are indicated by the status `Operator card` and the card's label. A fixed token is always available and is listed as slot 0.

28.2. ckcheckinst output examples: invalid cards

If you insert a blank card or an unrecognized card (for example, an Operator Card from a different Security World or an Administrator Card), this is indicated in the `Status` column. The corresponding slot number is not available.



If you are using the preload command-line utility in conjunction with the nShield PKCS #11 library, you can only see the token that you loaded with the `preload` utility. In load-sharing mode, the loaded card set is used to set the environment variable `CKNFAST_CARDSET_HASH`, so only this card set is visible as a slot.

If there is no card in a slot, `ckcheckinst` displays `No token present` beside the relevant slot numbers. `ckcheckinst` gives you the following choices:

```
No removable tokens present.
Please insert an operator card into at least one available slot and
enter 'R' retry.
If you have not created an operator card or there are no physical slots, enter a fixed token slot number,
or 'E' to exit this program and create a card set before continuing.
```

If there are no available slots with cards in them, you can choose one of the following actions:

- Insert a valid Operator Card, and press **R**
- choose a fixed token slot
- Press **E** to quit, then create an OCS, and run `ckcheckinst` again.

When there is at least one slot with a valid token, input a slot number, and press **Enter**. In a

FIPS 140 Level 3 compliant Security World, `ckcheckinst` prompts you to enter the passphrase for the selected Operator Card. Type the passphrase, and press **Enter**.

`ckcheckinst` displays the results of the tests:

```
Test Pass/Failed
---- -----------
1 Generate RSA key pair Pass
2 Generate DSA key pair Pass
3 Encryption/Decryption Pass
4 Signing/Verify Pass
Deleted test keys ok
PKCS11 Library test successful.
```

If any tests fail, `ckcheckinst` displays a message indicating the failure and quits. It does not run any subsequent tests.

If `ckcheckinst` fails:

- Check that the hardserver is running
- Use the `enquiry` and `nfkminfo` world.

29. cknfkmid

```
cknfkmid [-p | -n] IDENT
```



Do not use PKCS #11 to perform any task that requires an Administrator Card. Use the equivalent nShield utilities instead.

Displays values of attributes of PKCS #11 objects.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`.  Exposes PIN, use for testing only.
Help options	
`-h, --help`	Displays help for `cknfkmid`.
`-u, --usage`	Displays a brief usage summary for `cknfkmid`.
`-V, --version`	Displays the version number of the Security World Software that deploys `cknfkmid`.

30. ckshahmac

```
ckshahmac [-p | -n]
```



Do not use PKCS #11 to perform any task that requires an Administrator Card. Use the equivalent nShield utilities instead.

Performs a PKCS #11 test for vendor-defined `SHA1_HMAC` key signing and verification capabilities.

Option	Description
`-n`, `nopin`	Doesn't call `C_Login`, makes key public object.
`-p`, `pin-for-testing=PIN`	Use PIN for `C_Login`.  Exposes PIN, use for testing only.
Help options	
`-h`, `--help`	Displays help for `ckshahmac`.
`-u`, `--usage`	Displays a brief usage summary for `ckshahmac`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `ckshahmac`.

31. cksigtest

```
cksigtest [options]
```



Do not use PKCS #11 to perform any task that requires an Administrator Card. Use the equivalent nShield utilities instead.

Test cryptographic performance of attached nCipher hardware using the nCipher PKCS #11 Library.

Option	Description
Program options	
`-s`, `--sign`	Test sign operation (default).
`-v`, `--verify`	Test verify operation.
`-d`, `--decrypt`	Test decrypt operation.
`-e`, `--encrypt`	Test encrypt operation (default for mechs that can't sign).
Key options	
`-S`, `--key-type=TYPE`	Select key type to use - `RSA` (default), `DSA`, `KCDSA`, `EC`, `ECDSA`, `EDDSA`, `MLDSA` or `SLHDSA`.
`-l`, `--key-size=BITS`	Sets the key size for `RSA` (default `2048`).
`-M`, `--mech=MECH`	Use mechanism `MECH` (`RSA_PKCS_PSS`, `RSA_PKCS`, `RSA_PKCS_OAEP`, `RSA_X_509`, `SHA256_RSA_PKCS`, `DSA`, `DSA_SHA1`, `KCDSA_HAS160`, `EC_SHA256`, `ECDSA`, `EDDSA`, `ML_DSA`, `HASH_ML_DSA_SHA256`, `HASH_ML_DSA_SHA512`, `HASH_ML_DSA_SHAKE128`, `HASH_ML_DSA_SHAKE256`, `HASH_ML_DSA`, `SLH_DSA`, `HASH_SLH_DSA_SHA256`, `HASH_SLH_DSA_SHA512`, `HASH_SLH_DSA_SHAKE128`, `HASH_SLH_DSA_SHAKE256`, `HASH_SLH_DSA`) Default for sign / verify: `RSA_PKCS_PSS`. Default for encrypt / decrypt: `RSA_PKCS_OAEP`.
`--eddsa-mode=EDDSA_MODE`	Choose mode for `EDDSA`. Valid values are: `pure` or `prehash` (default).
PQC parameters	

Option	Description
`-P`, `--parameter-set=PARAMETER-SET`	Set `CKA_PARAMETER_SET`. Valid values are: • `CKP_ML_DSA_44` (default) • `CKP_ML_DSA_65` • `CKP_ML_DSA_87` • `CKP_SLH_DSA_SHA2_128S` • `CKP_SLH_DSA_SHA2_128F` • `CKP_SLH_DSA_SHA2_192S` • `CKP_SLH_DSA_SHA2_192F` • `CKP_SLH_DSA_SHA2_256S` • `CKP_SLH_DSA_SHA2_256F` • `CKP_SLH_DSA_SHAKE_128S` • `CKP_SLH_DSA_SHAKE_128F` • `CKP_SLH_DSA_SHAKE_192S` • `CKP_SLH_DSA_SHAKE_192F` • `CKP_SLH_DSA_SHAKE_256S` • `CKP_SLH_DSA_SHAKE_256F`
`--context-string=CONTEXT-STRING`	Set `CONTEXT-STRING` (default is empty string).
Hedge variant selection (default `hedge-preferred`)	
• `--hedge-preferred` • `--hedge-required` • `--deterministic-required`	• `CKH_HEDGE_PREFERRED` • `CKH_HEDGE_REQUIRED` • `CKH_DETERMINISTIC_REQUIRED`
Other options	
`-c`, `--cardset=NAME`	Name of cardset to use
`-j`, `--threads=COUNT`	Set the max no. threads (default = `30`).
`-B`, `--unbuffered-stdout`	Always flush stdout.
`-t`, `--stop-after=TIME`	Set maximum time to run (default is `60` seconds).
`-p`, `--pin-for-testing=PIN`	Use PIN for `C_Login` Exposes PIN, use for testing only.
`-n`, `--nopin`	Don't call `C_Login`, Object will be public object.
`--sessionkeys`	Generate session keys instead of token keys.
Help options	
`-h`, `--help`	Display help for `cksigtest`.

Option	Description
`-V`, `--version`	Display the version number of `cksigtest`.
`-u`, `--usage`	Display a brief usage summary for `cksigtest`.

32. ckinfo

```
ckinfo [--repeat-count=COUNT --sleep-for=SECONDS]
```

Displays information about the nShield PKCS #11 library, slot, and token. Use this utility to verify that the library is functioning correctly.

Option	Description
`-r, --repeat-count=COUNT`	Repeats the count. Default: `1`.
`-s, --sleep-for=SECONDS.`	Waits between repeats, in seconds. Default: `0`.
Help options	
`-h`, `--help`	Displays help for `ckinfo`.
`-u`, `--usage`	Displays a brief usage summary for `ckinfo`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `ckinfo`.

33. ckkeyloop

```
ckkeyloop [ -n | -p PIN ] [ -t ]
```

Generates AES secret keys.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
Template options	
`-t, --token`	Sets CKA_Token to `true`. Default: `false`.
Help options	
`-h, --help`	Displays help for `ckkeyloop`.
`-u, --usage`	Displays a brief usage summary for `ckkeyloop`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckkeyloop`.

34. cklist

```
cklist [-p PIN | -n]
```

Views details of P11 objects on all slots. If invoked with a PIN argument, the utility lists public and private objects. If invoked with the `-n` (`--nopin`) option, the utility lists only the public objects.

This utility does not output any potentially sensitive attributes, even if the object has `CKA_SENSITIVE` set to `FALSE`.

Option	Description
`-n, --nopin`	Doesn't call `C_Login`, doesn't list private objects.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
Template options	
`--cka-encrypt=ENCRYPT`	Matches `CKA_ENCRYPT`.
`--cka-id=ID`	Matches `ID` (hex bytestring).
`--cka-issuer=ISSUER`	Matches `ISSUER` (hex bytestring).
`--cka-nfkm-hash=HASH`	Matches `HASH`.
`--cka-nfkm-ident=IDENT`	Matches `IDENT`.
`--cka-serial-number=NUMBER`	Matches `NUMBER` (hex bytestring).
`--cka-sign=SIGN`	Matches `CKA_SIGN`.
`-l, --cka-label=LABEL`	Matches `LABEL`.
`-r, --repeat-count=COUNT`	Repeats the count. Default: `1`.
`-s, --slot-name=SLOT`	Use only named SLOT.
`--verify-mode`	Flattens the output with security relevant attributes only.
Help options	
`-h, --help`	Displays help for `cklist`.
`-u, --usage`	Displays a brief usage summary for `cklist`.
`-V, --version`	Displays the version number of the Security World Software that deploys `cklist`.

35. ckmechinfo

```
ckmechinfo
```

Option	Description
Help options	
`-h`, `--help`	Displays help for `ckmechinfo`.
`-u`, `--usage`	Displays a brief usage summary for `ckmechinfo`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `ckmechinfo`.

Displays details of the supported PKCS #11 mechanisms provided by the module.

36. ckmldsa

```
ckmldsa [-p PIN] [-s SLOT] [-c CONTEXT-STRING] [ template options ] [ hedge variant selection ]
```

Tests MLDSA key generation and use.

Option	Description
`-p`, `--pin-for-testing=PIN`	Use PIN for `C_Login`  Exposes PIN, use for testing only.
`-s`, `--slot-name=SLOT`	Use only named SLOT.
`-c`, `--context-string=CONTEXT-STRING`	Set `CONTEXT-STRING` (default is empty string).
Template options	
`-L`, `--label=LABEL`	Set `CKA_LABEL` (default "Example label MLDSA").
`-P`, `--parameter-set=PARAMETER-SET`	Set `CKA_PARAMETER_SET`. Valid values are: • `CKP_ML_DSA_44` (default) • `CKP_ML_DSA_65` • `CKP_ML_DSA_87`
Mechanism selection	
`-m`, `--mechanism=MLDSA-MECHANISM`	• `CKM_ML_DSA` (default) • `CKM_HASH_ML_DSA` • `CKM_HASH_ML_DSA_SHA256` • `CKM_HASH_ML_DSA_SHA512` • `CKM_HASH_ML_DSA_SHAKE128` • `CKM_HASH_ML_DSA_SHAKE256`
`-H`, `--prehash-mechanism=PREHASH-MECH`	Hash mechanism for use when prehashing. Valid values are: • `CKM_HASH_ML_DSA_SHA256` (default) • `CKM_HASH_ML_DSA_SHA512` • `CKM_HASH_ML_DSA_SHAKE128` • `CKM_HASH_ML_DSA_SHAKE256`
Hedge variant selection (default `hedge-preferred`)	
• `--hedge-preferred` • `--hedge-required` • `--deterministic-required`	• `CKH_HEDGE_PREFERRED` • `CKH_HEDGE_REQUIRED` • `CKH_DETERMINISTIC_REQUIRED`

Option	Description
Help options	
`-h`, `--help`	Display help for `ckmldsa`.
`-V`, `--version`	Display the version number of `ckmldsa`.
`-u`, `--usage`	Display a brief usage summary for `ckmldsa`.

37. ckrestrictkey

```
ckrestrictkey [ -n | -p PIN ] [ -s token-name] [-L label] [ template options ]
```

Sets attributes.

Option	Description
`-k, --keytype=KEYTYPE`	Key type. Default: `AES`
`-L, --label=LABEL`	Uses key with this label. Default: `Example label`
`-n, --nopin`	Doesn't call `C_Login`, makes key public object.
`-p, --pin-for-testing=PIN`	Use PIN for `C_Login`. Exposes PIN, use for testing only.
`-s, --slot-name=SLOT`	Use only named SLOT.
Template options	
`--noencrypt`	Sets `CKA_ENCRYPT` to `false`.
`--nodecrypt`	Sets `CKA_DECRYPT` to `false`.
`--nosign`	Sets `CKA_SIGN` and `CKA_SIGN_RECOVER` to `false`.
`--noverify`	Sets `CKA_VERIFY` and `CKA_VERIFY_RECOVER` to `false`.
`--nowrap`	Sets `CKA_WRAP` to `false`.
`--nounwrap`	Sets `CKA_UNWRAP` to `false`.
`--noderive`	Sets `CKA_DERIVE` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false`.
`--nomodifiable`	Sets `CKA_MODIFIABLE` to `false`.
Help options	
`-h, --help`	Displays help for `ckrestrictkey`.
`-u, --usage`	Displays a brief usage summary for `ckrestrictkey`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ckrestrictkey`.

38. ckrsagen

```
ckrsagen [-p | -n]
```

Tests RSA key generation. You can use specific PKCS #11 attributes for generating RSA keys.

Option	Description
`-n`, `nopin`	Doesn't call `C_Login`, makes key public object.
`-p`, `pin-for-testing=PIN`	Use PIN for `C_Login`.  Exposes PIN, use for testing only.
`-s`, `slot-name=SLOT`	Use only named SLOT.
Template options	
`-l`, `keylength=MODULUSBITS`	Sets `CKA_MODULUS_BITS`. Default: `128`
`-L`, `label=LABEL`	Sets CKA_LABEL. Default: `Example label`
`--sign`	Sets `CKA_SIGN` and `CKA_VERIFY` to `true` (default).
`--nosign`	Sets `CKA_SIGN` and `CKA_VERIFY` to `false`.
`--encrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `true` (default).
`--noencrypt`	Sets `CKA_ENCRYPT` and `CKA_DECRYPT` to `false`.
`--wrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `true` (default).
`--nowrap`	Sets `CKA_WRAP` and `CKA_UNWRAP` to `false`.
`--sensitive`	Sets `CKA_SENSITIVE` to `true` (default).
`--nosensitive`	Sets `CKA_SENSITIVE` to `false`.
`--extractable`	Sets `CKA_EXTRACTABLE` to `true`.
`--noextractable`	Sets `CKA_EXTRACTABLE` to `false` (default).
`--strongprime`	Uses `CKM_RSA_X9_31_KEY_PAIR_GEN` to generate the key, rather than `CKM_RSA_PKCS_KEY_PAIR_GEN`.
`--pubexp17`	Uses alternate public exponent 17. Not valid in FIPS.
Help options	
`-h`, `--help`	Displays help for `ckrsagen`.

Option	Description
`-u`, `--usage`	Displays a brief usage summary for `ckrsagen`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `ckrsagen`.

39. ckslhdsa

```
ckslhdsa [-p PIN] [-s SLOT] [-c CONTEXT-STRING] [ template options ] [ hedge variant selection ]
```

Tests SLHDSA key generation and use.

Option	Description
-p, --pin-for-testing=PIN	Use PIN for C_Login  Exposes PIN, use for testing only.
-s, --slot-name=SLOT	Use only named SLOT.
-c, --context-string=CONTEXT-STRING	Set CONTEXT-STRING (default is empty string).
Template options	
-L, --label=LABEL	Set CKA_LABEL (default "Example label SLHDSA").
-P, --parameter-set=PARAMETER-SET	Set CKA_PARAMETER_SET. Valid values are: • CKP_SLH_DSA_SHA2_128S (default) • CKP_SLH_DSA_SHAKE_128S • CKP_SLH_DSA_SHA2_128F • CKP_SLH_DSA_SHAKE_128F • CKP_SLH_DSA_SHA2_192S • CKP_SLH_DSA_SHAKE_192S • CKP_SLH_DSA_SHA2_192F • CKP_SLH_DSA_SHAKE_192F • CKP_SLH_DSA_SHA2_256S • CKP_SLH_DSA_SHAKE_256S • CKP_SLH_DSA_SHA2_256F • CKP_SLH_DSA_SHAKE_256F
Mechanism selection	
-m, --mechanism=SLHDSA-MECHANISM	• CKM_SLH_DSA (default) • CKM_HASH_SLH_DSA • CKM_HASH_SLH_DSA_SHA256 • CKM_HASH_SLH_DSA_SHA512 • CKM_HASH_SLH_DSA_SHAKE128 • CKM_HASH_SLH_DSA_SHAKE256

Option	Description
`-H`, `--prehash-mechanism=PRE-HASH-MECH`	Hash mechanism for use when prehashing. Valid values are: • `CKM_HASH_SLH_DSA_SHA256` (default) • `CKM_HASH_SLH_DSA_SHA512`
Hedge variant selection (default `hedge-preferred`)	
• `--hedge-preferred` • `--hedge-required` • `--deterministic-required`	• `CKH_HEDGE_PREFERRED` • `CKH_HEDGE_REQUIRED` • `CKH_DETERMINISTIC_REQUIRED`
Help options	
`-h`, `--help`	Display help for `ckslhdsa`.
`-V`, `--version`	Display the version number of `ckslhdsa`.
`-u`, `--usage`	Display a brief usage summary for `ckslhdsa`.

40. cksotool

```
usage: cksotool [-h] [--version] [-m MODULE] [-c | -p | -i | --delete]
```

Creates a PKCS #11 Security Officer role and manages its PIN.

Option	Description
`-c`, `--create`	Creates the Security Officer role.
`--delete`	Deletes the Security Officer role.
`-i`, `--info`	Shows information about the Security Officer role.
`-p`, `--change-pin`	Changes the Security Officer PIN.
Option to address HSMs	
`-m`, `--module=<MODULE>`	Selects the MODULE for Security Officer artefact creation. If you only have one module, `<MODULE>` is 1. If you do not specify a module number, the utility uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `cksotool`.
`--version`	Displays the version number of the Security World Software that deploys `cksotool`.

41. ck-xfer-fix

```
ck-xfer-fix [-m MODULE] ident [ident [...]]
```

PKCS#11 fixup utility for migrated keys.

Option	Description
Module selection	
`-m`, `--module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `ck-xfer-fix` uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `ck-xfer-fix`.
`-u`, `--usage`	Displays a brief usage summary for `ck-xfer-fix`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `ck-xfer-fix`.

42. config-auditlogging

```
config-auditlogging [OPTIONS]
```

Edits the `[auditlog_settings]` section of the local hardserver config file. Unspecified fields are not changed. After making any changes you will be required to restart the hardserver in order for them to take effect.

Option	Description
`-l`, `--enable-logfile`	Saves audit messages to the hardserver log.
`-L`, `--disable-logfile`	Disables saving audit messages to hardserver logs. Default: configuration as shipped
`--port=PORT`	Specifies the UDP port to use for syslog messages. Default: 514
`-s`, `--enable-syslog`	Sends audit messages to a syslog server (requires `--server`)
`-S`, `--disable-syslog`	Disables sending audit messages to `syslog`. Default: configuration as shipped
`--server=IP`	Specifies the `syslog` server IP for `syslog` messages (implies `--enable-syslog`).
Help options	
`-h`, `--help`	Displays help for `config-auditlogging`.
`-u`, `--usage`	Displays a brief usage summary for `config-auditlogging`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `config-auditlogging`.

43. cpioc

```
cpioc output [file file ...]
```

Creates a POSIX.1 portable format CPIO archive containing the specified files. If no files are specified, filenames are read from standard input.

Option	Description
Help options	
`-h`, `--help`	Displays help for `cpioc`.
`--version`	Displays the version number of the Security World Software that deploys `cpioc`.

44. cngimport

Migrates Security World, CAPI and CNG keys to the Security World Key Storage Provider.

For more information, see:

- Importing a Microsoft CAPI key into the Security World Key Storage Provider.
- Importing a Microsoft CNG key into the Security World Key Storage Provider.
- Importing a Security World key into the Security World Key Storage Provider
- cngimport

Use this helper utility to manage keys and the interfaces between the CNG library and the HSM.

Utility names that end with `64` run only on 64-bit version of Microsoft Windows. All other utilities run on both 32-bit and 64-bit versions of Microsoft Windows.

Standard help options:

- `-h`|`--help` displays help for the utility.
- `-v`|`--version` displays the version number of the utility.
- `-u`|`--usage` displays a brief usage summary for the utility.

45. cnginstall32, cnginstall

(nShield CNG provider installer utility)

Removes or reinstalls the provider DLLs and associated registry entries manually.

For more information, see cnginstall.

Use this helper utility to manage keys and the interfaces between the CNG library and the HSM. For a list of utilities specific to the nShield CNG CSP, see Utilities for the CAPI CSP.

Utility names that end with `64` run only on 64-bit version of Microsoft Windows. All other utilities run on both 32-bit and 64-bit versions of Microsoft Windows.

Standard help options:

- `-h|--help` displays help for the utility.
- `-v|--version` displays the version number of the utility.
- `-u|--usage` displays a brief usage summary for the utility.

46. cnglist32, cnglist

Views information about CNG providers.

For more information, see:

- Migrating keys for CNG.
- Importing a Microsoft CAPI key into the Security World Key Storage Provider.
- Importing a Microsoft CNG key into the Security World Key Storage Provider.
- Importing a Security World key into the Security World Key Storage Provider
- cnglist.

Use this helper utility to manage keys and the interfaces between the CNG library and the HSM. For a list of utilities specific to the nShield CNG CSP, see Utilities for the CAPI CSP.

Utility names that end with `64` run only on 64-bit version of Microsoft Windows. All other utilities run on both 32-bit and 64-bit versions of Microsoft Windows.

Standard help options:

- `-h|--help` displays help for the utility.
- `-v|--version` displays the version number of the utility.
- `-u|--usage` displays a brief usage summary for the utility.

47. cngregister

(nShield CNG provider registration utility) to unregister and re-register the nShield providers manually.

For more information, see:

- Registering the CNG CSP.
- Unregistering or reregistering the CNG CSP.
- Uninstalling or reinstalling the CNG CSP.
- cngregister.

Use this helper utility to manage keys and the interfaces between the CNG library and the HSM. For a list of utilities specific to the nShield CNG CSP, see Utilities for the CAPI CSP.

Utility names that end with `64` run only on 64-bit version of Microsoft Windows. All other utilities run on both 32-bit and 64-bit versions of Microsoft Windows.

Standard help options:

- `-h|--help` displays help for the utility.
- `-v|--version` displays the version number of the utility.
- `-u|--usage` displays a brief usage summary for the utility.

48. cngsoak, cngsoak64

(nShield CNG soak tool)

Evaluates the performance of signing, key exchange, and key generation by using a user-defined number of threads.

For more information, see cngsoak.

Use this helper utility to manage keys and the interfaces between the CNG library and the HSM. For a list of utilities specific to the nShield CNG CSP, see Utilities for the CAPI CSP.

Utility names that end with `64` run only on 64-bit version of Microsoft Windows. All other utilities run on both 32-bit and 64-bit versions of Microsoft Windows.

Standard help options:

- `-h|--help` displays help for the utility.
- `-v|--version` displays the version number of the utility.
- `-u|--usage` displays a brief usage summary for the utility.

49. config-serverstartup

Edits the `[server_startup]` section of the configuration file for the client's hardserver to enable or disable TCP sockets.

For more information, see:

- After software installation.
- config-serverstartup.

Option	Description
`-p`, `--enable-privileged-tcp`	Enables listening for privileged local clients connecting by TCP.
`-P`, `--disable-privileged-tcp`	Disables listening for privileged local clients connecting by TCP. Default: configuration as shipped
`--port=PORT`	Specifies the TCP port on which to listen for unprivileged clients, if enabled. Default: `9000`
`--privport=PORT`	Specifies the TCP port on which to listen for privileged clients, if enabled Default: `9001`
`-s`, `--enable-tcp`	Enables listening for local clients connecting by TCP.
`-S`, `--disable-tcp`	Disables listening for local clients connecting by TCP. Default: configuration as shipped
Help options	
`-h`, `--help`	Displays help for `rocs`.
`-u`, `--usage`	Displays a brief usage summary for `rocs`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `rocs`.

50. configure-csp-poolmode, configure-csp-poolmode64

Configures HSM Pool mode for the nShield CAPI CSP.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They can also manage the interfaces between the MSCAPI library and the module.

See Utilities for the CAPI CSP.

51. createocs

```
createocs -m MODULE -Q K/N -N NAME [-MpPRqe] [-T TIME]
createocs -m MODULE -e [-e]
```

Creates operator cardsets or erases cards. When `createocs` has obtained the authorization from a valid card or if no authorization is required, it prompts you to insert a card.

Without `-e`, creates a new operator cardset. You must specify at least the module (with `--module`), the quorum (with `--ocs-quorum`) and the new cardset name (with `--name`).

By default when a new operator cardset is created:

- The cardset will NOT be persistent. Thus keys protected by it will only be usable while the last card remains inserted. Use the `--persist` option to change this.
- Passphrase recovery is enabled. Use the `--no-pp-recovery` option to make passphrase recovery impossible. This will make keys inaccessible if more than N-K passphrases are forgotten.
- Not remotely readable. Use the `--remotely-readable` option to allow the cardset to be used in remote slots. Remotely readable cardsets are always persistent.

For more information, see:

- Creating an Operator Card Set from the command line.
- Erasing cards from the command line.

Option	Description
`-e, --erase`	Erases a card (instead of creating a card set). This option cannot be used in conjunction with any of the 'New cardset properties' options.
`--ee`	Erases several cards. This option cannot be used in conjunction with any of the 'New cardset properties' options.
`-M, --name-cards`	Names individual cards within the card set. You can only use this option after the card set has been named by using the `--name=`NAME option. `createocs` prompts for the names of the cards as they are created. Not all applications can display individual card names.
`-N, --name=<NAME>`	Specifies a name for the card set. The card set must be named with this option before individual cards can be named using the `-M`/`--name-cards=<NAME>` options.
`-p, --persist`	Creates a persistent card set.

Option	Description
-P, --no-persist	Creates a non-persistent card set.
-q, --remotely-readable	Allows this card set to be read remotely. For information on configuring Remote OCSs, see Remote Operator. Not required for Remote Administration.
-Q, --ocs-quorum=<K>/<N>	<K> is the minimum required number of cards. If you do not specify the value <K>, the default is 1. Some applications do not have mechanisms for requesting that cards be inserted. Therefore any OCSs that you create for use with these applications must have <K>=1. <N> is the total number of cards. If you do not specify the value <N>, the default is 1.
-R, --no-pp-recovery	Specifies that passphrase replacement for this OCS is disabled. Setting this option overrides the default setting, which is that the card passphrases are replaceable. You can specify the enablement of passphrase replacement explic itly by setting the --pp-recovery option.
-T, --timeout=<TIME>	Sets the time-out for the card set. Use the suffix s to specify seconds, m for minutes, h for hours, and d for days. If the time-out is set to 0, the OCS never times out. Otherwise, the hardware security module automatically unloads the OCS when the amount of time spec ified by TIME has passed since the OCS was loaded.
Module selection	
-m, --module=MODULE	Specifies the number ID to use. If you only have one module, MODULE is 1. If you do not specify a module ID, createocs uses all modules by default.
Help options	
-h, --help	Displays help for createocs.
-u, --usage	Displays a brief usage summary for createocs.
-v, --version	Displays the version number of the Security World Software that deploys cre-ateocs.

51.1. Restrictions on using createocs

With Security World Software v11.72 and later, passphrases are limited to a maximum length of 254 characters, when using createocs. See Maximum passphrase length.

If you have created a FIPS 140 Level 3 compliant Security World, you must provide autho-rization to create new Operator Cards; createocs prompts you to insert a card that contains this authorization. Insert any card from the Administrator Card Set or any Operator Card

from the current Security World.

52. cryptest

```
cryptest [-esES] [-m BYTES] [-b BYTES]
```



Only supported in FIPS 140-2 Level 2 Security Worlds.

Tests all defined symmetric cryptographic mechanisms, encrypt/decrypt and sign/verify operations.

Option	Description
`-b`, `--channel-block-size=BYTES`	Specifies the block (chunk) size when using channels.
`-e`, `--encryption`	Tests the confidentiality (encryption) mechanisms.
`-E`, `--channel-encrypt`	Uses explicit symmetric channel operations to encrypt or sign.
`-m`, `--max-size=BYTES`	Specifies the maximum size of encryption tests to run.
`-s`, `--signature`	Tests the integrity (signature, hash, MAC, and HMAC) mechanisms.
`-S`, `--channel-decrypt`	Uses explicit symmetric channel operations to decrypt or verify.
Option to address HSMs	
`-M`, `--module=MODULE`	Specifies the number of the module to perform the tests on. If you only have one module, `<MODULE>` is 1. If you do not specify a module number, the utility uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `cryptest`.
`-u`, `--usage`	Displays a brief usage summary for `cryptest`.
`--version`	Displays the version number of the Security World Software that deploys `cryptest`.

53. csadmin

```
csadmin [-h] [-v] {image,load,start,stop,list,destroy,sshd,ids,log,config,stats}
```

Main utility to invoke CodeSafe 5 sub-utilities.

Positional arguments	
Option	Description
`config`	Manages the SEE machine configuration on an nShield 5 HSM.
`destroy`	Destroys a loaded SEE machine.
`ids`	Manages the CodeSafe Developer Authentication Certificates on an nShield 5 HSM.
`image`	Performs tasks related to loadable CodeSafe 5 images.
`list`	Lists the SEE machines loaded on an nShield 5 HSM.
`load`	Loads a Codesafe 5 image onto an nShield 5 HSM.
`log`	Manages the SEE machine logging state on an nShield 5 HSM.
`sshd`	Manages the SSH daemon dedicated to a specific SEE machine on an nShield 5 HSM.
`start`	Starts a previously loaded image on an nShield 5 HSM.
`stats`	Manages the SEE machine statistics on an nShield 5 HSM.
`stop`	Stops an SEE machine running on an nShield 5 HSM.
Help options	
`-h`, `--help`	Displays help for `csadmin`.
`--version`	Displays the version number of the Security World Software that deploys `csadmin`.

54. cspcheck, cspcheck64

Check that CSP container files and keys in the `%NFAST_KMDATA%` directory are intact and uncorrupted and that the referenced key files exist.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They also enable you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

55. cspimport, cspimport64

Insert keys manually into existing CSP containers. See Installing the CAPI CSP.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They also enable you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

56. cspmigrate, cspmigrate64

Move CSP container information for an existing Security World from the registry into the Security World.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They also enable you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

57. cspnvfix, cspnvfix64

Regenerate the NVRAM key counter area for a specified nShield CSP key.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They also enable you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

58. csptest, csptest64

Test the installed Cryptographic Service Providers.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They also enable you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

59. csputils, csputils64

Obtain detailed information about CSP containers.



You must have Administrator privileges to view or delete machine containers or containers that belong to other users.

Use these utilities to migrate from Windows registry-based CSP container storage to the new CSP formats. They also enable you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

60. date

```
date [MMDDhhmm[YYYY][.ss]]
```

Gets or sets the HSM system time

Option	Description
`date`	Date in `[MMDDhhmm[YYYY][.ss]]` format. `YYYY` has to be between `2000` and `2037`.

61. des_kat

```
des_kat
```

Only supported in FIPS 140-2 Level 2 Security Worlds.

Performs DES known-answer tests and indicates if any of them fail.

Option	Description
Option to address HSMs	
-m, --module=<MODULE>	Specifies the number of the module to erase cards. May be repeated, default = all.
Help options	
-h, --help	Displays help for des_kat.
-u, --usage	Displays a brief usage summary for des_kat.
-v, --version	Displays the version number of the Security World Software that deploys des_kat.

62. display-pubkey

```
display-pubkey [OPTIONS] APP IDENT
```

Displays the public key.

Option	Description
Option to address HSMs	
`-m`, `--module=<MODULE>`	Specifies the number of the module to erase cards. May be repeated, default = all.
Help options	
`-h`, `--help`	Displays help for `display-pubkey`.
`-u`, `--usage`	Displays a brief usage summary for `display-pubkey`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `display-pubkey`.

63. dump-marshalled

```
dump-marshalled [-p|-r|-i] [-a|-b TAG|-B TAG|-x TYPE] [-o FILE] [-t TAG]
                TYPE [FILE...]
```

With the `-p` option, dump-marshalled reads binary data from each FILE on its command line (or just standard input if no FILE is given), interprets it as an object of the given TYPE, and writes a printed dump to standard output. The names in the dump are prefixed by the TYPE name.

With `-r`, dump-marshalled reads a printed dump of an object of the given TYPE from each of the FILEs on its command line (or just standard input), and writes the corresponding binary data to standard output. The names in the dump must be prefixed by the TYPE name.

With `-i`, dump-marshalled reads an object of the given TYPE from standard input interactively, and writes the corresponding binary data to standard output.

This behaviour is modified by the other options as follows:

- a TAG specified by `-t` is used in place of the named TYPE as a prefix for the object`s member names
- an output FILE named by `-o` is used in place of standard output as the place where out put (of whatever kind) is written
- the `-a`, `-b`, `-B` and `-x` options perform encoding or decoding of binary data, as appropriate

Option	Description
Mode selection	
`-c`, `--check`	Exits zero if `TYPE` is recognized, nonzero otherwise.
`-i`, `--interactive-read`	Reads interactively, writes binary.
`-l`, `--list`	Lists the types currently supported.
`-p`, `--print`	Reads binary, writes print dump (default).
`-r`, `--read`	Reads print dump, writes binary.
Binary encoding options	
`-a`, `--base64`	Base-64 encoding.
`-b`, `--byteblock=TAG`	ByteBlock, in print dump format, with `TAG`.
`-B`, `--interactive-byte-block=TAG`	ByteBlock, reads interactively, with `TAG`.

Option	Description
`-x`, `--nchex=TYPE`	nCHex format, with `TYPE`.
Other options	
`-o`, `--output=FILE`	Writes the output to `FILE` rather than `stdout`.
`-t`, `--tag=TAG`	Uses `TAG` as the prefix for textual data.
Help options	
`-h`, `--help`	Displays help for `dump-marshalled`.
`-u`, `--usage`	Displays a brief usage summary for `dump-marshalled`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `dump-marshalled`.

64. edit-world

Edits parameters on an existing security world.

```
edit-world [ACTION]
```

64.1. Prerequisites for using edit-world

- A pre-existing Security World must exist and be loaded on at least 1 HSM.
- If changing the world, the Administrative cardset will be needed to authorize the changes.

64.2. edit-world syntax

Without any parameters specified the `edit-world` utility will list the editable Security World parameters.

64.3. edit-world NAME=VALUE syntax

The `edit-world` utility allows the changing of parameters using NAME=VALUE as the input.

- NAME is the target parameter to be altered
- VALUE is either 0 (disable) or 1 (enable)

64.4. edit-world [OPTIONS]

|`-h`, `--help` |Displays help for `new-world`.

|`-v`, `--version` |Displays the version number of the Security World Software that deploys `edit-world`.

|`-u`, `--usage` |Displays a brief usage summary for `edit-world`.

64.5. edit-world examples

Example 1

```
edit-world
```

```
StrictSP80056Ar3=0|1 Enforce strict SP800-56Ar3 compliance
```

List editable Security World parameters. Currently the only support parameter is SP800-56Ar3 compliance.

Example 2

```
edit-world StrictSP80056Ar3=0
```

Disable SP800-56Ar3 compliance in the Security World (FIPS-140 Level 3 worlds only)

Example 3

```
edit-world StrictSP80056Ar3=1
```

Enable SP800-56Ar3 compliance in the Security World (FIPS-140 Level 3 worlds only)

65. elftool

Converts ELF format executables into a format suitable for loading as an SEE machine.

66. enquiry

```
enquiry [-m MODULE]
```

Obtain information about the hardserver (Security World Software server) and the modules connected to it.

- Check if the software has been installed correctly
- Check the firmware version
- Check if the Remote Operator feature is enabled
- **On a network-attached HSM**: Check if the Serial Console feature is available

  On a PCIe or USB-attached HSM: Check the hardware status of the HSM
- **On a network-attached HSM**: Check the hardware status of internal security modules

See:

- **network-attached HSM**: Testing the installation
- **PCIe or USB-attached HSM**: Checking the installation

Option	Description
Connection options	
`--pool`	Views the pool of HSMs as a single resource.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to perform the tests on. If you only have one module, `<MODULE>` is `1`. If you specify module `0`, `enquiry` prints data from only the hardserver. If you do not specify a module number, the utility uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `enquiry`.
`-u`, `--usage`	Displays a brief usage summary for `enquiry`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `enquiry`.

66.1. enquiry output info

`enquiry` displays information similar to that shown in the following example:



The output for remote modules contains the `connection status` and `connection info` fields. These fields are absent for local modules.

```
Server:
 enquiry reply flags    none
 enquiry reply level    Six
 serial number          A815-03E0-D947
 mode                   operational
 version                12.81.2
 speed index            478
 rec. queue             374..574
 level one flags        Hardware HasTokens SupportsCommandState
 version string         12.81.2-393-7b3f83e, 13.3.1-210-bfe23daa, Bootloader: 1.2.3, Security Processor: 13.3.1 ,
13.4.3-349-5a0b72d8
 checked in             00000000623c858f Thu Mar 24 10:51:59 2022
 level two flags        none
 max. write size        8192
 level three flags      KeyStorage
 level four flags       OrderlyClearUnit HasRTC HasNVRAM HasNSOPermsCmd ServerHasPollCmds FastPollSlotList HasSEE
HasKLF HasShareACL HasFeatureEnable HasFileOp HasLongJobs ServerHasLongJobs AESModuleKeys NTokenCmds
JobFragmentation LongJobsPreferred Type2Smartcard ServerHasCreateClient HasInitialiseUnitEx AlwaysUseStrongPrimes
Type3Smartcard HasKLF2
 module type code       0
 product name           nFast server
 device name
 EnquirySix version     8
 impath kx groups
 feature ctrl flags     none
 features enabled       none
 version serial         0
 level six flags        none
 remote port (IPv4)     9004
 kneti hash             5e2ade32b47dde562a4b3f6a9c11eb75b0f40b47
 rec. LongJobs queue    0
 SEE machine type       None
 supported KML types
 active modes           none
 remote port (IPv6)     9004

Module #1:
 enquiry reply flags    none
 enquiry reply level    Six
 serial number          A815-03E0-D947
 mode                   operational
 version                13.3.1
 speed index            478
 rec. queue             43..152
 level one flags        Hardware HasTokens SupportsCommandState SupportsHotReset
 version string         13.3.1-210-bfe23daa, Bootloader: 1.2.3, Security Processor: 13.3.1 , 13.4.3-349-5a0b72d8
 checked in             0000000063b6f493 Thu Jan 5 11:02:27 2023
 level two flags        none
 max. write size        8192
 level three flags      KeyStorage
 level four flags       OrderlyClearUnit HasRTC HasNVRAM HasNSOPermsCmd ServerHasPollCmds FastPollSlotList HasSEE
HasKLF HasShareACL HasFeatureEnable HasFileOp HasLongJobs ServerHasLongJobs AESModuleKeys NTokenCmds
JobFragmentation LongJobsPreferred Type2Smartcard ServerHasCreateClient HasInitialiseUnitEx AlwaysUseStrongPrimes
Type3Smartcard HasKLF2
 module type code       12
 product name           nC3025E/nC4035E/nC4335N
 device name            Rt1
 EnquirySix version     7
 impath kx groups       DHPrime1024 DHPrime3072 DHPrime3072Ex DHPrimeMODP3072
 feature ctrl flags     LongTerm
 features enabled       RemoteShare GeneralSEE StandardKM EllipticCurve ECCMQV AcceleratedECC HSMBaseSpeed
 version serial         37
```

```
 connection status      OK
 connection info        esn = A815-03E0-D947; addr = INET/192.168.156.32/9004; ku hash =
3a75d883a3bca6e3d277ea3ca0f9179b31ed40c3, mech = Any
 image version          13.4.3-294-5a0b72d8
 level six flags        SerialConsoleAvailable
 max exported modules   4100
 rec. LongJobs queue    42
 SEE machine type       PowerPCELF
 supported KML types    DSAp1024s160 DSAp3072s256
 using impath kx grp    DHPrimeMODP3072
 active modes           UseFIPSApprovedInternalMechanisms AlwaysUseStrongPrimes  FIPSLevel3Enforcedv2
 hardware status        OK
```

66.2. Flag explanations

66.2.1. Level one flags

Flag	Explanation
`Hardware`	Set if this is a hardware module.
`HasTokens`	Set if the module has a hardware token interface, such as a smart card reader.
`MaintenanceMode`	The module is in maintenance mode.
`InitialisationMode`	The module is in initialisation mode.
`PreMaintInitMode`	The module is in pre-maintenance or pre-initialisation mode.
`Uninitialised`	**Firmware versions earlier than 13.5:** The module enters this state following a firmware upgrade. When in this state it cannot be used, it can only be changed into the pre-maintenance or pre-initialisation states to load new firmware or be initialised. **Firmware versions 13.5 and later:** This flag is never set. The module enters pre-initialisation mode following a firmware upgrade.
`SupportsCommandState`	The firmware supports the `state` field in `Command` (for HSM Pool Mode).
`SupportsHotReset`	The firmware supports hot reset (for `nopclearfail -S` with Solo XC).

66.2.2. Level two flags

These flags are not used in practise. The `Level two flags` value will always be `none`.

66.2.3. Level three flags

Flag	Explanation
KeyStorage	The module is capable of key management functions.

66.2.4. Level four flags

Flag	Explanation
OrderlyClearUnit	The module supports Cmd_ClearUnit. If this flag is set, the server will clear the module whenever the server is started.
HasRTC	The module has an onboard real-time clock.
HasNVRAM	The module has onboard nonvolatile memory.
HasNSOPermsCmd	The module supports the SetNSOPermsCmd command.
ServerHasPollCmds	The server supports the PollModuleState and PollSlotList commands.
FastPollSlotList	The module issues asynchronous notifications to the server when tokens are inserted, removed, or modified.
HasSEE	The module supports the Secure Execution Engine (SEE).
HasKLF	The module has a KLF long-term fixed signing key.
HasShareACL	The module supports setting ACLs on logical token shares, the impath commands, and the Send and Receive commands.
HasFeatureEnable	The module supports feature-enabled functions.
HasFileOp	The module supports operations using nonvolatile memory, and the FileCopy, FileCreate, FileErase, FileOp, LoadBlob and MakeBlob commands.
HasPCIPush	The module supports the PCI push interface. This increases the speed of commands on the PCI bus, improving performance for certain channel commands.
HasKernelInterface	The module has a separate logical interface capable of receiving jobs from, for instance, the OS kernel. This facility requires support from the driver.
HasLongJobs	The module supports the command flag Command_flags_LongJob and will not time out commands with this flag set.
ServerHasLongJobs	The hardserver understands the command flag Command_flag_LongJobs and will correctly wait for the module to complete commands that have this flag set. Clients must only set the Command_flag_LongJobs flag if the server supports it; otherwise the server may declare the module to have failed. For a job to be processed as a LongJob, the module and all servers handling the job must support long jobs.

Flag	Explanation
AESModuleKeys	The module supports AES module keys.
NTokenCmds	The module is an nToken.
JobFragmentation	The module supports fragmentation of large commands and replies to and from the host.
LongJobsPreferred	The module is happy to receive all commands as LongJobs, that is jobs with no timeout.
Type2Smartcard	The module supports type 2 (Payflex) smartcards.
ServerHasCreateClient	The server can accept the CreateClient command in place of NewClient, and store information about the process for associating connections with applications.
HasInitialiseUnitEx	The module supports the Cmd_InitialiseUnitEx command.
AlwaysUseStrongPrimes	The module is behaving as if the UseStrongPrimes flag was present for all RSA key generations.
Type3Smartcard	The module supports type 3 smartcards (original Remote Administration Ready Athena Javacards supported v12.0 onwards).
HasKLF2	The module has a KLF2 long-term fixed signing key.
DisablePKCS1Padding	All cryptographic mechanisms which use PKCS #1 v1.5 padding are disabled. If this is enabled, raw RSA encryption/decryption is still supported by the RSA OAEP mechanisms.
HasPCIPushPull	The module supports the PCI push pull interface. This increases the speed of commands on the PCI bus in both directions, improving performance for certain channel commands.

66.2.5. Level six flags

Flag	Explanation
SerialConsoleAvailable	This is a remote module with a serial console.
Type3SmartcardRevB	The module supports type 3 revision B smartcards (NXP JCOP Javacards, second generation of Remote Administration Ready).

67. esn

```
esn
```

Shows the Electronic Serial Number (ESN) of the HSM.

Option	Description
esn	Shows the Electronic Serial Number (ESN) of the HSM.

68. factorystate

```
factorystate
```

Resets the module to its original (factory) state.

Option	Description
`factorystate`	Resets the module to its original (factory) state.  Also resets the IP address and the serial console settings of the module.

69. fet

```
fet [-m MODULE] [-e ESN] [-c FILENAME] [-r|-R] [-s] [-a]
```

- Activates features on an nShield module connected to the host
- Lists the status of features on a connected module
- Verifies that a feature has been successfully enabled on a connected module

To view the status of features, run the tool without a smart card. If a FEM card is not present, or if any of the features are not enabled successfully, the utility prompts you to indicate what to do next.



To enable features, and view the status of or verify features on an nShield HSM, use the front panel rather than the `fet` utility.

For more information, see Optional features

Option	Description
`-a`, `--show-all`	Shows all features, including obsolete ones. Default: `0`
`-c`, `--cert=FILENAME`	Certificate file.
`-e`, `--esn=ESN`	ESN the certificate(s) are destined for.
`-r`, `--reset-module`	Forces module reset when needed.
`-R`, `--skip-reset`	Does not ask for module reset.
`-s`, `--show-only`	Shows the features then exits. Default: `0`
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to perform the tests on. If you only have one module, `<MODULE>` is `1`. If you do not specify a module number, the utility uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `fet`.
`-u`, `--usage`	Displays a brief usage summary for `fet`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `fet`.

70. floodtest

```
floodtest [options]
```

Performs hardware speed-testing by using modular exponentiation.

If skew or threshold checking is enabled (they are mutually exclusive), the average number of operations per second is recorded at `TIME`.

If skew checking is enabled, each subsequent operation must be within `SKEW` of the recorded average. If threshold checking is enabled, the average must stay above `COUNT` after checking starts. If either of these conditions is not met, the application terminates.

Option	Description
Program options	
`--crt`	Performs ModExps using the Chinese Remainder Theorem (default).
`--no-crt`	Perform ModExps without using CRT.
`-j`, `--outstanding-jobs=COUNT`	Sets the maximum number of outstanding jobs. Default: minimum number recommended for the hardserver + 1.
`-l`, `--job-size=BITS`	Sets the size of each ModExp in bits. Default = `1024`
`-L`, `--longjobs`	Sets the `LongJobs` flag in crypto commands.
`-n`, `--jobs-count=COUNT`	Sets the maximum number of jobs. Default: infinite.
`-Q`, `--query`	Uses Query mode (spinlock) rather than Wait mode.
`-R`, `--no-round-robin`	Accepts replies in any order. Default: `round-robin`
`-t`, `--stop-after=LENGTH`	Sets the maximum time to run, in seconds. Default: infinite.
Automatic checking options	
`-C`, `--check-start=TIME`	Specifies when skew or threshold checking commences, in seconds. Default: `15`, rounded up to nearest multiple of `INTERVAL`.
`-K`, `--skew-check=SKEW`	Turns skew checking on.
`-T`, `--min-check=COUNT`	Turns threshold checking on.
Output options	
`-o`, `--output=FILE`	Send output to named file as well as to stdout.

Option	Description
`--overprint`	Print results all on one line, using \r rather than \n.
`-r`, `--report-interval=INTERVAL`	Set the statistics reporting interval in seconds (default = 1).
Help options	
`-h`, `--help`	Displays help for `floodtest`.
`-u`, `--usage`	Displays a brief usage summary for `floodtest`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `floodtest`.

71. fwcheck

```
fwcheck [-v] [-m MODULE] [-c HEX] <firmware file>
```

Verifies the firmware installed on a module. Supported firmware files: `.nff` and `.ftv`.

Option	Description
`-c`, `--challenge=CHAL`	Uses `CHAL` as the firmware challenge. Default: a random challenge.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to perform the tests on. If you only have one module, `<MODULE>` is `1`. If you do not specify a module number, the utility uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `fwcheck`.
`-u`, `--usage`	Displays a brief usage summary for `fwcheck`.
`-v`, `--verbose`	Prints verbose output for `fwcheck`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `fwcheck`.

72. fwversion

```
fwversion
```

Shows the firmware version for nShield 5s inside the nShield 5c HSM.

Option	Description
`fwversion`	Shows the firmware version for nShield 5c HSMs.

73. gateway

```
gateway [address]
```

Gets or sets the default IPv4 gateway address.

Option	Description
address	The IPv4 address for the gateway.

74. gateway6

```
gateway6
gateway6 [gateway=::] [linklocal_if=0]
```

Gets or sets the default IPv6 gateway address of the module.

Option	Description
`gateway`	IPv6 address for the gateway.
`linklocal_if`	The ethernet interface (`0` or `1`) to use if the IPv6 default gateway address is a link-local address. The information is not used if the IPv6 default gateway is not a link-local address. Default: `0`.

75. generatekey

```
generatekey [OPTIONS] APP [NAME=VALUE...]
```

Generates, imports, or retargets keys.

This utility is included in the `Core Tools` bundle, which contains all the Security World Software utilities. For more information, see:

- Generating keys with the command line.
- Importing keys from the command line.
- Example of key generation with generatekey, for an example of key generation in batch mode.
- Example of key importation with generatekey, for an example of importing an RSA key.
- Listing supported applications with generatekey.
- Retargeting keys with generatekey.

Option	Description
Modes of operation	
`-a`, `--list-apps`	Lists all recognised `APPNAMEs`.
`-g`, `--generate`	Generates a key (default).
`-i`, `--import`	Imports a key.
`-l`, `--list-params`	Lists all available parameters for a given `APPNAME`.
`-r`, `--retarget`	Retargets a key.
`-t`, `--list-key-types`	Lists all available key types for a given `APPNAME`.
Module and cardset selection	
`-c`, `--cardset=NAME`	Selects the cardset to protect key.
`-m`, `--module=MODULE`	Specifies the number of the module to perform the tests on. If you only have one module, `<MODULE>` is `1`. If you do not specify a module number, the utility uses all modules by default.
`-s`, `--slot=SLOT`	Selects the slot to use.
Key protection options	
`--force-see`	Offers SEE options even with non-SEE modules.
`--trusted-certifier=HASH`	Trust the seeinteg certifier with this hash. This option can also take a list of key hashes separated by commas or spaces. It instructs `generatekey` to regard these keys as trusted even if they cannot be verified.

Option	Description
`-N`, `--no-verify`	Does NOT verifies the security of the key.
`--verify`	Verifies the security of the key (default).
Other settings	
`-b`, `--batch`	Runs in non-interactive mode.
`-I`, `--interactive`	Runs in interactive mode (default).
`-n`, `--check`	Checks the settings.
`-q`, `--quiet`	Runs in quiet mode.
`-v`, `--verbose`	Prints verbose output for `generatekey`.
Help options	
`-h`, `--help`	Displays help for `generatekey`.
`-H`, `--help-parameters`	Lists all `generatekey` parameters.
`-u`, `--usage`	Displays a brief usage summary for `generatekey`.
`-v`, `--verbose`	Prints verbose output for `generatekey`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `generatekey`.

76. getrtc

```
getrtc
```

Gets the real-time clock of an nShield 5c HSM.

Option	Description
`getrtc`	Gets the real-time clock of an nShield 5c HSM. Only supported in Security World Software v13.3 or later.

77. hakever

```
hakever [option|file]
```

Option	Description
`--ecr2csr`	Next input file is a GCR and will be converted to a standard CSR on stdout, if successful.
`--ecr-chk`	Next input file is a GCR and will only be checked.
`--ev-chk`	Next input file is an evidence chain and will only be checked.
`--csr-chk`	Next input file is a standard X.509 certificate request and will only be checked for consistency of public key value with other input files.
`--cert-chk`	Next input file is a standard X.509 certificate and will only be checked for consistency of public key value with other input files.
`-Vstuff`	Passes information as an argument to `check-extract`.
`-T TABLE`	Uses `TABLE` instead of `/opt/nfast/lib/hardkey/vendors`.

78. help

```
help
```

Prints the help for the given command or list all commands if no command has been given.

79. hsc_codesafe

```
hsc_codesafe
```

Automatically loads, configures, and starts a CodeSafe 5 container on a specified nShield module. It is typically executed during module startup by the hardserver, but it can also be run interactively for troubleshooting or testing purposes.

79.1. Privileges and access requirements

All nCore operations require a privileged client. On Windows, run the command from an elevated command prompt. On Linux, the invoking user must be a member of the `nfast` group.

Launcher (`csadmin`) operations additionally require membership of the `nfastadmin` group on Linux, or an elevated command prompt on Windows.

For auto-enrollment to succeed with an nShield 5 HSM, the module must be enrolled as privileged on the local hard server. You must run this utility as `nfast` or `root` on Linux, or from an elevated command prompt on Windows.

79.2. Options

Option	Description
`-m MODULE`, `--module MODULE`	Module number of the HSM on which to load the CodeSafe container.
`-c CONFIG`, `--config CONFIG`	Configuration file path. You only need this option if the configuration file is not in the standard location.
`-h`, `--help`	Displays help for `hsc_codesafe`.

79.3. Positional arguments and interactive execution

`hsc_codesafe` accepts optional positional arguments in the form `name=value`. These arguments override fields from the `[codesafe]` section of the configuration file for the current invocation. The argument names must exactly match field names in the CodeSafe configuration.

If a `--` argument separator is supplied, all subsequent arguments are treated as a command line to run after the CodeSafe container has been loaded. The following special values are supported:

- `pause` Runs until interrupted with Ctrl-C.
- `exit` Terminates successfully after the container has finished loading.

When a command line is executed in this way, the SEEJobs SSH tunnel and any published objects are removed after the command terminates. The exit code of `hsc_codesafe` is the exit code of the executed command.

If no command line is provided, the behavior matches execution during hardserver startup. The process continues running only if an SSH tunnel has been established for SEEJobs support. In this case, any published object is removed automatically.

80. hsc_configurepoolmodule

```
hsc_configurepoolmodule
```

Checks a single module and either adds it to or removes it from the logical module.

Option	Description
Option to address HSMs	
`-m, --module=MODULE`	Number of the module to configure. Default is to configure all eligible.
Help options	
`-h, --help`	Displays help for `hsc_configurepoolmodule`.
`-u, --usage`	Displays a brief usage summary for `hsc_configurepoolmodule`.
`-v, --version`	Displays the version number of the Security World Software that deploys `hsc_configurepoolmodule`.

81. hsc_configureslots

```
hsc_configureslots
```

Performs slot setup for modules served by this hardserver according to the values in the configuration file through the following operations:

- Dynamic slots are created
- Specified remote slots are imported
- Specified slot exports are permitted
- Slot mapping is applied

Option	Description
`-c, --configfile=FILENAME`	Name of the config file to read.
`-d, --debug`	Outputs debug information.
Option to address HSMs	
`-m, --module=MODULE`	Number of the module to configure. Default is to configure all eligible.
Help options	
`-h, --help`	Displays help for `hsc_configureslots`.
`-u, --usage`	Displays a brief usage summary for `hsc_configureslots`.
`-v, --version`	Displays the version number of the Security World Software that deploys `hsc_configureslots`.

82. hsc_loadseemachine

```
hsc_loadseemachine
```

Loads an SEE machine into each module that is configured to receive one, then publishes a newly created SEE World, if appropriate.

Option	Description
`-c, --configfile=FILENAME`	Name of the config file to read.
Option to address HSMs	
`-m, --module=MODULE`	Select one module to read config data for (default = `0`). Only with this option will a SEE machine be loaded. `hsc_loadseemachine` does nothing when called without `-m`.
Help options	
`-h`, `--help`	Displays help for `hsc_loadseemachine`.
`-u`, `--usage`	Displays a brief usage summary for `hsc_loadseemachine`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `hsc_loadseemachine`.

83. hsc_loadwarrants

```
hsc_loadwarrants
```

Loads installed warrants for modules served by this hardserver.

Option	Description
Option to address HSMs	
`-m, --module=MODULE`	Number of the module to load a warrant for
Help options	
`-h, --help`	Displays help for `hsc_loadwarrants`.
`-u, --usage`	Displays a brief usage summary for `hsc_loadwarrants`.
`-v, --version`	Displays the version number of the Security World Software that deploys `hsc_loadwarrants`.

84. hsc_nethsmexports

```
hsc_nethsmexports
```

Checks the FEM certificate and setup hardserver permissions to reflect the exports declared in the configuration file.

Option	Description
`-c, --configfile=FILENAME`	Name of the config file to read.
`--setmaxnumclients`	Always issue the `SetMaxNumClients` command.
Option to address HSMs	
`-m, --module=MODULE`	Select the module to use for verification (default = `1`).
Help options	
`-h`, `--help`	Displays help for `hsc_nethsmexports`.
`-u`, `--usage`	Displays a brief usage summary for `hsc_nethsmexports`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `hsc_nethsmexports`.

85. hsc_nethsmimports

```
hsc_nethsmimports
```

Configures the hardserver according to the `[nethsm_imports]` section in the configuration file.

Option	Description
`-c, --configfile=FILENAME`	Name of the config file to read.
`-w, --wait`	Waits for remote modules (default)
`-W, --no-wait`	Doesn't wait for remote modules
Help options	
`-h, --help`	Displays help for `hsc_nethsmimports`.
`-u, --usage`	Displays a brief usage summary for `hsc_nethsmimports`.
`-v, --version`	Displays the version number of the Security World Software that deploys `hsc_nethsmimports`.

86. hsc_remotefilesystem

```
hsc_remotefilesystem
```

Configures the remote server permissions related to file transfer in the hardserver.

Option	Description
`-c, --configfile=FILENAME`	Name of the config file to read.
Help options	
`-h, --help`	Displays help for `hsc_remotefilesystem`.
`-u, --usage`	Displays a brief usage summary for `hsc_remotefilesystem`.
`-v, --version`	Displays the version number of the Security World Software that deploys `hsc_remotefilesystem`.

87. hsc_serverremotecomms

```
hsc_serverremotecomms
```

Reads the `[server_remotecomms]` section of configuration file and restart the hardserver if any are different from the values in use.

Option	Description
`-c, --configfile=FILENAME`	Name of the config file to read.
Help options	
`-h`, `--help`	Displays help for `hsc_serverremotecomms`.
`-u`, `--usage`	Displays a brief usage summary for `hsc_serverremotecomms`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `hsc_serverremotecomms`.

88. hsc_serversettings

```
hsc_serversettings
```

Tells the hardserver to reread the [server_settings] and [module_settings] sections in the configuration file.

Option	Description
-n, --newconfig	Tells the hardserver to read the proposed new configuration file.
Help options	
-h, --help	Displays help for hsc_serversettings.
-u, --usage	Displays a brief usage summary for hsc_serversettings.
-v, --version	Displays the version number of the Security World Software that deploys hsc_serversettings.

89. Administration of platform services (nShield 5 HSMs)

nShield 5s platform services are administered through the unified utility `hsmadmin`, which directs the command to the service that implements the command.

Some commands require elevated privileges by default because both the permissions and the protection settings have an impact on the usability of the keys by non-administrative users. Commands that create keys or modify configuration always require elevated privileges. Elevated privileges mean `root` on Linux, and the built-in local Administrators group (running in an elevated shell) on Windows. If a command requires elevated privileges, this is indicated in the command description.

You can modify the permissions and protection options on service keys to allow particular groups of users to execute commands that require the private key for a given service. See Permissions on SSH keys and Setting protection on SSH keys.

All of the platform services are administered by a unified utility called `hsmadmin`

89.1. hsmadmin

The `hsmadmin` utility manages the administration of nShield HSMs using different subcommands.

```
hsmadmin <subcommand> [-h] [-v] [--no-reset]
```

You can use one of the following subcommands each time you run `hsmadmin`:

- factorystate
- status
- npkginfo
- upgrade
- reset
- enroll
- keys
- logs
- info
- esn
- settime

- gettime
- setminvsn
- getenvstats
- cs5
- select

You can use the following options with `hsmadmin`:

- Reset options:
  - `--no-reset`: Does not trigger an immediate reset of any HSM. If a reset is required you must reboot the host system.
- Help options:
  - `-h`, `--help`: Displays help for `hsmadmin`.
  - `-v`, `--version`: Displays the version number of the Security World Software that deploys `fwcheck`.

89.1.1. hsmadmin factorystate

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`.

This command returns an HSM to the state it was in when it left the factory. This securely erases all user credentials and information. It resets the `sshadmin` SSH credential to the default.

```
hsmadmin factorystate [-h] [--timeout <TIMEOUT>] [--esn <ESN>] [--verbose]
```

This command takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.

Parameter	Description
`--esn`	Resets specific modules to factory state. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin factorystate --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If no ESNs are specified, the command resets all connected modules.
`--verbose`	Prints verbose logs.

89.1.2. hsmadmin status

This command displays the ESN and currently loaded firmware version for discovered HSMs. It also displays whether the current image is a primary or a recovery image. When used with the `--json` option it displays primary firmware version, recovery firmware version, and uboot version.

```
hsmadmin status [-h] [--esn <ESN>] [--timeout <TIMEOUT>] [--verbose] [--json]
```

This command takes the following parameters:

Parameter	Description
`--verbose`	Prints verbose logs.
`--json`	Prints the HSM firmware version and image version in JSON format.
`--esn`	Displays information for specified HSMs. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin status --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If you do not specify any ESNs, the command displays information for all connected HSMs.
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.

89.1.3. hsmadmin npkginfo

This command inspects the `npkg` file and displays the metadata.

```
hsmadmin npkginfo [--json] <NPKGFILE>
```

This command takes the following parameters:

Parameter	Description
`--json`	Prints metadata in JSON format.
`<NPKGFILE>`	Specifies the NPKG-format file to inspect.

89.1.4. hsmadmin upgrade

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`.

This command installs firmware packages in npkg format. The command can install both primary and recovery firmware.

```
hsmadmin upgrade [-h] --esn <ESN> [--timeout <TIMEOUT>] [--dry-run] [--force] [--verbose] [--json] <NPKGFILE>
```

This command takes the following parameters:

Parameter	Description
`--verbose`	Prints verbose logs.
`--json`	Prints metadata in JSON format.
`--dry-run`	Don't load the package, just validate it.
`--force`	Ignore warnings and force upgrade to proceed.
`--esn`	Specifies the HSMs in which to load the NPKG file. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin upgrade --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321 <NPKGFILE>`
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`<NPKGFILE>`	Specifies the npkg file to load in to the HSMs.

89.1.5. hsmadmin reset



Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`. If you run the command while in operational mode, it creates a failed state and you will need to run `nopclearfail -r -m <MODULEID>` to correct it.

This command resets the nShield HSM.

```
hsmadmin reset [-h] [--esn <ESN>]
```

This command takes the following parameters:

Parameter	Description
`--esn`	Specifies the HSMs to reset. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin reset --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If you do not specify any ESNs, all connected HSMs will be reset.

89.1.6. hsmadmin enroll

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`.

This command configures the SSH keys for the nShield HSM.



Linux-only

The install script calls this command automatically as `hsmadmin enroll --sshadmin-key /root/.ssh/id_nshield5_sshadmin`. This will generate SSH client keys and register them with the units if they have not been previously set up. If the `sshadmin` key is not found in its usual location under `/opt/nfast` then `/root/.ssh/id_nshield5_sshadmin` will be tried instead, so it is convenient to use `hsmadmin keys backup` to backup the key to this location (use the `--passphrase` option to make the key portable to other machines rather than restricted to the local machine).

> If `/root/.ssh/id_nshield5_sshadmin` does not already exist, and the `/root/.ssh` directory exists, `hsmadmin keys backup --local /root/.ssh/id_nshield5_sshadmin` will be called automatically by the install script if `hsmadmin enroll` succeeds to create a backup of the `sshadmin` key that is restricted to the local machine, as a precaution against accidental deletion of the `/opt/nfast/services/client` directory.

```
hsmadmin enroll [--timeout <TIMEOUT>] [--verbose] [--sshadmin-key <SSHADMIN_KEY>]
```

This command takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--verbose`	Prints verbose logs.
`--sshadmin-key`	Path to backup of `sshadmin` key to use if not present in the standard location.

89.1.7. hsmadmin keys

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`.

This command is used to manage the SSH keys currently loaded on a module.

```
hsmadmin keys [--timeout <TIMEOUT>] <subcommand>
```

This command takes the following parameter:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.

You can use one of the following subcommands with this command:

- show
- migrate

- roll
- backup
- restore
- remote-set
- remote-remove

89.1.7.1. hsmadmin keys show

This subcommand displays the public client and server keys used to communicate with the HSMs. For client keys, it also displays the time stamp held on the associated key file in the host file system.

```
hsmadmin keys show [--json] [--verbose]
```

This subcommand takes the following parameters:

Parameter	Description
`--json`	Prints output in JSON format.
`--verbose`	Prints verbose logs.

89.1.7.2. hsmadmin keys migrate

This subcommand changes the SSHAdmin client key on all connected modules to match a public key. The public key is derived from the private key specified in the subcommand.

```
hsmadmin keys migrate --privkeyfile <PRIVKEYFILE> [--json] [--verbose]
```

This subcommand takes the following parameters:

Parameter	Description
`--json`	Prints output in JSON format.
`--verbose`	Prints verbose logs.
`--privkeyfile`	Specifies the file containing the private key to be migrated to.

89.1.7.3. hsmadmin keys roll

This subcommand changes the client keys for all services.

See SSH Client Key Protection (nShield 5s HSMs) for information about protection options that can be set on keys during generation.

```
hsmadmin keys roll [--json] [--verbose]
```

This subcommand takes the following parameters:

Parameter	Description
`--json`	Prints output in JSON format.
`--verbose`	Prints verbose logs.

On Linux, the hardserver must be restarted in order to be able to use the new `ncoreapi` SSH client key after performing this operation, for example, with `/opt/nfast/sbin/init.d-ncipher restart`.

89.1.7.4. hsmadmin keys backup

This subcommand makes a backup of the private client key for the `sshadmin` service.



The backup key should be protected against unauthorized access. Refer to your security procedures for information on how to store the backup file.

```
hsmadmin keys backup [--passphrase] <FILE>
```

This subcommand takes the following parameters:

Parameter	Description
`--passphrase`, `-p`	Replace host key protection with passphrase protection.
`<FILE>`	Path to file in which to store backup.

If the `--passphrase` option is not supplied, then the existing `sshadmin` key file will be copied verbatim with whatever existing protections it has. By default, the `sshadmin` key is tied to the host machine and OS install, and will not be usable on another machine. A warning about this restriction to the local machine will be printed if the `--passphrase` option is omitted (add `--local` to indicate that this is the explicit choice in order to prevent the warning).

If the `--passphrase` option is used, then the `sshadmin` key will be loaded and re-encrypted using a user passphrase that must be supplied at the prompt. If the existing `sshadmin` key was also protected with a user passphrase (this is not the case by default), then there will

be a prompt for that key's passphrase too. The backup key will not be tied to the host machine in this case, and can be used to re-install the HSM on another machine.

On Linux, the backup file will be generated with owner and group matching the directory in which it is created, and readable by owner only.

89.1.7.5. hsmadmin keys restore

This subcommand restores the private client key for the `sshadmin` service from a backup file that has previously been created with the `hsmadmin keys backup` command.

Once the private client key for the `sshadmin` service has been successfully restored, this command will automatically configure all other SSH keys for the HSM.

```
hsmadmin keys restore <FILE>
```

This subcommand takes the following parameter:

Parameter	Description
`<FILE>`	Path to file previously created by `hsmadmin keys backup`

89.1.7.6. hsmadmin keys remote-set

This subcommand installs a specific SSH public key for remote access to one HSM service.

```
hsmadmin keys remote-set <SERVICE> <KEYTYPE> <KEYDATA>
```

This subcommand takes the following parameters:

Parameter	Description
`<SERVICE>`	HSM service to be accessed remotely
`<KEYTYPE>`	SSH public key type
`<KEYDATA>`	SSH public key

89.1.7.7. hsmadmin keys remote-remove

This subcommand removes a specific SSH public key that had previously been set for remote access and restores the local client key.

```
hsmadmin keys remote-remove <SERVICE>
```

This subcommand takes the following parameter:

Parameter	Description
<SERVICE>	HSM service from which to remove remote access

89.1.8. hsmadmin logs

This command manages the system logs of connected HSMs. These logs are separate from the `ncoreapi` logs. See Platform services and (nShield 5 HSMs) for more information about platform services and `ncoreapi`.

For more information about system logs, see System logging (nShield 5 HSMs).

For more information about managing `ncoreapi` logs, see Audit Logging.

```
hsmadmin logs <subcommand>
```

You can use one of the following subcommands with this command:

- get
- clear
- export
- expire
- getkey

89.1.8.1. hsmadmin logs get

This subcommand retrieves logs from a connected HSM.



HSMs running firmware version 13.5 or later can produce logs in either a signed or unsigned format. This subcommand will retrieve unsigned logs. To retrieve logs in a signed format, use the `export` subcommand.

```
hsmadmin logs get [-h] [--verbose] [--timeout <TIMEOUT>] --esn <ESN> --log <LOG> [--json | --out <OUTFILE>]
```

This subcommand takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Specifies the HSM from which to retrieve logs. Only one ESN can be used in the command to retrieve the logs of one specific HSM.
`--json`	Prints output in JSON format.
`--out`	Write logs to file specified by OUTFILE
`--verbose`	Prints verbose logs.
`--log`	Selects log to be retrieved. Options are `system`, `init`

89.1.8.2. hsmadmin logs clear

This subcommand clears logs from connected HSMs.



The `system` log can only be cleared using this command on firmware versions earlier than 13.5. The `system` log on HSMs running firmware version 13.5 or later is cleared using the `expire` command. See Logging, debugging, and diagnostics for more information. The `init` log can be cleared on all firmware versions using this command.

Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`.

```
hsmadmin logs clear [-h] [--verbose] [--timeout <TIMEOUT>] [--esn <ESN] --log <LOG> [--json]
```

This subcommand takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Specifies the HSMs from which to clear logs. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin logs clear --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321 --log <LOG>` If you do not specify any ESNs, logs will be cleared from all connected HSMs.

Parameter	Description
`--json`	Prints output in JSON format.
`--verbose`	Prints verbose logs.
`--log`	Selects log to be cleared. Options are `system`, `init`

89.1.8.3. hsmadmin logs export

This subcommand retrieves and validates signed logs from a connected HSM.



The directory used for storing the log files must exist before running this command.

```
hsmadmin logs export [-h] [--verbose] [--timeout <TIMEOUT>] [--esn <ESN>] [--saved] [--expire] [--json | --outdir <OUTDIR>]
```

This subcommand takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Specifies the HSM from which to export logs.
`--json`	Prints metadata in JSON format.
`--outdir`	Write logs to directory specified by OUTDIR
`--verbose`	Prints verbose logs.
`--expire`	Expire the log after exporting it
`--saved`	If not expired, re-export a previously saved log

89.1.8.4. hsmadmin logs expire

This subcommand expires saved system logs from a connected HSM.

```
hsmadmin logs expire [-h] [--verbose] [--timeout <TIMEOUT>] [--esn <ESN>] --seq <SEQ_NO> [--json]
```

This subcommand takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Specifies the HSM from which to expire logs.
`--json`	Prints output in JSON format.
`--seq`	expire the log identified by `<SEQ_NO>`
`--verbose`	Prints verbose logs.

89.1.8.5. hsmadmin logs getkey

This subcommand retrieves the system log signing key from a connected HSM.

```
hsmadmin logs getkey [-h] [--verbose] [--timeout <TIMEOUT>] [--esn <ESN>] [--json | --out <OUTFILE>]
```

This subcommand takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Specifies the HSM from which to retrieve the log signing key
`--json`	Prints output in JSON format.
`--out`	Write key to file specified by `<OUTFILE>`.
`--verbose`	Prints verbose logs.

89.1.9. hsmadmin info

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

This command returns information that was loaded in the HSM during manufacturing. This information is persistent even after returning the HSM to factory state.

```
hsmadmin info [-h] [--timeout <TIMEOUT>] [--esn <ESN>] [--verbose] [--json]
```

This command takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Returns information for the HSM identified by <ESN>. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin info --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If no ESNs are specified, the command returns information for all connected modules.
`--verbose`	Prints verbose logs.
`--json`	Prints output in JSON format.

89.1.10. hsmadmin esn

This command will translate between the Electronic Serial Number (ESN) for a nShield 5s HSM, and its corresponding Physical Serial Number (PSN).

```
hsmadmin esn [-h] [--from] [--json] serialno [serialno ...]
```

Parameter	Description
`--from`	If this option is given, `hsmadmin esn` converts the supplied PSN(s) to ESN values. If it is not given, it will convert ESN(s) to PSN(s).
`--json`	If this option is given, the command writes a JSON object to *stdout*, which is an object mapping the input `serialno` values to their translated counterparts.
`serialno`	If `--from` is specified, this is a PSN in the format `NN-AANNNN` where `N` is a decimal digit, and `A` is a letter or decimal digit. If `--from` is not specified, this is an ESN in the format `XXXX-XXXX-XXXX` where each `X` is a hexadecimal digit. More than one `serialno` can be specified. These values are not case-sensitive.



The nShield 5 HSMs use a fixed transformation to derive the ESN from the physical serial number, which is assigned during manufacturing. This allows one to be derived from the other, offline, without needing access

> to the HSM itself. The physical serial number is printed on the nShield 5s's label, which differs from the PSN on the outer case of a nShield 5c or 5c 10G unit. So, `hsmadmin esn` is mostly useful for users of nShield 5s HSMs.

89.1.11. hsmadmin settime

This command is used to synchronize the HSM system clock with the clock in the host PC.

See Setting the system clock for more information on managing the system clock.

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

To use this command without the `--adjust` parameter, the HSM must be in maintenance mode.



> Setting the system date and time without the `--adjust` parameter automatically resets the HSM.

```
hsmadmin settime [-h] [--adjust <adjust>] [--timeout <TIMEOUT>] [--esn <ESN>] [--verbose]
```

This command takes the following parameters:

Parameter	Description
`--adjust`	Optional parameter. If specified an HSM System clock drift calibration is executed.
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Sets the system date and time of specific modules. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin settime --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If no ESNs are specified, the command resets all connected modules. If the adjust parameter is specified, a module reset is not required.
`--verbose`	Prints verbose logs.

89.1.12. hsmadmin gettime

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

It returns the system date and time of the HSM.

```
hsmadmin gettime [-h] [--timeout <TIMEOUT>] [--esn <ESN>] [--verbose] [--json]
```

This command takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Returns information for the HSM identified by <ESN>. You need to add `--esn` before each ESN you include in the command, for example: `hsmadmin gettime --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If no ESNs are specified, the command returns the HSM system date and time for all connected modules.
`--verbose`	Prints verbose logs.
`--json`	Prints output in JSON format.

89.1.13. hsmadmin setminvsn

This command requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

Before running this command, place the unit in maintenance mode using `nopclearfail -M -m <MODULEID> -w`.

This command sets the minimum VSN number of the firmware which the HSM will in the future accept as an upgrade.

```
hsmadmin setminvsn [-h] [--timeout <TIMEOUT>] [--esn <ESN>] [--verbose] [--json] <VSN>
```

This command takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Sets the minimum VSN on the HSM identified by <ESN>. You need to add `--esn` before each ESN you include in the command, for example: `>hsmadmin setminvsn --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321 2` If no ESNs are specified, the command sets the minimum VSN on all connected HSMs.
`--verbose`	Prints verbose logs.
`--json`	Prints output in JSON format.
`<VSN>`	The minimum VSN to set. Once this command is executed, the HSM will no longer accept a command to upgrade to a firmware with a VSN lower than <VSN>. The new minimum VSN cannot be lower than the HSM's current VSN, and cannot be higher than the VSN of the firmware currently installed on the HSM.

89.1.14. hsmadmin getenvstats

This command returns the environmental monitoring statistics of the HSM.

Environmental monitoring statistics available depend on the model of the HSM, the hardware revision and the version of the firmware installed on the HSM.

For the nShield5 with firmware version 13.3 the available statistics are:

uptime	The time since the HSM was last rebooted, in seconds.
current_time	The current system time of the HSM.
mem_total	Total amount of physical RAM, in kilobytes.
msp_temp	Temperature recorded by the MSP sensor, in degrees C.
cpu_temp	Temperature recorded by the CPU sensor, in degrees C.
crypto_co_proc_temp	Temperature recorded by the cryptographic co-processor sensor, in degrees C.
voltage_t1022_core	Voltage drawn by the T1022 core chip.

voltage_t1022_ifc_io	Voltage drawn by the T1022 IFC I/O chip.
voltage_t1022_serdes	Voltage drawn by the T1022 SERDES chip.
voltage_t1022_serdes_io	Voltage drawn by the T1022 SERDES I/O chip.
voltage_c292_serdes	Voltage drawn by the C292 SERDES chip.
voltage_fpga_serdes	Voltage drawn by the FPGA SERDES chip.
voltage_c292_serdes_io	Voltage drawn by the C292 SERDES I/O chip.
voltage_fpga_serdes_io	Voltage drawn by the FPGA SERDES I/O chip.
voltage_msp_avcc	MSP Analogue Vcc.
voltage_ddr4_io_access	Voltage drawn by the DDR4 I/O access chip.
voltage_ddr4_io	Voltage drawn by the DDR4 I/O chip.
voltage_battery	Voltage supplied by the on-board battery.
voltage_pci_bus	Voltage drawn by the PCI bus.
max_temp	Highest temperature recorded by any temperature sensor since statistics were reset.
min_temp	Lowest temperature recorded by any temperature sensor since statistics were reset.
ais31_preliminary_alarm_count	AIS31 (RNG) preliminary alarm count.
spi_retries	SPI protocol failure count.
sp_i2c_total_failures	MSP430 I2C total failures.
sp_i2c_slave_failures	MSP430 I2C slave failures.
sp_temp_failures	MSP430 temperature failures.
sp_voltage_failures	MSP430 voltage failures.
host_bus_exceptions	PCI0 (Host) NPE and PE error count.
crypto_bus_exceptions	PCI1 (Crypto) NPE error count.
sp_sensor_cmd_failures	Read security processor handshake line failure count.
nvm_free_space	Free space on user NVRAM.
nvm_wear_level	Wear level on user NVRAM.
nvm_worn_blocks	Worn block count on user NVRAM.
bios_code	Not used; always reports 'None'
dfs_throttling	Whether CPU performance is currently degraded due to excessive heat.

```
hsmadmin getenvstats [-h] [--timeout <TIMEOUT>] [--esn <ESN>] [--verbose] [--json]
```

This command takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--esn`	Returns information for the HSM identified by <ESN>. You need to add `--esn` before each ESN you include in the command, for example: `>hsmadmin getenvstats --esn 1A23-BC45-6789 --esn 9Z87-YX65-4321` If no ESNs are specified, the command returns the environmental monitoring statistics of all connected modules.
`--verbose`	Prints verbose logs.
`--json`	Prints output in JSON format.

89.1.15. hsmadmin cs5

This command is used to manage some aspects of CodeSafe SEE machines running on the HSM.

See also csadmin for additional commands related to managing CodeSafe SEE machines.

```
hsmadmin cs5 <subcommand>
```

You can use the following subcommand with this command:

- stats

The following subcommands are only relevant to the nShield 5c. See CodeSafe setup for the nShield 5c for more detail about the subcommands.

- clientinfo
- genclientinfo
- enroll
- unenroll
- list

89.1.15.1. hsmadmin cs5 stats

This subcommand gets statistics from active SEE machines.

```
hsmadmin cs5 stats [--timeout TIMEOUT] [-u UUID] [--esn ESN] [--json]
```

This subcommand takes the following parameters:

Parameter	Description
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.
`--u`	UUID of SEE machine from which to obtain statistics. If no UUID is specified, statistics will be retrieved for all running SEE machines.
`--esn`	Returns statistics for the HSM identified by <ESN>. If no ESNs are specified, the command returns statistics for all connected modules.
`--json`	Prints output in JSON format.

89.1.16. hsmadmin select

This command is used to select configuration options that are not controlled by licenses.

This command can only be used when the unit is in maintenance mode. It requires `root` privileges on Linux and the privileges of the built-in local Administrators group on Windows.

```
hsmadmin select <option> [--esn ESN] [--json] [--timeout]
```

All select <option> commands support the following parameters:

Parameter	Description
`--esn`	Select this option on the HSM identified by <ESN>. If no ESNs are specified, the selection is applied to all connected modules.
`--json`	Prints output in JSON format. You can use the following options with this command:
`--timeout`	Time to wait for service response, in seconds. Default 30 seconds, minimum 3s, maximum 120s.

89.1.16.1. hsmadmin select acceleration

This option selects which algorithms will be accelerated.

```
hsmadmin select acceleration [--set ID | --show]
```

Profile IDs are case-sensitive and must be uppercase (for example, `HY`, `PK`).

This option takes the following parameters:

Parameter	Description
`--set`	Set the accelerator to use. Specify the profile ID in upper case (for example, `HY` or `PK`). To revert to the default profile, use `--set PK`. The module must be reset for the change to take effect.
`--show`	Show the available accelerators, the currently loaded accelerator, and any staged selection pending a module reset.

Example output from `--show`:

```
Available accelerators: PK HY
Loaded accelerator:     PK
Staged accelerator:     HY
```

See Cryptographic Acceleration Profiles for more information about the available profiles.

90. hsmdiagnose

Runs automated health check against nShield 5s or nShield 5c HSMs and saves the information to an XML file.



On Solo XC, Connect +, Connect XC and Edge HSMs, use nfdiag.

Option	Description
`-h, --help`	Shows the help message and exits.

91. initunit

```
initunit [-m MODULE]
```

Re-initializes modules into their factory state.

For more information, see Erasing a module with initunit.

Option	Description
`-k, --kml-type=KML-TYPE`	Configures KML type and key hash mechanism.
`-K, --list-kml-types`	Lists recognized KML types.
`-n, --ntoken`	Makes the module(s) into nTokens.
Option to address HSMs	
`-m, --module=<MODULE>`	Re-initializes module `<MODULE>`. If you only have one module, `<MODULE>` is `1`. If you do not specify a module number, the utility re-initializes all suitable modules.
Help options	
`-h, --help`	Displays help for `initunit`.
`-u, --usage`	Displays a brief usage summary for `initunit`.
`-v, --version`	Displays the version number of the Security World Software that deploys `initunit`.

92. killrecov

```
killrecov
```

Disables retrospective key recovery (operator cardset replacement) for Security Worlds. `killrecov` removes the relevant token shares from administrator cards.



This operation cannot be undone. Contact Support for more information about using this utility.

Option	Description
`-s, --slot=SLOT`	Selects the slot with card to use. Default: slot `0`
Option to address HSMs	
`-m, --module=<MODULE>`	Uses module `<MODULE>` to disable key recovery for the Security World. If you only have one module, `<MODULE>` is `1`. If you do not specify a module number, `killrecov` uses the most suitable module.
Help options	
`-h, --help`	Displays help for `killrecov`.
`-u, --usage`	Displays a brief usage summary for `killrecov`.
`-v, --version`	Displays the version number of the Security World Software that deploys `killrecov`.

93. km-plode

```
km-plode [OPTIONS] SRCPATH/SRCFILE --dest-dir DESTDIR
km-plode [OPTIONS] SRCDIR --dest-file DESTFILE
```

Breaks up a `kmdata` file into multiple files.

Option	Description
`SRCPATH/SRCFILE --dest-dir DESTDIR`	Extracts key data from `SRCPATH/SRCFILE` into `DESTDIR`. Default: `./SRCFILE.d`
`SRCDIR --dest-file DESTFILE`	Implodes `kmdata`. Default: current `kmdata`
Help options	
`-h`, `--help`	Displays help for `km-plode`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `fwcheck`.

94. kmfile-dump

```
kmfile-dump [-bpv] worldfile [worldfile ...]
```

Displays key management information from a Security World's `kmdata` file.

Option	Description
`b, --binary`	Displays all entries in binary.
`-N, --no-worldinfo`	Does not attempt to read Security World data files in the kmdata/local directory. This allows `kmfile-dump` to work when no hardserver is running at the cost of disabling annotations (`KM_sw`) on the output.
`-p, --plain`	Uses plain format for binary output: no offsetsor ASCII.
`-v, --verbose`	Displays binary dumps of key blobs.
`<worldfile>`	The file storing the World data, usually `/opt/nfast/kmdata/local/world` (**Linux**) or `%NFAST_KM-DATA%\local\world` (**Windows**). If no **WorldVersion** is received as a result of the command then the World is either version 1 or version 2. If a **WorldVersion** of either '2' or '3' is received then the World is version 3.
Help options	
`-h, --help`	Displays help for `kmfile-dump`.
`-u, --usage`	Displays a brief usage summary for `kmfile-dump`.
`-V, --version`	Displays the version number of the Security World Software that deploys `kmfile-dump`.

95. kneti

```
kneti
```

Show the Kneti hash, which then can be used for enrolling the HSM with clients.

96. kptest

Only supported in FIPS 140-2 Level 2 Security Worlds.

```
kptest [options]
```

Tests the consistency of encryption and decryption, or of signature and verification, with the RSA and DSA algorithms.

If skew or threshold checking is enabled (they are mutually exclusive), the average number of operations per second is recorded at TIME.

If skew checking is enabled, each subsequent operation must be within SKEW of the recorded average. If the condition is not met, the application terminates

If threshold checking is enabled, the average must stay above COUNT after checking starts. If the condition is not met, the application terminates.

Option	Description
Program options	
`-e, --encrypt-decrypt`	Tests the encrypt and decrypt operations. Default: for RSA.
`-i, --plain-size=SIZE`	Uses plaintext of `SIZE` sized bits. Default: `160`.
`-k, --key-regenerate=CHECKS`	Regenerates the key for every `CHECKS` number of checks. Default: never.
`-L, --longjobs`	Sets the `LongJobs` flag in crypto commands.
`-n, --jobs-count=COUNT`	Sets the maximum number of jobs. Default: infinite.
`-s, --sign-verify`	Tests the sign and verify operations. Default: for DSA/KCDSA/ECDSA.
`-t, --stop-after=LENGTH`	Sets the maximum time to run, in seconds. Default: infinite.
Key options	
`-c, --curve=CURVENAME`	Uses the curve named NAME. Default: `NISTP192`.
`-l, --key-size=BITS`	Sets the key size (default 1024).
`-M, --mechanism=MECH`	Uses mechanism MECH.

Option	Description
`-p, --plain-type=TYPE`	Uses plaintext type TYPE (Bignum, Hash or Bytes). The mechanism and plaintext types must be compatible with the key type.
`--pairwise-check`	Sets `PairwiseCheck` in the key generation command.
`-S, --key-type=TYPE`	Selects the key type to use — RSA (default), DSA, KCDSA, or ECDSA
`--strong`	For RSA, uses strong (ANSI X9.31) primes. For DSA, uses the `Strict` flag.
Automatic checking options	
`-C, --check-start=TIME`	Specifies when skew or threshold checking commences, in seconds, rounded up to nearest multiple of INTERVAL. Default: `15`.
`-K, --skew-check=SKEW`	Turns on skew checking.
`-T, --min-check=COUNT`	Turns on threshold checking.
Output options	
`--overprint`	Prints the results all on one line, using \r rather than \n.
`-o, --output=FILE`	Sends the output to a named file as well as to `stdout`.
`-r, --report-interval=INTERVAL`	Sets the statistics reporting interval in seconds. Default: `1`.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `kptest` uses all modules by default.
Help options	
`-h, --help`	Displays help for `kptest`.
`-u, --usage`	Displays a brief usage summary for `kptest`.
`-v, --version`	Displays the version number of the Security World Software that deploys `kptest`.

97. keytst, keytst64

Creates, tests, and displays information about keys and CSP key containers.

Use this utility to migrate from Windows registry-based CSP container storage to the new CSP formats. The utility also enables you to manage the interfaces between the MSCAPI library and the module. See Utilities for the CAPI CSP.

98. loadmache

```
loadmache [-U|-e IDENT] [-a HASH] [OPTIONS] [MACHINE-FILENAME]
```

Prepares a module for SEE applications by loading an SEE machine image.

Which machine to load:

If no machine filename is specified this program will pick a default as follows. If `NFAST_SEE_MACHINEIMAGE_<module>` is set in the environment, where `<module>` is the module number, then that will be used. Otherwise if `NFAST_SEE_MACHINEIMAGE_DEFAULT` is set in the environment then that will be used. Finally if even that fails then `/opt/nfast/see/machine/see-jvm.sar` is the default. If this file doesn't exist then loadmache will fail.

Machine encryption key:

If `--unencrypted` is specified then the machine is assumed to be unencrypted. If `--encryptionkey IDENT` is specified the machine is assumed to be encrypted with `seeconf key IDENT`.

If neither of these options are specified `NFAST_SEE_MACHINEENCKEY_*` are checked following the same pattern as above; if no environment variables are set then the machine is assumed to be unencrypted.

Machine signing key:

For encrypted machines if you are use a dynamic SEE feature enable then `--sighash HASH` must be specified with the hash of the key used to sign the SEE machine. `NFAST_SEE_MACHINESIGHASH_*` are checked following the same pattern as above. For unencrypted machines, or if you have the General SEE feature, then this is not required at all.

Option	Description
`-s, --slot=SLOT`	Select the slot from which to load cards when `-n` is specified.
SEE machine loading options	
`-a, --sighash=HASH`	Loads a SEE machine signed with the key whose hash is `HASH`.
`-e, --encryptionkey=IDENT`	Loads a SEE machine encrypted with key IDENT.
`-n, --noprompt`	Never prompts for missing smartcards or passphrases.
`-U, --unencrypted`	Loads an unencrypted SEE machine (default)
Module selection	

Option	Description
`-m`, `--module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `loacmache` uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `loadmache`.
`-u`, `--usage`	Displays a brief usage summary for `loadmache`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `loadmache`.

99. loadrom

- Obtains information about the firmware installed on a module
- Upgrades the module firmware

To determine the version security number of the firmware in a file and for more information, see Firmware on the installation media.

Option	Description
`-b, --maxblocksize=SIZE`	Sets the maximum block size (in bytes) for module programming.
`-i, --ioboard`	File is I/O board reprogramming firmware.
`-n, --notypecheck`	Omits the module type check.
`--noauditcheck`	Omits the check for an unfinished audit logging session.
`-v, --view`	Just displays information about the NFF file, it doesn't load it.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `loadrom` uses all modules by default.
Help options	
`-h, --help`	Displays help for `loadrom`.
`-u, --usage`	Displays a brief usage summary for `loadrom`.
`-V, --version`	Displays the version number of the Security World Software that deploys `loadrom`.

100. loadsee-setup

```
loadsee-setup --setup -m MODULE <OPTIONS>
loadsee-setup --remove -m MODULE
loadsee-setup --display [-m MODULE]
```

Manages the configuration of auto-loaded SEE machines.

Option	Description
Action selection	
`-s, --setup`	Installs a new auto-loaded SEE machine configuration onto the module identified with the `-m` option in the command. All options except for `--machine` are optional.
`r, --remove`	Removes auto-loaded SEE machine configuration from the module identified with the `-m` option in the command. No SEE machines will be automatically loaded onto that module.
`-d, --display`	Outputs the configuration for auto-loaded SEE machines on the module identified with the `-m` option in the command.
--setup options	
`-M, --machine=MACHINE`	Filename of the SEE machine file (required).
`-U, --userdata=USERDATA`	Filename of the userdata file to pass to the SEE machine.
`-k, --key=IDENT`	Ident of seeconf key protecting SEE machine. Required if the SEE machine is encrypted.
`-S, --sighash=HASH`	Hash of key signing SEE machine. Only required if the SEE machine is encrypted and you are using the dynamic SEE feature enable rather than the static feature.
`-p, --published-object=NAME`	Name to publish `WorldID` of the SEE machine with.
`-P, --postload-prog=PROG`	Post-init program to run after machine load.
`-A, --postload-args=ARGS`	String to pass to `--postload-prog`.
General options, including HSM selection	
`-c, --configfile=FILENAME`	Selects the configuration file to use. Default: `NFAST_KMDATA/config/config`.
`-f, --force`	Allow config changes without prompting.
`-m, --module=MODULE`	Selects the module to use when configuring auto-load SEE machines (required for `--setup` and `--remove`). If you only have one module, `<MODULE>` is 1. If you do not specify a module number, the utility uses all modules by default.

Option	Description
`--no-reset`	Does not reset modules with changed configurations.  Unless the `--no-reset` option was given `loadsee-setup` will reset all modules whose configurations have changed .
Help options	
`-h`, `--help`	Displays help for `loadsee-setup`.
`-u`, `--usage`	Displays a brief usage summary for `loadsee-setup`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `loadsee-setup`.

101. logout

```
logout
```

Logs you out of the serial console session.

102. sbin/logrotate-hardserver

Linux

Archive the existing hardserver log from `/opt/nfast/log/hardserver.log` and re-open as a fresh log file.

When run with no arguments, it will automatically archive the existing log to `/opt/nfast/log/archive/hardserver.DATETIME.log` (where *DATETIME* is the current date and time). The directory `/opt/nfast/log/archive/` is created if it does not already exist.

Optionally, a single argument can be provided with the full file name to archive the exist ing hardserver log to.

This script must be run as root.

Windows

Extract Windows event log entries and output them to the console or a text file.

As required, specify:

- `-s` \| `--source`: The event log source. The default is the `nCipherlog`
- `-c` \| `--count`: The number of records read from the event log. The default is `10000`
- `-f` \| `--file`: The output filename.

103. logs

```
logs
logs [lines=1000] [timeout=enable]
```

Prints logs from the HSM.

Only available for nShield 5c models.

Option	Description
`lines`	Optional argument to only print the given number of lines from the end of the log. Default: prints the whole log.
`timeout`	Whether the log printing should have a timeout (enable or disable). With a time out, the log printing will be interrupted if it does not complete before the normal console timeout would occur. Default: enable timeout.

104. maintmode

```
maintmode <enable/disable>
```

Sets the mode of the nShield 5c to Maintenance mode.

Option	Description
`enable`	Switches the nShield 5c to Maintenance mode.
`disable`	Switches the nShield 5c to Operational mode.

105. makecspyuserdata

```
makecspyuserdata [options] conffile.py output.cpio
```

Packages Python files, and anything else required, into a `userdata` file for SEE machines.

Option	Description
`-f, --force`	Overwrites the existing `output.cpio` file.
Help options	
`-h, --help`	Displays help for `makecspyuserdata`.
`--version`	Displays the version number of the Security World Software that deploys `makecspyuserdata`.

106. migrate-world

Migrate existing keys to a destination Security World.

```
migrate-world [OPTIONS] --src-module=<source_module> --dst-module=<dest_module> --source=<source-kmdata-path>
--debug --dst-warrant=<dst-warrant-path> --src-warrant=<src-warrant-path [--plan | --perform] --key-logging
```

106.1. Prerequisites for using migrate-world

In order to use the `migrate-world` utility the following will be needed:

- Two HSMs. These can be any of the currently supported HSM types and the two HSMs do not need to be of the same type.
- A quorum of ACS cards for the source world.
- A quorum of ACS cards for the destination world.
- Sufficient blank cards to create new OCS cards for any keys that are OCS protected.
- **For PCIe HSMs**: Remote mode switching must be enabled on both HSMs used for the migration.
- **For network-attached HSMs**: Remote mode switching must be enabled on both HSMs used for the migration. See `enable_remote_mode` in the server_settings section or the *Top-level menu* chapter of the HSM *Install Guide*.

106.2. migrate-world modes

- **Plan mode**: Returns a list of steps for migration and the required card sets and passphrases but does not migrate any keys.
- **Perform mode**: Runs the plan mode prior to presenting the option to proceed and migrate keys according to the plan.

Option	Description
`-c <CARDSETS>|--cardsets-at-once=<CARDSETS>`	Migrates keys protected by this number of card sets or softcards per ACS loading. This is useful to prevent ACS time-outs if you have a large number of different card sets or softcards to migrate. (0=unlimited, default=0).
`--debug`	Outputs debug messages and stack traces in case of errors. It is recommended to use this only for testing as it will slow down operation and make card timeouts more likely to occur. A large volume of output is produced for each key that is migrated, so it is recommended to migrate a single key at a time when using this option.
`--dst-module=<ModuleId>`	Specifies which module ID to use as the destination module.

Option	Description
`--dst-prots=<list of destination protections>`	Specifies a comma-separated list of OCS or softcard names in the destination security world. These will be the target protections for the keys that are protected with methods specified with `--src-prots` in the source security world.
`--dst-warrant=<dst-warrant-file>`	Specifies the location of the warrant file of the destination module.
`-k <KEYS> --keys-at-once=<KEYS>`	Migrates no more than this number of keys per ACS loading. This is useful to prevent ACS time-outs if you have a large number of keys to migrate. (0=unlimited, default=0). It is recommended to limit the number of keys to be migrated at any one time to no more than 100.
`--key-logging`	Enables key usage logging on all migrated keys. If the destination world does not support audit logging the keys will still be migrated but LogKeyUsage logging will not be set in the ACL of the migrated keys.
`--perform`	Migrates keys interactively.
`--plan`	Displays the steps that will be carried out.
`--prots-config=<PATH>`	Specifies a configuration file that lists the source and destination protection pairs for migration. The file must contain pairs of tab-separated protection names `src_prot dst_prot`, one pair per line.
`--source=<SOURCE>`	Specifies the path to the folder that contains the source world data.
`--src-module=<MODULE>`	Specifies which module ID to use as the source module.
`--src-prots=<list of source protections>`	Specifies a comma-separated list of OCS or softcard names in the source security world. The keys will be migrated to the corresponding protections specified with `--dst-prots`.
`--src-warrant=<src-warrant-file>`	Specifies the location of the warrant file of the source module.
Help options	
`-h`, `--help`	Displays help for `migrate-world`.
`-u`, `--usage`	Displays a brief usage summary for `migrate-world`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `migrate-world`.



Do not terminate path names in the command parameters with a backslash character. If this is not possible then either terminate with a double backslash or insert a blank space between the backslash and the terminating quotation mark.

106.3. Restrictions on using migrate-keys

- The source module must be running firmware version 12.50 or later.
- The destination module must be running firmware version 12.50 or later.
- Only recoverable keys can be migrated. If your source keys are non-recoverable, you cannot use the migration utility to migrate keys.
- You can change some, but not all, security world properties during migration:

Property	Up to 13.3	13.4 and later
Key protection method whether softcard or OCS is used	Fixed	Fixed
Key protection name softcard name or cardset name	Fixed	Editable
Quorum	Editable	Editable

  If the name or quorum is to be changed, you must create the softcard or OCS in the destination world before migration begins.

- Replacement cards should be of the same or newer generation than the cards that they replace.
- The source and destination modules must both have KLF2 warrants in the correct location.

  The migration process directly downloads the warrants from the module for the nShield 5s and nShield 5c HSMs. You do not need to take any action.

  For pre-nShield5 HSMs:

  - If one or both of the modules have a KLF warrant, you must request an upgrade to a KLF2 warrant from nshield.support@entrust.com before you start the migration.
  - For Solo + and Solo XC, the default location is `NFAST_KMDATA/warrants/` (**Linux**) or `NFAST_KMDATA\warrants\` (**Windows**).
  - For Connect + and Connect XC, the default location is `NFAST_KMDATA/hsm-<ESN>/warrants/` (**Linux**) or `NFAST_KMDATA\hsm-<ESN>\warrants\` (**Windows**).
  - After adding or upgrading to a KLF2 warrant, you must reboot the HSM before the warrant file will appear in the warrants directory.

  See Warrant Management.

- The operator running the migrate-world utility must have the access rights to create a privileged connection to the hardserver.

- The migration tool must have exclusive use of the modules during migration. Do not use them for any other purpose during migration and if either module is an nShield network-attached HSM, do not enter anything via the front panel during migration.
- If the destination world is `fips-140-level-3`, then some keys that were usable in the source world may not be usable in the destination world due to those algorithms or key lengths being restricted. The migration tool might not be able to successfully migrate these keys so they should be removed from the source world before attempting the migration. Any keys of this type that do migrate successfully will be restricted at the point of use.
- If the destination world is `fips-140-level-3` or `common-criteria-cmts` the migration tool will automatically remove ExportAsPlain from the ACL of any migrated key during the migration process.
- If the destination world does not support audit logging the migration tool will automatically remove LogKeyUsage from the ACL of any migrated key during the migration process.

106.4. migrate-world to migrate keys using custom protection pairs

Regular security world migration will create new card sets and softcards in the destination world with the same names as the source protections or it will use existing destination protections if they share a name and type (card set or softcard) with the source protection.

You can specify custom protection pairs if you want to change the name, the quorum, or the properties of the protection. You can also combine multiple source protections of the same type into one destination protection. You cannot diffuse keys from one source protection to multiple destination protections.

The source-destination protection pairs can be selected either as:

- Two comma-separated lists `--src-prots <source protections>` and `--dst-prots <destination protections>`.
- Tab-separated pairs "source destination", one per line, in a configuration file `--prots-config <file path>`.

The protections can be referred to by their name, 40-character hash, or "c:name" and "s:name" when a source card set and softcard share a name. The source and destination protection types must match.

The following example shows the two ways of specifying a set of protection pairs and the different ways each protection can be referred to. The example hashes are shortened for

readability.

Protection type	Source protection to be migrated	Target destination protection
card set	`ocs 1`	`ocstarget1`
softcard	`softcard 1`	`softcardtarget`
card set	`name1` (duplicate name)	`ocstarget1`
softcard	`name1` (duplicate name)	`softcardtarget`
card set	`name2` (duplicate name and type) `hash: XXXXXXX1`	`ocstarget1`
card set	`name2` (duplicate name and type) `hash: XXXXXXX2`	`ocstarget2`

By specifying the lists using the `--src-prots` and `--dst-prots` options:

```
migrate-world [OPTIONS] \
--src-prots "ocs 1,softcard 1,c:name1,s:name1,XXXXXXX1,XXXXXXX2" \
--dst-prots "ocstarget1,softcardtarget,ocstarget1,softcardtarget,ocstarget1,ocstarget2"
```

By using a configuration file specified with the `--prots-config` option:

```
migrate-world [OPTIONS] --prots-config=migration.cfg

--- migration.cfg ---
ocs 1    ocstarget1
softcard 1    softcardtarget
c:name1    ocstarget1
s:name1    softcardtarget
XXXXXXX1    ocstarget1
XXXXXXX2    ocstarget2
---------------------
```

106.5. Troubleshoot migrate-world

If you encounter any errors that are not listed in the following table, contact Support.

Error	Explanation	Action
There are no keys requiring migration.	Any migrate-able keys found in the source world already exist in the destination world. The migration utility returns this error if: • The keys have already been migrated • All keys are non-recoverable and therefore cannot be migrated.	None.
Source module must be specified. Destination module must be specified. Source and Destination modules must be different. Module is not usable.	`migrate-world` requires you to specify both a source and destination module which must be different modules and both must be usable.	Specify the correct modules.
Source world has indistinguishable cardsets or softcards. Destination world has indistinguishable keys.	There are irregularities in one of the worlds, but these irregularities do not affect the migration process.	None.
Destination world has indistinguishable cardsets or softcards. Source world has indistinguishable keys. Cannot determine protection of keys.	There are problems with one of the worlds.	Contact Support.
Source world not recoverable.	The source world is not recoverable and the keys therefore cannot be migrated.	If the source world is not recoverable, you cannot use the migration utility to migrate keys. Contact Support.
Missing security world at `PATH`. Source world must be specified.	The path for the source world is wrong. There is no world data at the location that was specified when running the migration utility.	Supply the correct path to the source world. If you have supplied the correct path to the directory that contains the source world data, the migration utility has not found a destination world.

Error	Explanation	Action
Source world is the same as the destination world.	An incorrect path was supplied for the source world data when running the utility. The destination world data does not exist in the default location defined by the environment variable `NFAST_KMLOCAL` or `NFAST_KMDATA`.	Run the utility with the correct path to the source world data. Move the source world data to a different location and then copy the destination world data to the default location. If the default location is defined by an environment variable, configure the variable to point to the location of the destination world, which then becomes the new default location.
Cannot find `<NAME>` utility, needed by this utility. `<NAME>` utility is too old, need at least version `<VERSION NUMBER>`.	The software installation is partially completed. The path (in the environment variable for the operating system) might be pointing to an old version of the software.	Reinstall the software. Ensure that the path points to the latest version of the software.
nFast error: TimeLimitExceeded; in response to SetKM...	The ACS time-out limit has expired.	Restart the key migration process; see Key migration.
Destination world does not support audit logging.	You have specified the `--key-logging` option but the destination world does not support audit logging.	None. The keys will be migrated but LogKeyUsage will not be set in the ACL of migrated keys.
Failed to load warrant file <FILE>.	There is a problem reading the warrant file.	Check that your warrant files are in the correct location and have not been edited in any way.

107. mkaclx

```
mkaclx [-kKCMrRqviGA] [-a IDENT[:MECH]] [-t TYPE] [-b BITS] [-g BITS]
       [-O OPPERMS] [-m MODULE] [-N NAME] [-T TIME] [-U N] IDENT
```

Generates non-standard cryptographic keys that can be used to perform specific functions, for example, to wrap keys and derive mechanisms. This utility includes options that are not available with the `generate-key` utility.



Ensure that you run `mkaclx` with the options that are appropriate for your security infrastructure. If the appropriate options are not chosen, the security of existing keys might potentially be compromised.

Option	Description
Key generation parameters	
`-b`, `--bits=BITS`	Generates a key with length `BITS`. Default: depends on key-type.
`-g`, `--group-size=BITS`	Group size is `BITS` long for Diffie-Hellman keys.
`-k`, `--keygen-cert`	Stores a key generation certificate (default).
`-K`, `--no-keygen-cert`	Doesn't store a key generation certificate.
`-O`, `--deny-oppermis-sions=OPFLAGS`	Disables listing `OpPermissions` as a comma-separated list.
`-t`, `--type=KEYTYPE`	Selects the type of the generated key. Default: `RSA`.
Key protection options	
`-a`, `--see-app -key=IDENT[:MECH]`	Restricts the use of key to SEE programs signed by SEE integrity key `IDENT`, optionally with mechanism `MECH`.
`-A`, `--assigned`	Requires the key to be assigned (`common-criteria-cmts` worlds only).
`-C`, `--cardset-protected`	Generates a cardset-protected key.
`-G`, `--logkeyusage`	Requires logging of usage of the key.
`-i`, `--kitb`	Writes the blob to the module's NVRAM.
`-M`, `--module-protected`	Generates a module-protected key (default).
`-r`, `--recovery`	Allows key to be recoverable (default).
`-R`, `--no-recovery`	Doesn't allow key to be recoverable.

Option	Description
-S, --softcard-protected=NAME	Generates a softcard-protected module key using softcard NAME.
-T, --timeout=TIME	Sets the time limit of TIME seconds on main-use operations.
-U, --use-limit=N	Sets per-auth use limit of N on main-use operations.
Other settings	
--confirm	Shows the command and requests confirmation.
-N, --name=NAME	Sets the key's name. Default: no name.
-q, --quiet	Produces fewer messages on successful runs.
-v, --verbose	Produces more messages on successful runs.
Option to address HSMs	
-m, --module=MODULE	Specifies the number of the module to use. If you only have one module, <MODULE> is 1. If you do not specify a module number, the utility uses all modules by default.
Help options	
-h, --help	Displays help for mkaclx.
-u, --usage	Displays a brief usage summary for mkaclx.
-V, --version	Displays the version number of the Security World Software that deploys mkaclx.

108. modstate

```
modstate [-m MODULE] [--kml|--klf2]
```

Option	Description
`--klf`	Use `SignerType_KLF` (deprecated).
`--klf2`	Use `SignerType_KLF2`.
`--kml`	Use `SignerType_KML` (default).
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to use. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `modstate`.
`-u`, `--usage`	Displays a brief usage summary for `modstate`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `modstate`.

109. ncdate

```
ncdate [-m MODULE]
       ncdate --set    [-m MODULE] hh:mm:ss [yyyy.mm.dd]
       ncdate --adjust [-m MODULE] hh:mm:ss [yyyy.mm.dd]
```

Views, sets, and updates the time on a module's real-time clock.

Option	Description
`-a`, `--adjust`	Adjusts the module time.
`-d`, `--display`	Displays the current module time (default).
`-s`, `--set`	Sets the module time.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to use. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `ncdate`.
`-u`, `--usage`	Displays a brief usage summary for `ncdate`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `ncdate`.

110. ncperftest

```
ncperftest [options]
```



Only supported in FIPS 140-2 Level 2 Security Worlds.

Tests the performance of various crypto commands using attached nShield hardware.



Performance results vary depending on the selected cryptographic acceleration profile. See Cryptographic Acceleration Profiles for details.

Supported since Security World version v12.10, it contains all the functionality in sigtest and floodtest. It also supports other tests, and provides greater accuracy and throughput capability in performance management.

The default action is `--sign`. However, if a mechanism or key type is selected, then the default action is changed to be appropriate to the key type.

For `--mechanism` the default depends on the action. For example for signing the default is `RSApPKCS1`.

`--key-type` selects a default mechanism for the requested key type. For example `-S RSA` is equivalent to `-M RSApPKCS1`. For `-S HMAC`, the default is `HMACSHA256`.

The default for the `--key-size` option depends on the mechanism. For RSA and (KC)DSA keys it is `1024`, unless a mechanism was selected which requires a larger key. For AES it is `128`.

Option	Description
Action options	
`--channel-decrypt`	Tests the `channel decrypt` operation.
`-d`, `--decrypt`	Tests the `Decrypt` operation.
`-D`, `--rsa-sign-decrypt`	Tests the `RSAImmedSignDecrypt` operation.
`-e`, `--encrypt`	Tests the `Encrypt` operation.
`-E`, `--rsa-verify-encrypt`	Tests the `RSAImmedVerifyEncrypt` operation.
`-H`, `--hash`	Tests the `Hash` operation. No Key Options available, always load-balanced due to no module option.
`-N`, `--nop`	Tests the `NoOp` operation. Load-balanced across all modules unless only one module is specified. No Key Options available.

Option	Description
-R, --mod-exp-crt	Test ModExpCrt operation. Always load-balanced.
-s, --sign	Test Sign operation (default).
-U, --channel	Test channel encrypt operation.
-v, --verify	Tests the Verify operation.
-x, --mod-exp	Tests the ModExp operation. Always load-balanced.
Key and mechanism options	
-c, --curve=CURVENAME	Uses the curve named NAME. Default: NISTP192.
-l, --key-size=BITS	Sets the key size. Default: depends on the key type
-M, --mechanism=MECH	Use nCore mechanism MECH.
-p, --plain-type=TYPE	Uses plaintext type TYPE (Bignum, Hash, or Bytes). The mechanism and plaintext types must be compatible with the key type.
--pairwise-check	Set PairwiseCheck in key generation command.
-S, --key-type=TYPE	Select key type to use: RSA, DSA, KCDSA, ECDSA, ECDH, X25519, Ed25519, AES, DES3, HMAC
--strong	For RSA, use strong (ANSI X9.31) primes. For DSA, use the Strict flag.
Measurement options	
--bytes	Measure data throughput in (bytes/second).
--initial-delay=SECONDS	Starts timing after an initial delay of SECONDS. Default: 1 No delay: 0
--latency	Measures and reports latences between submitting jobs and receiving replies.
--operations	Measures the operation throughput (default).
Behavior options	
--display-pubkey	Displays public keys
-F, --no-failover	Doesn't failover if the loaded key becomes unusable.
--format=FORMAT	Selects the output format (text, json).
-G, --logging	Attempts audit logging. For this to succeed, all specified modules must report audit logging as active.
-j, --outstanding-jobs=COUNT	Sets the maximum number of outstanding jobs shared over all threads. Default: minimum hardserver recommended, but see also max-jobs-multiplier.
-L, --longjobs	Sets the LongJobs flag in crypto commands.

Option	Description
`--max-jobs-multiplier=MULTIPLIER`	Applies a multiplier to the maximum outstanding jobs. Default: `2`, which in conjunction with the default for outstanding-jobs means a default of twice the hardserver's recommended minimum queue across all threads.
`-n, --jobs-count=COUNT`	Sets the maximum number of jobs. Default: infinite.
`-t, --stop-after=LENGTH`	Sets the maximum time to run, in seconds. Default: infinite.
`--threads=THREADS`	Number of threads to use. Default: `4`.
Module and cardset selection	
`-m, --module=MODULE`	Specifies the number of the module to perform the tests on, it can repeated. If you only have one module, `<MODULE>` is `1`. If you do not specify a module number, the utility uses all modules by default.
Help options	
`-h, --help`	Displays help for `ncperftest`.
`-u, --usage`	Displays a brief usage summary for `ncperftest`.
`-V, --version`	Displays the version number of the Security World Software that deploys `ncperftest`.

111. ncsvcdep

nShield Service dependency tool to configure service-based applications such as Microsoft Certificate Services and IIS to use the nShield CNG CSP. Use this tool to add the nFast Server to the dependency list of such services.

For more information, see:

- Uninstalling or reinstalling the CNG CSP.
- ncsvcdep.

Use this helper utility to manage keys and the interfaces between the CNG library and the HSM. For a list of utilities specific to the nShield CNG CSP, see Utilities for the CAPI CSP.

Utility names that end with `64` run only on 64-bit version of Microsoft Windows. All other utilities run on both 32-bit and 64-bit versions of Microsoft Windows.

112. ncversions

```
ncversions
```

Lists the following information:

- Version of all Security World components, irrespective of whether they are installed individually or as part of a component bundle
- Version of each component bundle

Option	Description
Help options	
`-h`, `--help`	Displays help for `ncversions`.
`-u`, `--usage`	Displays a brief usage summary for `ncversions`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `ncversions`.

113. ncthread-test

```
ncthread-test [-q] [-d THREADS] [-c SIZE] [-r THREADS] [-b SIZE] [-s FREQ]
```



Only supported in FIPS 140-2 Level 2 Security Worlds.

Stress tests modules and tests the nCore API concurrent connection support.

If any threads have not returned a value since the last thread check a warning is generated. This may simply indicate that the interval between checks is too short. However, if a thread persistently fails to return a value this indicates that an error has occurred.

Option	Description
Thread options	
`-d`, `--des-threads=THREADS`	Creates `THREADS` DES3 threads for symmetric encryption. Default: `4`.
`-r`, `--rsa-threads=THREADS`	Creates `THREADS` RSA threads for digital signing. Default: `4`.
`-s`, `--check-every=PERIOD`	Checks threads every `PERIOD` seconds. Default: `10`.
Other options	
`-b`, `--rsa-size=SIZE`	Uses RSA key of size `SIZE` bits. Default: `1024`.
`-c`, `--block-size=SIZE`	Encrypts blocks of up to `SIZE` characters long. Default: `16384`.
`-q`, `--quiet`	Runs quietly, outputting only errors and warnings.
`-t`, `--time=DURATION`	Runs test for `DURATION` seconds. Default: `60`.
Help options	
`-h`, `--help`	Displays help for `ncthread-test`.
`-u`, `--usage`	Displays a brief usage summary for `ncthread-test`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `ncthread-test`.

114. netcfg

```
netcfg
netcfg [iface=0]
netcfg iface=0 [addr=0.0.0.0 netmask=0.0.0.0] [linkspeed=auto]
```

Configures an IPv4 network interface, one at a time.

Both `addr` and `netmask` must be specified at the same time to avoid bad network configuration. Either `addr` and `netmask` together or `linkspeed` alone can be configured separately or all at the same time for the specified interface.

Option	Description
`iface`	The Ethernet interface number (`0` or `1`). • If `iface` is not specified on the command line, the current setting of both interfaces will be displayed. • If `iface` is specified without arguments, then the current setting of that interface will be displayed.
`addr`	Static IPv4 address for the interface. Default: `0.0.0.0`.
`netmask`	IPv4 netmask for the interface. Default: `0.0.0.0`.
`linkspeed`	Link speed setting, one of: `auto`, `10BaseT`, `10BaseT-FDX`, `` `100BaseTX `` and `100BaseTX-FDX`. `auto includes negotation of speeds up to and including 1Gb.` `Default: `auto`.

115. netcfg6

```
netcfg6
netcfg6 [iface=0]
netcfg6 iface=0 [addr=:: netmask=64]
```

Configures an IPv6 network interface, one at a time.

Both `addr` and `netmask` must be specified at the same time to avoid bad network configuration.

Option	Description
`iface`	The Ethernet interface number (`0` or `1`). • If `iface` is not specified on the command line, the current setting of both interfaces will be displayed. • If `iface` is specified without arguments, then the current setting of that interface will be displayed.
`addr`	Static IPv6 address for the interface. Default: `::`.
`netmask`	Subnet prefix length of the static IPv6 address for the interface. Default: `64`.

116. netdiagnose

```
netdiagnose iface=0
netdiagnose iface=1
netdiagnose iface=bond
```

Runs network diagnostics on the specified network interface and prints out the status.

Option	Description
`iface`	The network interface index number or bond (`0`, `1` or `bond`). `bond` is only allowed if the bond link interface is enabled.

117. netenable

```
netenable
netenable [iface=0]
netenable iface=0 [enable_ipv4=yes] [enable_ipv6=no] [ipv6_conf_addr=no]
```

Gets or sets the `nethsm_enable` configuration.

Option	Description
`iface`	The Ethernet interface number (`0` or `1`). • If `iface` is not specified on the command line, the current setting of both interfaces will be displayed. • If `iface` is specified without arguments, then the current setting of that interface will be displayed.
`enable_ipv4`	Indicator of whether the IPv4 protocol on the interface is enabled. Default: `yes`.
`enable_ipv6`	Indicator of whether the IPv6 protocol on the interface is enabled. Default: `no`.
`ipv6_conf_addr`	Indicator of whether the interface uses IPv6 static addresses or SLAAC. Default: `static`.

118. nethsmadmin

```
nethsmadmin [-m MODULE] [-c|-w|-r|-e|-g]
nethsmadmin -l -s RFS_IP [options]                          # List images on the RFS
nethsmadmin [-m MODULE] -f -s RFS_IP [options]              # List features on the RFS
nethsmadmin [-m MODULE] -a FEATURE_FILE -s RFS_IP [options] # Apply feature file to RFS
nethsmadmin [-m MODULE] -i IMAGE                            # Upgrade image
nethsmadmin [-m MODULE] -d MMDDhhmmYYYY                     # Get date
```

Administers an HSM without using the front panel.

Options include:

- Check the Security World files on a specified nShield HSM.
- Copy Security World files from the RFS to the nShield HSM.
- Command the specified nShield HSM to reboot. This restarts the hardserver.
- Command the nShield HSM to upgrade using the specified image file from its RFS.
- Retrieve a list of image files available on the RFS.
- Retrieve a list of feature certificates available on the RFS for a specified nShield HSM.
- Command the nShield HSM to apply a specified feature certificate from the RFS.
- Erase the Security World on the nShield HSM and re-initialize the HSM.
- Get the date and time on the nShield HSM.
- Set the date and time on the nShield HSM.
- Enable dynamic features, including client licenses remotely.

You must use a privileged connection to use this utility with the following parameters:

- Reboot the HSM (`nethsmadmin -r`)
- Erase the Security World (`nethsmadmin -e`)
- Upgrade the HSM firmware (`nethsmadmin -i`)

For more information, see:

- Using nethsmadmin to copy a Security World to an nShield HSM and check the current version.
- Upgrading the image file and firmware from a privileged client.
- Remotely enabling dynamic feature certificates including nShield HSM client licenses.

Option	Description
RFS options	
`-p`, `--port=PORT`	Overrides the default RFS port `9004`.

Option	Description
`-s`, `--rfs=RFS_IP`	IP address of the remote file server (RFS).
Authentication options	
`-k`, `--kneti-module=LOCAL_MODULE`	Optional. Specifies the local module whose KNETI authentication key will be used to authenticate this client to the RFS. If omitted or `0`, this client will authenticate itself to the RFS using the hardserver's software KNETI authentication key. Default: `0`.
`--rfs-esn=ESN`	Sets the ESN of the remote module used to authenticate the RFS when using module KNETI authentication.
`--rfs-hkneti=HKNETI`	Required. Sets the software or module KNETI hash used to authenticate the RFS.
Admin operations	
`-a`, `--apply-feature=FEATURE_FILENAME`	Applies the specified feature file to the newtork-attached HSM. The path to the feature file must be a full path as `--list-features` retrieved it.
`-c`, `--check-world`	Prints the state of the security world/files on the specified remote module.
`-d`, `--set-date=DATE`	Sets the date and time on the specified remote module to the specified date. Format: `MMDDhhmmYYYY` (`MM` is the month, `mm` is the minutes, `YYYY` must be between 2000 and 2037).
`-e`, `--erase-world`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Erases the security world on the specified remote module.
`-f`, `--list-features`	Lists the nethsm features on the remote filesystem.
`-g`, `--get-date`	Rerieves the date and time on the specified remote module.
`-i`, `--upgrade-image=IMAGE`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Instructs the module to upgrade using the specified image file from the remote filesystem. The path to the image must be a full path as `--list-images` retrieved it.
`-l`, `--list-images`	Lists the nethsm images on the remote filesystem.
`-r`, `--reboot`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Instructs the specified module to remotely reboot.
`-w`, `--update-world`	Instruct the specified module to fetch its world files from its RFS.
Option to address HSMs	

Option	Description
`-m`, `--module=MODULE`	Specifies the number of the module to use. If you only have one module, `<MODULE>` is `1`. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `nethsmadmin`.
`-u`, `--usage`	Displays a brief usage summary for `nethsmadmin`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nethsmadmin`.

119. nethsmenroll

```
nethsmenroll [OPTIONS] NETHSM-IP [ESN HKNETI]
```

As an alternative to hand-editing a client's hardserver configuration file, you can run `nethsmenroll` on a client to configure it to access an nShield HSM. For example:

- Enroll an HSM, without needing to restart the hardserver
- Unenroll an HSM (`nethsmenroll -r`), then restart the hardserver to update the information about the HSM estate

A network-attached HSM for this kind of configuration file editing can be either an nShield Connect or nShield 5c, or a remote hardserver that has been configured to export a local HSM. If the network-attached HSM's ESN and HKNETI are not specified, attempts to contact the HSM to determine them and requests confirmation. ESN and HKNETI must be specified if the HSM is a remote hardserver with more than one HSM.

For more information, see:

- nethsm_imports.
- Configuring the unit to use the client.
- nethsmenroll.

Option	Description
`-f`, `--force`	Forces reconfiguration of an already known HSM.
`-n`, `--ntoken-esn=ESN`	Specifies the `ESN` of the nToken to be used to authenticate this client. If the option is omitted, then software authentication will be used instead.
`--no-hkneti-confirmation`	Does not request confirmation when automatically determining the nethsm's ESN and HKNETI. Only use this option on secure networks.
`-p`, `--privileged`	Causes the hardserver to request a privileged connection to the HSM. Default: unprivileged.
`-P`, `--port=PORT`	Specifies the port to use when connecting to the HSM. Default `9004`.
`-r`, `--remove`	Deconfigures the HSM.
`-V`, `--verify-nethsm-details`	When the ESN and HKNETI have been provided on the command line, verifies that the HSM is alive, reachable and matches those details.
Option to address HSMs	

Option	Description
`-m`, `--module=MODULE`	Specifies the number of the module whose hardserver configuration file to use. If you only have one module, `<MODULE>` is `1`. Default: `0` for dynamic configuration by the hardserver.
Help options	
`-h`, `--help`	Displays help for `nethsmenroll`.
`-u`, `--usage`	Displays a brief usage summary for `nethsmenroll`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nethsmenroll`.

120. netlink

```
netlink
netlink iface=0
netlink iface=1 [enable/disable]
```

Gets or sets the network interface link.

Option	Description
`iface`	The Ethernet interface number (`0` or `1`). You can enable or disable `iface=1` only. `iface=0` is always enabled. • If `iface` is not specified, the current setting of both interfaces will be dis played. • If `iface` is specified without arguments, then the current setting of that interface will be displayed.
`action`	The action to take for the interface (`enable` or `disable`).

121. new-world

Creates a new security world, or adds or restores a HSM to an existing security world.

```
new-world [ACTION] [OPTION] [-m MODULE] [FEATURES]
```

121.1. Prerequisites for using new-world

- Most options of `new-world` require a privileged connection between the host machine on which you run it and the HSM that it uses to administer the security world.
- The HSM must be in pre-initialization mode for the `new-world` utility to work with the HSM to create, configure, erase a security world or to enrol the HSM into the security world. Furthermore, if you use `new-world` to re-configure an HSM, you will have to restart the HSM into operational state.
  If the HSM is not in the pre-initialization mode, `new-world` advises you that you must put the HSM in this mode and waits until you have changed the HSM mode before con tinuing. See
  - Network-attached HSMs: Checking and changing the mode on a network-attached HSM
  - nShield Solo and Solo XC: Checking and changing the mode on nShield Solo and nShield Solo XC HSMs
  - nShield 5s: nShield 5s modes of operation
  - USB HSMs: Checking and changing the mode on an nShield Edge
- If the HSM is ready for creating, repprogramming, or erasing a security world (that is, it's in the pre-initialization mode), `new-world` prompts you for smart cards and passphrases as required.
- If `new-world` cannot interpret the command line, it displays its usage message and exits. It does not create a security world and does not modify the existing security world.
- If you attempt to set a quorum for a feature that you have disabled or if you attempt to set a quorum too high, `new-world` displays an error and exits.
- If `new-world` cannot find the key-management data, it displays the message `new-world: no existing world to load`

121.2. new-world [ACTIONS]

You can use `new-world` in three different ways to perform three different ACTIONs. Each

ACTION has its own options.

If you do not enter a specific action, `new-world` selects one of `[-i]` or `[-l]`, depending on whether you have already created a security world.

Action	Description
`-i, --initialize`	Initializes a new security world according to the specified parameters and programs it into the given module. `new-world [-i] [-SRG] [-m MODULE] [--mode=SWORLD-MODE] [-c CIPHER-SUITE] -Q K/N [FEATURES]` Creating a new security world replaces any existing `/opt/nfast/kmdata/local/` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directory. Replacing an existing security world in this way does not delete the security world's host data and recovery and replacement data, but renames the existing `/opt/nfast/kmdata/local/` (**Linux**) or `%NFAST_KMDATA%\local` (**Windows**) directory in which these reside as `%NFAST_KMDATA%\localN` (**Linux**) or `/opt/nfast/kmdata/localN` (**Windows**) where *N* is an integer assigned depending on how many security worlds have been previously saved during overwrites.
`-l --program`	Programs a module with an existing security world (enrols the module into the security world). `new-world [-l] [-S] [-m MODULE]` Adds an HSM to an existing security world in the Key Management Data directory. If you have multiple HSMs available, you can use the `-m` option to specify one HSM. If you do not specify an HSM, `new-world` adds all available HSMs to the security world.
`-e --factory`	Restores a module to its factory default condition: • the only module key is `KM0` • no operations require `NSO` certificates • `KNSO` is the single-DES key `0101010101010101` You must run `new-world -m=<module-id>` separately for each HSM that you want to factory state. `new-world [-e] [-m MODULE]`

121.3. new-world [OPTIONS]

Option	Description
`-c, --cipher-suite=<CIPHER-SUITE>`	Specifies the cipher suite and type of key that is used to protect the new Security World. Permitted values: • DLf3072s256mAEScSP800131Ar1 (default) • ECp521mAES
`-k, --kml-type=KML-TYPE`	Select the KML type. If not specified, the default KML type for the ciphersuite is used when creating a new world, or the KML type selected when creating the world is used when loading an existing world. Permitted value(s): • `DSAp3072s256` • `NISTp256hSHA1` • `NISTp521hSHA1`
`-R, --no-recovery`	Disables OCS and softcard replacement; see Replacing Operator Card Sets. Equivalent to setting `!r`. By default, `new-world` creates key recovery and replacement data that is protected by the cryptographic keys on the ACS. This option does not give Entrust or any other third party access to your keys. Keys can only be recovered if authorization from the ACS is available. We recommend that you leave OCS and softcard recovery and replacement functionality enabled.  Entrust recommend that you do not disable the recovery and replacement option because if you set the `--no-recovery` option, you can never replace lost or damaged OCSs generated for that security world. Therefore, you could never recover any keys protected by lost or damaged OCSs, even if the keys themselves were generated as recoverable (which is the default for key generation). OCS and softcard replacement cannot be enabled after security world creation without reinitializing the security world and discarding all the existing keys within it. **Default**: disabled

Option	Description
`--reduced-features`	Uses a reduced default feature set when it creates the security world: • no passphrase recovery (`-R`) • no NVRAM • no RTC • no FTO • no NSO delegate keys Such a reduced-features security world can perform many operations faster than more fully featured security worlds.  This option must be the first option on the command line: `new-world -i --reduced-features` **Default**: *dis*abled
`-S, --no-remoteshare-cert`	Prevents the HSM from becoming a target for remote shares. If you do not want an HSM to be able to read remote card sets, initialize it by running `new-world -S <module-id>.
`--mode=MODE`	Available modes: **(none specified)** — Compliant with FIPS 140-2 Level 2 **fips-140-level-3** — Compliant with FIPS 140-2 Level 3 **common-criteria-cmts** — Supports _Common Criteria PP 419 221-5 In this mode, `new-world` requires a minimum K of 2 **Default**: no mode is specified, a FIPS 140-2 Level 2 compliant security world is created

121.4. new-world [FEATURE] syntax

The feature expressions in the `new-world` utility is a comma-separated list of *<feature-terms>*, each of which can optionally be flanked by an *<operator>* and the *<quorum-info>* for the ACS that is required to manage the feature:

```
<operator><feature-term><quorum-info>
```

Term	Description
<feature-term>	Name of the feature, see new-world [FEATURES].

Term	Description
<operator>	double dash (`--`) to enable the feature exclamation point (`!`), or `no-` to turn off the feature Three features remain available for use on presentation of the standard ACS quorum, even if turned off using the `!` operator. Setting the quorum of these features to `0` has the same effect as turning them off using the `!` operator. • `nvram` • `rtc` • `fto`  Some Linux shells interpret `!` character as history expansion. You must escape and must be escaped with a backslash, `\!`. The dash may be interpreted as being the start of an command-line option unless you have used the `-f` option or specified an HSM without including the `-m` flag.
<quorum-info>	`=<n>` The quorum of cards from the ACS required to use the feature

121.5. new-world [FEATURES]

Feature	Description
`--disablepkcs1pad`	Disables the use of PKCS#1 v1.5 padding. All attempts to use PKCS#1 v1.5 padding for encryption or decryption operations will be rejected. PKCS#1 v1.5 signature operations are not affected. PSS and OAEP are not affected. **Default**: enabled
`--dsee`	Specifies that that ACS authorization is needed to enable SEE World debugging. See Debugging SEE machines. **Default**: enabled

Feature	Description
`--dseeall`	Enables SEE World debugging for `all` users. See Debugging SEE machines. If you try to set the `Cmd_CreateSEEWorld_Args_flags_EnableDebug` CodeSafe flag in a security world that does not allow SEE debugging, the `CreateSEE-World` command returns `AccessDenied`. This also occurs if you call `CreateSEE-World` in a security world where SEE debugging is restricted and an appropriate certifier is not present.  The `dseeall` option is designed for testing purposes only and to use extended debugging for the HSM, you must enable `dseeall`. Do not enable this feature on pro duction security worlds as it may enable SEE applications to leak security information. **Default**: *dis*abled
`-G`, `--audit-logging`	Configures the security world and the HSM on which it is being created for audit logging, creating a log signing key for each HSM. Additional configuration is required for audit offload (see Audit Logging). This option is enabled by default for all world types in v13.9 and later software, and enabled by default only when the security world is created in `common-criteria-cmts` mode in prior versions. **Default**: enabled
`-g`, `--no-audit-logging`	Disable audit logging. This option is required to explicitly create a security world without audit logging enabled in v13.9 and later software (otherwise an audit logging world will be created). This option is ignored if `common-criteria-cmts` mode is enabled as such worlds always require audit logging. This option is not recommended for new security worlds unless HSM Pool Mode support is required. **Default**: *dis*abled
`--fto`	Specifies that ACS authorization is needed to enable foreign token operations (FTO). If you set the `!fto` flag, that is, turn off FTO, you will not be able to use smart cards to import keys. + This feature remains available for use on presentation of the standard ACS quorum, even if turned off using the `!` operator. Setting the quorum of this feature to `0` has the same effect as turning it off using the `!` operator. **Default**: enabled

Feature	Description
`--max-keyusage`	Specifies a maximum reauthorization condition in terms of number of key usages since authorization for Assigned keys in common-criteria-cmts mode. A use limit compatible with the specified maximum will be applied at key creation time and can be verified for Assigned keys. If this is not set then no `--max-keyusage` limit is applied to Assigned keys on creation. **Only in `common-criteria-cmts` mode** Satisfies the Protection Profile requirement for the administrator to set reauthorization conditions when they are creating an Assigned Key. **Default**: enabled
`--max-keytimeout`	Specifies a maximum reauthorization condition in terms of a TIMEOUT since authorization for Assigned keys in common-criteria-cmts mode. By default, an integer given for TIMEOUT is interpreted in seconds, but you can supply values for TIMEOUT in the form *Ns*, *Nh*, or *Nd* where *N* is an integer and *s* specifies second, *h* specifies hours, and *d* specifies days. A use limit compatible with the specified maximum will be applied at key creation time and can be verified for Assigned keys. If this is not set then no limit is applied to Assigned keys on creation. **Only in `common-criteria-cmts` mode** Satisfies the Protection Profile requirement for the administrator to set reauthorization conditions when they are creating an Assigned Key. **Default**: enabled
`--no-remoteshare-cert`	This option prevents making the HSM from becoming a target for remote shares. **Default**: enabled
`--no-strict-rsa-keygen`	If you have not specified a mode parameter you can use the `-no-strict-rsa-keygen` flag to disable the `UseStrongPrimes` setting. Otherwise it will be enabled by default. See Security World options. **Default**: enabled
`t, --nso-timeout=<TIMEOUT>`	This option allows you to specify the time-out for new security worlds. By default, an integer given for *TIMEOUT* is interpreted in seconds, but you can supply values for *TIMEOUT* in the form *N* `s`, *N* `h`, or *N* `d` where *N* is an integer and `s` specifies second, `h` specifies hours, and `d` specifies days. See Access Control. **Default**: enabled
`nvram`	This feature specifies that ACS authorization is needed to enable nonvolatile memory (NVRAM) allocation. See Designing SEE machines and SEE-ready HSMs. This feature remains available for use on presentation of the standard ACS quorum, even if turned off using the `!` operator. Setting the quorum of this feature to `0` has the same effect as turning it off using the `!` operator. **Default**: enabled

Feature	Description
`p`	This feature enables passphrase replacement; see passphrase replacement and Changing card and softcard passphrase. **Default**: *dis*abled

`--pp-min=LENGTH`	Enables a minimum passphrase length check for the Administrator Card Set (ACS) the Operator Card Set (OCS) and any associated softcards when you create a security world. The minimum passphrase length check is then applied after the security world is created. When enabled and you attempt to create a card passphrase with fewer characters than the specified minimum length, the following warning message displays:
	`WARNING`
	`Warning: short passphrase.`
	`---- +`
	`However, the passphrase can still be used. +`
	`Example: +`
	`[source]`
	new-world --initialize --acs-quorum=K/N --pp-min=14 ----
	If `--pp-min=<length>` is not used, the minimum passphrase length is set to the default value (0).

`--pp-strength`	Enables passphrases to have at least one uppercase, lowercase, number, and symbol. If the `--pp-strength` argument is omitted, the complexity requirements are not enforced. **Default**: enabled
`-Q, --acs-quorum=<K>/<N>`	This feature only takes effect if you are creating a new security world. `<K>` specifies the minimum number of smart cards needed from the ACS to authorize a feature. You can specify lower *K* values for a particular feature. All the *K* values must be less than or equal to the total number of cards in the set. If a value for *K* is not specified, `new-world` creates an ACS that requires a single card for authorization. When the security world is created in `common-criteria-cmts` mode, `new-world` requires a minimum K of 2. Some applications do not have mechanisms for requesting that cards be inserted. Therefore any OCSs that you create for use with these applications must have *K*=1. `<N>` specifies the total number of smart cards to be used in the ACS. This must be a value in the range 1-64. If a value for this option is not specified, `new-world` creates an ACS that contains a single card.+ We recommend that you do not create an ACS for which the required number of cards is equal to the total number of cards because you will not be able to replace the ACS if even a single card is lost or damaged. **Default**: enabled

Feature	Description
`rtc`	This feature specifies that ACS authorization is needed to set the real-time clock (RTC); (see rtc). This feature remains available for use on presentation of the standard ACS quorum, even if turned off using the `!` operator. Setting the quorum of this feature to `0` has the same effect as turning it off using the `!` operator. **Default**: enabled + **Not available on the nShield 5c.**
Module selection	
`-m`, `--module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `new-world` uses all modules by default. You must reference an HSM with `-m` whenever you run the `new-world` utility. See the examples for the impact of `new-world` on the HSM.
Help options	
`-h`, `--help`	Displays help for `new-world`.
`-u`, `--usage`	Displays a brief usage summary for `new-world`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `new-world`.

121.6. new-world examples

Example 1

```
new-world m=1, r, !p, nv=2, rtc=1
```

Create a security world for which:

m=1	y
r	
!p	passphrase replacement is not enabled
nv=2	Two cards are required to allocate nonvolatile memory
rtc=1	1 card is required to set the real-time clock (applies to SEE only)
(--acs-quorum is omitted)	The default number is required to replace an OCS

(--acs-quorum is omitted)	A single card from the ACS is required to add a new HSM

122. nfcp

```
nfcp [OPTIONS] SOURCE... DESTINATION
```

Performs file transfer operations with another hardserver using an interface compatible with `rcp`.

Each non-option argument can have the form `HOST:PATH` where `HOST` is the dotted decimal IPv4 address, or colon-separated hexadecimal IPv6 address, or DNS name of the target machine; and `PATH` is the location of the file or directory to be copied. IPv6 addresses must be surrounded by square brackets.

If the filename contains an unquoted *, ?, or [, it's considered to be a glob pattern; in this case, `DESTINATION` must be a directory.

Option	Description
`-a`, `--append`	Appends to destination file. Implies `--force`.
`-f`, `--force`	Overwrites files that exist at the destination.
`-k`, `--use-kneti`	Uses a local module KNETI to authenticate this client to the remote server. If this option is not specified, the hardserver's software KNETI is used instead.
`-P`, `--port=PORT`	Sets the port through which to connect to the remote server. Default: `9004`.
`--remote-esn=ESN`	Specifies the ESN of the nToken to be used to authenticate the remote hard-server. Cannot be specified without the corresponding remote KNETI hash option.
`--remote-hkneti=HKNETI`	Specifies the KNETI hash to authenticate the remote hardserver.
`-s`, `--sync`	Accepts less performance for greater reliability.
`-t`, `--text`	Converts line endings for locally read files.
`--target-directory=DIRECTORY`	Moves all source arguments to `DIRECTORY`.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module use KNETI to use. This option is ignored unless `--use-kneti` has been used as well. If you only have one module, `<MODULE>` is `1`. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `nfcp`.

Option	Description
`-u`, `--usage`	Displays a brief usage summary for `nfcp`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfcp`.
`-V`, `--verbose`	Shows more information on copy progress.

123. nfdiag

```
nfdiag [-h] [-u] [-v] [-f FILE] [-l KBYTES] [-e EXTRAINFO] [-q]
       [--home-directories "DIR1:DIR2:DIR3"] [-a]
```

Obtains information about the module and the host on which it is installed. This diagnostic utility can save information to either a `ZIP` file or a text file.

Under normal operating conditions, you do not need to run `nfdiag`. Run this utility only if requested to do so by Support.



On nShield 5s, use hsmdiagnose.

Option	Description
`-a`, `--check-all`	Checks for nShield logs in all user home directories.
`-e EXTRAINFO`, `--extrainfo EXTRAINFO`	Path to the file that contains additional information to include in the `nfdiag.zip` file.
`-f FILE`, `--file FILE`	Output file name. Default: `nfdiag.zip`
`--home-directories "DIR1:DIR2:DIR3"`	Absolute paths of nShield logs directories of any users with per-user logs.
`-l KBYTES`, `--logsize KBYTES`	Maximum logfile size in bytes.
`-q`, `--quiet`	Suppresses verbose output.
Help options	
`-h`, `--help`	Displays help for `nfdiag`.
`-u`, `--usage`	Displays a brief usage summary for `nfdiag`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfdiag`.

123.1. Include additional files for Support in the zip output of nfdiag

If you want to supply additional diagnostic files when you submit the `nfdiag` output to Entrust, run:

```
nfdiag -e|--extrainfo <your-plaintext-file>
```

By default, `nfdiag` runs in verbose mode, providing feedback on each command that it executes and which log files are available. If the system is unable to execute a command, the verbose output from `nfdiag` shows where commands are stalling or waiting to time out.

At any time while `nfdiag` is running, you can type **Ctrl**-**C** to cancel its current commands and re-run it.

123.2. Content of the text output of nfdiag

`nfdiag` generates a plain text output file and displays its file name. It does NOT capture any passphrases in the output file.

If the file `opt/nfast/log/logfile` (**Linux**) or `%NFAST_HOME%\log\logfile` (**Windows**) exists, `nfdiag` automatically includes this file in its output. If this file does not exist, `nfdiag` warns you that it could not process this file. This warning does not affect the validity of the generated output file.

When complete, this output file contains the following:

- Details about the client machine
- Details about any environment variables
- Output from the following command-line utilities:
  - `enquiry`
  - `stattree`
  - `ncversions`
  - `nfkminfo`
- The contents of the following log files (if they are available):
  - `hardserver.log`
  - `cmdadp.log`
  - `ncsnmpd.log`

124. nfkmattest

```
nfkmattest [OPTIONS] COMMAND [ARGS]...
```

Creates or verifies an attestation bundle.

Option	Description
`bundle`	Creates an attestation bundle for a key.
`verify`	Verifies an attestation bundle.
Help options	
`-h`, `--help`	Displays help for `nfkmattest`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfkmattest`.

125. nfkmcheck

```
nfkmcheck
```

Checks the security world data for consistency.

Option	Description
Help options	
`-h`, `--help`	Displays help for `nfkmcheck`.
`-u`, `--usage`	Displays a brief usage summary for `nfkmcheck`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfkmcheck`.

126. nfkminfo

```
nfkminfo -w|--world-info [-r|--repeat] [-p|--preload-client-id]
```

Shows you information about the current security world state, the stored security world keys, or the available operator card sets.

For more information, see:

- Viewing card sets from the command line.
- Viewing softcards with nfkminfo.
- Viewing keys using the command line.
- nfkminfo: information utility.

Option	Description
`-c, --cardset-list`	Lists cardsets.
`-k, --key-list`	Lists keys.
`-l, --name-list`	Lists keys and names, ordered by protection.
`--pool`	Displays the pool of HSMs as a single resource.
`-s, --softcard-list`	Lists softcards.
`-w, --world-info`	Specifies that you want to display general information about the security world. This option is set by default, so you do not need to include it explicitly. Flags for this option: **-r, --repeat** — Prints out the information repeatedly, pausing for a line from stdin each time. The next batch of information is displayed when you press **Enter**. **-p, --preload-client-id** — Shows preloaded client ID value, if any.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `nfkminfo` uses all modules by default.
Help options	
`-h, --help`	Displays help for `nfkminfo`.
`-u, --usage`	Displays a brief usage summary for `nfkminfo`.

Option	Description
-v, --version	Displays the version number of the Security World Software that deploys nfkminfo.

126.1. Front panel flags mapped to nfkminfo fields

The following table maps the flags visible on the front panel when you select **3 Security World mgmt > 3-1 Display World Info** to the flags in the output of nfkminfo.

Front panel	nfkminfo
admin	k-out-of-n
nCore flags	slotlistflags
NFKM flags	flags
Module slots	nflags
Initialized	Initialised
ForeignTokenOpen	FTO

126.1.1. nfkminfo: information utility

The nfkminfo utility displays information about the Security World and the keys and card sets associated with it.

126.1.1.1. Usage

```
nfkminfo -w|--world-info [-r|--repeat] [-p|--preload-client-id]
```

```
nfkminfo -k|--key-list [<APPNAME> [<IDENT>]]
```

```
nfkminfo -l|--name-list [<APPNAME> [<APPNAME>...]]
```

```
nfkminfo [-c|--cardset-list]|[-s|--softcard-list] [<TOKENHASH>]
```

```
nfkminfo --cardset-list [<TOKENHASH>] --key-list [<APPNAME>[<APPNAME>]]|--name-list <APPNAME>[<IDENT>...]]
```

126.1.1.1.1. Security World options

`-w|--world-info`

This option specifies that you want to display general information about the Security World. These options are the default and need not be included explicitly.

`-r|--repeat`

This option displays the information repeatedly. There is a pause at the end of each set of information. The information is displayed again when you press Enter.

`-p|--preload-client-id`

This option displays the preloaded client ID value, if any.

126.1.1.1.2. Key, card set, and softcard options

`-k|--key-list [<APPNAME>[<APPNAME>]]`

This option lists keys without key names. If `<APPNAME>` is specified, only keys for these applications are listed.

`-l|--name-list [<APPNAME>[<IDENT>]]`

This option lists keys with their names. If `<APPNAME>` is specified, only keys for these applications are listed. If `<IDENT>` is listed, only the keys with the specified identifier are listed.

`-c|--cardset-list [<TOKENHASH>]`

If `<TOKENHASH>` is not specified, this option lists the card sets associated with the Security World. The output is similar to this:

```
Cardset list - 1 cardsets: (P)ersistent/(N)ot, (R)emoteable/(L)ocal-only
Operator logical token hash      k/n timeout name <hash>                              1/1 none-PL <name>
```

If `<TOKENHASH>` is specified, these options list the details of the card identified by *hash*. The output is similar to this:

```
Cardset
 name        "name"
 k-out-of-n  1/1
 flags       Persistent PINRecoveryForbidden(disabled) !RemoteEnabled
 timeout     none
 card names   ""
 hkltu       hash
 gentime 2005-10-14 10:56:54
Keys protected by cardset hash:
 AppName app Ident keyident
 AppName app Ident keyident
 ...     ...  ...   ...
```

`-s|--softcard-list TOKENHASH`

This option works like the `-c|--cardset-list` option, except it lists softcards instead of card sets. If `<TOKENHASH>` is not specified, this option lists the softcards associated with the Security World.

126.1.1.2. Security World output info

If you run `nfkminfo` with the `-w|--world-info` option, it displays information similar to that shown in these examples:

```
World
 generation  1
 state       0x70000 Initialised Usable Recovery !PINRecovery !ExistingClient !RTC !NVRAM !FTO !SEEDebug
 n_modules   1
 hknso       hash_knso
 hkm         hash_km
 hkmwk       hash_knwk
 hkre        hash_kre
 hkra        hash_kra
 hkmc        hash_kmc
 hkrtc       hash_krtc
 hknv        hash_knv
 hkdsee      hash_kdsee
 hkfto       hash_kfto
 hkmnull     hash_kmnull
 ex.client   none
 k-out-of-n  1/1
 other quora m=1 r=1 nv=1 rtc=1 dsee=1 fto=1
 createtime  2024-11-04 16:25:39
 nso timeout 10 min
 ciphersuite ECp521mAES
 kmltype     NISTp521hSHA1
 min pp      0 chars
 mode        none

Module #1
 generation    1
 state         0x1 Usable
 flags         0x10000 ShareTarget
 n_slots       2
 esn           34F3-9CB4-753B
 hkml          hash_kml
 kmltype       NISTp521hSHA1

Module #1 Slot #0 IC 11
 generation    1
 phystype      SmartCard
 slotlistflags 0x2
 state         0x4 Operator
 flags         0x20000 RemoteEnabled
 shareno       2
 shares
 error         OK
 Cardset
 name          "fred"
 k-out-of-n    1/2
 flags         NotPersistent
 timeout       none
 card names    "" ""
```

```
 hkltu         hash_kt

Module #1 Slot #1 IC 0
 generation    1
 phystype      SmartCard
 slotlistflags 0x2 SupportsAuthentication
 state         0x4 Admin
 flags         0x10000 passphrase
 shareno       1
 shares        LTNSO(PIN) LTM(PIN) LTR(PIN) LTNV(PIN) LTRTC(PIN) LTDSEE(PIN)
 LTFTO(PIN)
 error         OK
 No Cardset

No Pre-Loaded Objects
```

126.1.1.2.1. World

`nfkminfo` reports the following information about the Security World:

`generation`

This indicates the internal number.

`state`

This indicates the status of the current world:

`Initialised`	This indicates that the Security World has been initialized.
`Usable`	This indicates that there is at least one usable HSM in this Security World on this host.
`!Usable`	This indicates that there are no usable HSMs in this Security World on this host.
`Recovery`	This indicates that the Security World has the OCS and softcard replacement and the key recovery features enabled.
`!Recovery`	This indicates that the Security World has the OCS and softcard replacement and the key recovery features disabled.
`AdminAuthRequired`	This indicates that additional authorization is required for the following operations: • Key generation • Public key import • Operator cardset creation • Softcard creation. This authorization is supplied by presenting any operator or administration card from the same Security World. A passphrase is not required:

ExistingClient	This indicates that there is a Client ID set, for example, by preload. This Client ID is given in the ex.client output if the --preload-client-id flag was supplied.
!ExistingClient	This indicates that no Client ID is set. The ex.client output will be empty.
AlwaysUseStrongPrimes	This indicates that the Security World always generates RSA keys in a manner compliant with FIPS 186-3.
!AlwaysUseStrongPrimes	This indicates that the Security World leaves the choice of RSA key generation algorithm to individual clients.
SEEDebug	This indicates that the Security World has an SEE Debugging delegation key.
!SEEDebug	This indicates the Security World has no SEE Debugging delegation key.
SEEDebugForAll	This indicates no authorization is required for SEE Debugging.
PINRecovery	This indicates that the Security World has the passphrase replacement feature enabled.
!PINRecovery	This indicates that the Security World has the passphrase replacement feature disabled.
FTO	This indicates that the Security World has an FTO delegation key.
!FTO	This indicates that the Security World has no FTO delegation key.
NVRAM	This indicates that the Security World has an NVRAM delegation key.
!NVRAM	This indicates that the Security World has no NVRAM delegation key.
RTC	This indicates that the Security World has an RTC delegation key.
!RTC	This indicates that the Security World has no RTC delegation key.
AuditLogging	This indicates that Audit Logging is enabled for this Security World.
!AuditLogging	This indicates that Audit Logging is not enabled for this Security World.

n_modules

This indicates the number of nShield HSMs connected to this computer.

hknso

This indicates the SHA-1 hash of the Security Officer's key.

hkm

This indicates the SHA-1 hash of the Security World key.

hkmwk

This indicates the SHA-1 hash of a dummy key used to load the Administrator Card Set (the

dummy key is the same on all HSMs that use Security Worlds and is not secret).

hkre

This indicates the SHA-1 hash of the recovery key pair.

hkra

This indicates the SHA-1 hash of the recovery authorization key.

ex.client

This indicates the `ClientID` required to use any pre-loaded keys and tokens.

k-out-of-n

This indicates the values of *K* and *N* for this Security World.

other quora

This indicates the number (quora) of Administrator Cards (*K*) required to perform certain other functions as configured for this Security World.

ciphersuite

This indicates the name of the Cipher suite that the Security World uses.

kmltype

This indicates the KML type used when the Security World was created.

Mode

none	This indicates that the Security World is in an unregulated mode. The Security World can be configured to meet the needs of your security policy. This includes, but is not limited to, creating a Security World that is compliant with FIPS140 Level 2.
fips1402level3	This indicates that the Security World is in a mode compliant with FIPS 140 Level 3.
commoncriteriacmts	This indicates that the Security World is in a mode compliant with Common Criteria Protection Profile EN 419 221-5, for Cryptographic Modules for Trust Services.

Assigned Keys

max usage	This indicates the maximum key usage reauthorization condition for Assigned Keys. (common-criteria-cmts mode only).

`max timeout`	This indicates the maximum key timeout reauthorization condition for Assigned Keys (common-criteria-cmts mode only).

126.1.1.2.2. Module

For each HSM in the Security World, `nfkminfo` reports:

`generation`

This indicates the version of the HSM data.

`state`

This indicates one of the following:

`PreInitMode`	This indicates that the HSM is in the pre-initialization state.
`InitMode`	This indicates that the HSM is in the initialization state.
`Unknown`	This indicates that the HSM's state could not be determined.
`Usable`	This indicates that the HSM is programmed in the current Security World and can be used.
`Uninitialized`	This indicates that the HSM does not have the Security Officer's key set and that the HSM must be initialized before use.
`Factory`	This indicates that the HSM has module key zero only and that the Security Officer's key is set to the factory default.
`Foreign`	This indicates that the HSM is from an unknown Security World.
`AccelOnly`	This indicates that the HSM is acceleration only.
`Unchecked`	This indicates that, although the HSM appears to be in the current Security World, `nfkminfo` could not find a module initialization certificate (a `module_<ESN>` file) for this HSM.
`Failed`	This indicates that the HSM has failed. For nShield HSMs running firmware 2.61.2 and above, use the `enquiry` utility for further information about the failure reason. On network-attached HSMs, you can also look for hardware errors in the hardserver log.
`MaintMode`	This indicates that the HSM is in the maintenance state.

`flags`

This displays ShareTarget if the HSM has been initialized to allow reading of remote card sets.

`n_slots`

This indicates the number of slots on the HSM (there is one slot for each physical smart card reader, one slot for each soft token, one slot for each available Remote Operator slot and one slot for each associated Dynamic Slots).

`esn`

This indicates the electronic serial number of the HSM (if the HSM is not in the `Usable` state, the electronic serial number may not be available).

`hkml`

This indicates the hash of the HSM signing key (if the HSM is not in the `Usable` state, this value may not be available).

`kmltype`

This indicates the KML type used by the HSM (if the HSM is not in the `Usable` state, this value may not be available).

126.1.1.2.3. Slot

For each slot on the HSM, `nfkminfo` reports:

`IC`

This indicates the insertion count for this slot (which is 0 if there is no card in the slot).

`generation`

This indicates the version of the `slotinfo` structure.

`phystype`

This indicates the type of slot, which can be one of:

- `SmartCard`
- `SoftToken`

`slotlistflags`

These are flags describing the capabilities of the slot. Single letters in parentheses are the flag codes reported by the `slotinfo` utility:

`0x2`	`(A) SupportsAuthentication` This indicates that the slot supports token-level challenge-response authentication.

0x40000	(R) RemoteSlot This indicates that the slot is a Remote Operator slot that has been imported from a remote HSM.
0x80000	(D) DynamicSlot This indicates that it is a Dynamic Slot.
0x100000	(a) Associated This indicates that a Remote Administration Client has associated a card reader with this
0x200000	(t) TimedOut This indicates that no response has been received from the smartcard in this Dynamic Slot within the configured timeout.
0x400000	(f) SecureChannelFailed This indicates that the secure channel between the HSM and the smartcard in this Dynamic Slot has failed in some way.

state

This can be one or more of the following flags:

Blank	This indicates that the smart card in the reader is unformatted.
Admin	This indicates that the smart card in the reader is part of the Administrator Card Set.
Empty	This indicates that there is no smart card in the reader.
Error	This indicates that the smart card in the reader could not be read (the card may be from a different Security World).
Operator	This indicates that the smart card in the reader is an Operator Card.

flags

This displays passphrase if the smart card requires a passphrase.

shareno

This indicates the number of the card within the card set.

shares

If the card in the slot is an Operator Card, no values are displayed for shares.

If the card in the slot is an Administrator Card, values are displayed indicating what key shares are stored on the card. Each share is prefixed with the letters LT (Logical Token), and the remaining letters identify the key (for example, the value LTNSO indicates that a share of K_{NSO}, the Security Officer's key, is stored on the card).

`error`

This indicates the error status encountered if the smart card could not be read:

`OK`	This indicates that there were no errors.
`TokenAuthFailed`	This indicates that the smart card in the reader failed challenge response authentication (the card may come from a different Security World).
`PhysTokenNotPresent`	This indicates that there is no card in the reader.

If you purchased a developer kit, you can refer to the relevant developer documentation for a full list of error codes.

126.1.1.2.4. Card set

If there is an Operator Card in the reader, `nfkminfo` reports:

`name`

This indicates the name given to this card set.

`k-out-of-n`

This indicates the values of *K* and *N* for this card.

`flags`

This displays one or more of each of the following pairs of flags:

`NotPersistent`	This indicates that the Operator Card is not persistent.
`Persistent`	This indicates that the Operator Card is persistent.
`NotRemoteEnabled`	This indicates that the card in the slot is not from a Remote Operator Card Set.
`RemoteEnabled`	This indicates that the card in the slot is from a Remote Operator Card Set.
`PINRecoveryForbidden(disabled)`	This indicates that the card in the slot does not have passphrase replacement enabled. This is always true if passphrase replacement is disabled for the Security World.
`PINRecoveryRequired(enabled)`	This indicates that the card in the slot does have passphrase replacement enabled.

`timeout`

the period of time in seconds after which the HSM automatically removes the Operator Card Set. If timeout is set to `none`, the Operator Card Set does not time out.

`card`

lists the names of the cards in the set, not all software can give names to individual cards in a set.

`hkltu`

the SHA-1 hash of the secret on the card.

127. nfkmverify

```
nfkmverify [-fvU] [-m MODULE] [appname ident [appname ident [...]]]
```

Establishes the soundness of security world infrastructure and application keys.

127.1. nfkmverify options

Option	Description
Program options	
-A, --assigned	In a common-criteria-cmts world, checks whether the key is assigned.
-f, --force	Forces the display of possibly-wrong output report.
-v, --verbose	Prints full public keys and generation parameters.
--trusted-certifier=HASH	Trust the seeinteg certifier with this hash. This option can also take a list of key hashes separated by commas or spaces. It instructs nfkmverify to regard these keys as trusted even if they cannot be verified.
Key checking options	
-C, --certificate	Check original ACL for the key using key generation certificate. (Default)
-L, --loaded	Checks the ACL of the loaded key instead of the generation certificate.
-R, --recov	Checks the ACL of the key loaded from the recovery blob.
Option to accept particular discrepancies	
--allow-dh-unknown-sg-group	Proceeds if a Diffie-Hellman key uses an unrecognized Sophie-Germain group.
-U, --unverifiable	Proceeds even if the security world is unverifiable.
Option to address HSMs	
-m, --module=MODULE	Specifies the number of the module to perform the test with. If you only have one module, <MODULE> is 1.
Help options	
-h, --help	Displays help for nfkmverify.
-u, --usage	Displays a brief usage summary for nfkmverify.
-V, --version	Displays the version number of the Security World Software that deploys nfkmverify.

127.2. Verify a migrated key

To verify a migrated key, you must preload the key and use `nfkmverify` with either `-L|--loaded` or `-R|--recov` options.

By default, `nfkmverify` compares the original Access Control List (ACL) that was provided when a key was generated to the current Security World. If the key was migrated, then the key hashes and mechanisms in the original ACL will not be consistent with the current Security World and `nfkmverify` will report a discrepancy. It might also be unable to load the KML blob necessary to verify the original ACL.

If the key is protected by a foreign `seeinteg` key, that is, a `seeinteg` key from another security world, you must use the `--trusted-certifier` option. Otherwise verification will fail because the `seeinteg` key cannot be verified.

128. nfloadmon

```
nfloadmon [-m MODULE] <monitor file> <firmware file>
```



`nfloadmon` upgrades the module monitor and the firmware for the module. Read the firmware upgrade documentation and the release notes for the firmware version before you run `nfloadmon`.

Upgrades the module monitor and firmware of the HSM.

For more information, see Upgrade firmware: nShield Solo, Solo XC, and Edge HSMs.

Option	Description
`--automode`	Tries to automatically switch module mode. You might still need to change module mode manually.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to perform the upgrade on. If you only have one module, `<MODULE>` is `1`. Default: module `1`.
Help options	
`-h`, `--help`	Displays help for `nfloadmon`.
`-u`, `--usage`	Displays a brief usage summary for `nfloadmon`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfloadmon`.

129. nfls

```
nfls [OPTIONS] FILENAME [FILENAME ...]
```

Lists the remote file or files specified by FILENAME in the following format:

IPv4 addresses A.B.C.D:volume/file

IPv6 addresses [A:B:C:D]:volume/file

Option	Description
`-c`, `--checksum`	Shows the SHA-1 file checksums.
`-k`, `--use-kneti`	Uses a local module KNETI to authenticate this client to the remote server. If this option is not specified, the hardserver's software KNETI is used instead.
`-n`, `--no-list`	Stats a directory rather than listing the contents.
`-P`, `--port=PORT`	Sets the port through which to connect to the remote server. Default: `9004`.
`--remote-esn=ESN`	Specifies the ESN of the nToken to be used to authenticate the remote hard-server. Cannot be specified without the corresponding remote KNETI hash option.
`--remote-hkneti=HKNETI`	Specifies the KNETI hash to authenticate the remote hardserver.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module use KNETI to use. This option is ignored unless `--use-kneti` has been used as well. If you only have one module, `<MODULE>` is `1`. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `nfls`.
`-u`, `--usage`	Displays a brief usage summary for `nfls`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfls`.
`-V`, `--verbose`	Be verbose.

130. nfrm

```
nfrm [OPTIONS] FILE [FILE [...]]
```

Removes files either locally or remotely using hardserver file transfer commands. Doesn't accept glob patterns. IPv6 addresses must be surrounded by square brackets.

Option	Description
`-f`, `--force`	Ignores nonexistent files.
`-k`, `--use-kneti`	Uses a local module KNETI to authenticate this client to the remote server. If this option is not specified, the hardserver's software KNETI is used instead.
`-P`, `--port=PORT`	Sets the port through which to connect to the remote server. Default: `9004`.
`--remote-esn=ESN`	Specifies the ESN of the nToken to be used to authenticate the remote hard-server. Cannot be specified without the corresponding remote KNETI hash option.
`--remote-hkneti=HKNETI`	Specifies the KNETI hash to authenticate the remote hardserver.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module use KNETI to use. This option is ignored unless `--use-kneti` has been used as well. If you only have one module, `<MODULE>` is `1`. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `nfrm`.
`-u`, `--usage`	Displays a brief usage summary for `nfrm`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nfrm`.
`-V`, `--verbose`	Be verbose.

131. nfwarrant

```
nfwarrant [-h] [--list] [--check] [--warrant] [--csr] [--details= FILE]
          [--install= FILE] [--req= MODULE] [--out= FILE] [--verbose]
          [--version]
```

Ensures that a suitable warrant is available to allow a security world to be dynamically managed using an nShield PCIe or USB-attached HSM.

Run `nfwarrant` to:

- Identify modules that have the appropriate firmware/KLF2 key
- Identify modules that need their KLF2 key to be warranted by Entrust
- Generate a warrant upgrade request for a specific module, as required
- Install an upgraded warrant
- List KLF2 warrants

See Warrant Management for more information.

Option	Description
`--check`	Lists the ESNs of known modules and their warrant state.
`--csr`	Performs CSR operations.
`--details= FILE`	Displays the module ESN found in the CSR/warrant `FILE`.
`--install= FILE`	Installs the warrant from `FILE`.
`--list`	Lists the ESNs of installed warrants.
`--out= FILE`	Saves the new requested CSR to `FILE`.
`--warrant`	Performs warrant operations.
Option to address HSMs	
`--req= MODULE`	Requests a warrant CSR for the module number/ESN.
Help options	
`-h`, `--help`	Displays help for `nfwarrant`.
`--verbose`	Prints extra information about CSR and warrant files.
`--version`	Displays the version number of the Security World Software that deploys `nfwarrant`.

132. nopclearfail

```
nopclearfail -n|-c|-f|-S|-r|-M|-O|-I -w -a|-m MODULE
```

Clear an HSM, put an HSM into the error state, retry a failed HSM, or change the HSM mode.

For information about changing the nShield HSM mode, see

- Network-attached HSMs: Checking and changing the mode on a network-attached HSM
- nShield Solo and Solo XC: Checking and changing the mode on nShield Solo and nShield Solo XC HSMs
- nShield 5s: nShield 5s modes of operation
- USB HSMs: Checking and changing the mode on an nShield Edge

Solo XC only: Reboot the Solo XC, for example after a firmware upgrade, without needing to reboot the host, see Upgrading firmware only.

Option	Description
Action selection	
`-c`, `--clear`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Sends the `ClearUnit` command to the module.
`-f`, `--fail`	Sends the `Fail` command to the module.
`-I`, `--initialization`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Puts the module into pre-initialization mode.
`-M`, `--maintenance`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Puts the module into maintenance mode.
`-n`, `--no-op`	Sends the `NoOp` command to the module.
`-O`, `--operational`	Can only be executed as privileged user (a user with a privileged connection to the HSM). Puts the module into operational mode.
`-r`, `--retry`	Sends the `RetryFailedModule` command to restore a failed local or remote module and its remote slots, if possible.
`-S`, `--hot-reset`	Performs a hot reset. The module must be in Maintenance mode before you run `nopclearfail` with this option.

Option	Description
Miscellaneous	
`-w`, `--wait`	Sends a `NoOp` to the affected modules and waits for a reply before exiting.
Options to address HSMs	
`-a`, `--all`	Sends the command to all modules.
`-m`, `--module=MODULE`	Sends the command to module `MODULE`. If you only have one module, `MODULE` is `1`.
Help options	
`-h`, `--help`	Displays help for `nopclearfail`.
`-u`, `--usage`	Displays a brief usage summary for `nopclearfail`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `nopclearfail`.

133. npkgtool

```
npkgtool [-h]
        {create,inspect,hash,presig,encrypt,sign,addsig,unpack,create-keyfile,generate-key,import-key,json-
import-keys,drop-key}
        ...
```

Manages NPKG files.

Option	Description
Positional arguments	
`addsig`	Imports the `ContentHash` and `ContentSignature` records from one package file into another.
`create`	Creates a new package from the specified contents file.
`create-keyfile`	Creates an empty keyfile.
`drop-key`	Removes a key or key pair from a keyfile.
`encrypt`	Encrypts the contents of a package.
`hash`	Generates a new, small package file containing `ContentHash` records for the content in the input file, but not including the content itself, for off-line signing.
`generate-key`	Generates a key or key pair and add it to a keyfile.
`import-key`	Imports a public key pair from a security world to a keyfile.
`inspect`	Displays the header records of a package file.
`json-import-keys`	Imports keys from a json file.
`presig`	Generates a new, small package file containing `ContentHash` records for the content in the input file, but not including the content itself, for off-line signing.
`sign`	Signs the contents of a package.
`unpack`	Retrieves the contents of a package file, decrypting if necessary.
Help option	
`-h`, `--help`	Displays help for `npkgtool`.

134. nshieldaudit

```
nshieldaudit [-h] [-v]
nshieldaudit syslogs [-h] {query,export,info} [parameters]
nshieldaudit ncore [-h] {query,export} [parameters]
nshieldaudit serveradmin [-h] [parameters]
```

Reads the nShield Audit Log Service database. This utility is described in detail here: Read the audit log database

Option	Description
Subcommands	
`syslogs`	Enables the user to access the signed system logs stored in the database. See Reading signed system logs
`ncore`	Enables the user to access the nCore audit logs stored in the database. See Reading nCore audit logs
`serveradmin`	Enables the user to check and repair the temporary database for nCore audit logs that haven't been processed by the nShield Audit Log Service. See Checking or repairing hardserver audit database
Help options	
`-h`, `--help`	Displays help for `nshieldaudit`.
`-v`, `--version`	Prints the `nshieldaudit` version.

135. ntokenenroll

```
ntokenenroll [OPTIONS]
```

Enrolls a locally attached nToken with an nShield HSM. `ntokenenroll` installs the Electronic Serial Number (ESN) of the nToken within the client configuration file and displays the module's ESN and the hash of the key to be used in nToken authentication. The network-attached HSM will need to be able to connect to TCP port `9004` on this host for this to work. For more information, see Configuring the unit to use the client.

Option	Description
`-a`, `--add`	Enrolls with the remote module (default)
`-c`, `--configfile=FILENAME`	Name of the configuration file to read and write.
`-H`, `--hashes`	Displays key hashes for all local modules.
`-q`, `--quiet`	Quiet operation.
`-r`, `--remove`	De-enrolls from the remote module.
`-t`, `--token=MODULE`	Selects the local module to use.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the remote module to enroll with or de-enroll from.
Help options	
`-h`, `--help`	Displays help for `ntokenenroll`.
`-u`, `--usage`	Displays a brief usage summary for `ntokenenroll`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `ntokenenroll`.

136. nvram-backup

```
nvram-backup -l|-c|-d -M|-S [-fvx] [-m MODULE] [-s SLOT] FILES
```

Copies files between a module's NVRAM and a smartcard, allowing the files to be backed up and restored.

Option	Description
Action selection	
`-c`, `--copy`	Copies files.
`-d`, `--delete`	Deletes files (from the module only).
`-l`, `--list`	Lists files.
Transfer direction	
`-M`, `--from-module`	Backs up files from the module to the smartcard or list the files on the module.
`-S`, `--from-smartcard`	Restores files from the smartcard to the module or lists the files on the smartcard.
General options	
`-f`, `--force`	Forces the operation. Otherwise, `nvram-backup` prompts before deleting or overwriting a file.
`--no-length`	Doesn't print file length for `--list`.
`-x`, `--hex`	Uses hex notation for filename glob patterns.
Option to address HSMs	
`-m`, `--module=MODULE`	Read files and cards from module `MODULE`. If you only have one module, `<MODULE>` is `1`. Default: `1`.
`-s`, `--slot=SLOT`	Reads files from or writes files to slot `SLOT`. Default: `0`. This must be a local slot. This option is not required for listing or deleting files on a module.
Help options	
`-h`, `--help`	Displays help for `nvram-backup`.
`-u`, `--usage`	Displays a brief usage summary for `nvram-backup`.
`-v`, `--verbose`	Prints verbose output for `nvram-backup`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `nvram-backup`.

137. nvram-sw

```
nvram-sw --alloc [-m MODULE] [-b BYTES] [-n ID] [-k APPNAME,IDENT]
nvram-sw --delete [-m MODULE] [-n ID]
nvram-sw --write [-m MODULE] [-f FILE]
nvram-sw --read [-m MODULE] [-f FILE]
nvram-sw --delete-noadmin [-m MODULE] [-n ID]
nvram-sw --acl [-m MODULE] [-n ID]
nvram-sw --list [-m MODULE]
```

Views or modifies information about NVRAM areas.

Option	Description
Action selection	
`-a`, `--alloc`	Allocates a new NVRAM area. ACS is required.
`-c`, `--delete-noadmin`	Deletes an NVRAM area without the ACS. OCS may be required.
`-d`, `--delete`	Deletes an NVRAM area. ACS is required.
`-i`, `--acl`	Displays the ACL of the NVRAM area. OCS may be required.
`-l`, `--list`	Lists the entire contents of the NVRAM.
`-r`, `--read`	Prints data from the NVRAM area to a file or `stdout`. OCS may be required.
`-w`, `--write`	Writes data to the NVRAM area from a file or `stdin`. OCS may be required.
General option	
`-x`, `--hex`	Uses hex notation for `--nvram-id`.
Action-specific options	
`-b`, `--bytes=BYTES`	Number of bytes. Default: `100` for allocation and size of file for reading.
`-f`, `--file=FILE`	File for input/output. Default = `stdin`/`stdout`.
`-k`, `--key=APPNAME,IDENT`	Specify a key during allocation; this will be required on all subsequent reads or writes on the file.
`-n`, `--nvram-id=ID`	NVRAM file ID (default = "test-file").
`--no-copy`	Allocate file with an ACL that disallows copying.

Option	Description
`-p`, `--persistent`	For the `list' and 'delete' actions, list/delete only persistent files.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to use. If you only have one module, `<MODULE>` is `1`. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `nvram-sw`.
`-u`, `--usage`	Displays a brief usage summary for `nvram-sw`.
`-v`, `--verbose`	Prints verbose output for `nvram-sw`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `nvram-sw`.

138. openssl

139. p11hyper

140. passwd

```
passwd
```

Sets the serial console password.

141. perfcheck

```
perfcheck [OPTIONS] [SUITE:TEST_NUMBER ...]
```



Only supported in FIPS 140-2 Level 2 Security Worlds.

Runs various tests to measure the cryptographic performance of a module.



Performance results vary depending on the selected cryptographic acceleration profile. See Cryptographic Acceleration Profiles for details.

By default, tests are run 3 times and with both `max-queue` and the queue of 1. To run tests just once each with `max-queue`, use the `-s` or `--single` parameter.

141.1. perfcheck example command lines

Run default set of tests	`perfcheck`
Run particular suites of tests	`perfcheck kx signing`
Run particular tests by ID	`perfcheck kx:1 kx:2`
Run particular tests by exact name	`perfcheck "signing:RSA using RSApPKCS1 with 2048-bit n."`
Run particular tests by prefix	`perfcheck signing:RSA*`
Run a list of tests from file	`perfcheck --testlist-file=tests.txt`
List default set of tests	`perfcheck --list`
List the tests within particular suites	`perfcheck --list kx signing`
List the available suites	`perfcheck --help-suites`
Compare two results sets	`perfcheck --diff --old=OLD_PATH --new=NEW_PATH`

141.2. perfcheck syntax

Option	Description
Output options for test execution and report diffs	

Option	Description
`--capture-raw-data`	Raw data for each test case will be written to the `raw_results` subdirectory of the output directory (`--outputdir`) which must be specified. Input parameters and overall results for each run of a test case are stored in the `JSON` files, and the timings of each individual job are stored in the `CSV` files.
`-o OUTPUT`, `--outputdir=OUTPUT`	Output destination directory name. A subdirectory named `perfcheck.DATE_AND_TIME` based on current time will be created.
`-n`, `--nosubdir`	Does not create a subdirectory of the `outputdir`. Reports will be written directly to that path which must not already exist.
`-p`, `--show-progress`	Prints a summary of the latest results for a test every second during execution. No interim results will be printed for tests that take less than a second to complete.
Diff options to produce a comparison report between two sets of test results to identify significant regressions or improvements in performance. The `-o`, `--outputdir` and `-n`, `--nosubdir` output options can be used in conjunction with this.	
`--diff`	Creates a comparison report.
`--new=NEW_RESULTS_PATH`	Path to the new result set.
`--old=OLD_RESULTS_PATH`	Path to the old result set.
`--threshold=THRESHOLD`	Threshold percentage regression to report as error. The threshold will be applied only to results that have a significance rating of at least 50% and to tests in suites other than Miscellaneous and Key Generation. The error will be reported on `stderr` and reflected in the exit code, but does not otherwise affect report generation.
Testing options to configure the test execution behavior.	
`--accuracy=base|default|high`	Accuracy level. Options are `base`, `default` and `high` in order of increasing test accuracy. More accurate tests take longer to execute. This adjusts the default values of `--target-test-rse`, `--max-test-time`, and `--run`s, but if any of those options are set explicitly, the explicit value will take precedence.
`--client-throttle-spins=SPIN_COUNT`	Throttle input to system by spinning for the specified number of instructions before submitting each job. Default: `0` for no throttling
`-d headline|overview|core|default|full`, `--depth=headline|overview|core|default|full`	Depth of testing. Options are `headline`, `overview`, `core`, `default` and `full` in order of increasing test depth. The higher the depth of testing, the more test cases are run. This option is not relevant if individual tests are specified directly by name or test id, but controls the depth of testing where all tests are requested or a particular suite. This option also affects listing tests, so to see all available tests specify `--depth=full` when listing; this will show some additional parametrizations that are not run by default.

Option	Description
`-f TESTLIST_FILE`, `--testlist-file=TESTLIST_FILE`	Path to a file containing a list of tests to run. Each line should contain a test in the same format as is supported on the command-line. For example, `suite:description, suite` to run a whole suite or `suite:PREFIX*` to run all tests whose name starts with `PREFIX`. To run the same tests as from a previous run, pass the path to the `testlist.txt` file that was automatically written to the output directory of that run.
`--max-test-time=TIME_SECONDS`	Time limit for individual test cases or 0 for no limit. Defaults based on accuracy level: base: `30` seconds, default: `60` seconds, high: `150` seconds.
`-q QUEUE_SIZE`, `--queue=QUEUE_SIZE`	Specifies the request queue size or `0` to run with max queue reported by `enquiry`. Default: run operations both with queue of `1` and with the max queue reported by `enquiry`, that is, both one-off and bandwidth measurements.
`-r REPS`, `--repetitions=REPS`	Runs this many repetitions instead of the default. More may be run due to other constraints.
`--runs=COUNT`	Runs each selected test case the specified number of times. Values above `1` allow variance between runs to be detected. Defaults based on accuracy level: base: `1`, default: `3`, high: `5`.
`-s`, `--single`	Does a single run of each test case. This is a shorthand for `--runs=1 --queue=0` (that, is `max-q`) but if either of those options are specified explicitly the explicit value takes precedence.
`-t TIME_SECONDS`, `--min-test-time=TIME_SECONDS`	Minimum time to run individual test cases. If set to `10` or above, the `--show-progress` option will be turned on automatically. Default: `0` seconds.
`--target-test-rse=RSE_PERCENT`	Target relative standard error percentage or `0` for none. Each test will keep running until this error target is met or `--max-test-time` is reached. Defaults based on accuracy level: base: `1.0`%, default/high: `0.1`%.
`--thread-count=THREAD_COUNT`	Number of client threads from which to fill the queue. The queue will be split evenly across the threads. Default: `1`.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the number of the module to perform the test with. If you only have one module, `<MODULE>` is `1`. Default: `1`.
Help options	
`-h`, `--help`	Displays help for `perfcheck`.
`--help-suites`	displays help for the available test suites.

Option	Description
`-l`, `--list`	Lists all tests that will be run in the selected suites.
`-u`, `--usage`	Displays a brief usage summary for `perfcheck`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `perfcheck`.

141.3. perfcheck tests

The available tests are grouped into suites:

- `kx` (key exchange)
- `keygen` (key generation)
- `signing` (signing)
- `verify` (verification)
- `enc` (encryption)
- `dec` (decryption)
- `misc` (miscellaneous)

To see the list of tests run by default in a particular suite, run a command of the form:

```
perfcheck --list suite
```

To see all available tests in a particular suite, including those not run by default, run a command of the form:

```
perfcheck --list --depth=full suite
```

For example, to list all the `signing` tests, run the command:

```
perfcheck --list --depth=full signing
>>> Suite `signing' -- Signing (374 tests)
>>>    signing 1 - DSA using RIPEMD160 with 1024-bit p and 160-bit q.
>>>    signing 2 - DSA using RIPEMD160 with 2048-bit p and 160-bit q.
>>>    signing 3 - DSA using RIPEMD160 with 3072-bit p and 160-bit q.
>>>    signing 4 - DSA using SHA1 with 1024-bit p and 160-bit q.
>>>    signing 5 - DSA using SHA1 with 2048-bit p and 160-bit q.
>>>    signing 6 - DSA using SHA1 with 3072-bit p and 160-bit q.
```

In the output, each listed test in the suite is identified with a number.

You can reference a test either by its number or by its name:

- by test number:

```
perfcheck suite:test_number
```

To use test 16 of the signing suite:

```
perfcheck signing:16
```

- by test name:

```
perfcheck "signing:RSA using RSAhSHA3b512pPSS with 4096-bit n."
```

Example:

```
perfcheck "signing:RSA using RSApPKCS1 with 2048-bit n."
```

The test numbers change between releases. If you want to rerun tests for comparison, refer ence the tests by their names.

perfcheck prints the results of individual tests to output as it goes along, and then prints a full report at the end. By default, perfcheck runs each test three times for both minimum and maximum queue sizes, and then collates the results in the final report. See --help for the options to adjust this behavior.

Optionally, perfcheck can write its output to a directory in multiple formats using the --out putdir option to specify a directory name. This will create a new subdirectory under the specified directory to write the output. The --nosubdir option can be added as well to write output to the specified directory directly, in which case that directory must not already exist. The output directory will contain perfcheck.html, perfcheck.txt, perfcheck.csv, and perfcheck.json files that contain the report in HTML, text, CSV, and JSON format respec- tively. JSON files that contain the detailed results of individual tests will also be written to the output directory.

Output reports from test suites include the following information about each test:

Value	Description
CV (%)	This value is the coefficient of variation expressed as a percentage of the mean latency. It gives an indication of the variability in the time it takes individ ual jobs to complete. If a test has been rerun, this is the mean of the CV (%) values from each run.
Max latency (ms)	This value is the time in milliseconds that the slowest individual job across all the test runs took to round-trip.

Value	Description
Max rate (tps)	This is the estimated upper bound of the throughput for this queue size in transactions per second. The value becomes more accurate if more test runs of the same test are done. When it is compared against Min rate (tps) and Mean rate (tps), Max rate (tps) gives an indication of the variability between runs.
Mean latency (ms)	This value is the mean time in milliseconds that jobs took to round-trip. If a test has been rerun, this is the mean of the mean latency values from each run.
Mean rate (tps)	This is a measure of throughput. Unlike Rate (Units/s), it is expressed in transactions per second, that is, as the number of jobs that round-trip per second. Mean rate (tps) is included for comparison against the Min rate (tps) and Max rate (tps) figures.
Min latency (ms)	This value is the time in milliseconds that the quickest individual job across all the test runs took to round-trip.
Min rate (tps)	This is the estimated lower bound of the throughput for this queue size in transactions per second. The value becomes more accurate if more test runs of the same test are done. When it is compared against Mean rate (tps) and Max rate (tps), Min rate (tps) gives an indication of the variability between runs.
Queue	This value is the number of outstanding jobs in the queue when the test was run. By default, most tests run both with a queue of 1, and with a fully maxed out module queue, to give an indication of both one-at-a-time performance and the bandwidth for the operation. The queue can be set differently using the `--queue` option, in which case only that queue length will be run with, except for some `misc` suite tests which set their own queue.
Rate (Units/s)	This value is a measure of throughput. It is calculated by dividing the number of repetitions by total time. If a test has been rerun to improve accuracy, as is the case by default, then this is the mean across all the runs. Some tests, for example `enc`, set the Unit to something other than an operation, for example KB, to indicate the amount of data that can be encrypted.
Reps	This value is the number of repetitions that were actually carried out, that is, the number of jobs that were round-tripped over all tests of this operation for this queue size. If a test was rerun, this is the sum of the repetitions for each run. The target repetitions for an individual run can be set using the `--repetitions` option but note that in most cases more repetitions will be run depending on the `--accuracy` setting provided that the timeout is not reached. It is recommended to set `--accuracy` rather than `--repetitions` to control the accuracy of the test instead of adjusting the repetitions.

141.4. How perfcheck calculates statistics

The `perfcheck` utility sends multiple simultaneous nCore commands to keep the HSM busy. It can send more commands if a required number of repetitions has not yet been reached.

After sending some initial commands, `perfcheck` begins marking commands with the time at which are submitted. When a command comes back with a timestamp, `perfcheck` checks the amount of time needed to complete the command and updates the values for `Std dev` and `Latency`. The value of `Total time` is the amount of time from sending the first job to receiving the final one.

When an nCore command is submitted to an HSM by a client application, it is processed as follows:

PCIe and USB HSMs

Because an HSM can execute several commands at once, throughput is maximized by ensuring there is always at least one command in the hardserver queue (so that there are always commands available to give to the HSM).

1. The command is passed to the hardserver.
2. The client hardserver encrypts the command.
3. When the HSM is free, the command is submitted from the hardserver queue.
4. The command is executed by the HSM, and the reply is given to the hardserver.
5. The unit hardserver queues the reply.
6. The unit hardserver sends the command back to the client hardserver over the network.
7. When the client application is ready, the queued reply is returned to it.

network-attached HSMs

Because an HSM can execute several commands at once, throughput is maximized by ensuring there is always at least one command in the unit hardserver queue (so that there are always commands available to give to the HSM).

1. The command is passed to the client hardserver.
2. The client hardserver encrypts the command.
3. The encrypted command is sent to the unit hardserver over the network.
4. The unit hardserver decrypts the command and queues it.
5. When the internal security module is free, the command is submitted from the hardserver queue.

6. The command is executed by the HSM, and the reply is given to the unit hardserver.
7. The unit hardserver encrypts the command.
8. The unit hardserver sends the command back to the client hardserver over the network.
9. The client hardserver decrypts the reply and queues it.
10. When the client application is ready, the queued reply is returned to it.

142. ping

```
ping address
```

Pings a remote host.

Option	Description
address	The IPv4 or IPv6 address of the remote host.

143. pollbare

```
pollbare [-q] [-t TIME] [-m MODULE]
```

Obtains information about state changes by running `PollModuleState` and displaying the results in a terse format. se this utility to ensure that the HSMs are functioning as expected and to test the cryptographic functionality at the nCore level.

The functionality of this test utility depends on whether the server or an HSM supports nCore API poll commands. To check if your server or HSM supports nCore API poll commands, run `enquiry`.

Option	Description
`-q`, `--quiet`	Runs only once.
`-t`, `--time=TIME`	Updates every `TIME` seconds. Default: `1`.
Option to address HSMs	
`-m`, `--module=MODULE`	Specifies the ID of the module to send the command to. If you only have one module, `MODULE` is `1`. Default: all modules.
Help options	
`-h`, `--help`	Displays help for `pollbare`.
`-u`, `--usage`	Displays a brief usage summary for `pollbare`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `pollbare`.

144. postload-bsdlib

```
postload-bsdsee --module MOD --provision PROV [OPTIONS...]
```

Option	Description
`--provision PROV`	
Option to address HSMs	
`--module=MOD`	

145. postrocs

```
postrocs -m  MODULE -s SLOT
```

Transfers PKCS #11 keys to a new card set in the new security world. When transferring keys by using either the `key-xfer-im` utility or the `migrate-world` utility, run the `postrocs` utility if there are any PKCS #11 keys that are protected by OCSs.



The prefix of PKCS #11 keys is `keys_pkcs_um` or `key_pkcs_uc`.

Option	Description
`-m`, `--module=MODULE`	Specifies the number of the module to use. If you only have one module, `MODULE` is `1`.
`-s`, `--slot=SLOT`	Selects `SLOT` for the card to re-attach keys.
Help options	
`-h`, `--help`	Displays help for `postrocs`.
`-u`, `--usage`	Displays a brief usage summary for `postrocs`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `postrocs`.

146. ppmk

```
ppmk --list
ppmk --new [-R] [-m MODULE] [--force] NAME
ppmk --info NAME|IDENT
ppmk --check NAME|IDENT
ppmk --change NAME|IDENT
ppmk --recover NAME|IDENT
ppmk --delete NAME|IDENT
```

Creates and manages softcards. Passphrases are prompted for when required or, on secure networks, may be given with the `--oldpp` or `--newpp` options.

For more information, see:

- Creating a softcard with ppmk.
- Erasing softcards with ppmk.
- Viewing softcards with ppmk.
- Verifying the passphrase of a softcard with ppmk.
- Changing known softcard passphrases with ppmk.
- Replacing unknown passphrases with ppmk.

Option	Description
Modes of operation	
`-c`, `--check`	Checks the softcard's passphrase.
`-C`, `--change`	Changes the softcard's passphrase.
`-i`, `--info`	Shows the softcard details.
`-l`, `--list`	Lists the softcards.
`--delete`	Deletes the softcard.
`-n`, `--new`	Makes a new softcard.
`--newpp=PP`	New passphrase. Use this option only on secure networks.
`--oldpp=PP`	Existing passphrase. Use this option only on secure networks.
`-r`, `--recover`	Recovers a softcard's passphrase.
Options for --new	
`--force`	Forces the creation of a softcard with a duplicate name.

Option	Description
`-m`, `--module=MODULE`	Loads any necessary FIPS authorisation on `MODULE`. Default: the first usable module.
`--recoverable`	Makes the softcard passphrase recoverable.
`-R`, `--non-recoverable`	Makes the softcardpass phrase non-recoverable.
Help options	
`-h`, `--help`	Displays help for `fwcheck`.
`-u`, `--usage`	Displays a brief usage summary for `fwcheck`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `fwcheck`.

147. preload

```
preload [options] subprocess ...
preload [options] pause
preload [options] exit
```

Loads keys into a module before an application is run in another session.

With no options, does nothing.

By default, keys protected by explicitly requested (or interactively loaded) cardsets will be loaded and FIPS auth will be loaded on all modules that it is easily available on.

The `pause` argument makes `preload` pause after loading keys or cardsets. This is useful to load keys in one session and use them in another.

The `exit` argument causes `preload` to exit immediately after loading keys or cardsets. This is useful to add to an existing `preload` session.

By default, `preload` files are placed in directories private to the user that creates them. If `preload` files are to be shared between users then the `--preload-file` option must be used to specify an alternative location.

All file paths supplied as options to `preload` must be surrounded by quotations to avoid ambiguity.

Option	Description
Cardset selection options	
`-c IDENT, --cardset=IDENT`	Load all cardsets matching `IDENT`. If `IDENT` looks like a hash it will be interpreted as that, otherwise it will be interpreted as a name. If it's definitely a name, use `--cardset-name`.
`--cardset-name=NAME`	Loads cardset(s) named `NAME`.
`-i, --interactive`	Loads cardsets interactively until told to stop.
`-o, --any-one`	Loads a single cardset.
`-s IDENT, --softcard=IDENT`	Loads all softcards matching `IDENT`. If `IDENT` looks like a hash it will be interpreted as that, otherwise it will be interpreted as a name.
`--softcard-name=NAME`	Loads softcard(s) named `NAME`.
Key selection options	
`-A APP, --appname=APP`	Chooses the appname `APP` for subsequent `-K` options.

Option	Description
`--admin=KEYS`	Loads admin keys listed as a comma-separated list, or loads all admin keys if the value for `KEYS` is set to `all`.
`-K PATTERN, --key-ident=PATTERN`	Loads keys with ident matching `PATTERN` from most recently chosen appname.
`--list-admin`	Lists available admin key names for `--admin`.
`-M, --module-prot`	Loads all module protected keys, in addition to any others requested.
`-n PATTERN, --name-pattern=PATTERN`	Loads keys with the name matching `PATTERN`.
`--name-exact=NAME`	Loads keys with the name `NAME`.
`--no-cardset-keys`	Doesn't automatically load keys protected by requested tokens. This option has been superseded by the `--no-token-keys` option.
`--no-token-keys`	Doesn't automatically load keys protected by requested tokens.
FIPS options	
`-F, --require-fips`	Requires `FIPS-auth` to be loaded. Overrides a previous `-N`.
`-N, --no-fips`	Loads FIPS where required, but do not record `FIPS-auth`. Overrides a previous `-F`.
Loading options	
`-f PRELOAD_FILE, --preload-file=PRELOAD_FILE`	Uses specified preloaded objects file, instead of the default.
`-H, --high-availability`	High availability mode.
`--polling-interval=POLLING_INTERVAL`	Interval in seconds between polls for changes to the module list. Default: `60`. High availability mode only.
`-R, --reload-everything`	Reloads keys and tokens that are already loaded.
`--show-key-info`	Displays key information for keys as they are loaded.
Logging options	
`-l, --file-logging`	Enable log to file.
`--log-file=LOG_FILE`	The file destination for the log. Default: `preload_%pid.log` in the `nfast` log directory.
`--log-level=LOG_LEVEL`	The log level to log. One of DEBUG, INFO, WARNING, ERROR, CRITICAL. Default is INFO.
`-S, --no-stderr-logging`	Doesn't log to `stderr`. Independent of file logging.
Module selection	

Option	Description
`-m`, `--module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `preload` uses all modules by default.
Help options	
`-h`, `--help`	Displays help for `preload`.
`--version`	Displays the version number of the Security World Software that deploys `preload`.

147.1. Pattern matching in preload commands

Options to preload that use pattern matching can accept the following wildcards:

***** Everything

? A single character

[seq] Any character in seq

[!seq] Any character not in seq

Always surround arguments containing wildcards with quotations.

148. pubkey-find

```
pubkey-find [OPTIONS] - < CERT-OR-KEY-FILE
pubkey-find [OPTIONS] CERT-OR-KEY-FILE
```

Obtains information of the public key from a certificate or certificate request in a Base-64 encoded PEM file.

Option	Description
Options for the input format	
`--auto`	Guesses the input format (default).
`--cert`	Input is a (PEM X.509) certificate.
`--certreq`	Input is a (PEM X.509) certificate request
`--privkey`	Input is a private key in PEM format.
`--verbose`	Report more details if input cannot be parsed.
Options for the output	
You can use several output and processing options. They operate in the order they appear in the command.	
`--fingerprint`	Prints the certificate fingerprint, that is, the MD5 hash.
`--hash`	Prints the nCore keypair hash.
`--identify`	Prints the key's Security World appname(s) or ident(s).
`--summary`	Outputs summary information in human-readable format (default).
`--thumbprint`	Prints the certificate thumbprint, that is, the SHA-1 hash.
Options for further processing	
`--augment`	Augments the `kmdata` file with the public key value from the input file.
`--nfkmverify-options OPTIONS`	Passes `OPTIONS` to `nfkmverify`. `OPTIONS` must be a Tcl list. You can use several output and processing options. They operate in the order they appear in the Tcl list.
`--info`	Displays extensive general information about the key by running `nfkminfo -k`.
`--verify`	Verifies that the key was securely generated by running `nfkmverify`.
Options for controlling which `kmdata` key files are reported and processed	
Only keys that match the condition are processed	
`--all`	All key file(s) regardless of modification time (default).
`--earliest`	The least recently modified file.

Option	Description
`--latest`	Most recently modified file of those which otherwise match.
`--match-appname PATTERN, --match-ident PATTERN`	Whose `appname` or `ident` matches `PATTERN`. Default: no restriction
Option to address HSMs	
`--module N`	Specifies the number of the module to use.
Help options	
`-h`, `--help`	Displays help for `pubkey-find`.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `pubkey-find`.

149. push

```
push
push OFF
push ON ipv4address
push ON ipv6address
push ON [ipv6address]
push ON [ipv6address%eth0]
push ON keyhash
push ON ipv4address keyhash
push ON ipv6address keyhash
push ON [ipv6address] keyhash
push ON [ipv6address%eth0] keyhash
```

Gets or sets the configuration push setting. Corresponds to the `config_op` section of the network-attached HSM's configuration file.

Option	Description
`OFF`	If `OFF` is supplied on the command line, then any further arguments are ignored and the push configuration is set to `off`.
`ON`	If `ON` is supplied on the command line, then the push configuration is set to `on`. Either an IP address, a keyhash, or both an IP address and the corresponding keyhash may be supplied. If not already set, and not supplied, the IP address defaults to `0.0.0.0` (any address permitted).
`keyhash`	Default: 40 zeros.
`ipv4address` `ipv6address`	The IP address of the client allowed to push config files can be either an IPv4 address or an IPv6 address. IPv6 addresses may optionally be enclosed in square brackets. For link-local IPv6 addresses, the interface (`eth0` or `eth1`) is specified after the address with a percentage (`%`) character separating the address and the interface name. For example, `fe80::1%eth0`.

150. raccmd

```
raccmd [-h] [--address ADDRESS] [--port PORT] [-v] [--version]
              {listhsms,listreaders,associate} ...
```

nShield Remote Administration client command tool.

Option	Description
`--address ADDRESS`	Address of the Remote Administration Service. Default: `localhost`
`--port PORT`	Port of the Remote Administration Service. Default: `9005`
Positional arguments	
`associate`	Associates a local reader with an HSM identified by its ESN.
`listhsms`	Lists the HSMs provided by a Remote Administration Service.
`listreaders`	Lists readers attached to the local host.
Help options	
`-h`, `--help`	Displays help for `raccmd`.
`--version`	Displays the version number of nShield Remote Adminstration that deploys `raccmd`.
`-v`, `--verbose`	Increases the verbosity of the output.

151. racgui

```
racgui [-h] [-v] [-d] [-l LOGFILENAME]
```

nShield Remote Administration client GUI.

Option	Description
`-d`, `--debug`	Enables debug from the Remote Administration client GUI.
`-l LOGFILENAME`, `--logfile LOGFILENAME`	The full pathname of a file to log the debug to.
Help options	
`-h`, `--help`	Displays help for `racgui`.
`-v`, `--version`	Displays the version number of nShield Remote Adminstration that deploys `racgui`.

152. racs

```
racs [[-m] MODULE]
```

Creates a new administrator card set to replace an existing ACS. See Replacing an Administrator Card Set using racs.

Option	Description
-m, --module=MODULE	Uses module MODULE to create the new administrator card set. Default: the first available module.
Help options	
-h, --help	Displays help for racs.
-u, --usage	Displays a brief usage summary for racs.
-v, --version	Displays the version number of nShield Remote Adminstration that deploys racs.

153. randchk

```
randchk [L-min [L-max]]
```

Runs a universal statistical test on random numbers returned by the module.

Option	Description
`L-min`, `L-max`	By default, `randchk` runs one 6-bit test. If `L-min` and `L-max` are specified, they give the lowest and highest bit lengths to test. Both `L-min` and `L-max` must be in the range `1` to `16`.
Help options	
`-h`, `--help`	Displays help for `randchk`.
`-u`, `--usage`	Displays a brief usage summary for `randchk`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `randchk`.

154. reboot

```
reboot
```

Reboots the network-attached HSM.

155. retrievewarrants

```
retrievewarrants [-h] --module MODULE --klf2 KLF2WARPATH --klf3
                        KLF3WARPATH (--nchex | --bin) [--verbose]
```

Option	Description
`--bin`	Uses the `bin` format.
`--klf2 KLF2WARPATH`	Path for the retrieved KLF2 warrant file.
`--klf3 KLF3WARPATH`	Path for the retrieved KLF3 warrant file.
`--nchex`	Uses the `nchex` format.
`--verbose`	Prints verbose logs.
Option to address the HSM	
`-m`, `--module=MODULE`	Module to get the warrants from.
Help options	
`-h`, `--help`	Displays help for `retrievewarrants`.

Retrieves warrants stored within the nShield 5s HSM and writes them to a file.

156. rfs-setup

```
rfs-setup <ADDRESS> <ESN> <KEYHASH>
rfs-setup --gang-client <ADDRESS> [<ESN>] <KEYHASH>
rfs-setup --gang-client --write-noauth <ADDRESS>
rfs-setup --gang-client --readonly <ADDRESS> [<ESN>] <KEYHASH>
rfs-setup --gang-client --readonly --write-noauth <ADDRESS>
```

Creates a default RFS hardserver configuration.

Run this utility when you first configure the RFS.

`rfs-setup` creates all appropriate directories for the remote file system and edits the hardserver configuration file appropriately.



To revoke a networked-attached HSM or a ganged client, you must edit the hardserver configuration file manually.

For procedures, see:

- Configuring the remote file system (RFS).
- Client cooperation.

Option	Description
Action selection	
`-g`, `--gang-client`	Sets up a client machine to share the RFS. In this case `<ADDRESS>` is the IP address of the client.
`--readonly`	Limits the ganged client to read-only.
`--write-noauth`	Allows the ganged client to access the RFS without authentication. Do not use this option over insecure networks.
Options for the actions	
`<ADDRESS>`	
`-c`, `--configfile=FILENAME`	Default: `NFAST_KMDATA/config/config`.
`<ESN> <KEYHASH>`	If an option of `rfs-setup` allows a network-attached HSM to write to the RFS, which requires authentication from an HSM. The client can be authenticated by passing in its HSM's ESN and KNETI hash, or for software authentication by specifying its hardserver's KNETI hash only.
`-f`, `--force`	Removes old existing `remotefilesystem` config entries with the same ESN value.
Help options	

Option	Description
`-h`, `--help`	Displays help for `racs`.
`-u`, `--usage`	Displays a brief usage summary for `racs`.
`-v`, `--version`	Displays the version number of nShield Remote Adminstration that deploys `racs`.

157. rfs-sync

```
rfs-sync [-U|--update] [-c|--commit] [-s|--show] [--remove]
        [--setup [setup_options] ip_address]
```

Synchronizes your local key management data (`opt/nfast/kmdata/local` on **Linux**, or `%NFAST_KMDATA%\local` on **Windows**) with the remote file system it is configured to access by updating from it or committing changes to it. If you need to push changes to the remote file system with `rfs-sync`, the cooperating client from which you are pushing changes from must have write access to the remote file system.

Run `rfs-sync` to retrieve data from the remote file system when

- A cooperating client is initialized
- A client needs to update its local copy of the data

Option	Description
Action selection	
`-c`, `--commit`	Commits local key management data changes to the remote file system, and updates the client from the remote file system.
`--remove`	Removes the synchronization configuration. Reverting to a standalone configuration leaves the current contents of the Key Management Data directory in place.
`-s`, `--show`	Displays the current synchronization configuration.
`--setup`	Sets up a new synchronization configuration.
`-U`, `--update`	Updates local key management data from the remote file system. If a cooperating client has keys in its `kmdata/local` directory that are also on the remote file system, if these keys are deleted from the remote file system and then `rfs-sync --update` is run on the client, these keys remain on the client until manually removed.
Options for --setup	
`-a`, `--authenticate`	Specifies the use of a module KNETI key to authenticate this client to the RFS. Default: software KNETI key of the hardserver
`ip_address`	Specifies the IP address of the remote file system, which could be one of the following: • an IPv4 address • an IPv6 address, including a link-local IPv6 address • a hostname

Option	Description
`-m`, `--module=module`	Selects the local module to use for authentication. Default: `1`. This option can only be used with the `--authenticate` option.
`-p`, `--port=port`	Specifies the port on which to connect to the remote file system. Default: `9004`.
`--rfs-hkneti=HNETI`	Specifies the hash of the K_{NETI} key to use for nToken or software-based authentication of the RFS.
`--rfs-esn=ESN`	Specifiesthe `ESN` of an nToken to use for authentication of the RFS.
Options for a stuck lockfile that has been left behind by a failed `rfs-sync --commit` operation	
`--who-has-lock`	Displays the task ID of the lock owner.
`--kill-lock`	Forcibly removes the lock file. Only use this option as a last resort. For network-attached HSMs, the lock file can also be removed via menu item 3-3-2, **Remove RFS Lock**: this executes the `rfs-sync --kill-lock` command.
Help options	
`-f`, `--force`	Disable confirmation prompts for the `--setup` and `--remove` actions when over writing an existing configuration and with `--kill-lock` when removing a lock.
`-h`, `--help`	Displays help for `rfs-sync`.
`-q`, `--quiet`	Displays fewer messages.
`-u`, `--usage`	Displays a brief usage summary for `rfs-sync`.
`-v`, `--verbose`	Displays more messages.
`-V`, `--version`	Displays the version number of the Security World Software that deploys `rfs-sync`.

For more information, see:

- Client cooperation.

158. rfsaddr

```
rfsaddr
rfsaddr address[:port] [keyhash [esn]] [push]
```

Gets or sets the RFS IP address, port, and authentication settings.

Option	Description
address	The IP address of the RFS can be either an IPv4 address or an IPv6 address. IPv6 addresses may optionally be enclosed in square brackets. For link-local IPv6 addresses, the interface (eth0 or eth1) is specified after the address with a percentage (%) character separating the address and the interface name. For example, fe80::1%eth0.
esn	(optional) The ESN of the module to authenticate on the RFS end. Default: "" (no ESN authentication, the RFS software key is to be used). If you use the esn parameter, you must also specify the keyhash with a non-default value.
keyhash	(optional) A hash of the RFS KNETI key may be provided to authenticate the RFS when connecting to it. Default: 40 zeroes (no RFS authentication will take place).
port	(optional) The port of the RFS. IPv4: ipaddress:port. IPv6: enclose the IP address in square brackets to disambiguate from the port number, for example [fe80::1%eth0]:9004 Default: 9004.
push	(optional) The RFS can be set up as a client that can push configuration to the network-attached HSM using the push parameter. It takes one of the following values: ON, OFF or AUTO. Default: AUTO (the RFS can push if and only if a keyhash has been specified).

159. rocs

```
rocs -m|--module=<MODULE> [-t|--target=<CARDSET-SPEC>] [-k|--keys=<KEYS-SPEC>] [-c|--cardset=<CARDSET-SPEC>] [-i|--interactive]
```

- Restores an OCS from a quorum of its cards
- Restores softcards



Keys protected by an OCS can only be recovered to another OCS, and not to a softcard. Likewise, softcard-protected keys can only be recovered to another softcard, and not to an OCS.

If you run `rocs` without any parameters, it enters interactive mode, where it displays a `rocs` prompt. In interactive mode, it reads and executes commands from `stdin`:

rocs in interactive mode

```
'rocs' key recovery tool
Useful commands: 'help', 'help intro', 'quit'.
rocs >
```

For more information, see:

- Replacing OCSs or softcards with rocs.
- Using rocs from the command line.

Solo XC	nShield 5s	Connect +	Connect XC	nShield 5c	Edge	Remote Admin
n	y	n	n	n	n	n

Option	Description
`-c`, `--cardset=CARDSET-SPEC`	Specifies all keys protected by a cardset. You can use this option multiple times to specify multiple cardsets. The value of `CARDSET-SPEC` can have any of the following forms: • `[number]` *cardset-number*: A value of this form selects the OCS or softcard with the given number from the list produced by the `list cardsets` command. • `[name]` *cardset-name*: A value of this form selects card sets or softcards by their names (the card set or softcard name may be a wildcard pattern in order to select all matching OCSs or softcards). • `hash` *cardset-hash*: A value of this form selects the OCS or softcard with the given hash.

Option	Description
`-i`, `--interactive`	Reads commands interactively, even though keys are specified on the command-line.
`-k`, `--keys=KEY-SPEC`	Specifies the keys to recover (to create new passphrase for). The value of `KEYS-SPEC` can have one of the following forms: * *mark* `key-number`: A value of this form selects the key with the given number from the list produced by the list keys command. + Examples of usage are: [source] ---- rocs -t <target_OCS> -k <key_number> ---- + [source] ---- rocs -t <target_OCS> -k "mark 56" ---- * *appname_*:`keyident` : A value of this form selects keys by their internal application name and `ident`. You must supply at least one of `appname` or `keyident`, but you can use wildcard patterns for either or both in order to select all matching keys. An example of usage is: + [source] ---- rocs -t <target_OCS> --keys="simple:simplekey" ---- * `hash` *keyhash*: A value of this form selects the key with the given key hash. An example of usage is: + [source] ---- rocs -t <target_OCS> --keys="hash e364[...]" ----
`--cardset` *cardset-spec*	A value of this form selects all keys protected by a given card set.
`-t`, `--target=CARDSET-SPEC`	Specifies the cardset to recover (to create new passphrases for). You can use this option multiple times to specify multiple cardsets. See `-c`, `--cardset=CARDSET-SPEC` for the available forms for the `CARDSET-SPEC` value.
Option to address the HSM	
`-m`, `--module=MODULE`	Module to use for recovery (creating new passphrases).
Help options	
`-h`, `--help`	Displays help for `rocs`.
`-u`, `--usage`	Displays a brief usage summary for `rocs`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `rocs`.

159.1. rocs interactive mode commands

At the `rocs` prompt, you can use the following commands.



You can specify a command by typing enough characters to identify the command uniquely. For example, for the `status` command, you can type `st` and then press **Enter**.

Command	Description
`help`	Displays a list of available commands with brief usage messages and a list of other help topics. With an argument, `help` shows detailed help information about a given topic.
`help intro`	Displays a brief step-by-step guide to using `rocs`.
`list cardsets`	Lists the OCSs and softcards in the current Security World. For example: `No. Name Keys (recov) Sharing 1 test 6 (6) 3 of 5; 20 minute timeout 2 test2 3 (2) 2 of 3 3 test3 1 (1) 1 of 1; persistent` In this output: • `No.`: The card set or softcard number, which you can use to identify this card set in `rocs` commands. • `Name`: The OCS or softcard name. • `Keys`: The number of keys protected by this OCS or softcard. • `(recov)`: The number of keys protected by this OCS or softcard. • `Sharing`: The *K* of *N* parameters for this OCS. • `persistent`: The OCS is persistent and does not have a time-out set. • `### minute timeout`: The OCS is persistent and has a time-out set.

Command	Description

`list keys`	Lists the keys in the current Security World, as in the following example:
	
	`No.`
	`Name App Protected by`
	` 1 Id: uc63e0ca3cb032d71c1c pkcs11 test2`
	`R 2 Server-Cert pkcs11 test --> test2`
	` 3 Id: uc63e0ca3cb032d71c1c pkcs11 test --> test3`
	` 4 Server-Cert pkcs11 module (test ---> fred2)`
	
	In this output:
	
	• `No.`: The key number, which you can use in `mark` and `unmark` commands.
	• `Name`: The key name.
	• `App`: The application with which the key is associated.
	• `Protected by`: This indicates the protection method.
	
	Protection methods:
	
	• `module`: Key protected by the Security World.
	• `<name>`, for example `test2`: Key protected by the named OCS or softcard.
	• `<name> --> <name2>`, for example `test -→ test2`: Key protected by the OCS or softcard *name1* marked for recovery to OCS or softcard *name2*.
	• `module (<name>)`: PKCS #11 public object.
	
	These are protected by the Security World but associated with a specific OCS or softcard.
	
	• `module (<name> --> <name2>)`, for example `module (test --→ fred2)`: PKCS #11 public object marked for recovery.

`mark <key-spec>`	Marks the listed keys that are to be recovered to the target OCS or softcard. You can mark one or more keys by number, *ident*, OCS or softcard, or hash. To mark more than one key at a time, ensure that each *key-spec* is separated from the other by spaces, for example: [source] ---- mark key-spec1 key-spec2 key-spec3 ---- If you have not selected a target OCS or softcard, or if `rocs` cannot parse the *key-spec*, then `rocs` displays an error message. You can mark and remark the keys to be recovered to various target OCSs or softcards. Remarking a key displaces the first target in favor of the second target. [NOTE] Keys protected by an OCS can only be recovered to another OCS, and not to a softcard. Likewise, softcard-protected keys can only be recovered to another softcard, and not to an OCS.

Command	Description
`module <number>`	Selects the hardware security module to be used. The module `<number>` must correspond to a hardware security module in the current Security World. If the hardware security module does not exist, is not in the Security World, or is oth erwise unusable, then `rocs` displays an error message and does not change to the selected module.
`quit`	Allows you to leave `rocs`. If you attempt to `quit` when you have recovered keys but have not saved them, `rocs` displays a warning.
`recover`	Transfers the marked keys to their target OCSs or softcards. This operation is not permanent until you save these keys by using the `save` command.
`rescan`	Updates the card set and key information.
`revert <key-spec>`	Returns keys that have been recovered, but not saved, to being protected by the original protection method. If the selected keys have not been recovered, `rocs` displays an error message.
`save [<key-spec>]`	Writes the new key blobs to disk. If you specify `<key-spec>` values, only those keys are saved. Otherwise, all recovered keys are saved.
`status`	Lists the currently selected hardware security module and target OCS or soft-card.
`target <cardset-spec>`	Sselects a given OCS or softcard (`<cardset-spec>`) as the target. You can spec ify the card set or softcard name, the number returned by `list cardsets`, or the hash.
`unmark <key-spec>`	Uunmarks the listed keys. Unmarked keys are not recovered.

160. route

Gets or sets the IPv4 network routes.

If the gateway is unreachable, the routing entry will be added to the configuration file, but it will not be added in the kernel routing table.

Display all existing route entries:

```
route
```

Add a new route entry:

```
route addr=0.0.0.0 masklen=0 gateway=0.0.0.0
```

Change an existing route entry:

```
route entry=0 [addr=0.0.0.0] [masklen=0] [gateway=0.0.0.0]
```

Delete a route entry:

```
route entry=0 delete
```

Option	Description
addr	Routable IPv4 address block.
entry	Index of the route entry to be displayed or modified.
gateway	Route gateway's IPv4 address.
masklen	Number of leading 1 bits in the network mask.

161. route6

Gets or sets the IPv6 network routes.

If the gateway is unreachable, the routing entry will be added to the configuration file, but it will not be added in the kernel routing table.

Display all existing route entries:

```
route6
```

Add a new route entry:

```
route6 addr=:: masklen=0 gateway=:: linklocal_if=0
```

Change an existing route entry:

```
route6 entry=0 [addr=::] [masklen=0] [gateway=::] [linklocal_if=0]
```

Delete a route entry:

```
route6 entry=0 delete
```

Option	Description
addr	Routable IPv6 address block.
entry	Index of the route entry to be displayed or modified.
gateway	Route gateway's IPv6 address.
masklen	Number of leading 1 bits in the network mask.
linklocal_if	The ethernet interface (0 or 1) to use if the IPv6 route gateway address is a link-local address. The information is not used if the IPv6 route gateway is not a link-local address. Default: 0.

162. routing

```
routing
```

Shows the IPv4 routing table.

163. routing6

```
routing6
```

Shows the IPv6 routing table.

164. rserverperm

```
rserverperm --add [options] --exportslot
rserverperm --list --exportslot|--all
rserverperm --remove -p ID
```

Adds, lists, or removes remote module permissions to the local hardserver configuration.

The default is `rserverperm --add --exportslot`, which, without other options, exports the module 1 slot 0 to any remote host and any remote module.

Option	Description
Action selection	
`--add`	Adds a new permission.
`--list`	Lists the permissions.
`--remove`	Removes a permission.
Permitted remote server operations	
`--accessfiles`	Allows remote access to files.
`--all`	Show all permissions (only with `--list`).
`--exportmodule`	Allows remote use of a module.
`--exportslot`	Allows remote reading of a slot.
Add options:	
`-a`, `--address=ADDRESS`	Sets the IP address of the remote server. Default: any.
`-A`, `--Allowed=ALLOWED`	When are privileged commands allowed.
`-f`, `--force`	Tries to add the module even if it doesn't claim support.
`-l`, `--local=ESN`	From the local/remote pair, sets the local module ESN. Default: module 1.
`-r`, `--remote=ESN`	From the local/remote pair, sets the remote module ESN. Default: any.
`-s`, `--slot=SLOT`	Sets the slot to export to. Default: slot 0.
Add accessfiles options	
`-D`, `--dir`	The volume is a directory.
`-L`, `--allow-list`	Allows directory listings.

Option	Description
`-N`, `--nativepath=FILENAME`	Nativepath for the host volume.
`-R`, `--allow-read`	Allows read access to the file or files.
`-T`, `--text`	The volume is or contains text files.
`-V`, `--volume=STRING`	Name of the host volume.
`-W`, `--allow-write`	Allows write access to the file or files
Remove option	
`-p`, `--permission-id=ID`	Sets the permission ID (given by `rserverperm --list`).
Help options	
`-h`, `--help`	Displays help for `rserverperm`.
`-u`, `--usage`	Displays a brief usage summary for `rserverperm`.
`-v`, `--version`	Displays the version number of the Security World Software that deploys `rserverperm`.

165. rtc

```
rtc --get-clock [-m MODULE]
rtc --set-clock [-aA] [-m MODULE] [TIME]
```

Views and sets the module's real-time clock.

Option	Description
Action selection	
`-g, --get-clock`	Gets (reads) the module's clock time (default).
`-t, --set-clock`	Sets (writes) the module's clock. Setting the module's clock usually requires the insertion of administrator cards. To try anyway, without any admin cards, use `--no-admin-keys'. The module's clock is set to one of: • TIME, if it is provided as a list of six integers (in the order of `yyyy mm dd hh mm ss`), separated by non-digit characters • the host's current time
Clock setting options	
`a, --no-admin-keys`	Doesn't read admin cards.
`-A, --adjust`	Calibrates clock drift. The module uses the difference between its idea of the current time and the new time, together with how long it's been since the clock was last set, to com pute how much its clock is drifting. Assuming that the host has an accurate clock, for example, it runs an NTP client, you can calibrate the drift by running `rtc --set-clock', and then, about 24 hours later, `rtc --set-clock --adjust'.
Option to address HSMs	
`-m, --module=<MODULE>`	Read or write the clock of module MODULE (default = module 1).
Help options	
`-h, --help`	Displays help for `rtc`.
`-u, --usage`	Displays a brief usage summary for `rtc`.
`-v, --version`	Displays the version number of the Security World Software that deploys `rtc`.

166. see-sock-serv, see-stdioe-serv, see-stdioesock-serv, see-stdoe-serv

```
see-sock-serv -p <PUBL-NAME> | -o <KEYID> | -M <MACHINE>.sar
see-stdoe-serv -p <PUBL-NAME> | -o <KEYID> | -M <MACHINE>.sar
see-stdioe-serv -p <PUBL-NAME> | -o <KEYID> | -M <MACHINE>.sar
see-stdioesock-serv -p <PUBL-NAME> | -o <KEYID> | -M <MACHINE>.sar
```

`see-*-serv` utilities activate or enable standard IO and socket connections for SEE machines using the `glibc` architecture. Ensure that you select the appropriate utility for your SEE machine, because running a host-side utility with more provisions than the SEE machine was linked against causes the SEE machine to abort.

- `see-sock-serv`, for SEE machines that require only sockets.
- `see-stdoe-serv`, for SEE machines that require only standard output and error streams.
- `see-stdioe-serv`, for SEE machines that require standard input, output, and error streams.
  If you are using a nShield Connect, you must set the `--no-feature-check` option when running the `see-stdoe-serv` utility.
- `see-stdioesock-serv`, for SEE machines that require sockets in addition to standard input, output, and error streams.

Each utility can:

- Load the SAR file for the SEE machine
- Load the mandatory `userdata` file
- Provide a selection of socket and I/O streams

SEE machines that require the standard I/O streams or INET domain sockets must be serviced by one of the described host-side utilities. Without an appropriate host-side utility, SEE machine operations requiring any of these streams are blocked until the appropriate service becomes available.

All the `see-*-serv` host-side utilities take the same arguments.

Option	Description
Loading the SEE machine	
`-e, --encryptionkey=IDENT`	The SEE machine is encrypted with key IDENT.
`-s, --sighash=HASH`	The SEE machine is signed with key whose hash is HASH. Use this option together with the `-e` option and only if you have the dynamic SEE feature.

Option	Description
`-M, --machine=<MACHINE>.sar`	Specifies a SEE machine file (packed as a SAR). If you do not specify this option, the SEE machine must have been loaded previously by, for example, running loadmache.
Starting the SEE world	
`--userdata-raw <USER-DATA.bin>`	An unpacked `userdata` file to be passed to SEE machine. The raw file is internally made into an unsigned SAR file.
`--userdata-sar <USER-DATA>.sar`	The `userdata` file (packed as a SAR) to be passed to SEE machine.
`-V, --userdata-vuln`	Starts the SEE world, passing remaining arguments, which should include an `argv[0]` for the world in `userdata` to `vulnerability.o`.
Pre-started SEE world	
`-o, --object-id=<NAME>`	The `KeyID` of the started SEE machine. By default, a decimal value is expected. Use `0x` notation for hexadecimal values.
`-p, --published-object=<NAME>`	The `PublishedObject` name to use for publishing the `KeyID` of the started SEE machine.
Tracing	
`--trace`	Polls the security world's trace buffer. The contents are printed to `stderr` in dark red. If the configuration of the Security World requires it, you must supply authorization to poll the trace buffer when specifying this option. The `see-*-serv` host-utility prompts you to supply authorization if it is required.
`--plain-trace`	Functions like the `--trace` option to poll the security world's trace buffer, but the output from `--plain-trace` is not surrounded by terminal escape codes.
HSM options	
`-f, --no-feature-check`	Suppresses the default behavior of the `see-*-serv` host-side utilities to ensure that the HSM specified by the `-m, --module=<MODULE>` option has the `HasSEE` flag and the `GeneralSEE` feature before the utility tries to load an SEE machine. If you are using a network-attached HSM (an nShield Connect), you must set the `--no-feature-check` option when running the `see-stdoe-serv` utility.
`--job-prefix <PREFIX>`	This option is for debugging. For the host-side utilities that provide a single service (that is, `see-sock-serv`, `see-stdoe-serv`, and `see-stdioe-serv`), specifying this option forces the service to use the job prefix specified by `<PREFIX>`.
`-m, --module=<MODULE>`	The HSM onto which the SEE machine is to be loaded. Use enquiry to get information about the HSM.

Option	Description
`-r`, `--restrict`	Only permits userdata and machine-image files from the `nc-seemachines` or the `custom-seemachines` subdirectories of the `/opt/nfast` (**Linux**) or `%NFAST_HOME%` (**Windows**) directory to be loaded. When userdata is loaded auto matically by a privileged account, this option should be specified, for extra security.
Help options	
`-h`, `--help`	Displays help for the utility.
`-u`, `--usage`	Displays a brief usage summary for the utility.
`-v`, `--version`	Displays the version number of the Security World Software that deploys the utility.

166.1. Error output from SEE machine with SEElib architecture

You cannot use the `see-*-serv` host-side utilities to load SEE machines built with the `SEElib` architecture. If you try to do so, the utility returns a message similar to

```
FATAL: SeeHostCallProvision_Init (prefix `nC/HC/sock/INET ') failed:
SeeHostcallProvisionFailed
```

This is the expected behavior caused by the host utility sending `SEEJob`s that the SEE machine cannot understand or to which it cannot respond correctly.

You can use the `loadmache` command-line utility to manually load SEE machines built with the `SEElib` architecture.

167. setrtc

```
setrtc date=YYYY-mm-dd time=HH:MM:SS
```

Sets the real-time clock of the network-attached HSM. The HSM must be in Maintenance mode when you run `setrtc`.

Only supported in Security World Software v13.3 or later.

Option	Description
`date`	The date to set in the format `YYYY-mm-dd`.
`time`	The time to set in the format `HH:MM:SS`.

168. sigtest



Only supported in FIPS 140-2 Level 2 Security Worlds.

```
sigtest [options]
```

Measures module speed using RSA or DSA signatures or signature verifications. If skew or threshold checking is enabled (they are mutually exclusive), the average number of operations per second is recorded at TIME.

If skew checking is enabled, each subsequent operation must be within SKEW of the recorded average. If the condition is not met, the application terminates

If threshold checking is enabled, the average must stay above COUNT after checking starts. If the condition is not met, the application terminates.

Option	Description
Program options	
`-d, --decrypt`	Tests the decrypt operation.
`-F, --no-failover`	Doesn't failover if the loaded key becomes unusable.
`-G, --logging`	Attempts audit logging. For this to succeed, all specified modules must report audit logging as active.
`-j, --outstanding-jobs=COUNT`	Sets the maximum number of outstanding jobs. Default: minimum number of hardservers recommended + 1.
`-L, --longjobs`	Sets the `LongJobs` flag in crypto commands.
`-n, --jobs-count=COUNT`	Sets the maximum number of jobs. Default: infinite.
`-s, --sign`	Tests the sign operation (default).
`-t, --stop-after=LENGTH`	Sets the maximum time to run, in seconds. Default: infinite.
`-v, --verify`	Tests the verify operation.
`-x, --keyx`	Tests the key exchange operation.
Key options	
`-c, --curve=CURVENAME`	Uses the curve named NAME. Default: `NISTP192`.
`-l, --key-size=BITS`	Sets the key size (default 1024).

Option	Description
`-M, --mechanism=MECH`	Uses mechanism MECH.
`-p, --plain-type=TYPE`	Uses plaintext type TYPE (Bignum, Hash or Bytes). The mechanism and plaintext types must be compatible with the key type.
`--pairwise-check`	Sets `PairwiseCheck` in the key generation command.
`-S, --key-type=TYPE`	Selects the key type to use — RSA (default), DSA, KCDSA, or ECDSA
`--strong`	For RSA, uses strong (ANSI X9.31) primes. For DSA, uses the `Strict` flag.
Automatic checking options	
`-C, --check-start=TIME`	Specifies when skew or threshold checking commences, in seconds, rounded up to nearest multiple of INTERVAL. Default: `15`.
`-K, --skew-check=SKEW`	Turns on skew checking.
`-T, --min-check=COUNT`	Turns on threshold checking.
Output options	
`--overprint`	Prints the results all on one line, using \r rather than \n.
`-o, --output=FILE`	Sends the output to a named file as well as to `stdout`.
`-r, --report-interval=INTERVAL`	Sets the statistics reporting interval in seconds. Default: `1`.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `sigtest` uses all modules by default.
Help options	
`-h, --help`	Displays help for `sigtest`.
`-u, --usage`	Displays a brief usage summary for `sigtest`.
`-V, --version`	Displays the version number of the Security World Software that deploys `sigtest`.

169. slotinfo

```
slotinfo -m MODULE [-s SLOT]
slotinfo --format [--ignoreauth] -m MODULE -s SLOT
```

- Obtains information about tokens in a module
- Formats a smart card

Option	Description
`--format`	Formats the token in the slot. **!** Irreversibly destroys all data that was stored on the card.
`--ignoreauth`	Ignore any unrecognised token authentication key when formatting a token.
`-s, --slot=SLOT`	Reads slot SLOT.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `slotinfo` uses all modules by default.
Help options	
`-h, --help`	Displays help for `slotinfo`.
`-u, --usage`	Displays a brief usage summary for `slotinfo`.
`-v, --version`	Displays the version number of the Security World Software that deploys `slotinfo`.

169.1. slotinfo output

Example:

```
slotinfo -m 1

Slot Type           Token   IC    Flags   Details
#0   Smartcard      present 3     A
#1   Software Tkn   -       0
#2   Smartcard      -       0     AR
```

If only a module is specified, the following columns are displayed:

Column	Description
Slot	Slot number
Type	Type of this slot
Token	`present` if a token is present, otherwise `-`
IC	Insertion counter. 0 when no token is present and nonzero when one is present. Differs from its value at any previous insertion.
Flags	**A** Slot supports token-level challenge-response authentication **R** Slot is attached to a remote module **D** Slot is a dynamic slot for use with a remote card reader **a** dynamic slot is associated with a remote card reader **t** dynamic slot connection to remote card timed out **f** dynamic slot secure channel connection failed
Details	Any error string relevant to this slot

If both a module and a slot are specified in the command, information about the card and any shares and files stored on it are displayed.

170. stattree

```
stattree [<node> [<node> [[...]]]]
```

The output of `stattree` is statistics currently available on the host machine. Statistics are gathered both by the hardserver (relating to the server itself, and its current clients) and by each attached HSM.

A typical (abbreviated) output fragment from `stattree`. See the examples for more complete outputs.

```
$ stattree
+ServerGlobals:
   -Uptime               613631
   -CmdCount             153343
   [...]
+Connections:
   +#1:
      -Uptime               613631
      -CmdCount             0
      [...]
   +#25:
      -Uptime               0
      -CmdCount             13
      [...]
+PerModule:
   +#1:
      +ModuleObjStats:
         -ObjectCount          5
         [...]
      +ModuleCacheStats:
         -CacheEntryCount      0
         [...]
      +ModuleEnvStats:
         -SerialNumber         8ED1-2C9A-9331
         [...]
      +ModuleJobStats:
         -CmdCount             153352
         [...]
+RemoteServers:
+PerDevice:
   +#1:
      +ModuleDriverStats:
         -DriverIRQs           0
         [...]
      +ModuleServerStats:
         -JobsOutstanding      0
         [...]
```

170.1. Example outputs

▼ *+ServerGlobals*

```
+ServerGlobals:
   -Uptime              17577
```

```
   -CmdCount             1240357
   -CmdBytes             77927908
   -CmdMarshalErrors     0
   -ReplyCount           1240512
   -ReplyBytes           2036926904
   -ReplyMarshalErrors   0
   -ClientCount          9
   -MaxClients           17
   -DeviceFails          0
   -DeviceRestarts       0
   -CryptoClientCount    0
   -MaxCryptoClients     0
   -AuditDBFreeSpaceMB   486429
   -AuditDBUsedSpaceMB   0
```

▼ *+Connections*

```
+ServerGlobals:
   -Uptime               17577
   -CmdCount             1240357
   -CmdBytes             77927908
   -CmdMarshalErrors     0
   -ReplyCount           1240512
   -ReplyBytes           2036926904
   -ReplyMarshalErrors   0
   -ClientCount          9
   -MaxClients           17
   -DeviceFails          0
   -DeviceRestarts       0
   -CryptoClientCount    0
   -MaxCryptoClients     0
   -AuditDBFreeSpaceMB   486429
   -AuditDBUsedSpaceMB   0
```

▼ *+PerModule*

```
+PerModule:
   +#1:
      +ModuleObjStats:
         -ObjectCount          7
         -ObjectsCreated       51
         -ObjectsDestroyed     44
      +ModuleCacheStats:
         -CacheEntryCount      0
         -CacheEntriesInserted 0
         -CacheEntriesRemoved  0
      +ModuleEnvStats:
         -SerialNumber         AC03-03E0-D947
         -Uptime               11262
         -CurrentTime          1437708806
         -MemTotal             2030891008
         -MemAllocKernel       470183936
         -MemAllocUser         0
         -TempSP               39.00
         -CurrentCPUTemp1      53.00
         -CurrentCPUTemp2      45.00
         -CPUVoltage1          1.00
         -CPUVoltage2          1.80
         -CPUVoltage3          0.99
         -CPUVoltage4          1.35
         -CPUVoltage5          1.00
         -CPUVoltage6          1.51
         -CPUVoltage7          3.35
```

```
      -CPUVoltage8          2.53
      -CPUVoltage9          1.19
      -CPUVoltage10         2.92
      -CPUVoltage11         11.84
      -MaxTempC             53.00
      -MinTempC             38.00
      -AIS31PrelimAlarms    0
      -MceCount             0
      -SpiRetries           0
      -SpI2c1               0
      -SpI2c2               9
      -SpTempExcursion      0
      -SpVoltageExcursion   0
      -HostBusExceptions    0
      -CryptoBusExceptions  0
      -SpSensorCmdFails     0
      -NVMFreeSpace         272822272
      -NVMWearLevel         0.00
      -NVMWornBlocks        0.00
      -CurrentFanSpeed      4615
      -CurrentFanDuty       25
   +XCSecurityProcessorLog:
      +#18:
         -XCSpLogEventID       19
         -XCSpLogEventDate     1431650807
      +#17:
         -XCSpLogEventID       50
         -XCSpLogEventDate     4294967295
      +#16:
         -XCSpLogEventID       11
         -XCSpLogEventDate     1431650905
      +#15:
         -XCSpLogEventID       4
         -XCSpLogEventDate     1431652698
      +#14:
         -XCSpLogEventID       5
         -XCSpLogEventDate     1431652698
      +#13:
         -XCSpLogEventID       17
         -XCSpLogEventDate     1431652764
      +#12:
         -XCSpLogEventID       17
         -XCSpLogEventDate     1431661948
      +#11:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1434078570
      +#10:
         -XCSpLogEventID       38
         -XCSpLogEventDate     1434078570
      +#9:
         -XCSpLogEventID       39
         -XCSpLogEventDate     1434078570
      +#8:
         -XCSpLogEventID       11
         -XCSpLogEventDate     1434078710
      +#7:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1434158867
      +#6:
         -XCSpLogEventID       38
         -XCSpLogEventDate     1434158867
      +#5:
         -XCSpLogEventID       39
         -XCSpLogEventDate     1434158867
      +#4:
         -XCSpLogEventID       11
         -XCSpLogEventDate     1434158878
```

```
      +#3:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1436951986
      +#2:
         -XCSpLogEventID       38
         -XCSpLogEventDate     1436951986
      +#1:
         -XCSpLogEventID       39
         -XCSpLogEventDate     1436951986
      +#0:
         -XCSpLogEventID       11
         -XCSpLogEventDate     1436951993
   +ModuleJobStats:
      -CmdCount             8086
      -ReplyCount           8084
      -CmdBytes             175400
      -ReplyBytes           391520
      -JobsStarted          8086
      -JobsComplete         8085
      -RepliesQueued        8089
      -HostWriteCount       7693
      -HostWriteErrors      0
      -HostReadCount        15651
      -HostReadErrors       0
      -HostReadEmpty        0
      -HostReadDeferred     7798
      -HostReadTerminated   0
      -PFNIssued            4425
      -PFNRejected          0
      -PFNCompleted         4424
      -ANIssued             5
      -CPULoadPercent       4
   +ModulePCIStats:
      -HostIRQs             8876778
      -HostReadCount        3029341
      -HostReadDeferred     0
      -HostReadErrors       66
      -HostReadReconnect    0
      -HostWriteCount       5847385
      -HostWriteErrors      0
+#2:
   +ModuleObjStats:
      -ObjectCount          7
      -ObjectsCreated       83
      -ObjectsDestroyed     76
   +ModuleCacheStats:
      -CacheEntryCount      0
      -CacheEntriesInserted 0
      -CacheEntriesRemoved  0
   +ModuleEnvStats:
      -SerialNumber         E2D5-E4DD-7C59
      -Uptime               17570
      -MemTotal             62169088
      -MemAllocKernel       577536
      -MemAllocUser         0
      -CurrentTempC         58.50
      -MaxTempC             58.50
      -MinTempC             57.00
   +ModuleJobStats:
      -CmdCount             12676
      -ReplyCount           12674
      -CmdBytes             273284
      -ReplyBytes           533932
      -JobsStarted          12676
      -JobsComplete         12675
      -RepliesQueued        12802
      -HostWriteCount       12044
```

```
         -HostWriteErrors      0
         -HostReadCount        25369
         -HostReadErrors       0
         -HostReadEmpty        0
         -HostReadDeferred     12599
         -HostReadTerminated   0
         -PFNIssued            7010
         -PFNRejected          0
         -PFNCompleted         7009
         -ANIssued             128
         -CPULoadPercent       0
      +ModulePCIStats:
         -HostIRQs             24819
         -HostReadCount        12773
         -HostReadDeferred     12601
         -HostReadReconnect    12600
         -HostReadErrors       0
         -HostReadPushedDMA    80
         -HostReadPushedPIO    12690
         -HostWriteCount       12046
         -HostWriteErrors      0
         -HostDebugIRQs        0
         -HostUnhandledIRQs    0
         -HostKernReadCount    0
         -HostKernReadDeferred 0
         -HostKernReadReconnect 0
         -HostKernReadErrors   0
         -HostKernReadPushedDMA 0
         -HostKernReadPushedPIO 0
         -HostKernWriteCount   0
         -HostKernWriteErrors  0
```

▼ *+RemoteServers*

```
+RemoteServers:
   +#3:
      +ServerGlobals:
         -Uptime               557411
         -CmdCount             13499000
         -CmdBytes             1516570032
         -CmdMarshalErrors     0
         -ReplyCount           13499019
         -ReplyBytes           2836736908
         -ReplyMarshalErrors   0
         -ClientCount          12
         -MaxClients           34
         -DeviceFails          0
         -DeviceRestarts       0
         -CryptoClientCount    2
         -MaxCryptoClients     3
         -AuditDBFreeSpaceMB   11822
         -AuditDBUsedSpaceMB   0
      +HostEnvStats:
         -Uptime               557466
         -CPULoadPercent       0
         -MemAllocUser         1874176
         -MemAllocKernel       1874176
         -CurrentTempC         34.50
         -MinTempC             24.00
         -MaxTempC             41.50
         -CurrentTemp2C        34.00
         -MinTemp2C            24.50
         -MaxTemp2C            41.50
         -VoltageOn3p3VSupply  3.37
         -CurrentOn3p3VSupply  0.20
```

```
      -VoltageOn5VSupply    5.00
      -CurrentOn5VSupply    0.68
      -VoltageOn12VSupply   11.91
      -CurrentOn12VSupply   1.08
      -VoltageOn5VSBSupply  5.02
      -CurrentOn5VSBSupply  0.35
      -TamperBattery1       3.56
      -TamperBattery2       0.04
      -PSUFailure           0
      -CurrentFanRPM        6240
      -CurrentFan2RPM       6240
      -CurrentFan3RPM       6240
      -CurrentFan4RPM       6240
      -CurrentFan5RPM       0
      -CurrentFan6RPM       0
   +HostSysInfo:
      +SystemFans:
         +#1:
            -CurrentFanRPM        6240
         +#2:
            -CurrentFanRPM        6240
         +#3:
            -CurrentFanRPM        6240
         +#4:
            -CurrentFanRPM        6240
   +Connections:
      +#1:
         -Uptime               557411
         -CmdCount             0
         -CmdBytes             0
         -CmdMarshalErrors     0
         -ReplyCount           1713
         -ReplyBytes           47964
         -ReplyMarshalErrors   0
         -DevOutstanding       1
         -QOutstanding         0
         -LongOutstanding      0
         -RemoteIPAddr         (local)
         -ClientNumber         1
         -ClientProcessID      0
         -ClientProcessName
         -ObjectCountTotal     0
         +PerModule:
            +#1:
               -ObjectCount          0
      +#5:
         -Uptime               557357
         -CmdCount             7
         -CmdBytes             536
         -CmdMarshalErrors     0
         -ReplyCount           7
         -ReplyBytes           748
         -ReplyMarshalErrors   0
         -DevOutstanding       0
         -QOutstanding         0
         -LongOutstanding      0
         -RemoteIPAddr         (local)
         -ClientNumber         5
         -ClientProcessID      726
         -ClientProcessName    /opt/nfast/sbin/config-update
         -ObjectCountTotal     0
         +PerModule:
            +#1:
               -ObjectCount          0
   +PerModule:
      +#1:
         +ModuleObjStats:
```

```
      -ObjectCount          6
      -ObjectsCreated       211
      -ObjectsDestroyed     205
   +ModuleCacheStats:
      -CacheEntryCount      0
      -CacheEntriesInserted 0
      -CacheEntriesRemoved  0
   +ModuleEnvStats:
      -SerialNumber         4210-02E0-D947
      -Uptime               54491
      -CurrentTime          1713974041
      -MemTotal             2030891008
      -MemAllocKernel       473952256
      -MemAllocUser         0
      -TempSP               38.00
      -CurrentCPUTemp1      55.00
      -CurrentCPUTemp2      44.00
      -CPUVoltage1          1.00
      -CPUVoltage2          1.79
      -CPUVoltage3          0.99
      -CPUVoltage4          1.35
      -MaxTempC             55.00
      -MinTempC             36.00
      -AIS31PrelimAlarms    0
      -MceCount             0
      -SpiRetries           0
      -SpI2c1               0
      -SpI2c2               34
      -SpTempExcursion      0
      -SpVoltageExcursion   0
      -HostBusExceptions    0
      -CryptoBusExceptions  0
      -SpSensorCmdFails     0
      -NVMFreeSpace         273788928
      -NVMWearLevel         0.04
      -NVMWornBlocks        0.00
      -CurrentFanSpeed      4615
      -CurrentFanDuty       25
   +XCSecurityProcessorLog:
      +#4:
         -XCSpLogEventID       5
         -XCSpLogEventDate     1712803702
      +#3:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1712803759
      +#2:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1712803979
      +#1:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1712804362
      +#0:
         -XCSpLogEventID       18
         -XCSpLogEventDate     1713416534
   +ModuleJobStats:
      -CmdCount             4643573
      -ReplyCount           2290447
      -CmdBytes             1668
      -ReplyBytes           5251
      -JobsStarted          55
      -JobsComplete         2353182
      -RepliesQueued        55
      -HostWriteCount       198489
      -HostWriteErrors      0
      -HostReadCount        397806
      -HostReadErrors       0
      -HostReadEmpty        0
```

```
            -HostReadDeferred     198705
            -HostReadTerminated   0
            -PFNIssued            21456
            -PFNRejected          0
            -PFNCompleted         21455
            -ANIssued             6
            -CPULoadPercent       2
         +ModulePCIStats:
            -HostIRQs             4643632
            -HostReadCount        2290449
            -HostReadDeferred     0
            -HostReadErrors       3
            -HostReadReconnect    0
            -HostWriteCount       2353184
            -HostWriteErrors      0
   +PerDevice:
      +#1:
         +ModuleDriverStats:
            -DriverIRQs           4643585
            -DriverReadIRQs       2290453
            -DriverWriteIRQs      2353185
            -DriverWriteFails     0
            -DriverWriteBlocks    2353185
            -DriverWriteBytes     1489286200
            -DriverReadFails      0
            -DriverReadBlocks     0
            -DriverReadBytes      0
            -DriverEnsureFail     1
            -DriverEnsure         2290451
         +ModuleServerStats:
            -JobsOutstanding      1
            -LongJobsOutstanding  0
            -CmdCount             13438883
            -ReplyCount           13438882
```

▼ *+PerDevice*

```
+PerDevice:
   +#1:
      +ModuleDriverStats:
         -DriverIRQs           190493
         -DriverReadIRQs       96241
         -DriverWriteIRQs      94281
         -DriverWriteFails     0
         -DriverWriteBlocks    94281
         -DriverWriteBytes     3470424
         -DriverReadFails      0
         -DriverReadBlocks     0
         -DriverReadBytes      0
         -DriverEnsureFail     0
         -DriverEnsure         96236
      +ModuleServerStats:
         -JobsOutstanding      1
         -LongJobsOutstanding  0
         -CmdCount             12638
         -ReplyCount           12637
   +#2:
      +ModuleDriverStats:
         -DriverIRQs           193067
         -DriverReadIRQs       99010
         -DriverWriteIRQs      94580
         -DriverWriteFails     0
         -DriverWriteBlocks    94580
         -DriverWriteBytes     3043588
         -DriverReadFails      0
```

```
         -DriverReadBlocks     14
         -DriverReadBytes      114688
         -DriverEnsureFail     0
         -DriverEnsure         99011
      +ModuleServerStats:
         -JobsOutstanding      1
         -LongJobsOutstanding  0
         -CmdCount             12681
         -ReplyCount           12680
```

`PerModule`, `ModuleObjStats`, and `ModuleEnvStats` are `<node>` tags that identify classes of statistics for each hardwerver or HSM. `1` identifies an instance node.

`ObjectCount`, `MemTotal`, and the remaining items at the same level are pairs of `<statistics-id>`s and their values. Times are listed in seconds. Other numbers are integers, which are either real numbers, IP addresses, or counters. For example, a result `-CmdCount 74897` means that there have been 74,897 commands submitted.

If you provide a `<node>` on the command line, `stattree` uses it as the starting point of the tree and displays only information at or below that `<node>` in the tree. Values for `<node>` can be numeric or textual.

```
$ stattree PerModule 3 ModuleObjStats
+#PerModule:
   +#3:
      +#ModuleObjStats:
         -ObjectCount          6
         -ObjectsCreated       334
         -ObjectsDestroyed     328
```

If you use a `<statistics-id>` instead of a `<node>` on the command line, you get an error message:

```
$ stattree PerModule 3 ModuleObjStats ObjectCount
+#PerModule:
   +#3:
      +#ModuleObjStats:
Unable to convert 'ObjectCount' to number or tag name.
```

170.2. Node tags

`Connections`	Statistics for connections between clients and the hardserver. There is one node for each currently active connection. Each node has an instance number that matches the log message generated by the server when that client connected. For example, when the hardserver message is `Information: New client #24 connected`, the client's statistics appear under node #24 in the `stattree` output.
`HostEnvStats`	**Only in network-attached HSMs** Environmental statistics for the HSM.

HostSysInfo	**Only in network-attached HSMs** Further statistics for the HSM.
ModuleCacheStats	Statistics about cache entries.
ModuleEnvStats	General statistics for the HSM's operating environment.
ModuleJobStats	This tag holds statistics for the Security World Software commands (jobs) processed by this HSM.
ModuleObjStats	Statistics for the HSM's Object Store, which contains keys and other resources. These statistics may be useful in debugging applications that leak key handles, for example.
ModulePCIStats	This tag holds statistics for the PCI connection between the HSM and the host computer. It does not apply to nShield Edge HSMs.
ModuleSerialStats	This tag is for nShield Edge HSMs only. It holds statistics for the serial connection between the HSM and the host computer.
PerModule	Statistics kept by the HSMs. There is one instance node for each HSM, numbered using the standard HSM numbering. The statistics provided by each HSM depend on the HSM type and firmware version.
ServerGlobals	Aggregate statistics for all commands processed by the hardserver since it started. The standard statistics apply to the commands sent from the hardserver to HSMs. Commands processed internally by the server are not included here. The Uptime statistic gives the total running time of the server so far.
XCSecurityProcessorLog*	Lists Security Processor log events for XC HSMs.

170.3. Statistics IDs

AIS31PrelimAlarms	**nShield 5 HSMs** The total number of AIS31 (RNG) preliminary alarms. Does not necessarily indicate RNG failure.
ANIssued	The number of Asynchronous Notification messages issued by the HSM to the hardserver. These messages indicate such things as the clear key being pressed and the HSM being reset. In later firmware revisions inserting or removing the smartcard or changing the non-volatile memory also generate asynchronous notifi cations.
AuditDBFreeSpaceMB	**nShield 5 HSMs** The amount of free space available for the hardserver's temporary audit database, rounded to nearest megabyte.
AuditDBUsedSpaceMB	**nShield 5 HSMs** The amount of space consumed by the hardserver's temporary audit database, rounded to the nearest megabyte.

CacheEntriesInserted	Total number of entries inserted into the cache.
CacheEntriesRemoved	Total number of entries removed from the cache.
CacheEntryCount	The number of entries in the cache.
ChanJobsCompleted	The number of fast channel jobs completed by the HSM. The fast channel facility is unsupported on current HSMs. This number should always be 0.
ChanJobErrors	The number of low-level (principally data transport) errors encountered while processing fast channel jobs. Should always be 0 on current HSMs.
ChanJobsIssued	The number of fast channel jobs issued to the HSM. The fast channel facility is unsupported on current HSMs. This number should always be 0.
ClientCount	The number of clients currently connected to the hardserver. This also includes internal client objects. Remote nCipher Secure Transport/Impath connections are represented both by the outer remote protocol and the nested nCore client payload, which results in ClientCount being incremented twice. See CryptoClientCount if you only want the number of licensed cryptographic client sessions currently in use.
ClientNumber	The integer identifier for this hardserver client. It increments for each new client created.
ClientProcessID	The process ID of the client application connected to the hardserver, or 0 if not applicable or not available for this client type.
ClientProcessName	The process name of the client, if available. Usually the executable path.
CmdBytes	The total length of all the command blocks sent for processing.
CmdCount	The total number of commands sent for processing from a client to the server, or from the server to an HSM. Contains the number of commands currently being processed.
CmdMarshalErrors	The number of times a command block was not understood when it was received. A nonzero value indicates either that the parties at each end of a connection have mismatched version numbers (for example, a more recent hardserver has sent a command to a less recent HSM that the HSM does not understand), or that the data transfer mechanism is faulty.

`CPULoadPercent`	**PCIe HSMs** The current processing load on the HSM, represented as a number between 0 and 100. Because an HSM typically contains a number of different types of processing resources (for example, main CPU, and RSA acceleration), this figure is hard to interpret precisely. In general, HSMs report 100% CPU load when all RSA processing capacity is occupied; when performing non-RSA tasks the main CPU or another resource (such as the random number generator) can be saturated without this statistic reaching 100%. **Network-attached HSMs** The current utilization of the main CPU, across all cores. If you are on a firmware version earlier than 13.1, this instead reports a load average that is scaled by 100, but could be greater than 100% if there is an average of more than one runnable thread.
`CPUVoltage_N_`	The current battery voltage for each voltage rail in the HSM. Where `N` is the CPU number.
`CryptoBusExceptions`	PCI1 (Crypto) NPE (non-parity error) count.

`CryptoClientCount`	**Only relevant when reported from a hardserver with remote clients**
	The number of licensable clients connected (active and parked sessions).
	
	Use the `CryptoClientCount` value under `stattree RemoteServers #MODULE_NUMBER ServerGlobals`. This is the number of remote crypto clients for the specified HSM.
	
	`+RemoteServers:`
	`  +#3:`
	`  +ServerGlobals:`
	`  <...>`
	`  -CryptoClientCount  2`
	
	The value under `stattree ServerGlobals` is a count of remote crypto clients of the local hardserver. This value is typically `0`, unless you share local PCIe or USB HSMs with another client hardserver.
	
	Network-attached HSMs: If you have updated the client software to v13.6 or later but have not yet updated the nShield Connect or 5c image to v13.6 or later, then the only `CryptoClientCount` value reported by `stattree` will be the `stattree ServerGlobals` value for the local hardserver, which will not usually be relevant.

`CurrentCPUTemp_N_`	Temperature recorded by the CPU sensor, in degrees C. Where `N` is the CPU number.
`CurrentFanDuty`	Fan duty cycle.
`CurrentFanSpeed`	The fan speed, in RPM, for each fan in the HSM.
`CurrentFan_N_RPM`	The fan speed, in RPM. Where `N` is the fan number.

CurrentOn_X_Supply	The current on the power supply. Where X is the power supply voltage.
CurrentTempC	The current temperature (in degrees Celsius) of the HSM main circuit board. First-generation HSMs do not have a temperature sensor and do not return temperature statistics.
CurrentTime	The current time on the module RTC (Real Time Clock), in seconds-since-the-1970-epoch (Unix time) format.
DeviceFails	The number of times the hardserver has declared a device to have failed. The hardserver provides a diagnostic message when this occurs.
DeviceRestarts	The number of times the hardserver has attempted to restart an HSM after it has failed. The hardserver provides a Notice message when this occurs. The message does not indicate that the attempt was successful.
DevOutstanding	The number of commands sent by the specified client that are currently executing on one or more HSMs. When an HSM accepts a command from a client, this number decreases by 1 and QOutstanding increases by 1. Commands that are processed purely by the server are never included in this count.
HostBusExceptions	**nShield 5 HSMs** PCI0 (Host) NPE (non-parity error) and PE (parity error) count.
HostDebugIRQs	On PCI HSMs, the number of debug interrupts received. This is used only for driver testing, and should be 0 in any production environment.
HostIRQs	On PCI HSMs, the total number of interrupts received from the host. On current HSMs, approximately equal to the total of HostReadCount and HostWriteCount.
HostReadCount	The number of times a read operation to the HSM was attempted. The HSM can defer a read if it has no replies at the time, but expects some to be available later. Typically the HSM reports HostReadCount in two places: the number under ModuleJobStats counts a deferred read twice, once when it is initially deferred, and once when it finally returns some data. The number under ModulePCIStats counts this as one operation.
HostReadDeferred	The number of times a read operation to the HSM was suspended because it was waiting for more replies to become available. When the HSM is working at full capacity, a sizeable proportion of the total reads are likely to be deferred.
HostReadEmpty	The number of times a read from the HSM returned no data because there were no commands waiting for completion. In general, this only happens infrequently during HSM startup or reset. It can also happen if PauseForNotifications is disabled.
HostReadErrors	The number of times a read to an HSM failed because the parameters supplied with the read were incorrect. A nonzero value here typically indicates some problem with the host interface or device driver.

HostReadTerminated	The number of times an HSM had to cancel a read operation which has been deferred. This normally happens only if the clear key is pressed while the HSM is executing commands. Otherwise it might indicate a device driver, interface, or firmware problem.
HostReadUnderruns	Not currently reported by the HSM.
HostUnhandledIRQs	On PCI HSMs, the number of unidentified interrupts from the host. If this is nonzero, a driver or PCI bus problem is likely.
HostReadReconnect	On PCI HSMs, the number of deferred reads that have now completed. This should be the same as HostReadDeferred, or one less if a read is currently deferred.
HostWriteBadData	Not currently reported by the HSM. Attempts to write bad data to the HSM are reflected in HostWriteErrors.
HostWriteCount	The number of write operations (used to submit new commands) that have been received by the HSM from the host machine. One write operation may contain more than one command block. The operation is most efficient when this is the case.
HostWriteErrors	The number of times the HSM rejected the write data from the host. A nonzero value may indicate that data is being corrupted in transfer, or that the hard-server/device driver has got out of sync with the HSM's interface.
HostWriteNoMemory	Not currently reported by the HSM. Write failures due to a lack of memory are reflected in HostWriteErrors.
HostWriteOverruns	Not currently reported by the HSM. Write overruns are reflected in HostWriteErrors.
JobsComplete	The number of jobs completed in the HSM. This value includes all jobs on the module, including jobs from the SEE machine.
JobsOutstanding	The number of jobs that are currently in progress on the HSM. This value includes all jobs on the module, including jobs from the SEE machine.
JobsStarted	The number of jobs started in the HSM. This value includes all jobs on the module, including jobs from the SEE machine.
LongJobsOutstanding	The number of long jobs that are currently in progress on the HSM. This value includes all jobs on the module, including jobs from the SEE machine.
LongOutstanding	The number of LongJobs sent by the specified client that are currently executing on one or more HSMs. When an HSM accepts a LongJobs command from a client, this number increases by 1 and QOutstanding decreases by 1. Commands that are processed purely by the server are never included in this count.
MaxClients	The maximum number of client connections ever in use simultaneously to the hard server. This gives an indication of the peak load experienced so far by the server.
MaxCryptoClients	**Only in network-attached HSMs** The maximum number of client connections permitted by license.

MaxTempC	The maximum temperature recorded by the HSM's temperature sensor. This is stored in non-volatile memory, which is cleared only when the HSM is initialized. First-generation HSMs do not have a temperature sensor and do not return temperature statistics.
MCECount	The number of machine check exceptions (MCEs). This is the total of RAM ECC error counts in /sys/devices/system/edac/mc/mc0/ce_count and /sys/devices/system/edac/mc/mc0/ue_count.
MemAllocKernel	**PCIe HSMs** not supported **Connect network-attached HSMs** The total amount of RAM allocated for kernel (that is, non-SEE) use in an HSM. This is principally used for the object store (keys, logical tokens, and similar) and for big-number buffers. **nShield 5c and later network-attached HSMs** Obsolete, retained only for backwards compatibility. Shows the same value as MemTotal.
MemAllocUser	**PCIe HSMs** not supported **Connect network-attached HSMs** The total amount of RAM allocated for user-mode processes in the HSM (0 for non-SEE use). This includes the size of the SEE Machine image, and the total heap space available to it. **nShield 5c and later network-attached HSMs** Obsolete, retained only for backwards compatibility. Shows the same value as MemTotal.
MemTotal	The total amount of RAM (both allocated and free) available to the HSM. This is the installed RAM size minus various fixed overheads.
MinTempC	The minimum temperature recorded by the HSM's temperature sensor. This is stored in non-volatile memory, which is cleared only when the HSM is initialized. First-generation HSMs do not have a temperature sensor and do not return temperature statistics.
NVMFreeSpace	The total amount of free space in the NVRAM of the HSM, in bytes.
NVMWearLevel	The wear level of the HSM's NVRAM, expressed as a percentage of the ratio between the erase count and the endurance.
NVMWornBlocks	The percentage of worn blocks in the NVRAM of the HSM.
ObjectsCreated	The number of times a new object has been put into the object store. This appears under the HSM's ModuleObjStats node.
ObjectsDestroyed	The number of items in the HSM's object store that have been deleted and their corresponding memory released.

ObjectCount	The current number of objects (keys, logical tokens, buffers, SEE Worlds) in the object store. This is equal to ObjectsCreated minus ObjectsDestroyed. An empty HSM contains a small number of objects that are always present.
ObjectCountTotal	The number of objects loaded by this hardserver client across all modules. This is the sum of all the nested ObjectCount values.
PFNCompleted	The number of PauseForNotifications commands that have been completed by the HSM. Normally, this is one less than the PFNIssued figure because there is normally one such command outstanding.
PFNIssued	The number of PauseForNotifications commands accepted by the HSM from the hardserver. This normally increases at a rate of roughly one every two seconds. If the hardserver has this facility disabled (or a very early version), this does not occur.
PFNRejected	The number of PauseForNotifications commands rejected by the HSM when received from the hardserver. This can happen during HSM startup or reset, but not in normal use. It indicates a hardserver bug or configuration problem.
PSUFailure	The number of power supply unit (PSU) failures.
QOutstanding	The number of commands waiting for an HSM to become available on the specified client connection. When an HSM accepts a command from a client, this number decreases by 1 and DevOutstanding increases by 1. Commands that are processed purely by the server are never included in this count.
RemoteIPAddr	The remote IP address of a client who has this connection. A local client has the address 0.0.0.0.
RepliesQueued	The number of replies and notifications added to the output queue, waiting to be sent.
ReplyCount	The total number of replies returned from server to client, or from HSM to server.
ReplyBytes	The total length of all the reply blocks received after completion.
ReplyMarshalErrors	The number of times a reply was not understood when it was received. A nonzero value indicates either that the parties at each end of a connection have mismatched version numbers (for example, a more recent hardserver has sent a command to a less recent HSM that the HSM does not understand), or that the data transfer mechanism is faulty.
SPVoltageExcursion	The number of voltage excursions (sudden changes in voltage level).
SerialNumber	The unique serial number (ESN) of the HSM.
SpI2c1 and SpI2c2	The Security Processor total and secondary I2c errors.
SpSensorCmdFails	The number of SPI bus synchronization errors.
SpTempExcursion	The number of temperature excursions (sudden changes in temperature level).
SpiRetries	The number of times the T1022 SPI code has passed through an error path.

SystemFans	The fan speed (RPM) for each fan in the HSM.
TamperBattery1 and TamperBattery2	**Network-attached HSMs** The voltage of the batteries in the fan tray. When the unit has mains power, the readings will be slightly over 3.6v. If a voltage less than 3.5 is reported for a batttery, that battery has very little remaining capacity.
TempSP	The temperature of the Security Processor, in degrees C.
Uptime	The length of time (in seconds) since an HSM was last reset, the hardserver was started, or a client connection was made.
VoltageOn_X_Supply	The actual voltage on the power supply. Where X is the power supply voltage.
XCSplogEventDate	The date the log event was created.
XCSplogEventID	The ID of the log event.

170.4. ModuleDriverStats fields

DriverIRQs	Total number of interrupts
DriverReadIRQs	Read interrupts
DriverWriteIRQs	Write interrupts
DriverWriteFails	Write failures
DriverWriteBlocks	Blocks written
DriverWriteBytes	Bytes written
DriverReadFails	Read failures
DriverReadBlocks	Blocks read
DriverReadBytes	Bytes read
DriverEnsureFail	Read request failures
DriverEnsure	Read requests

171. sworldcheck

```
sworldcheck
```

Checks for any Security World data on the network-attached HSM.

172. tamperlog

```
tamperlog
```

Shows the tamper log of the network-attached HSM.

173. tct2

```
tct2 [[-S|--sign] | [-P|--pack] | [-E|--encrypt] | [--add-sig] | [--sign-and-pack] | [--print-sigs] | [--unpack-
skycert] | [--unpack-sar-payload]] [--sigfile=<NAME> ] [-k|--key=<IDENT>] [[--is-machine] |[--machine-
key=<HASH>]| [--machinekey-ident=<IDENT> ] [-T|--machine-type=<TYPE>]] [-m|--module=<MODULE> ] [-o|--
outfile=<OUTFILE>] [--non-interactive] [--show-metadata] [-v|--verbose] [-q|--quiet] [[-i|--infile=]<INFILE>]
```

Trusted Code Tool: enables users to sign, pack, and encrypt file archives so that they can be loaded onto an SEE-Ready nShield HSM. `tct2` uses keys that are protected by a Security World or an OCS and creates SAR files.

Examples of how tct2 can be used are provided in Example SEE machines.



Encrypted SEE machines are not supported for use with nShield Connect HSMs. When the `SEEMachine` binary is installed on the Connect itself for automated loading at boot, the SEE Confidentiality key is not available. However, when a client host loads a `SEEMachine`, it has access to the SEE Confidentiality key and can cause the binary to be decrypted. In this scenario, the Connect works fine with encrypted `SEEMachine` binaries.

Check the documentation supplied by the application vendor to see if you need to use `tct2` to set up and use the application.

Option	Description
Program options, use exactly one	
`--add-sig`	Creates a signed SAR file `--outfile=<OUTFILE>` from the unsigned SAR file `--infile=<INFILE>` and the key `--key=<IDENT>`.
`-E, --encrypt`	Encrypts the packed SAR file `--infile=<INFILE>`. `--key=<IDENT>` must be specified.
`-P, --pack`	Packs the file `--infile=<INFILE>` and any signatures `--sigfile=<NAME>` into a SAR file `--outfile=<OUTFILE>`. When creating an SEE machine image, the input file is a `.SXF` file produced by the `elftool` utility. When creating a SEE user data file, the input format is determined by the SEE machine type.
`--print-sigs`	Displays the key hashes used to sign the SAR file `--infile=<INFILE>`.

Option	Description
`-S, --sign`	Creates a signature on the file `--infile=<INFILE>`. You must specify `--key=<IDENT>` and one of • `--is-machine` • `--machine-key=<HASH>` • `--machine-key-ident=<IDENT>`
`--sign-and-pack`	Creates a signature on the file `--infile=<INFILE>` using `--key=<IDENT>` and one of `--is-machine`, `--machine-key=<HASH>`, or `--machine-key-ident=<IDENT>`, then to pack it in the file `--outfile=<OUTFILE>`.
`--unpack-sar-payload`	Retrieves the payload of the SAR file `--infile=<INFILE>`.
Packing and signing options	
`--sigfile=<NAME>`	File that contains the signature. This option can be repeated to specify multiple signatures.
Machine key specification options for signing operations	
`--is-machine`	Uses SEE machine signing mode.
`--machine-key=<HASH>`	Key hash of the SEE machine for which this signature is good.
`--machine-key-ident=<IDENT>`	Retrieves the hash of key `<IDENT>` then behaves like `--machine-key=<HASH>`. Only one machine key specification option can be specified.
`-T, --machine-type=<TYPE>`	SEE machine type. If you are not sure which SEE machine type is appropriate for your HSM, run enquiry and check the `SEE Machine Type` output. If you do not specify an SEE machine type with this option, `tct2` tries to determine the appropriate type by reading the format of the code to be signed. If `tct2` cannot determine the appropriate SEE machine type, it returns an error message. In such a case, run `tct2` again, explicitly setting the SEE machine type with this option. Machine type parameter (`<TYPE>`) for `tct2` as a string or a number: **SEE Machine Type** — `tct2` machine type parameter **PowerPCSXF** — `PowerPCSXF` or `2` **PowerPCELF** — `PowerPCELF` or `5`
Other options	
`-i, --infile=<INFILE>`	Name of the input `.sxf` file. You can also specify the input file without the using `--infile` option by including the file name at the end of the command.
`--non-interactive`	Sets non-interactive mode. If you have not already loaded any required card sets, `tct2` fails (instead of prompting you to load any required card sets).

Option	Description
`-o, --outfile=<OUTFILE>`	Name of the output `.sar` file to create. This option is valid only with the **Program options** that create an output file.
`-q, --quiet`	Decrease the verbosity level. Use repeatedly, for example as `-qqq` to jump-decrease the level.
`--show-metadata`	Shows the image metadata before signing.
`-v, --verbose`	Increase the verbosity level. Use repeatedly, for example as `-vvv` to jump-increase the level.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `tct2` uses all modules by default.
Help options	
`-h, --help`	Displays help for `tct2`.
`-u, --usage`	Displays a brief usage summary for `tct2`.
`-V, --version`	Displays the version number of the Security World Software that deploys `tct2`.

173.1. Sign with tct2

- When you are generating the key with the `generatekey` command-line utility, ensure you select the application type `seeinteg`.

Signing keys can be DSA or RSA. You can sign a file any number of times using different signing keys.

For information about key application types, see Key application type (APPNAME).

For information about generating keys, see Generating keys.

To create a signature, give a command of the form:

```
tct2 -S|--sign [-m|--module=<MODULE>] -k|--key=<IDENT> [--machine-key=<HASH>| --machine-key-ident=<IDENT> | --is
-machine] -o|--outfile=<OUTFILE> [-i|--infile=<INFILE>]
```

If the signing key is protected by an OCS, `tct2` prompts you for the passphrase for the inserted card.

173.2. Pack with tct2

All files must be packed even if you are not adding signatures. The packing operation must be performed once and only once. Your application vendor may have supplied a pre-packed SAR file.

Packing a file creates a new SAR file. The packed file contains:

- The original file
- Specified signatures, if any.

To pack a file and any signatures, give a command of the form:

```
tct2 -P|--pack -o|--outfile=<OUTFILE> [-i|--infile=]<INFILE> [sigfile...]
```

173.3. Encrypt with tct2

Encrypted SEE machines are not currently supported for use with nShield Connects. When the `SEEMachine` binary is installed on the Connect itself for automated loading at boot, the SEE Confidentiality key is not available. However, when a client host loads a `SEEMachine`, it has access to the SEE Confidentiality key and can cause the binary to be decrypted. In this scenario, the Connect works fine with encrypted `SEEMachine` binaries.

When you are generating the key with the `generatekey` command-line utility, ensure you select the application type `seeconf`.

Encryption keys can be either Triple DES or AES keys. Encryption keys can be protected by the Security World or by a 1/*N* OCS.

For information about key application types, see Key application type (APPNAME).

For information about generating keys, see the *User Guide*.

A `.sar` file can be encrypted only once. To encrypt a `.sar` file, use the command:

```
tct2 -E|--encrypt -k|--key=<IDENT> [-m|--module=<MODULE>] -o|--outfile=<OUTFILE> [-i|--infile=]<INFILE>
```

174. trial

```
trial -i [flags]
trial -p [flags] >tracefile
trial -c [flags] <tracefile
trial -t [flags] <tracefile
```

Checks that the HSMs are functioning as expected and to test the cryptographic functionality at the nCore level. Tests the nCore API commands.

You can use this utility interactively or from a script file.

Default:

`stdin` is not a `tty`? → `-c`
  `stdout` is a `tty`? → `-i`
    `stdout` is not `tty`? → `-p`

Option	Description
Action selection	
`-c, --check`	Runs non-interactively from file and check answers, expecting them in the file.
`-i, --prompt-nolog`	Runs interactively on `stdout` without logging.
`-p, --prompt-log`	Runs interactively on `stdout` with logging.
`-t, --test`	Runs non-interactively from a file and prints the results.
Other options	
`--force-client`	Reports errors relating to `ClientID` immediately.
`--no-client`	Does not set the `ClientID` on connection.
`--no-preload`	Ignores preloaded tokens or keys.
`-P, --privileged`	Uses a privileged connection.
`-U, --unbuffered`	Makes `stdin` and `stdout` unbuffered.
Module selection	
`-m, --module=MODULE`	Specifies the number ID to use. If you only have one module, `MODULE` is `1`. If you do not specify a module ID, `trial` uses all modules by default.
Help options	
`-h, --help`	Displays help for `trial`.

Option	Description
`-u, --usage` `--help-labels`	Display a brief usage summary for trial.
`-v, --version`	Displays the version number of the Security World Software that deploys `floodtest`.

175. uptime

```
uptime
```

Shows how long the network-attached HSM has been running since the last boot.

176. version

```
version
```

Shows the serial console version of the network-attached HSM.