THALES

cpl.thalesgroup.com

payShield 9000 v3.5

Console Reference Manual

1270A544-038

26 July 2021

Contents

End User License Agreement

Use of this product is subject to the Thales Cloud Protection & Licensing *End User License Agreement* found at:

https://cpl.thalesgroup.com/legal

Revision Status

Document No.	Manual Set	Software Version	Release Date
1270A544-038	Issue 38	payShield 9000 v3.5	July 2120

Chapter 1 – Introduction

About this Manual

This manual is a reference document containing details of all commands that can be used on the HSM console. For other payShield 9000 information, see the following manuals:

- *payShield 9000 Security Operations Manual*
- *payShield 9000 Installation Manual*
- *payShield 9000 Host Programmer's Manual*
- *payShield 9000 Host Command Reference Manual*

List of Console Commands (Alphabetical)

Command	Function	Chapter	Page
XK	Make an RACC left or right key	5	261
XR	Commission a smartcard	5	262
XT	Transfer existing LMK to RLMK	5	263
XX	Decommission a smartcard	5	265
XY	HSM commissioning status	5	266
XZ	Duplicate CTA share	5	267
YA	Generate a CSCK	4	248
YB	Export a CSCK	4	249
Z	Encrypt a Clear ZMK Component	4	230
$	Double-Length Key Calculator	4	224

Note: The following Console commands are no longer available and have been superseded by newer commands:

Console Command	Replaced by
DB Import a KML	IK Import Key
DF Import a BDK	IK Import Key
K Encrypt a Key Under LMK Variants 14-15	FK Form Key from Components
YC Import a CSCK	IK Import Key

List of Console Commands (Functional)

Note: The following Console commands are no longer available and have been superseded by newer commands:

Console Command	Replaced by
DB Import a KML	IK Import Key
DF Import a BDK	IK Import Key
K Encrypt a Key Under LMK Variants 14-15	FK Form Key from Components
YC Import a CSCK	IK Import Key

Chapter 2 – Configuration Commands

This chapter describes the commands used to configure a payShield 9000 HSM to work with the host system. It also includes those commands that provide information to assist with installation and configuration.

Configuration Commands

The payShield 9000 HSM provides the following console commands to support configuration operations:

Reset to Factory Settings

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **RESET**

Function: Returns the HSM to the state it was in when it was shipped from the factory, so that it can be securely taken out of service – e.g. for return to Thales for repair.
Any configuration changes (including port settings) that the customer has applied will be reversed, and any customer data and logs will be erased.
If the HSM is to be returned (e.g. after it has been repaired), a record of all the settings should be made before using this command such that the settings can be re-applied after the HSM's return.
This command also reports whether the HSM is currently configured as it left the factory.

Authorization:
- Authorization is not required.
- The HSM must be in the secure state.

Inputs:
- Confirmation that Reset is required.

Outputs:
- Whether HSM is currently in its factory default state.
- Confirmation of Reset.

Notes:
- This utility cannot reset firmware or licenses installed on the HSM. Therefore after use of this facility, the HSM will still have the most recently installed firmware and license – which may be different from the firmware and license when the HSM was shipped from the factory.
- At the end of the reset process, the payShield 9000 will automatically perform a restart. If the console does not display correctly after this, the payShield 9000 should be restarted manually by using the "Restart" button on the front panel.

Example 1:

```
Secure> RESET <Return>

Reset HSM to factory settings? [Y/N]: Y <Return>

The unit is currently in its factory default state: NO

Resetting the unit will remove all customer data,
including logs, port settings, keys, etc.  This may cause
the console to stop functioning.

This operation should only be performed if this unit is being
taken out of normal operation.

Do you want to reset to the factory default settings? [Y/N]: Y
<Return>

You selected Yes; please confirm to Proceed with reset? [Y/N]:
Y <Return>

Return to factory default state complete

The HSM will now reboot automatically. Press any key to
continue: <Return>

Secure>
```

Example 2:

```
Secure> RESET <Return>

Reset HSM to factory settings? [Y/N]: Y <Return>
The unit is currently in its factory default state: YES

Resetting the unit will remove all customer data,
including logs, port settings, keys, etc.  This may cause
the console to stop functioning.

This operation should only be performed if this unit is being
taken out of normal operation.

Do you want to reset to the factory default settings? [Y/N]: N
<Return>

Secure>
```

Configure Commands

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CONFIGCMDS**

Function: To view the list of enabled host and console commands, and (if in secure state) to enable or disable host and console commands. All available commands are enabled by default.

Commands are enabled or disabled using the following syntax:

[+ or -] [C or H] [<Command Code>]

\+ indicates that the specified command should be enabled.
\- indicates that the specified command should be disabled.
C indicates that <Command Code> is a Console command.
H indicates that <Command Code> is a Host command.
<Command Code> is the command code to be enabled or disabled, and may contain the wildcard character '*'. If the first character is '*', then the second character is absent, and this matches all command codes of the specified type. If the second character is '*', then this matches all command codes of the specified type starting with the given first character.

Authorization: The HSM must be in the secure state to enable/disable host and console commands. The current status of enablement of host and console commands can be viewed in any state.

Inputs:
- List of host commands to enable.
- List of console commands to enable.
- List of host commands to disable.
- List of console commands to disable.

Outputs:
- Complete list of enabled host commands.
- Complete list of enabled console commands.

Errors: Invalid entry

Notes:
- When a disabled host command is invoked, error code 68 is returned.
- When a disabled console command is invoked, the message "Function not defined or not allowed" is displayed.

Example 1: *This example demonstrates the use of the **CONFIGCMDS** console command to view the list of enabled host and console commands.*

```
Online> CONFIGCMDS <Return>

List of enabled Console commands:
GC     GS     EC     FK
List of enabled Host commands:
A0 A4 GG GY

Online>
```

Example 2: *This example demonstrates the use of the **CONFIGCMDS** console command to enable one console command (DE) and disable one host command (A4).*

```
Secure> CONFIGCMDS <Return>

List of enabled Console commands:
GC     GS     EC     FK
List of enabled Host commands:
A0 A4 GG GY
Enter command code (e.g. +CDE) or Q to Quit: +CDE <Return>

List of enabled Console commands:
GC     GS     EC     FK     DE
List of enabled Host commands:
A0 A4 GG GY
Enter command code (e.g. +CDE) or Q to Quit: -HA4 <Return>

List of enabled Console commands:
GC     GS     EC     FK     DE
List of enabled Host commands:
A0 GG GY
Enter command code (e.g. +CDE) or Q to Quit: Q <Return>
Save COMMAND settings to smart card? [Y/N]: N <Return>

Secure>
```

Example 3: *This example demonstrates the use of the **CONFIGCMDS** console command using the wildcard character '*' to disable all non-core host commands, and then enable just those host commands beginning with 'A'.*

```
Secure> CONFIGCMDS <Return>

List of enabled Console commands:
GC     GS     EC     FK
List of enabled Host commands:
A0 A4 GG GY
Enter command code (e.g. +CDE) or Q to Quit: -H* <Return>

List of enabled Console commands:
GC     GS     EC     FK     DE
List of enabled Host commands:
Enter command code (e.g. +CDE) or Q to Quit: +HA* <Return>
List of enabled Console commands:
GC     GS     EC     FK     DE

List of enabled Host commands:
A0 A2 A4 A6 A8 AA AC AE AG AS AU AW AY
Enter command code (e.g. +CDE) or Q to Quit: Q <Return>
Save COMMAND settings to smart card? [Y/N]: Y <Return>

Insert card and press ENTER: <Return>
COMMAND settings saved to the smartcard.

Secure>
```

Configure PIN Block Formats

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CONFIGPB**

Function: To view the list of enabled PIN block formats, and (if in secure state) to enable or disable individual PIN block formats.
The default settings for the available PIN block formats are listed in Chapter 14 of the *payShield 9000 Host Programmer's Manual*.

Authorization: The HSM must be in the secure state to enable/disable PIN block formats. The current status of PIN Block format enablement can be viewed in any state.

Inputs:
- PIN block format identifier.

Outputs:
- List of enabled PIN block formats.

Errors:
- Invalid entry

Example 1: *This example demonstrates the use of the **CONFIGPB** console command to view the list of enabled PIN block formats.*

```
Online> CONFIGPB <Return>

List of enabled PIN Block formats:
01 - ISO 9564-1 & ANSI X9.8 format 0
05 - ISO 9564-1 format 1
35 - MasterCard Pay Now & Pay Later format
41 - Visa/Amex new PIN only format
42 - Visa/Amex new & old PIN format
47 - ISO 9564-1 & ANSI X9.8 format 3
48 - ISO 9564-1 format 4

Online>
```

Example 2**:** *This example demonstrates the use of the **CONFIGPB** console command to enable the use of HSM PIN Block format 03.*

```
Secure> CONFIGPB <Return>

List of enabled PIN Block formats:
01 - ISO 9564-1 & ANSI X9.8 format 0
05 - ISO 9564-1 format 1
35 - MasterCard Pay Now & Pay Later format
41 - Visa/Amex new PIN only format
42 - Visa/Amex new & old PIN format
47 - ISO 9564-1 & ANSI X9.8 format 3
48 - ISO 9564-1 format 4

Enter + or - followed by PIN Block format or Q to Quit: +03 <Return>

List of enabled PIN Block formats:
01 - ISO 9564-1 & ANSI X9.8 format 0
03 - Diebold & IBM ATM format
05 - ISO 9564-1 format 1
35 - MasterCard Pay Now & Pay Later format
41 - Visa/Amex new PIN only format
42 - Visa/Amex new & old PIN format
47 - ISO 9564-1 & ANSI X9.8 format 3
48 - ISO 9564-1 format 4
```

```
Enter + or - followed by PIN Block format or Q to Quit: Q <Return>
Save PIN BLOCK settings to smart card? [Y/N]: Y <Return>

Insert card and press ENTER: <Return>
PIN BLOCK settings saved to the smartcard.

Secure>
```

Configure Security

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **CS**

Function: To set the security configuration of the HSM and some processing parameters. CS converts all lower-case alpha values to upper case for display purposes, except for the Card issuer Password. Operation is menu-driven, as shown in the examples. The security settings can optionally be saved to a smartcard. The settings are described in Chapter 2 of the Security Operations manual, together with their default values.

Authorization: The HSM must be in the secure state to run this command.

Inputs:

- PIN length [4-12]: a one or two-digit number in the range 4 to 12.
- Echo [oN/ofF]: N or F
- Atalla ZMK variant support [oN/ofF]: N or F
- Transaction key scheme: Racal, Australian or None? [R/A/N]: R or A or N
- User storage key length [S/D/T/V]: S, D, T, or V
- Display general information on payShield Manager Landing page? [Y/N]: Y or N
- Default LMK identifier [0-x]: Integer between 0 and x
- Management LMK identifier [0-x] : Integer between 0 and x
- Whether to erase the installed LMKs to enable the following settings to be changed.
- Enforce Atalla variant match to Thales key type [Y/N]
- Select clear PINs? [Y/N]: Y or N
- Enable ZMK translate command? [Y/N]: Y or N
- Enable X9.17 for import? [Y/N]: Y or N
- Enable X9.17 for export? [Y/N]: Y or N
- Solicitation batch size [1-1024]: a one to four-digit number, range 1 to 1024.
- Prevent Single-DES keys masquerading as double or triple-length key? [Y/N]: Y or N
- Single/double length ZMKs [S/D]: S or D
- Decimalization table Encrypted/Plaintext [E/P]: E
- Enable decimalization table checks? [Y/N]: Y or N
- PIN encryption algorithm: A or B
- Whether to use the default Card Issuer password or to enter a different value (of 8 alphanumeric printable characters).
- Authorized State required when importing DES key under RSA key? [Y/N]: Y or N
- Minimum HMAC verification length in bytes [5-64]: number, range 5-64
- Enable PKCS#11 import and export for HMAC keys? [Y/N]: Y or N
- Enable ANSI X9.17 import and export for HMAC keys? [Y/N]: Y or N

- Enable ZEK/TEK encryption of ASCII data or Binary data or None? [A/B/N]: A or B or N
- Restrict Key Check Values to 6 hex chars? [Y/N]: Y or N
- Enable multiple authorized activities? [Y/N]: Y or N
- Enable variable length PIN offset? [Y/N]: Y or N
- Enable weak PIN checking? [Y/N]: Y or N
- Check new PINs using global list of weak PINs? [Y/N]: Y or N
- Check new PINs using local list of weak PINs? [Y/N]: Y or N
- Check new PINs using rules? [Y/N]: Y or N
- Enable PIN Block format 34 as output format for PIN translations to ZPK? [Y/N]: Y or N
- Enable translation of account number for LMK encrypted PINs [Y/N]: Y or N.
- Enable 2DES LMK encryption of 3DES/2048-bit RSA keys [Y/N] : Y or N
- Use HSM clock for date/time validation? [Y/N]: Y or N
- Additional padding to disguise key length? [Y/N] : Y or N
- Key export and import in trusted format only? [Y/N] : Y or N
- Protect MULTOS Cipher Data Checksums? [Y/N]
- Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK? [Y/N] : Y or N
- Enable use of Tokens in PIN Translation? [Y/N]: Y or N
- Enable use of Tokens in PIN Verification? [Y/N]: Y or N
- Allow Error light to be extinguished when viewing Error Log? [Y/N]: Y or N
- Ensure LMK Identifier in command corresponds with host port? [Y/N]: Y or N
- Ignore LMK ID in Key Block Header? [Y/N]: Y or N
- Enforce NIST recommendations when encrypting AES keys using RSA? [Y/N]: Y or N
- Enable import and export of RSA Private keys? [Y/N]: Y or N
- Enable import of a ZMK? [Y/N]: Y or N
- Enable export of a ZMK? [Y/N]: Y or N
- Enable Single-DES [Y/N]: Y or N
- Card/password authorization (local) [C/P]: C or P (Card or Password).
- Restrict PIN block usage for PCI HSM compliance? [Y/N]: Y or N.
- Enforce key type 002 separation for PCI HSM compliance [Y/N]: Y or N. *See notes below.*
- Enforce Authorization Time Limit? [Y/N]: Y or N.
- Enforce Multiple Key Components? [Y/N]: Y or N.
- Save SECURITY settings to smartcard? [Y/N]: Y or N

Outputs:

- Prompts according to the settings chosen (see examples below).

Errors:

Invalid Entry
Card not formatted to save/retrieve HSM settings.
Attempt with another card? [Y/N]

Notes:

- For software versions which have been PCI HSM certified, in

order to be PCI HSM compliant a number of security settings must have specific values as follows:
  - Card/password authorization (local) – must be "C"
  - Restrict PIN block usage for PCI HSM compliance – must be "Y"
  - Enforce key type 002 separation for PCI HSM compliance –must be "Y"
  - Enforce Authorization Time Limit – must be "Y"
  - Enforce Multiple Key Components – must be "Y"

  See Chapter 14 of the *payShield 9000 Host Programmer's Manual* for further information about PIN Block formats.
  See Chapter 10 of the *payShield 9000 General Information Manual* for further information about PCI HSM Compliance.
- Once all of these settings are at the PCI HSM compliant value, they cannot be changed unless the RESET command is used.
- If the value of the setting "Enforce key type 002 separation for PCI HSM compliance" is "Y", then:
  - Key Type Table 2 (see Chapter 7 of the *payShield 9000 Host Programmer's Manual*) is in effect. If the setting has a value of "N", then the HSM is not being operated in a PCI HSM compliant manner and Key Type Table 1 is in effect.
  - The following Host commands are disabled: AA, AE, FC, FE, FG, HC, KA, OE

Example 1: *Erasing LMKs not selected by the user*

```
Secure> CS <Return>
PIN Length [4-12]: 8 <Return>
Echo [oN/ofF]: N <Return>
Atalla ZMK variant support [oN/ofF]: F <Return>
Transaction Key Scheme: Racal, Australian or None [R/A/N]: N <Return>
User storage key length [S/D/T/V](SINGLE): <Return>
Display general information on payShield Manager Landing page? [Y/N]:
Y <Return>
Default LMK identifier [0-4](0): <Return>
Management LMK identifier [0-4](0): <Return>

LMKs must be erased before remaining parameters can be set.

Erase LMKs? [Y/N]: N <Return>

Save SECURITY settings to smartcard? [Y/N]: N <Return>
Secure>
```

Example 2: *Settings affecting PCI HSM compliance do not have compliant values. The user wishes to use the default card issuer password.*

```
Secure> CS <Return>

Please make a selection.  The current setting is in parentheses.
Press ENTER to keep the current setting.
PIN length [4-12](4): <Return>
Echo [oN/ofF](ON): <Return>
Atalla ZMK variant support [oN/ofF](ON): <Return>
Transaction key scheme: Racal, Australian or None? [R/A/N](R): <Return>
User storage key length [S/D/T/V](SINGLE): <Return>
Display general information on payShield Manager Landing page? [Y/N]: Y
<Return>
Default LMK identifier [0-4](0): <Return>
Management LMK identifier [0-4](0): <Return>

LMKs must be erased before remaining parameters can be set

Erase LMKs? [Y/N]: Y <Return>

Enforce Atalla variant match to Thales key type? [Y/N](NO): <Return>
Select clear PINs? [Y/N](YES): <Return>
Enable ZMK translate command? [Y/N](YES): <Return>
Enable X9.17 for import? [Y/N](YES): <Return>
Enable X9.17 for export? [Y/N](YES): <Return>
Solicitation batch size [1-1024](5): <Return>

Prevent Single-DES keys masquerading
as double or triple-length key? [Y/N](YES): <Return>
Single/double length ZMKs [S/D](DOUBLE): d<Return>
Decimalization table Encrypted/Plaintext [E/P](P): <Return>
Enable Decimalization Table Checks? [Y/N](YES): <Return>

PIN encryption algorithm [A/B](A): <Return>
Use default card issuer password [Y/N](YES): <Return>
Authorized State required when importing a key under an RSA key?
[Y/N](YES): <Return>
Minimum HMAC key length in bytes [5-64](10): <Return>
Enable PKCS#11 import and export for HMAC keys [Y/N](YES): <Return>
Enable ANSI X9.17 import and export for HMAC keys [Y/N](YES): <Return>
Enable ZEK/TEK encryption of ASCII data or Binary data or None? [A/B/N]
(NONE): <Return>
Restrict Key Check Values to 6 hex chars [Y/N](YES): <Return>
Enable multiple authorized activities [Y/N](NO): <Return>
Allow persistent authorized activities [Y/N](YES): <Return>
Enable support for variable length PIN offset [Y/N](NO): <Return>
Enable weak PIN checking [Y/N](YES): <Return>
Check new PINs using global list of weak PINs? [Y/N](NO): <Return>
Check new PINs using local list of weak PINs? [Y/N](NO): <Return>
Check new PINs using rules? [Y/N](YES): <Return>
Enable PIN Block Format 34 as output format for PIN Translations to ZPK
[Y/N](NO): <Return>
Enable translation of account number for LMK encrypted PINs [Y/N](YES):
<Return>
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys [Y/N](YES): <Return>

Use HSM clock for date/time validation? [Y/N](YES): <Return>
Additional padding to disguise key length? [Y/N](NO): <Return>
Key export and import in trusted format only? [Y/N](NO): <Return>
Protect MULTOS Cipher Data Checksums? [Y/N](YES): <Return>
Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK? [Y/N](NO):
<Return>
Enable use of Tokens in PIN Translation? [Y/N](NO): <Return>
Enable use of Tokens in PIN Verification? [Y/N](NO): <Return>
Allow Error light to be extinguished when viewing Error Log? [Y/N](NO):
<Return>
Ensure LMK Identifier in command corresponds with host port? [Y/N](NO):
<Return>
```

```
Ignore LMK ID in Key Block Header? [Y/N](NO): <Return>
Enforce NIST recommendations when encrypting AES keys using RSA?
[Y/N](YES): <Return>
Enable import and export of RSA Private keys? [Y/N](NO): <Return>
Enable import of a ZMK? [Y/N](NO): <Return>
Enable export of a ZMK? [Y/N](NO): <Return>

The following settings affect PCI HSM compliance - see Console Reference
Manual:

The following setting is not PCI HSM compliant:
Enable Single-DES? [Y/N](YES): <Return>

The following setting is not PCI HSM compliant:
Card/password authorization (local) [C/P](P): C <Return>

The following setting is not PCI HSM compliant:
Restrict PIN block usage for PCI HSM compliance? [Y/N](NO): N <Return>

Note that this setting is not PCI HSM compliant.
Confirm? [Y/N]: Y <Return>

The following setting is not PCI HSM compliant:
Enforce key type 002 separation for PCI HSM compliance? [Y/N](NO):
<Return>

The following setting is not PCI HSM compliant:
Enforce Authorization Time Limit? [Y/N](NO): <Return>

The following setting is not PCI HSM compliant:
Enforce Multiple Key Components? [Y/N](NO): <Return>

Save SECURITY settings to smartcard? [Y/N]: N <Return>

Secure>
```

Example 3: *Final setting affecting PCI HSM compliance is about to be set to compliant value. The user is specifying a different card issuer software.*

```
Secure> CS <Return>

Please make a selection.  The current setting is in parentheses.
Press ENTER to keep the current setting.
PIN length [4-12](4): <Return>
Echo [oN/ofF](ON): <Return>
Atalla ZMK variant support [oN/ofF](ON): <Return>
Transaction key scheme: Racal, Australian or None? [R/A/N](R): <Return>
User storage key length [S/D/T/V](SINGLE): <Return>
Display general information on payShield Manager Landing Page? [Y/N](Y):
<Return>
Default LMK identifier [0-4](0): <Return>
Management LMK identifier [0-4](0): <Return>

LMKs must be erased before remaining parameters can be set

Erase LMKs? [Y/N]: Y <Return>

Enforce Atalla variant match to Thales key type? [Y/N](NO): <Return>
Select clear PINs? [Y/N](YES): <Return>
Enable ZMK translate command? [Y/N](YES): <Return>
Enable X9.17 for import? [Y/N](YES): <Return>
Enable X9.17 for export? [Y/N](YES): <Return>
Solicitation batch size [1-1024](5): <Return>

Prevent Single-DES keys masquerading
as double or triple-length key? [Y/N](YES): <Return>
Single/double length ZMKs [S/D](DOUBLE): <Return>
Decimalization table Encrypted/Plaintext [E/P](P): <Return>
Enable Decimalization Table Checks? [Y/N](YES): <Return>

PIN encryption algorithm [A/B](A): <Return>
Use default card issuer password [Y/N](YES): N <Return>
Enter card issuer password (local):***** <Return>
Password must be 8 characters.
Enter card issuer password (local):******** <Return>
Confirm card issuer password: ******** <Return>
Authorized State required when importing a key under an RSA key?
[Y/N](YES): <Return>
Minimum HMAC key length in bytes [5-64](10): <Return>
Enable PKCS#11 import and export for HMAC keys [Y/N](YES): <Return>
Enable ANSI X9.17 import and export for HMAC keys [Y/N](YES): <Return>
Enable ZEK/TEK encryption of ASCII data or Binary data or None? [A/B/N]
(NONE): <Return>
Restrict Key Check Values to 6 hex chars [Y/N](YES): <Return>
Enable multiple authorized activities [Y/N](NO): <Return>
Allow persistent authorized activities [Y/N](YES): <Return>
Enable support for variable length PIN offset [Y/N](NO): <Return>
Enable weak PIN checking [Y/N](YES): <Return>
Check new PINs using global list of weak PINs? [Y/N](NO): <Return>
Check new PINs using local list of weak PINs? [Y/N](NO): <Return>
Check new PINs using rules? [Y/N](YES): <Return>
Enable PIN Block Format 34 as output format
for PIN Translations to ZPK [Y/N](NO): <Return>
Enable translation of account number for LMK encrypted PINs [Y/N](YES):
<Return>
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys [Y/N](YES): <Return>

Use HSM clock for date/time validation? [Y/N](YES): <Return>
Additional padding to disguise key length? [Y/N](NO): <Return>
Key export and import in trusted format only? [Y/N](NO): <Return>
Protect MULTOS Cipher Data Checksums? [Y/N](YES): <Return>

Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK? [Y/N](NO):
<Return>
Enable use of Tokens in PIN Translation? [Y/N](NO): <Return>
```

```
Enable use of Tokens in PIN Verification? [Y/N](NO): <Return>
Allow Error light to be extinguished when viewing Error Log? [Y/N](NO):
<Return>
Ensure LMK Identifier in command corresponds with host port? [Y/N](NO):
<Return>
Ignore LMK ID in Key Block Header? [Y/N](NO): <Return>
Enforce NIST recommendations when encrypting AES keys using RSA?
[Y/N](YES): <Return>
Enable import and export of RSA Private keys? [Y/N](NO): <Return>
Enable import of a ZMK? [Y/N](NO): <Return>
Enable export of a ZMK? [Y/N](NO): <Return>

The following settings affect PCI HSM compliance - see Console Reference
Manual:

The following setting is not PCI HSM compliant:
Enable Single-DES? [Y/N](YES): N <Return>

The following setting is not PCI HSM compliant:
Card/password authorization (local) [C/P](P): C <Return>

The following setting is not PCI HSM compliant:
Restrict PIN block usage for PCI HSM compliance? [Y/N](NO): Y <Return>

The following setting is not PCI HSM compliant:
Enforce key type 002 separation for PCI HSM compliance? [Y/N](NO): Y
<Return>

The following setting is not PCI HSM compliant:
Enforce Authorization Time Limit? [Y/N](NO): Y <Return>

The following setting is not PCI HSM compliant:
Enforce Multiple Key Components? [Y/N](NO): Y <Return>

These settings will all become PCI HSM compliant.
No further changes will be allowed to these options:
  Single-DES: DISABLED
  Card/password authorization = 'C'
  Restrict PIN block usage    = 'Y'
  Enforce key type separation = 'Y'
  Enforce Authorization Time Limit = 'Y'
  Enforce Multiple Key Components = 'Y'

Confirm? [Y/N]: Y <Return>

KTKs must be erased by the HSM can be set to PCI compliant mode.
Erase KTKs and switch to PCI compliant mode? [Y/N]: Y <Return>

Save SECURITY settings to smartcard? [Y/N]: Y <Return>
Insert card and press ENTER: <Return>
SECURITY settings saved to the smartcard.

Secure>
```

Example 4: *All settings affecting PCI HSM compliance have compliant values*

```
Secure> CS <Return>

Please make a selection.  The current setting is in parentheses.
Press ENTER to keep the current setting.
PIN length [4-12](4): <Return>
Echo [oN/ofF](ON): <Return>
Atalla ZMK variant support [oN/ofF](ON): <Return>
Transaction key scheme: Racal, Australian or None? [R/A/N](R): <Return>
User storage key length [S/D/T/V](SINGLE): <Return>
Display general information on payShield Manager Landing Page? [Y/N](Y):
<Return>
Default LMK identifier [0-4](0): <Return>
Management LMK identifier [0-4](0): <Return>

LMKs must be erased before remaining parameters can be set

Erase LMKs? [Y/N]: Y <Return>

Enforce Atalla variant match to Thales key type? [Y/N](NO): <Return>
Select clear PINs? [Y/N](YES): <Return>
Enable ZMK translate command? [Y/N](YES): <Return>
Enable X9.17 for import? [Y/N](YES): <Return>
Enable X9.17 for export? [Y/N](YES): <Return>
Solicitation batch size [1-1024](5): <Return>

Prevent Single-DES keys masquerading
as double or triple-length key? [Y/N](NO): <Return>
Making default length for ZMKs: Double
Decimalization table Encrypted/Plaintext [E/P](P): <Return>
Enable Decimalization Table Checks? [Y/N](YES): <Return>

PIN encryption algorithm [A/B](A): <Return>
Use default card issuer password [Y/N](YES): <Return>
Authorized State required when importing a key under an RSA key?
[Y/N](YES): <Return>
Minimum HMAC key length in bytes [5-64](10): <Return>
Enable PKCS#11 import and export for HMAC keys [Y/N](YES): <Return>
Enable ANSI X9.17 import and export for HMAC keys [Y/N](YES): <Return>
Enable ZEK/TEK encryption of ASCII data or Binary data or None? [A/B/N]
(NONE): <Return>

Restrict Key Check Values to 6 hex chars [Y/N](YES): <Return>
Enable multiple authorized activities [Y/N](NO): <Return>
Allow persistent authorized activities [Y/N](YES): <Return>
Enable support for variable length PIN offset [Y/N](NO): <Return>
Enable weak PIN checking [Y/N](NO): <Return>
Enable PIN Block Format 34 as output format
for PIN Translations to ZPK [Y/N](NO): <Return>
Enable translation of account number for LMK encrypted PINs [Y/N](YES):
<Return>
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys [Y/N](YES): <Return>

Use HSM clock for date/time validation? [Y/N](YES): <Return>
Additional padding to disguise key length? [Y/N](NO): <Return>
Key export and import in trusted format only? [Y/N](NO): <Return>
Protect MULTOS Cipher Data Checksums? [Y/N](YES): <Return>
Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK? [Y/N](NO):
<Return>
Enable use of Tokens in PIN Translation? [Y/N](NO): <Return>
Enable use of Tokens in PIN Verification? [Y/N](NO): <Return>
Allow Error light to be extinguished when viewing Error Log? [Y/N](NO):
<Return>
Ensure LMK Identifier in command corresponds with host port? [Y/N](NO):
<Return>
Ignore LMK ID in Key Block Header? [Y/N](NO): <Return>
Enforce NIST recommendations when encrypting AES keys using RSA?
[Y/N](YES): <Return>
```

```
Enable import and export of RSA Private keys? [Y/N](NO): <Return>
Enable import of a ZMK? [Y/N](NO): <Return>
Enable export of a ZMK? [Y/N](NO): <Return>

The following settings are all PCI HSM compliant and cannot be changed.
  Single-DES: DISABLED
  Card/password authorization (local): C
  Restrict PIN block usage for PCI HSM Compliance: YES
  Enforce key type separation for PCI HSM compliance: YES
  Enforce Authorization Time Limit: YES
  Enforce Multiple Key Components: YES


Save SECURITY settings to smartcard? [Y/N]: Y <Return>
Insert card and press ENTER: <Return>
SECURITY settings saved to the smartcard.

Secure>
```

View Security Configuration

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QS**

Function: Reports the security configuration of the HSM and some processing parameters, plus the LMK check value.

Authorization: This command does not require any authorization.

Inputs: None.

Outputs:
- See examples below.

Errors: None.

Notes:
- Where the software has been PCI HSM certified, in order to be PCI HSM compliant a number of security settings must have specific values as follows:
  - Card/password authorization (local) – must be "C"
  - Restrict PIN block usage for PCI compliance – must be "YES" (see Chapter 14 of the *payShield 9000 Host Programmer's Manual* and Chapter 10 of the *payShield 9000 General Information Manual*)
  - Enforce key type 002 separation for PCI HSM compliance –must be "YES"

Once all of these settings are at the PCI HSM compliant value, they cannot be changed. See Chapter 10 of the *payShield 9000 General Information Manual* for further information.

Example 1: *Settings affecting PCI HSM compliance do not all have compliant values:*

```
Online> QS <Return>

PIN length: 04
Encrypted PIN length: 05
Echo: ON
Atalla ZMK variant support: ON
Transaction key support: RACAL
User storage key length: SINGLE
Display general information on payShield Manager Landing Page: YES
Default LMK identifier: 0
Management LMK identifier: 0

Enforce Atalla variant match to Thales key type: NO
Select clear PINs: YES
Enable ZMK translate command: YES
Enable X9.17 for import: YES
Enable X9.17 for export: YES
Solicitation batch size: 0005
Prevent Single-DES keys masquerading as double or triple-length keys: YES
ZMK length: SINGLE
Decimalization tables: PLAINTEXT
Decimalization table checks: ENABLED
PIN encryption algorithm: A

Press "Enter" to view additional security settings... <Return>

Authorized state required when importing a key under an RSA key: YES
Minimum HMAC key length in bytes: 10
Enable PKCS#11 import and export for HMAC keys: YES
Enable ANSI X9.17 import and export for HMAC keys: YES
Enable ZEK/TEK encryption of ASCII data or Binary data or None: NONE
Restrict Key Check Values to 6 hex chars: YES
Enable multiple authorized activities: NO
Allow persistent authorized activities: YES
Enable variable length PIN offset: NO
Enable weak PIN checking: YES
Check new PINs using global list of weak PINs: NO
Check new PINs using local list of weak PINs: NO
Check new PINs using rules: YES
Enable PIN Block Format 34 as output format for PIN Translations to ZPK: NO
Enable translation of account number for LMK encrypted PINs: YES
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys: YES

Use HSM clock for date/time validation: YES
Additional padding to disguise key length: NO
Key export and import in trusted format only: NO
Protect MULTOS Cipher Data Checksums: YES
Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK: NO
Enable use of Tokens in PIN Translation: NO
Enable use of Tokens in PIN Verification: NO
Allow Error light to be extinguished when viewing Error Log: NO
Ensure LMK Identifier in command corresponds with host port: NO
Ignore LMK ID in Key Block Header: NO
Enforce NIST recommendations when encrypting AES keys using RSA: NO
Enable import and export of RSA Private keys: NO
Enable import of a ZMK: NO
Enable export of a ZMK: NO

NOTE: The following settings are not all PCI HSM compliant.
Single-DES: DISABLED
Card/password authorization (local): P
Restrict PIN block usage for PCI HSM Compliance: NO
Enforce key type 002 separation for PCI HSM compliance: NO
Enforce Authorization Time Limit: YES
Enforce Multiple Key Components: YES

Online>
```

Example 2: *Settings affecting PCI HSM compliance have compliant values*

```
Online> QS <Return>

PIN length: 04
Encrypted PIN length: 05
Echo: ON
Atalla ZMK variant support: ON
Transaction key support: RACAL
User storage key length: SINGLE
Display general information on payShield Manager Landing Page: YES
Default LMK identifier: 0
Management LMK identifier: 0

Enforce Atalla variant match to Thales key type: NO
Select clear PINs: YES
Enable ZMK translate command: YES
Enable X9.17 for import: YES
Enable X9.17 for export: YES
Solicitation batch size: 0005
Prevent Single-DES keys masquerading as double or triple-length keys: YES
ZMK length: SINGLE
Decimalization tables: PLAINTEXT
Decimalization table checks: ENABLED
PIN encryption algorithm: A

Press "Enter" to view additional security settings... <Return>

Authorized state required when importing a key under an RSA key: YES
Minimum HMAC key length in bytes: 10
Enable PKCS#11 import and export for HMAC keys: YES
Enable ANSI X9.17 import and export for HMAC keys: YES
Enable ZEK/TEK encryption of ASCII data or Binary data or None: NONE
Restrict Key Check Values to 6 hex chars: YES
Enable multiple authorized activities: NO
Allow persistent authorized activities: YES
Enable variable length PIN offset: NO
Enable weak PIN checking: YES
Check new PINs using global list of weak PINs: NO
Check new PINs using local list of weak PINs: NO
Check new PINs using rules: YES
Enable PIN Block Format 34 as output format for PIN Translations to ZPK: NO
Enable translation of account number for LMK encrypted PINs: YES
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys: YES

Use HSM clock for date/time validation: YES
Additional padding to disguise key length: NO
Key export and import in trusted format only: NO
Protect MULTOS Cipher Data Checksums: YES
Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK: NO
Enable use of Tokens in PIN Translation: NO
Enable use of Tokens in PIN Verification: NO
Allow Error light to be extinguished when viewing Error Log: NO
Ensure LMK Identifier in command corresponds with host port: NO
Ignore LMK ID in Key Block Header: NO
Enforce NIST recommendations when encrypting AES keys using RSA: NO
Enable import and export of RSA Private keys: NO
Enable import of a ZMK: NO
Enable export of a ZMK: NO

The following settings are all PCI HSM compliant and cannot be changed.
Single-DES: DISABLED
Card/password authorization (local): C
Restrict PIN block usage for PCI HSM Compliance: YES
Enforce key type separation for PCI HSM compliance: YES
Enforce Authorization Time Limit: YES
Enforce Multiple Key Components: YES

Online>
```

Example 3: *Software has not been PCI HSM certified*

```
Online> QS <Return>

PIN length: 04
Encrypted PIN length: 05
Echo: ON
Atalla ZMK variant support: ON
Transaction key support: RACAL
User storage key length: SINGLE
Display general information on payShield Manager Landing Page: YES
Default LMK identifier: 0
Management LMK identifier: 0

Select clear PINs: YES
Enable ZMK translate command: YES
Enable X9.17 for import: YES
Enable X9.17 for export: YES
Solicitation batch size: 0005
Single-DES: ENABLED
Prevent Single-DES keys masquerading as double or triple-length keys: YES
ZMK length: SINGLE
Decimalization tables: PLAINTEXT
Decimalization table checks: ENABLED
PIN encryption algorithm: A

Press "Enter" to view additional security settings... <Return>

Authorized state required when importing a key under an RSA key: YES
Minimum HMAC key length in bytes: 10
Enable PKCS#11 import and export for HMAC keys: YES
Enable ANSI X9.17 import and export for HMAC keys: YES
Enable ZEK/TEK encryption of ASCII data or Binary data or None: NONE
Restrict Key Check Values to 6 hex chars: YES
Enable multiple authorized activities: NO
Allow persistent authorized activities: YES
Enable variable length PIN offset: NO
Enable weak PIN checking: NO
Enable PIN Block Format 34 as output format for PIN Translations to ZPK: NO
Enable translation of account number for LMK encrypted PINs: YES
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys: YES

Use HSM clock for date/time validation: YES
Additional padding to disguise key length: NO
Key export and import in trusted format only: NO
Protect MULTOS Cipher Data Checksums: YES
Enable Key Scheme Tag 'X' (X9.17) for storing keys under LMK: NO
Enable use of Tokens in PIN Translation: NO
Enable use of Tokens in PIN Verification: NO
Allow Error light to be extinguished when viewing Error Log: NO
Ensure LMK Identifier in command corresponds with host port: NO
Ignore LMK ID in Key Block Header: NO
Enforce NIST recommendations when encrypting AES keys using RSA: NO
Enable import and export of RSA Private keys: NO
Enable import of a ZMK: NO
Enable export of a ZMK: NO

The following settings are all PCI HSM compliant and cannot be changed.
Single-DES: DISABLED
Card/password authorization (local): C
Restrict PIN block usage for PCI HSM Compliance: YES
Enforce key type separation for PCI HSM compliance: YES
Enforce Authorization Time Limit: YES
Enforce Multiple Key Components: YES

Online>
```

Configure Console Port

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CC**

Function: To set the baud rate and word format for the console port.

The new settings come into effect immediately after the command has completed.

Authorization: This command does not require any authorization.

Inputs:
- Console baud rate.
- Console word format.
- Console parity.
- Console flow control.

Outputs: None.

Errors: None.

Notes:
- The default settings for the console port are:
  - 9600 baud
  - 8 data bits
  - 1 stop bit
  - No parity
  - No flow control
- A USB port which has been configured for printer connection cannot be used for Console connection.

Example:

```
Offline> CC <Return>

Serial Port:

    BAUD RATES
 1.   1200
 2.   2400
 3.   4800
 4.   9600 (current value)
 5.  19200
 6.  38400
 7.  57600
 8. 115200
Console baud rate (enter for no change): <Return>

  DATA BITS
 1. 5
 2. 6
 3. 7
 4. 8 (current value)
Console data bits (enter for no change): <Return>

  STOP BITS
 1. 1 (current value)
 2. 2
Console stop bits (enter for no change): <Return>

  PARITY
 1. none (current value)
 2.  odd
 3. even
Console parity (enter for no change): <Return>

  FLOW Control
 1. none (current value)
 2. software
 3. hardware
Console flow_ctl (enter for no change): <Return>

Serial Port will be configured as:
 Baud: 9600
 Word format: 8 bits, none parity, 1 stop
 Flow control: none

Offline>
```

View Console Port Configuration

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QC**

Function: To display details of the console port configuration of the HSM.

Authorization: This command does not require any authorization.

Inputs: None.

Outputs:
- The console baud rate.
- The console word format.
- The console flow control.

Errors: None.

Example:

```
Online> QC <Return>

Serial Port:
 Baud: 9600
 Word format: 8 bits, no parity, 1 stop
 Flow control: none

Online>
```

Configure Host Port

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CH**

Function: To configure the Host port to emulate a type of data communications equipment and control equipment, which can be one of the following:

- Standard asynchronous emulation.
- Transparent asynchronous emulation.
- Ethernet.
- FICON

The Host port setting can optionally be saved to a smartcard. The new settings come into effect a few seconds after the command has completed.
It is recommended that the Host Ethernet Ports, the Management Ethernet Port, and the Auxiliary Ethernet Port are all on different IP subnets from each other.

Authorization:
- The HSM must be in the offline or secure state to run this command.
- If settings relating to Secure Host Communications (TLS) or Access Control Lists are to be changed, the payShield 9000 must be in Secure state.

Inputs:
- The options are menu driven and the inputs vary depending on the communication mode selected. See examples below.
- When configuring an asynchronous port to accept non-transparent traffic, you will be prompted for "*Terminating characters (4 hex):*". You can specify a sequence of 1 or 2 bytes. If only a single byte is required, the second byte should be set to 00. For example, if you wish to use the standard ETX character ('03' in hex) then you should enter "0300".
- Inputs specific to the FICON interface have the following definitions:
  - Control Unit Image:
    - Valid Range: 0-255; Default=0
    - This is the actual control unit image defined in the mainframe I/O gens.
  - Unit Address:
    - Valid Range: 0-255; Default=0
    - The unit address for this control unit.
  - Missing Interrupt Handler (mih) Minutes
    - Valid Range: 0-60; Default=0

    This specifies the missing interrupt handler value to be used in the read device characteristics CCW for the mainframe. If set to 0, the mainframe setting is used.

Outputs: None.

Notes:

- To achieve maximum throughput on the HSM, the TCP/IP and FICON interfaces need to be driven with multiple connections (or threads). Optimum performance is normally achieved with 4 - 8 connections (depending on the HSM performance model and the commands being processed), although for FICON on the 1500 tps model the performance improves right up to the maximum of 16 connections. Running with only a single thread can significantly reduce the throughput of the HSM, and means that you will not be able to reach the rated throughput for the machine.
- A USB port which has been configured for printer connection cannot be used for Asynchronous Host Communications.
- It is recommended that the Management Ethernet Port, Host 1 Ethernet Port, and Host 2 Ethernet Port are all on different IP subnets from each other.
- Where dual Ethernet host ports are in use, 2 different IP addresses at the Host computer must be used to drive the 2 ports on the HSM.
- The ROUTE Console command can be used to set up static routes from the HSM's Host ports to a Host IP address on a different subnet from the HSM.
- The use of TLS is supported from v2.2a of payShield 9000:
    - TLS traffic can be supported at the same time as non-TLS traffic.
    - The specified number of connections are shared between TLS and non-TLS traffic.
    - The HSM can be forced to accept only TLS traffic by setting the *UDP* and *TCP* options to "*N*".
    - From v3.3a, support for SSL v3, TLS v1.0, TLS v1.1 has been removed. Only TLS v1.2 is supported.

  For Ethernet communications (not protected by TLS), a Well-Known Port Address is defined (default value 1500). See Chapter 1 of the *payShield 9000 Host Command Reference Manual* for information on how port addresses can be used to select the LMK to be used with Host Commands.

  If TLS is enabled, a Well-Known Port Address is also required (default value 2500). This works in the same way as the Well-Known Port Address for non-TLS traffic.
- The facility to apply ACLs (Access Control Lists) to Host Ports is available in software versions 2.3a and later.
- Host ports can be configured from software v2.3a to get IP addresses from a DHCP server and to support the use of network names.
- When upgrading from a version of payShield 9000 software that does not support Default Gateways (i.e. versions up to 1.3) to a version that does support Default Gateways (i.e. versions 1.4 onwards), a default value for the Default Gateway IP address will be provided by the software. If the IP address for the port that was previously set up was A.B.C.D, then the default value of the Default Gateway IP address will be A.B.C.1.

Errors: None.

Example 1: *In this example, Ethernet communications using TCP/IP and TLS are selected – all types of traffic are allowed. The IP addresses are set up as static, manually-entered addresses. Access Control Lists are to be used, and will be set up using the CONFIGACL console command. Secure state is required to change TLS or ACL settings.*

```
Secure> CH <Return>

Please make a selection.  The current setting is in parentheses.
Message header length [1-255] (4): <Return>
Disable host connections when no LMKs are installed? [Y/N] (N):
<Return>
Host interface [[A]sync, [E]thernet, [F]icon] (E): <Return>
Enter Well-Known-Port (1500): <Return>
Enter Well-Known-TLS-Port (2500): <Return>
UDP [Y/N] (Y): <Return>
TCP [Y/N] (Y): <Return>
Enable TLS [Y/N] (N): Y <Return>
ACL Enabled [Y/N] (N): Y <Return>
Number of connections [1-64] (64): 5 <Return>
Enter TCP keep alive timeout [1-120 minutes] (120): <Return>
Number of interfaces [1/2] (2): <Return>

Interface Number 1:

IP Configuration Method? [D]HCP or [S]tatic (DHCP): S <Return>
Enter IP Address (192.168.200.36):192.168.200.100 <Return>
Enter subnet mask (255.255.255.0): <Return>
Enter Default Gateway Address (192.168.200.3): <Return>

Enter speed setting for this port:

    SPEED OPTIONS:
0   Autoselect
1   10BaseT half-duplex
2   10BaseT full-duplex
3   100BaseTX half-duplex
4   100BaseTX full-duplex
5   1000BaseT half-duplex
6   1000BaseT full-duplex

Speed setting (4): 6 <Return>

Interface Number 2:

IP Configuration Method? [D]HCP or [S]tatic (DHCP): S <Return>
Enter IP Address (192.168.202.110): <Return>
Enter subnet mask (255.255.255.0): <Return>
Enter Default Gateway Address (192.168.202.3): <Return>

Enter speed setting for this port:

    SPEED OPTIONS:
0   Autoselect
1   10BaseT half-duplex
2   10BaseT full-duplex
3   100BaseTX half-duplex
4   100BaseTX full-duplex
5   1000BaseT half-duplex
6   1000BaseT full-duplex

Speed setting (4): 6 <Return>

Save HOST settings to smart card? [Y/N]: N <Return>

Secure>
```

Example 2:

In this example, Ethernet communications using TLS is enabled - but UDP, and unprotected TCP are not allowed (i.e. all traffic must be protected using TLS). The IP addresses are set up as dynamic addresses to be obtained from a DHCP server. Access Control Lists are not being used. Only one host port is being configured. Secure state is required to change TLS or ACL settings.

```
Secure> CH <Return>

Please make a selection.  The current setting is in parentheses.
Message header length [1-255] (4): <Return>
Disable host connections when no LMKs are installed? [Y/N] (N):
<Return>
Host interface [[A]sync, [E]thernet, [F]icon] (E): <Return>
Enter Well-Known-Port (1500): <Return>
Enter Well-Known-TLS-Port (2500): <Return>
UDP [Y/N] (Y): N <Return>
TCP [Y/N] (Y): N <Return>
Enable TLS [Y/N] (Y): Y <Return>
ACL Enabled [Y/N] (N): N <Return>
Number of connections [1-64] (64): 5 <Return>
Enter TCP keep alive timeout [1-120 minutes] (120): <Return>
Number of interfaces [1/2] (2): 1 <Return>

Interface Number 1:

IP Configuration Method? [D]HCP or [S]tatic (static): D <Return>
Network Name (A4665275320Q-host1): HSM1-Host-1 <Return>

Enter speed setting for this port:

    SPEED OPTIONS:
0   Autoselect
1   10BaseT half-duplex
2   10BaseT full-duplex
3   100BaseTX half-duplex
4   100BaseTX full-duplex
5   1000BaseT half-duplex
6   1000BaseT full-duplex

Speed setting (4): 6 <Return>

Save HOST settings to smartcard? [Y/N]: N <Return>

Secure>
```

Example 3: *In this example, transparent asynchronous communications is enabled and the message header length is set to 6 characters. The Host baud is changed to 115200 bps and the word format is set to 8 data bits, no parity and 1 stop bit.*

```
Offline> CH <Return>
Please make a selection.  The current setting is in
parentheses.
Message header length (1-255): 6 <Return>
Disable host connections when no LMKs are installed? [Y/N] (N):
<Return>
Host interface [[A]sync, [E]thernet, [F]icon] (E): A <Return>
Transparent mode (Y/N): Y <Return>
* No interface device configured *

The following possible asynchronous interface devices were
found in the system:
 1. USB-Serial Controller by Prolific Technology Inc. located
at Rear 3
Your selection (enter for no change): 1 <Return>
You must configure the serial parameters for this device:

   BAUD RATES
1.   1200
2.   2400
3.   4800
4.   9600 (current value)
5.  19200
6.  38400
7.  57600
8. 115200
Device baud rate (enter for no change):  8 <Return>

   DATA BITS
1. 5
2. 6
3. 7
4. 8 (current value)
Device data bits (enter for no change): <Return>

   STOP BITS
1. 1 (current value)
2. 2
Device stop bits (enter for no change): <Return>

   PARITY
1. none (current value)
2.  odd
3. even
Device parity (enter for no change): <Return>

   FLOW CONTROL
1. none (current value)
2. hardware
3. software
Host flow control (enter for no change): <Return>
Save HOST settings to smartcard? [Y/N]: Y <Return>
Insert card and press ENTER: <Return>
HOST settings saved to the smartcard.

Offline>
```

Example 4: *In this example, FICON communications is selected.*

```
Secure> CH <Return>

Please make a selection.  The current setting is in parentheses.
Message header length [1-255] (4): 4 <Return>
Disable host connections when no LMKs are installed? [Y/N] (N):
<Return>
Host interface [[A]sync, [E]thernet, [F]icon] (E): F <Return>
Control Unit Image [0-255] (0): <Return>
Unit address [0-255] (0): <Return>
Missing Interrupt Handler (mih) Minutes [0-60] (0): <Return>
Save HOST settings to smart card? [Y/N]: N <Return>

Secure>
```

View Host Port Configuration

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QH**

Function: To display details of the Host port configuration of the HSM.

Authorization: This command does not require any authorization.

Inputs: None.

Outputs: For all systems:
- The message header length. This is the number of characters at the front of each command from the Host to the HSM (after the STX character). The HSM returns the message header in the response.
- The protocol used.
- Whether to disable the processing of host commands when no LMKs are installed.

For an asynchronous system:
- The Host baud rate.
- The Host word format.
- The response delay. This is the delay before the HSM responds to the Host. It allows use of half-duplex Host communications that require a defined delay between the transmission of a command and the response from the HSM.

For an Ethernet system:
- The Well-Known Port. This is the publicized TCP Port address of the HSM.
- Transport method: TCP or UDP.
- Number of TCP connections. Each host interface supports this number of connections.
- The IP address for each host interface, and how they are derived. This is the IP address of the HSM in the system.
- Subnet mask for each host interface. This is the subnet mask of the attached TCP/IP network. It is recommended that the Management Ethernet Port, Host 1 Ethernet Port, and Host 2 Ethernet Port are all on different IP subnets from each other.
- The port speed for each host interface.
- Whether Secure Host Communications is being used.
- Whether ACLs are being used.

Errors: None.

Example 1: *In this example, Ethernet communications using TCP/IP and TLS are selected – all types of traffic are allowed. The IP addresses are set up as static, manually-entered addresses. Access Control Lists are to be used, and will be set up using the CONFIGACL console command.*

```
Online> QH <Return>

Message header length: 04
Disable host connections when no LMKs are installed: No
Protocol: Ethernet
Well-Known-Port: 01500
Well-Known-TLS-Port: 02500
Transport: UDP TCP TLS, 64 connections
TCP Keep_Alive value (minutes): 120 minutes
ACL: Enabled
Number of interfaces : (2)

Interface Number: 1
IP Configuration Method: static
IP address: 192.168.200.036
Subnet mask: 255.255.255.000
Default Gateway: 192.168.200.003
MAC address: 00:d0:fa:04:27:62
Port speed:  Ethernet autoselect (1000baseT full-duplex)

Interface Number: 2
IP Configuration Method: static
IP address: 192.168.202.110
Subnet mask: 255.255.255.0
Default Gateway: 192.168.202.3
MAC address: 00:d0:fa:04:27:63
Port speed:  Ethernet autoselect (1000baseT full-duplex)

Online>
```

Example 2: *In this example, Ethernet communications using TCP/IP and TLS are selected - but UDP, and unprotected TCP traffic is not allowed (i.e. all traffic must be TLS protected). The IP addresses are set up as dynamic addresses to be obtained from a DHCP server. Access Control Lists are not being used. Only one host port has been configured.*

```
Online> QH <Return>

Message header length: 04
Disable host connections when no LMKs are installed: No
Protocol: Ethernet
Well-Known-Port: 01500
Well-Known-TLS-Port: 02500
Transport: TLS, 64 connections
TCP Keep_Alive value (minutes): 120 minutes
ACL: Disabled
Number of interfaces : (1)

Interface Number: 1
IP Configuration Method: DHCP
Network Name: HSM1-Host-1
IP address: 192.168.200.036
Subnet mask: 255.255.255.000
Default Gateway: 192.168.200.003
MAC address: 00:d0:fa:04:3b:4a
Port speed:  Ethernet autoselect (1000baseT full-duplex)

Online>
```

Example 3: *In this example, the host interface has been configured for transparent asynchronous communications.*

```
Online> QH <Return>

Message header length: 04
Disable host connections when no LMKs are installed: No
Protocol: Transparent Asynchronous
Terminating Sequence: 03 00
Response delay (ms): 00
Interface device: USB-Serial Controller by Prolific Technology
Inc. located at Rear 2 (ready)
Baud: 19200
Word format: 8 bits, 1 stop bit, no parity
Flow control: none

Online>
```

Example 4: *In this example, the host interface has been configured for FICON communications.*

```
Online> QH <Return>

Message header length: 04
Disable host connections when no LMKs are installed: No
Protocol: FICON
Control Unit Image: 0
Control Unit Address: 0
Missing Interrupt Handler (mih): 0 minutes

Online>
```

Host Port Access Control list (ACL) Configuration

Variant ☑	Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**	

Command: **CONFIGACL**

Function: To display and amend the Access Control Lists (ACLs) for the HSM's host ports. When ACL checking is enabled using the CH console command, traffic from hosts is accepted only where the host's IP address is included in one of the ACL entries set up using this command.

Authorization: This command does not require any authorization.
The HSM must be in Secure state.

Inputs:
- The user can view/add/delete entries. Entries cannot be amended.
- Each of the 2 host ports has its own ACL set.
- Entries can be of the following types:
  - A single IP address
  - An IP address range
  - An IP address mask
- Multiple types of entry can co-exist.
- Multiple entries of each type are allowed.
- The IP addresses in an entry can overlap with IP addresses in other entries.

Outputs:
- Confirmations and errors only.

Errors:
- IP address formats are validated.

Notes:
- This command sets up the IP addresses and ranges that will be used when checking traffic against the ACL, but the use of ACLs must be enabled in the CH console command before the ACLs configured in this command are applied.
- If the CH console command enables ACL checking but no ACL entries have been configured using CONFIGACL, then all host traffic will be blocked.
- ACLs apply only to Ethernet (including TLS) host traffic. They have no effect when asynchronous or FICON host communications are being used.

Example 1: *In this example, only one host interface has been configured in the CH command. There are no existing ACL entries. The user sets up a single address ACL entry, then adds a mask ACL entry, then adds a range ACL entry, and finally deletes the single address ACL entry.*

```
Secure> CONFIGACL <Return>

Access control list for Interface 1:
Single:
        None
Range:
        None
Mask:
        None

Add/Delete/Quit [A/D/Q]: A <Return>

Type - Single/Range/Mask [S/R/M]: S <Return>

IP Address: 10.10.41.10 <Return>

Access control list for Interface 1:
Single:
        1)  10.10.41.10
Range:
        None
Mask:
        None

Add/Delete/Quit [A/D/Q]: A <Return>

Type - Single/Range/Mask [S/R/M]: M <Return>

Base IP Address: 10.10.40.0 <Return>

Mask: 255.255.255.0 <Return>

Access control list for Interface 1:
Single:
        1)  10.10.41.10
Range:
        None
Mask:
        2)  10.10.40.0 to 10.10.40.255 (Mask:255.255.255.0)

Add/Delete/Quit [A/D/Q]: A <Return>

Type - Single/Range/Mask [S/R/M]: R <Return>

From IP Address: 192.168.0.0 <Return>

To IP Address: 192.168.0.92 <Return>

Access control list for Interface 1:
Single:
        1)  10.10.41.10
Range:
        2)  192.168.0.0 to 192.168.0.92
Mask:
```

```
        3)  10.10.40.0 to 10.10.40.255 (Mask:255.255.255.0)

Add/Delete/Quit [A/D/Q]: D <Return>
Entry to delete [1/3]: 1 <Return>

Access control list for Interface 1:
Single:
        None
Range:
        1)  192.168.0.0 to 192.168.0.92
Mask:
        2)  10.10.40.0 to 10.10.40.255 (Mask:255.255.255.0)

Add/Delete/Quit [A/D/Q]: Q <Return>

Secure>
```

Example 2: *In this example, both host interfaces have been configured in the CH command. The user simply views the existing ACL for host interface 2, and then exits..*

```
Secure> CONFIGACL <Return>

Interface 1: 10.10.100.216

Interface 2: 10.10.101.216

Select Interface [1/2]: 2 <Return>

Access control list for Interface 2:

Single:

        1)  10.10.40.22

        2)  10.10.40.23

        3)  10.10.40.23

Range:

        4)  10.10.40.200 to 10.10.40.220

Mask:

        None

WARNING: Duplicate - Single: Entries 2 and 3

Add/Delete/Quit [A/D/Q]: Q <Return>

Secure>
```

Configure Printer Port

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CP**

Function: To select and configure a connection to a printer attached to the HSM via a USB port. The HSM is compatible with most printers via its USB interfaces:
- A serial printer may be connected using a USB-to-serial converter cable available from Thales
- A parallel printer may be connected using a USB-to-parallel converter cable available from Thales

The new settings come into effect immediately after the command has completed.

Authorization: This command does not require any authorization.

Inputs:
- CR/LF order (standard or reversed): Y or N
- Selected printer connection.
- Setup Parameters, dependent on printer type.
- Whether to print a test page.

Outputs:
- Test page.

Errors: None.

Notes: A printer must be connected to the HSM before the CP command is invoked.
If a USB port is configured for a printer, it cannot subsequently be used for other purposes such as Console connection or Asynchronous Host Communications.

Example 1: *This example demonstrates the configuration of a printer attached to the HSM via a USB-to-serial cable.*

```
Offline> CP <Return>

Reverse the <LF><CR> order? [Y/N]: N <Return>

The following possible printer devices were found in the
system:
    0.  No printer
    1.  USB-Serial Controller by Prolific Technology Inc. located
        at Rear 4 (current selection)
Your selection (ENTER for no change): 1 <Return>
You must configure the serial parameters for this device:

   BAUD RATES
1.   1200
2.   2400
3.   4800
4.   9600 (current value)
5.  19200
6.  38400
7.  57600
8. 115200
Device baud rate (ENTER for no change):  8 <Return>

   DATA BITS
1. 5
2. 6
3. 7
4. 8 (current value)
Device data bits (ENTER for no change): <Return>

   STOP BITS
1. 1 (current value)
2. 2
Device stop bits (ENTER for no change): <Return>

   PARITY
1. none (current value)
2.  odd
3. even
Device parity (ENTER for no change): <Return>

   Flow Control
1. none
2. software (current value)
3. hardware
Printer flow_ctl (ENTER for no change): <Return>

   Printer Offline Control
1. none (current value)
2. RTS
3. DTR
Printer offline control (ENTER for no change): <Return>
Timeout [in milliseconds, min=1000, max=86400000] (12000):
<Return>
Delay [in milliseconds, min = 0, max=7200000] (0): <Return>

Print test page? [Y/N]: Y <Return>

Offline>
```

Example 2: *This example demonstrates the configuration of a printer attached to the HSM via a USB-to-parallel cable.*

```
Offline> CP <Return>

Reverse the <LF><CR> order? [Y/N]: N <Return>

The following possible printer devices were found in the
system:
    0.  No printer
    1.  USB2.0-Print  by  located at Rear 1
Your selection (enter for no change): 1 <Return>
Timeout [in milliseconds, min=1000, max=86400000] (1000):
1000<Return>
Delay [in milliseconds, min = 0, max=7200000] (0): <Return>
Print test page? [Y/N]: Y <Return>

Offline>
```

Example 3: *This example demonstrates the configuration of a native USB printer attached to the HSM.*

```
Offline> CP <Return>

Reverse the <LF><CR> order? [Y/N]: N <Return>

The following possible printer devices were found in the
system:
   0. No printer
   1. USB Printer by EPSON located at Front left (current
selection)
Your selection (ENTER for no change): 1 <Return>
Timeout [in milliseconds, min=1000, max=86400000] (1000):
1000<Return>
Delay [in milliseconds, min = 0, max=7200000] (0): <Return>
Print test page? [Y/N]: n<Return>

Offline>
```

View Printer Port Configuration

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QP**

Function: To display details of the HSM's printer configuration.

Authorization: This command does not require any authorization.

Inputs:
- Print test page: Y or N

Outputs:
- <LF><CR> order revered: YES or NO.
- Validation of current printer configuration.
- The serial configuration settings (serial printer only).

Errors: None.

Example 1: *This example demonstrates viewing the configuration of a printer attached to the HSM via a USB-to-serial cable.*

```
Online> QP <Return>

The configured printer, USB-Serial Controller by Prolific
Technology Inc. located at Rear 1, has been validated
   BAUD RATE:       38400
   DATA BITS:       8
   STOP BITS:       1
   PARITY:          none
   Flow Control:    XON/XOFF
   Offline Control: none
<LF><CR> order reversed: NO
Timeout:  12000 milliseconds
Delay:    0 milliseconds
Print test page? [Y/N]: N <Return>

Online>
```

Example 2: *This example demonstrates viewing the configuration of a printer attached to the HSM via a USB-to-parallel cable.*

```
Online> QP <Return>

The configured printer, USB2.0-Print  by  located at Rear 1, has
been validated.
<LF><CR> order reversed: NO
Timeout:  12000 milliseconds
Delay:    0 milliseconds
Print test page? [Y/N]: N <Return>

Online>
```

Configure Management Port

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CM**

Function: To configure the Management port, which is an Ethernet port used only for management of the HSM. If connection to the host is via Ethernet then the Ethernet host port is used for that purpose. The Management Ethernet port is used to update the HSM's internal software, updating licensing information, and for enabling management of a HSM via the payShield Manager.

The new settings come into effect a few seconds after the command has completed.

It is recommended that the Management Ethernet Port, the Auxiliary Ethernet Port and the Host Ethernet Ports are all on different IP subnets.

Authorization: The HSM must be in the offline or secure state to run this command.

Inputs:
- Whether IP address is manually or automatically derived.
  - If manually derived, then the address details must be entered.
  - If using DHCP, then a network name may be entered.
- Ethernet speed setting.
- Enable (local or remote) payShield Manager connection?

Outputs: None.

Errors: None.

Notes:
- When upgrading from a version of payShield 9000 software that does not support Default Gateways (i.e. versions up to 1.3) to a version that does support Default Gateways (i.e. versions 1.4 onwards), a default value for the Default Gateway IP address will be provided by the software. If the IP address for the port that was previously set up was A.B.C.D, then the default value of the Default Gateway IP address will be A.B.C.1.

Example 1: *In this example, the management port has its IP address set up manually.*

```
Offline> CM <Return>

Management Ethernet Port:
IP Configuration Method? [D]HCP or [S]tatic (DHCP): s<Return>
Enter IP address (192.168.100.200): 192.168.200.90 <Return>
Enter subnet mask (255.255.255.0): <Return>
Enter Default Gateway Address (192.168.200.1): <Return>

  Enter speed setting for this port:

      SPEED OPTIONS:
  0   Autoselect
```

```
  1   10BaseT half-duplex
  2   10BaseT full-duplex
  3   100BaseTX half-duplex
  4   100BaseTX full-duplex
  5   1000BaseT half-duplex
  6   1000BaseT full-duplex

  Speed setting (4): 6 <Return>


Enable payShield Manager connection:
  Enable or Disabled? (E): D <Return>

Would you like to apply the changes now? [Y/N]: Y <Return>

Offline>
```

Example 2: *In this example, the management port has its IP address set up automatically by a DHCP server.*

```
Secure> CM <Return>

Management Ethernet Port:
IP Configuration Method? [D]HCP or [S]tatic (DHCP): <Return>
Network Name (B4665271226O-mgmt): HSM-Mngmnt <Return>

Enter speed setting for this port:

    SPEED OPTIONS:
0   Autoselect
1   10BaseT half-duplex
2   10BaseT full-duplex
3   100BaseTX half-duplex
4   100BaseTX full-duplex
5   1000BaseT half-duplex
6   1000BaseT full-duplex

Speed setting (0): <Return>

Enable payShield Manager connection: <Return>
  Enable or Disabled? (E): <Return>

Would you like to apply the changes now? [Y/N]: Y <Return>

Secure>
```

View Management Port Configuration

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QM**

Function: To display details of the Management port parameters.

Authorization: This command does not require any authorization.

Inputs: None.

Outputs:
- IP address.
- Subnet mask.
- Default gateway.
- MAC address.
- Ethernet speed setting.
- Enable (local or remote) payShield Manager connection?

Errors: None.

Example 1:

```
Online> QM <Return>

Management Ethernet Port:
IP Configuration Method: static
IP address: 192.168.200.90
Subnet mask: 255.255.255.0
Default Gateway: 192.168.200.1
MAC address: 00:d0:fa:04:27:64
Port speed:  Ethernet 1000baseT full-duplex

payShield Manager connection: Disabled

Online>
```

Example 2: *In this example, the management port has its IP address set up automatically by a DHCP server.*

```
Online> QM <Return>

Management Ethernet port:
IP Configuration Method: DHCP
Network Name: HSM-Mngmnt
IP address: 192.168.1.3
Subnet mask: 255.255.255.0
Default Gateway: 192.168.1.1
MAC address: 00:d0:fa:04:27:64
Port speed:  Ethernet autoselect (100baseTX full-duplex)

payShield Manager connection: Enabled

Online>
```

Configure Auxiliary Port

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CA**

Function: To configure the Auxiliary port, which is an Ethernet port currently used only for transmission of SNMP traffic from the HSM.

The new settings come into effect a few seconds after the command has completed.

It is recommended that the Auxiliary Ethernet Port, the Management Ethernet Port and the Host Ethernet Ports are all on different IP subnets.

Authorization: The HSM must be in the offline or secure state to run this command.

Inputs:
- Whether IP address is manually or automatically derived.
  - If manually derived, then the address details must be entered.
  - If using DHCP, then a network name may be entered.
- Ethernet speed setting.

Outputs: None.

Errors: None.

Example 1: *In this example, the auxiliary port has its IP address set up manually.*

```
Offline> CA <Return>

Auxiliary Ethernet Port:
IP Configuration Method? [D]HCP or [S]tatic (DHCP): S <Return>
Enter IP address (192.168.300.200): 192.168.300.90 <Return>
Enter subnet mask (255.255.255.0): <Return>
Enter Default Gateway Address (192.168.300.1): <Return>

  Enter speed setting for this port:

      SPEED OPTIONS:
  0   Autoselect
  1   10BaseT half-duplex
  2   10BaseT full-duplex
  3   100BaseTX half-duplex
  4   100BaseTX full-duplex
  5   1000BaseT half-duplex
  6   1000BaseT full-duplex

  Speed setting (4): 6 <Return>

Would you like to apply the changes now? [Y/N]: Y <Return>

Offline>
```

Example 2: *In this example, the auxiliary port has its IP address set up automatically by a DHCP server.*

```
Secure> CA <Return>

Auxiliary Ethernet Port:
IP Configuration Method? [D]HCP or [S]tatic (DHCP): <Return>
Network Name (B46652712260-Aux): HSM-Aux <Return>

Enter speed setting for this port:

    SPEED OPTIONS:
0   Autoselect
1   10BaseT half-duplex
2   10BaseT full-duplex
3   100BaseTX half-duplex
4   100BaseTX full-duplex
5   1000BaseT half-duplex
6   1000BaseT full-duplex

Speed setting (0): <Return>

Would you like to apply the changes now? [Y/N]: Y <Return>

Secure>
```

View Auxiliary Port Configuration

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QA**

Function: To display details of the Auxiliary port parameters.

Authorization: This command does not require any authorization.

Inputs: None.

Outputs:
- IP address.
- Subnet mask.
- Default gateway.
- MAC address.
- Ethernet speed setting.

Errors: None.

Example 1:

```
Online> QA <Return>

Auxiliary Ethernet Port:
IP Configuration Method: static
IP address: 192.168.300.90
Subnet mask: 255.255.255.0
Default Gateway: 192.168.300.1
MAC address: 00:d0:fa:04:43:33
Port speed:  Ethernet 1000baseT full-duplex

Online>
```

Example 2: *In this example, the auxiliary port has its IP address set up automatically by a DHCP server.*

```
Online> QA <Return>

Auxiliary ethernet port:
IP Configuration Method: DHCP
Network Name: HSM-Aux
IP address: 192.168.1.3
Subnet mask: 255.255.255.0
Default Gateway: 192.168.1.1
MAC address: 00:d0:fa:04:43:33
Port speed:  Ethernet autoselect (100baseTX full-duplex)

Online>
```

Configure Alarms

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **CL**

Function: To enable or disable the motion alarm. The HSM alarm circuitry typically needs to be turned off if the HSM is to be moved. The alarm should be turned on while the HSM is in service or being stored. The alarm setting can optionally be saved to a smartcard. (*In software versions up to v2.1, the temperature alarm could also be turned on or off. From version 2.2 onwards the temperature alarm is permanently enabled.*)

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Motion alarm status: Low, Medium, High or Off.
- Save settings to smartcard: Yes or No.

Outputs: None.

Errors:
- Card not formatted to save/retrieve HSM settings. Attempt with another card? [Y/N]

Example 1: *In this example, the setting is being made to a **less** secure setting.*

```
Secure> CL <Return>

Please make a selection.  The current setting is in
parentheses.
Motion alarm [Low/Med/High/ofF] (MED): F<Return>
LMKs must be erased before proceeding.
Erase LMKs? [Y/N]: Y<Return>
Save ALARM settings to smart card? [Y/N]: N<Return>

Secure>
```

Example 2: *In this example, the setting is being made to a **more** secure setting.*

```
Secure> CL <Return>

Please make a selection.  The current setting is in
parentheses.
Motion alarm [Low/Med/High/ofF] (OFF): H<Return>
Save ALARM settings to smart card? [Y/N]: n<Return>

Secure>
```

View Alarm Configuration

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **QL**

Function: To display details of the alarm configuration of the HSM.

Authorization: This command does not require any authorization.

Inputs: None.

Outputs:
- The Temperature alarm status.
- The Motion alarm status.

Errors: None.

Example:

```
Online> QL <Return>

Temperature alarm enabled

Motion alarm enabled high sensitivity

Online>
```

Add Static TCP/IP Route

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **ROUTE**

Function: To configure static routes for routing TCP/IP traffic.

Authorization: The HSM does not require any authorization to run this command.

Inputs:

Syntax:
[-f] [-n] [-q] [-v] command { [[modifiers] args] }

Options:

-I *iface* The interface that ROUTE command is to be applied to. ***This is a required parameter***.

iface Value	HSM Port
h1	Host Port #1
h2	Host Port #2
m	Management Port

-f Remove all routes (as per flush). If used in conjunction with the add, change, delete, or get commands, route removes the routes before performing the command.

-n Don't print host and network names symbolically when reporting actions. (The process of translating between symbolic names and numerical equivalents can be quite time consuming, and may require correct operation of the network; thus it may be expedient to forgo this, especially when attempting to repair networking operations.)

-q Be quiet: suppress all output.

-v Be verbose: print additional details.

Command Options:

Add a route:
add [-net|-host] *destination gateway*

Change aspects of a route (such as its gateway):
change [-net|-host] *destination gateway*

Delete a specific route:
delete [-net|-host] *destination gateway*

(INET and INET6 only) Flush the routing tables of all gateway entries. If you want to delete only routes having destinations with addresses in a specified family, specify INET or INET6 as the *family* variable.
flush [family]

Look up and display the route for a destination:
get [-net|-host] *destination gateway*

Report changes to the routing information on a continuing basis:
monitor

Display route table (similar to netstat -r):
show

Specify the netmask to use when adding a network route:

netmask XXX.XXX.XXX.XXX

destination The destination host or network.

Gateway The next-hop gateway that packets should be addressed to.

If the keyword, default, or the network address, 0.0.0.0, is specified, then all packets sent to a remote network that's not defined in the routing tables, are sent to the specified gateway. Routes to a particular host are distinguished from those to a network by interpreting the Internet address associated with destination. Specifying the optional keywords -net and -host force the destination to be interpreted as a network or a host, respectively.
If the destination has a "local address part" of INADDR_ANY, or if the destination is the symbolic name of a network, then the route is assumed to be to a network; otherwise, the route is assumed to be to a host. For example:

This destination:	Is interpreted as:
128.32	-host 128.0.0.32
128.32.130	-host 128.32.0.130
-net 128.32	128.32.0.0
-net 128.32.130	128.32.130.0.

If the route is via an interface rather than via a gateway, you should specify the -interface modifier; the gateway given is the address of this host on the common network, indicating the interface to be used for transmission.

Option questions.

When certain command options are used, confirmation relating to route persistence is asked for.

Command Option	Requested confirmation
Add	Make route persistent? [Y/N]
Delete	Remove persistent route? [Y/N]
flush	Remove all persistent routes? [Y/N]

Modifiers

You can use the optional -netmask modifier to specify an additional address parameter that's interpreted as a network mask. You can use this like an OSI ESIS redirect with the netmask option, or to manually add subnet routes with netmasks different from that of the implied network interface (as would otherwise be communicated using the OSPF or ISIS routing protocols). After -netmask, enter the address parameter you want interpreted as the network mask.
You can override the implicit network mask generated in the INET case by placing this option after the destination parameter. Similarly, you can use the -prefixlen modifier for IPv6.
The optional modifiers:

- -expire
- -hopcount
- -mtu
- -recvpipe
- -rtt
- -rttvar
- -sendpipe

- -ssthresh

provide initial values to metrics maintained in the routing entry. To lock any of these modifiers, precede the modifier with the -lock meta-modifier; you can also specify the -lockrest meta-modifier to lock all ensuing metrics.

Diagnostics

add [host | network] %s: gateway %s flags %x
: The specified route is being added to the tables. The values printed are from the routing table entry supplied in the ioctl() call. If the gateway address used isn't the primary address of the gateway—the first one returned by gethostname() — the gateway address is printed numerically as well as symbolically.

delete [host &| network] %s: gateway %s flags %x
: As above, but when deleting an entry.

%s %s done
: A routing table entry is being deleted by the flush command.

Network is unreachable
: An attempt to add a route failed because the gateway listed wasn't on a directly connected network. The next-hop gateway must be given.

not in table
: A delete operation was attempted for an entry not present in the tables.

routing table overflow
: An add operation was attempted, but the system was low on resources and couldn't allocate memory to create the new entry.

Permission denied
The attempted operation is privileged. Only root may modify the routing tables. These privileges are enforced by the kernel.

Outputs: Text messages as appropriate.

Notes: When upgrading from a version of payShield 9000 software that does not support Default Gateways (i.e. versions up to 1.3) to a version that does support Default Gateways (i.e. versions 1.4 onwards), any existing routes previously set up using the ROUTE command will be deleted. If it is required to continue using static routes (despite the availability of Default Gateways), these should be re-entered using the ROUTE command.

Example:

```
Offline> ROUTE -I h1 add 20.20.20.0/24  10.10.10.1 <Return>

add net 20.20.20.0: gateway 10.10.10.1
Make route persistent? [Y/N]:

Offline>
```

View/Change Instantaneous Utilization Period

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **UTILCFG**

Function: To display the current setting of the period over which utilization statistics is to be collected when Instantaneous Utilization Data is requested. This command also allows the setting to be amended (in Offline/Secure states only).

Authorization: The HSM does not require any authorization to run this command.

Inputs: Amended value for Instantaneous Utilization Period. (It is suggested that the period should not be set to less than 10 seconds, as data collected over very short periods will not be indicative of actual activity.)

Outputs: Text messages as in example below.
Note that resetting of the value requires the HSM to be in Offline or Secure state.

Example:

```
Online> UTILCFG <Return>

Measurement period for instantaneous statistics is 60 seconds

Online>

…

Offline> UTILCFG <Return>

Measurement period for instantaneous statistics is 60 seconds

Change? [Y/N]: Y <Return>
Enter new value in seconds (1-60): 10 <Return>

Offline>
```

Suspend/Resume Collection of Utilization Data

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **UTILENABLE**

Function: To suspend or resume the collection of Utilization Data and the incrementing of the count of seconds over which the data is being collected. This allows data collection to be suspended if, for example, the HSM is taken out of service or temporarily re-purposed. It ensures that tps rates are not diluted by averaging command volumes over the total elapsed time, but only over the time that data is being collected

Authorization: The HSM does not require any authorization to run this command.

Inputs: Whether to change the current state.

Outputs: Text messages as in example below.

Notes: Following a software load, collection of Utilization Data will be suspended.
Data collection is automatically suspended while the HSM is not online.

Example:

```
Offline> UTILENABLE <Return>

Utilization statistics gathering is currently turned ON.
Suspend? [Y/N] Y <Return>

Offline> UTILENABLE <Return>

Utilization statistics gathering is currently turned OFF.
Resume? [Y/N] Y <Return>

Offline>
```

Suspend/Resume Collection of Health Check Counts

Variant ☑		Key Block ☑
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **HEALTHENABLE**

Function: To suspend or resume the collection of Health Check counts. This allows data collection to be suspended if, for example, data is not required.

Authorization: The HSM does not require any authorization to run this command.

Inputs: Whether to change the current state.

Outputs: Text messages as in example below.

Notes: Following a software load, the collection of Health Check counts will be disabled.

Example:

```
Offline> HEALTHENABLE <Return>

Health check statistics gathering is currently turned ON.
Suspend? [Y/N] Y <Return>

Offline> HEALTHENABLE <Return>

Health check statistics gathering is currently turned OFF.
Resume? [Y/N] Y <Return>

Offline>
```

View SNMP Settings

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **SNMP**

Function: To display the current SNMP settings, and to enable/disable provision of Utilisation and Health Check data via SNMP.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- Whether to Enable/Disable provision of Utilisation and Health Check data via SNMP.
- Which Ethernet port to use for SNMP traffic.

Outputs: Text messages as in example below. Information on Communities relates to SNMP versions 1 and 2; information on Users relates to SNMP version 3.

Notes: The HSM is delivered with no Users or Communities set up.

Example:

```
Online> SNMP <Return>

V1/V2 Communities:
Read=public
Read=public
Read=private

V3 Users:
public: Authentication=none, Privacy=none
shades: Authentication=SHA, Privacy=DES
none: Authentication=none, Privacy=none
md5: Authentication=MD5, Privacy=none


SNMP is currently enabled
Disable? [Y/N]: n <Return>


SNMP is currently enabled on Host Port 2
Change SNMP port? [Y/N]: y <Return>
 0. Host Port 1
 1. Host Port 2
 2. Management Port


SNMP port [0-2] (ENTER for no change): 1 <Return>

Online>
```

Add a SNMP Community or User

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SNMPADD**

Function: Add an SNMP Community (for SNMP versions 1 or 2) or User (for SNMP version 3).

Authorization:
- The HSM does not require any authorization to run this command.
- The HSM must be in Secure state.

Inputs:
- For an SNMP Community – the community name and security name (i.e. the community read strings).
- For an SNMP User – the user name, authentication algorithm, and privacy algorithm.

Outputs: Text messages as in example below.

Notes: The HSM is delivered with no Users or Communities set up.

Example:

```
Secure> SNMPADD <Return>

Add Community or User? [C/U]: C <Return>
Enter read string (Less than 20 characters): PUBLIC <Return>

The following entry will be added to the table
    'Read=public'.
Confirm? [Y/N]: Y <Return>

Community added successfully

Enter additional users or communities? [Y/N]: Y <Return>

Add Community or User? [C/U]: U <Return>
Enter user name: SHADES <Return>
Authentication algorithm [[N]one, [M]D5, [S]HA]: S <Return>
Enter authentication password: SHA <Return>
Privacy algorithm [[N]one, [D]ES]: D <Return>
Enter privacy password: DES <Return>
The following entry will be added to the table:
    'shades: Authentication=SHA,Privacy=DES'.
Confirm? [Y/N]: Y <Return>

User added successfully

Enter additional users or communities? [Y/N]: N <Return>

Save and exit? [Y/N]: Y <Return>

SNMP configuration updated
Secure>
```

Delete a SNMP Community or User

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SNMPDEL**

Function: Delete an SNMP Community (for SNMP versions 1 or 2) or User (for SNMP version 3).

Authorization:
- The HSM does not require any authorization to run this command.
- The HSM must be in Secure state.

Inputs: The index of the community or user to be deleted.

Outputs: Text messages as in example below.

Notes: The HSM is delivered with no Users or Communities set up.

Example:

```
Secure> SNMPDEL <Return>

Remove Community or User? [C/U]: C <Return>
SNMP community table:
        0: Read=public
        1: Read=public
        2: Read=private

Select community to delete [0-2]: 1 <Return>

Community public/private deleted successfully

Remove additional users or communities? [Y/N]: Y <Return>

Remove Community or User? [C/U]: U <Return>
SNMP user table:
        0: User=public, Authentication=none, Privacy=none
        1: User=shades, Authentication=SHA, Privacy=DES
        2: User=none, Authentication=none, Privacy=none
        3: User=md5, Authentication=MD5, Privacy=none

Select user to delete [0-3]: 1 <Return>

User shades deleted successfully

Remove additional users or communities? [Y/N]: N <Return>

Save and exit? [Y/N]: Y <Return>

SNMP configuration updated

Secure>
```

Configure SNMP Traps

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **TRAP**

Function: To display the current SNMP Trap configuration and to enable/disable individual SNMP Traps.

Authorization: The HSM does not require any authorization to run this command.

Inputs: Whether to Enable/Disable individual trap configurations.

Outputs: Text messages as in the example below.

Notes: The HSM is delivered with no SNMP Traps configured.

Example 1:

```
Offline> TRAP <Return>

Trap table is empty, no SNMP traps are configured.

Enable? [Y/N]: Y <Return>

Offline>
```

Example 2:

```
Offline> TRAP <Return>

Entry    IP Address:Port      Version   User/Comm name
 1       192.168.100.133:162  V3        User1

Disable? [Y/N]: N <Return>

Offline>
```

Add a new SNMP Trap

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **TRAPADD**

Function: Add an SNMP Trap.

Authorization:
- Authorization is not required.
- The HSM must be in the Secure state.

Inputs: Trap configuration data & confirmation.

Outputs: Text messages as in example below.

Notes: The HSM is delivered with no SNMP traps configured.

Example 1:

```
Secure> TRAPADD <Return>

Enter IP Address: 192.168.100.133 <Return>
Enter Port: 162 <Return>
Enter version number (1-3): 3 <Return>
SNMP user table:
        0: User=User1, Authentication=SHA, Privacy=DES

Select user [0-0]: 0 <Return>

The following entry will be added to the table:
        '192.168.100.133:162, V3, User1'.
Confirm? [Y/N]: Y <Return>

Trap destination added successfully

Configure additional traps? [Y/N]: N <Return>

Save and exit? [Y/N]: Y <Return>

SNMP configuration updated

Secure>
```

Delete an SNMP Trap

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **TRAPDEL**

Function: Add an SNMP Trap.

Authorization:
- Authorization is not required.
- The HSM must be in the Secure state.

Inputs: Confirmation of deletion.

Outputs: Text messages as in example below.

Notes: The HSM is delivered with no SNMP traps configured.

Example 1:

```
Secure> TRAPDEL <Return>

SNMP Trap table:
    0: Address=192.168.100.133, Port=162, Version=3, User=User1

Select trap to delete [0-0]: 0 <Return>

Trap destination deleted successfully

Delete additional traps? [Y/N]: N <Return>

Save and exit? [Y/N]: Y <Return>

SNMP configuration updated

Secure>
```

Fraud Detection Commands

The payShield 9000 HSM provides the following console commands to support fraud detection operations:

Configure Fraud Detection

Variant ☑		Key Block ☑
Online ☒	Offline ☑	Secure ☑
Authorization: **Required** Activity: **audit.console**		

Command: **A5**

Function: To set the configuration of the HSM fraud detection function.

Authorization: If the Fraud Detection settings are to be edited, the HSM must be:
- in the offline or secure state to run this command, and
- either in the Authorized State, or the activity **audit.console** must be authorized, using the Authorizing Officer cards of the Management LMK.

Inputs:
- Whether and how to respond to Fraud Detection
- Limit on number of PIN verification failures per minute.
- Limit on number of PIN verification failures per hour.
- Limit on number of PIN attacks detected.

Outputs: None.

Errors:
- Not Authorized - the HSM is not authorized to perform this operation.
- Not Offline - the HSM must be offline to run this command.
- Invalid Entry - the value entered is invalid.

Notes:
- See the description of the Fraud Detection facility at Chapter 7 of the *payShield 9000 General Information Manual*.
- If any of the limits set by this command are exceeded, an entry will be made in the Audit Log, and console command A7 must be used to re-enable PIN verification.
- Setting the HSM reaction to Logging only and the limits to zero will result in Fraud Detection not being recorded in the Health Check data. (*The term "Logging" as used in the screen prompt refers to logging in the Health Check data, not in the Audit Log.*)

Example:

```
Offline-AUTH> A5 <Return>

HSM reaction to Exceeding Fraud Limits is : ON

The following limits are set:
PIN verifies per minute: 100
PIN verifies per hour: 1000
PIN Attack Limit: 100

HSM reaction to Exceeding Fraud Limits? ([O]n/[L]ogging only): L
<Return>

Note that logging is supported only if enabled via the
HEALTHENABLE console command (or its payShield Manager
equivalent)

Enter limit on PIN verifies per minute: 200 <Return>
Enter limit on PIN verifies per hour: 2000 <Return>
Enter PIN Attack Limit: 200 <Return>

Offline-AUTH>
```

Re-enable PIN Verification

Variant ☑		Key Block ☑
Online ☒	Offline ☑	Secure ☑
Authorization: **Required** Activity: **audit.console**		

Command: **A7**

Function: To reset the configuration of the HSM fraud detection function.

Authorization: The HSM must be in the offline state to run this command. The HSM must be either in the Authorized State, or the activity **audit.console** must be authorized, using the Authorizing Officer cards of the Management LMK.

Inputs: None.

Outputs: None.

Errors:
- Not Authorized - the HSM is not authorized to perform this operation.
- Not Offline - the HSM must be offline to run this command.
- PIN Verification is not currently disabled

Example:

```
Offline-AUTH> A7 <Return>
PIN verification has been re-enabled
Offline-AUTH>
```

Diagnostic Commands

The payShield 9000 HSM provides the following console commands to support diagnostic operations:

Diagnostic Test

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **DT**

Function: To perform diagnostic tests.
The DT command tests the following parts of the HSM:
- Battery voltage level
- Various cryptographic algorithms (DES, AES, RSA, SHA-1, etc.)
- Working memory areas
- Power Supplies
- Random Number Generator
- Real-time clock
- Smartcard reader
- Operating temperature
- Operating fan speeds
- Operating voltages

The command also initiates the Health Check Instantaneous Status report as described in Chapter 9 of the *payShield 9000 General Information Manual*.

Authorization: The HSM does not require any authorization for this command.

Inputs: Optional qualifiers to modify scope and detail of output.
Options are:

all	run all the commands (default option)
verbose	be verbose in the output
battery	run the battery diagnostics
des	run the DES diagnostics
aes	run the AES diagnostics
health	run the health check diagnostics
md5	run the MD5 KAT
mem	run the memory diagnostics
psu	run the power supply diagnostics
rng	run the random number generator diagnostics
rsa	run the RSA KAT
rtc	run the real-time clock diagnostics
scr	run the smart card reader diagnostics
sha	run the SHA KAT
temp	run the temperature diagnostics
fans	run the fans diagnostics
volt	run the voltage diagnostics

Note that the multiple options can be combined (e.g." dt temp verbose"; "dt volt rsa")
Note that whilst the command code ("dt") is not case sensitive, the options listed above are.

Outputs: Status report on each item.

Errors: None.

Notes:
- The diagnostics are run automatically on a daily basis at the time specified using the ST Console command.

Example 1:

```
Offline> DT <Return>

Battery:                OK
AES:                    OK
DES:                    OK
MD5:                    OK
Memory:                 OK
Power Supply:           OK
RNG:                    OK
RSA:                    OK
Real-Time Clock:        OK
SHA:                    OK
SCR:                    OK
Temperature:            OK
Fans:                   OK
Voltages:               OK

Health Check Status

TCP Server:                         Up
UDP Server:                         Up
Async Server:                       Not Enabled
FICON Server:                       Not Enabled
Local/Remote Manager Server:        Up
Host Ethernet Link 1:               Up
Host Ethernet Link 2:               Not Enabled
Host Async Link:                    Not Enabled
Host FICON Link:                    Not Enabled
Unit Tampered?:                     No
Fraud limits exceeded?:             No
PIN attack limit exceeded?:         No

Diagnostics complete

Offline>
```

Example 2:

```
Online>DT verbose <Return>


Battery:          OK
   Voltage: 3050 mV
   HSM will enter tamper state if voltage drops below 2500 mV

AES:              OK
DES:              OK
MD5:              OK
Memory:           OK
Power Supply:     OK
RNG:              OK
RSA:              OK
Real-Time Clock: OK
   Current Time: Fri Aug 10 09:32:27 2012

SHA:              OK
SCR:              OK
Temperature:      OK
   Inlet:               27 C (80 F)
   Internal Device 1: 31 C (87 F)
   Internal Device 2: 31 C (87 F)
   Internal Device 3: 30 C (86 F)
Fans:             OK
   Fan 1:               3950 RPM
   Fan 2:               4047 RPM
Voltages:         OK
   Expected   Actual   Deviation
    3300 mV   3313 mV     0%
   12000 mV  11925 mV     0%
    5000 mV   5023 mV     0%
    2500 mV   2507 mV     0%
    1100 mV   1100 mV     0%


Health Check Status

TCP Server:                      Up
UDP Server:                      Up
Async Server:                    Not Enabled
FICON Server:                    Not Enabled
Local/Remote Manager Server:     Up
Host Ethernet Link 1:            Up
Host Ethernet Link 2:            Not Enabled
Host Async Link:                 Not Enabled
Host FICON Link:                 Not Enabled
Unit Tampered?:                  No
Fraud limits exceeded?:          No
PIN attack limit exceeded?:      No

Diagnostics complete

Online>
```

View Software Revision Number

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **VR**

Function: To display details of the software release number, revision number and build number.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs: Software revision numbers.

Errors: None.

Notes: The software revision reported by the VR command will have one of the following forms:

- xxxx-19xx – this indicates that this software has been PCI HSM certified and that the appropriate security settings have been set (e.g. by using the CS Console command) to the required values.
- xxxx-09xx – this indicates that either:
  - this version of software is not PCI HSM certified, or
  - this version of software is PCI HSM certified but one or more of the appropriate security settings have been set (e.g. by using the CS Console command) to the required values.

For software versions 3.0 and above, all RSA operations are automatically boosted, and therefore, the RSA Booster optional license (HSM9-LIC033) is no longer in use, and will not appear in the output of the VR command.

Example 1: *Some security settings are not PCI HSM compliant. FICON host interface is installed.*

```
Online> VR <Return>

Base release: X.Xx
Revision:     XXXX-X9XX
Build Number: XXXX


PCI HSM Compliance: Some security settings are not PCI HSM
compliant


HSM Core API Version:    6.0.1


Serial Number:  C4665271228Q
Unit info:      Licenced


Host Configuration: Async,Ethernet,FICON
Licence Issue No:   1
Performance:        1500 TPS
Base Software:      Version 2
Ship Counter:       1
Crypto:             3DES,AES,RSA
LMKs Enabled:       5 LMKs


Press "Enter" to view additional information... <Return>


HSM9-LIC001 Base Software
HSM9-LIC013 5 LMKs
HSM9-LIC024 Mag Stripe Issuers
HSM9-LIC025 Mag Stripe Trans Processing
HSM9-LIC026 EMV Trans Processing
HSM9-LIC027 PIN/Key Mailer
HSM9-LIC028 Visa Cash Processing
HSM9-LIC029 Legacy Functions
HSM9-LIC030 Miscellaneous HSM8000 Base Commands License


Bootstrap Version:          1.10.2
Bootmanager Version:        1.16.12
LBC Version:                1.6
Microcontroller Version:    1.33

AGS Cryptographic Library:          1.10.C644
FIPS Validated DRBG/RNG Algorithm:  TSPP-DRBG v1.1
FIPS Validated SHA Algorithm:       TSPP-SHA v1.0
FIPS Validated HMAC Algorithm:      TSPP-HMAC v1.0
FIPS Validated TDES Algorithm:      TSPP-TDES v1.0
FIPS Validated RSA Algorithm:       TSPP-RSA v1.0
FIPS Validated AES Algorithm:       TSPP-AES v1.0
FIPS Validated CMAC Algorithm:      TSPP-CMAC v1.0

Online>
```

Example 2: *All security settings compliant with PCI HSM:*

```
Online> VR <Return>

Base release: X.Xx
Revision:     XXXX-X9XX
Build Number: XXXX


PCI HSM Compliance: Refer to the PCI web site
(https://www.pcisecuritystandards.org/approved_companies_provid
ers/approved_pin_transaction_security.php) for current
certification status of this version of payShield 9000
software.


Security settings are consistent with the requirements of PCI
HSM.


HSM Core API Version:    6.0.1


Serial Number:  C4665271228Q
Unit info:      Licenced


Host Configuration: Async,Ethernet,FICON
Licence Issue No:   1
Performance:        1500 TPS
Base Software:      Version 2
Ship Counter:       1
Crypto:             3DES,AES,RSA
LMKs Enabled:       5 LMKs


Press "Enter" to view additional information... <Return>


HSM9-LIC001 Base Software
HSM9-LIC013 5 LMKs
HSM9-LIC024 Mag Stripe Issuers
HSM9-LIC025 Mag Stripe Trans Processing
HSM9-LIC026 EMV Trans Processing
HSM9-LIC027 PIN/Key Mailer
HSM9-LIC028 Visa Cash Processing
HSM9-LIC029 Legacy Functions
HSM9-LIC030 Miscellaneous HSM8000 Base Commands License


Bootstrap Version:         1.10.2
Bootmanager Version:       1.16.12
LBC Version:               1.6
Microcontroller Version:   1.33

AGS Cryptographic Library:          1.10.C644
FIPS Validated DRBG/RNG Algorithm:  TSPP-DRBG v1.1
FIPS Validated SHA Algorithm:       TSPP-SHA v1.0
FIPS Validated HMAC Algorithm:      TSPP-HMAC v1.0
FIPS Validated TDES Algorithm:      TSPP-TDES v1.0
FIPS Validated RSA Algorithm:       TSPP-RSA v1.0
FIPS Validated AES Algorithm:       TSPP-AES v1.0
FIPS Validated CMAC Algorithm:      TSPP-CMAC v1.0

Online>
```

Example 3: *Software which has not been PCI HSM certified. TLS protection of host communications is enabled.*

```
Online> VR <Return>

Base release: X.XX
Revision:     XXXX-09XX
Build Number: XXXX

HSM Core API Version:    6.0.1

Serial Number:  A46652754970
Unit info:      Licenced

Host Configuration: Async,Ethernet,(optional) TLS
Licence Issue No:   1
Performance:        1500 TPS
Base Software:      Version 2
Ship Counter:       1
Crypto:             3DES,AES,RSA
LMKs Enabled:       5 LMKs

Press "Enter" to view additional information... <Return>

HSM9-LIC001 Base Software
HSM9-LIC013 5 LMKs
HSM9-LIC024 Mag Stripe Issuers
HSM9-LIC025 Mag Stripe Trans Processing
HSM9-LIC026 EMV Trans Processing
HSM9-LIC027 PIN/Key Mailer
HSM9-LIC028 Visa Cash Processing
HSM9-LIC029 Legacy Functions
HSM9-LIC030 Miscellaneous HSM8000 Base Commands License
HSM9-LIC036 Secure Host Comms

Bootstrap Version:         1.10.2
Bootmanager Version:       1.16.12
LBC Version:               1.6
Microcontroller Version:   1.33

AGS Cryptographic Library:          1.10.C644
FIPS Validated DRBG/RNG Algorithm:  TSPP-DRBG v1.1
FIPS Validated SHA Algorithm:       TSPP-SHA v1.0
FIPS Validated HMAC Algorithm:      TSPP-HMAC v1.0
FIPS Validated TDES Algorithm:      TSPP-TDES v1.0
FIPS Validated RSA Algorithm:       TSPP-RSA v1.0
FIPS Validated AES Algorithm:       TSPP-AES v1.0
FIPS Validated CMAC Algorithm:      TSPP-CMAC v1.0

Online>
```

View Available Commands

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **GETCMDS**

Function: To display a list of available host & console commands.

There are three attributes which control whether individual commands are available for use:

- Whether the command is *implemented* in the installed firmware;
- Whether the command is *licensed* for use – i.e. included in the installed license. Note that only host commands (not console commands) are controlled in this way;
- Whether the command is *enabled* using the CONFIGCMDS console commands (or via payShield Manager).

Note: Some of the commands listed may require additional license options enabled. For example the command EI requires the RSA algorithm to be included in the license in order to function correctly.

Authorization: The HSM does not require any authorization to run this command.

Inputs: [-hl]

Switch	Description
<blank>	Display a list of all host & console commands that are *implemented* AND *licensed* AND *enabled*.
-h	Display a hash of the host & console commands that are *implemented* AND *licensed* AND *enabled*. (The hash is affected by enabling/disabling commands using the CONFIGCMDS console command.)
-l	Display a list of all host & console commands that are *implemented* AND *licensed*. (This list is not affected by enabling/disabling commands using the CONFIGCMDS console command.)

Outputs: A list of available HSM commands (depending on options).

Errors: None.

Example:

```
Online> GETCMDS -h -l <Return>

List of enabled Host commands:

A0  A2  A4  A6  A8  AA  AC  AE  AG  AI  AK  AM  AO  AS  AU
AW  AY  B0  B2  BA  BC  BE  BG  BI  BK  BM  BQ  BS  BU  BW
BY  C0  C2  C4  C6  C8  CA  CC  CE  CG  CI  CK  CM  CO  CQ
CS  CU  CW  CY  D0  D2  D4  D6  D8  DA  DC  DE  DG  DI  DK
DM  DO  DQ  DS  DU  DW  DY  E0  E2  E4  E6  E8  EA  EC  EE
EG  EI  EK  EM  EO  EQ  ES  EU  EW  EY  F0  F2  F4  F6  F8
FA  FC  FE  FG  FI  FK  FM  FO  FQ  FS  FU  FW  G0  G2  G4
G6  G8  GA  GC  GE  GG  GI  GK  GM  GO  GQ  GS  GU  GW  GY
H0  H2  H4  H6  H8  HA  HC  HE  HG  HI  HK  HM  HO  HQ  HS
HU  HW  I0  I2  I4  I6  I8  IA  IC  IE  J0  J2  J4  J6  J8
JA  JC  JE  JG  JI  JK  JO  JS  JU  K0  K2  K8  KA  KC  KE
KG  KI  KK  KM  KO  KQ  KS  KU  KW  KY  L0  LA  LC  LE  LG
LI  LK  LM  LO  LQ  LS  LU  LW  LY  M0  M2  M4  M6  M8  MA
MC  ME  MG  MI  MK  MM  MO  MQ  MS  MU  MW  MY  N0  NC  NE
NG  NI  NK  NO  NY  OA  OC  OE  OI  OK  OU  OW  P0  P2  P4
PA  PC  PE  PG  PI  PK  PM  PO  PQ  PS  PU  PW  PY  Q0  Q2
Q4  Q6  Q8  QA  QC  QI  QK  QM  QO  QQ  QS  QU  QW  QY  R2
R4  R6  R8  RA  RC  RE  RG  RI  RK  RM  RO  RQ  RS  RU  RW
RY  T0  T2  T4  T6  TA  U0  U2  U4  U6  U8  V0  V2  V4  V6
V8  W0  W2  W4  W6  W8  X0  X2  X4  X6  X8  XK  XM  XO  XQ
XS  XU  XW  Y0  Y2  Y4  Y6  Y8  Z0  ZA  ZE  ZK  ZM  ZU

List of enabled Console commands:

A         A5        A6        A7        AUDITLOG          AUDITOPTIONS
AUDITPRINT          B         BK        C         CA      CC
CH        CK        CL        CLEARERR          CLEARAUDIT        CM
CO        CONFIGACL           CONFIGCMDS        CONFIGPB          CP        CS
CV        D         DA        DC        DD      DE
DG        DM        DO        DT        EA      EC
ED        EJECT     ERRLOG    F         FC      FICONTEST
FK        GC        GETCMDS   GETTIME   GK      GS
GZ        HEALTHENABLE        HEALTHSTATS       IK        IV        KA
KB        KD        KE        KG        KK      KM
KN        KT        LK        LO        LN      MI
N         NETSTAT   NP        PING      PV      QA
QC        QH        QL        QM        QP      QS
R         RC        RESET     RH        RI      ROUTE
RS        RZ        SD        SE        SETTIME SG
SI        SK        SL        SP        SNMP    SNMPADD
SNMPDEL   SS        ST        SV        T         TD
TRAP      TRAPADD   TRAPDEL   TRACERT   UTILCFG   UTILENABLE
UTILSTATS           V         VA        VC        VR        VT
WK        XA        XD        XE        XH      XI
XK        XR        XT        XX        XY      XZ
YA        YB        Z         $

Host/Console Command Hash Value:  3aee4c

Online>
```

Show Network Statistics

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **NETSTAT**

Function: The HSM records details about network activity on both its Management and Host Ethernet ports for diagnostic and security purposes. As a diagnostic aid, it can provide useful information when configuring the unit. If reviewed periodically, it can also provide evidence of unexpected network activity, which may require further investigation.

The HSM collects information about each 'endpoint' that communicates with it. The information recorded will depend on the particular protocol that was used to send the packet.

Authorization: The HSM does not require any authorization to run this command.

Inputs:

Syntax:

Display a list of active sockets for each protocol:
netstat [-AanT] [-f address_family]

Display the contents of one of the other network data structures:
netstat [-dgiLmnrsTv] [-f address_family]

Continuously display (as per the wait interval) the information regarding packet traffic on the configured network interfaces:
netstat [-dnT] [-I interface] [-w wait]

Display statistics about the named protocol:
netstat [-T] [-p protocol]

Display per-interface statistics for the specified protocol:
netstat [-p protocol] [-iT] [-I interface]

Display per-interface statistics for the specified address family:
netstat [-sT] [-f address_family] [-i] [-I interface]

Options:

-A — Show the addresses of any protocol control blocks associated with sockets.

-a — Show the state of all sockets. Without -a, sockets used by server processes aren't shown.

-d — Show the number of dropped packets.

-f *address_family* — Limit the statistics or address control block reports to those of the specified address family.

Address family	address family value
AF_INET	inet
AF_INET6	inet6
AF_LOCAL	local or unix
AF_ARP	arp

-g — Show information related to multicast (group address) routing. By default, show the IP Multicast virtual-interface and routing tables. If -s is also specified, show the multicast routing statistics.

-I *interface* — If used with -w, show information about the specified interface only.
If used with -f *address_family* and -s, or with -p protocol, show per-interface statistics on the interface

for *address_family* or protocol, respectively.

Interface	HSM Port
h1	Host Port #1
h2	Host Port #2
m	Management Port

If the -I option is not specified, netstat will report on all
the interfaces.

-i — Show the state of interfaces that have been auto-configured. Interfaces statically configured into a system but not located at boot time aren't shown.
If you also specify -a, show multicast addresses currently in use for each Ethernet interface and for each IP interface address. Multicast addresses are shown on separate lines following the interface address with which they're associated.
If used with -f *address_family* and -s, or with -p protocol, show per-interface statistics on the interface for *address_family* or protocol, respectively

-L — Don't show link-level routes (e.g., IPv4 ARP or IPv6 neighbour cache).

-m — Show statistics recorded by the memory-management routines (the network manages a private pool of memory buffers).

-n — Show network addresses as numbers (normally netstat interprets addresses and attempts to display them symbolically).

-p *protocol* — Show statistics about *protocol*, which is either a well-known name for a protocol or an alias for it. A null response typically means that there are no interesting numbers to report. The utility complains if *protocol* is unknown or if there's no statistics routine for it.

-r — Show the routing tables. If -s is also specified, show the routing statistics instead.

-s — Show per-protocol statistics. If this option is repeated, counters with a value of zero are suppressed.

-T — Use TCP for name lookups (the default is UDP).

-v — Show extra (verbose) detail for the routing tables (-r), or avoid truncating long addresses.

-w *wait* — Specify the time interval for displaying network interface statistics.

Outputs: Text messages as appropriate.

The reported state can have the following values:

ESTABLISHED

The socket has an established connection.

SYN_SENT

The socket is actively attempting to establish a connection.

SYN_RECV

A connection request has been received from the network.

FIN_WAIT1

The socket is closed, and the connection is shutting down.

FIN_WAIT2

Connection is closed, and the socket is waiting for a shutdown from the remote end.

TIME_WAIT

The socket is waiting after close to handle packets still in the network.

CLOSED

The socket is not being used.

CLOSE_WAIT

The remote end has shut down, waiting for the socket to close.

LAST_ACK

The remote end has shut down, and the socket is closed. Waiting for acknowledgement.

LISTEN

The socket is listening for incoming connections.

CLOSING

Both sockets are shut down but we still don't have all our data sent.

UNKNOWN

The state of the socket is unknown

Example:

```
Offline> NETSTAT <Return>

Active Internet connections
Proto Recv-Q Send-Q  Local Address           Foreign Address
State
tcp        0      0  192.168.200.100.xserve  *.*
LISTEN
tcp        0      0  192.168.200.100.ftp     *.*
LISTEN
udp        0      0  *.*                     *.*
udp        0      0  *.syslog                *.*
udp        0      0  *.5002                  *.*
Offline>
```

Test TCP/IP Network

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **PING**

Function: To test the specified network node, and the route to it.

Authorization: The HSM does not require any authorization to run this command.

Inputs:

Syntax:

ping [-adDfLnoPqQrRv] [-c count] [-E policy] [-g gateway]
[-h host] [-I interface] [-i wait] [-l preload]
[-p pattern] [-s packetsize] [-t tos] [-T ttl]
[-w maxwait] host

Options:

-a	Emit an audible beep (by sending an ASCII BEL character to the terminal) after receiving each non-duplicate response.
-c *count*	Stop after sending (and receiving) this many ECHO_RESPONSE packets.
-D	Set the Don't Fragment bit in the IP header. This is meant to determine the path MTU.
-d	Set the SO_DEBUG option on the socket being used.
-E *policy*	Specify the IPsec policy for packets.
-g *gateway*	Use Loose Source Routing to send the ECHO_REQUEST packets via gateway. The default is to use the routing table.
-h *host*	Alternate way of specifying the target host instead of as the last argument.
-I *interface*	The interface that PING is to be sent from. *interface* Value — HSM Port h1 — Host Port #1 h2 — Host Port #2 m — Management Port (default)
-i *interval*	Wait interval seconds between sending each packet (default is one second). For the -f option, the interval is 0.01 seconds.
-l *preload*	Send this many packets as fast as possible before returning to normal behaviour.
-L	Disable loopback when sending to multicast destinations, so the transmitting host doesn't see the ICMP requests.
-n	Print numeric output only. No attempt is made to look up symbolic names for host addresses.
-o	Exit successfully after receiving one reply packet.
-P	Use a pseudo-random sequence for the data instead of the default, fixed sequence of incrementing 8-bit integers. This is useful to foil compression on PPP and other links.
-p *pattern*	Fill out the packet with this many "padding" bytes (maximum is 16). You should find this useful for diagnosing data-dependent problems in a network. For example, -p ff causes the sent packet to be filled with ones.
-Q	Don't display responses such as Network Unreachable ICMP messages concerning the ECHO_REQUESTs sent.
-q	Be quiet: display nothing except for the summary lines at startup time and when finished.
-R	Record the route.
-r	Bypass the normal routing tables and send directly to a host on an attached network. If the host isn't on a directly attached network, an error is returned. You can use this option to ping a local host through an interface that has no route through it.

-s *packetsize*	Send this many data bytes. The default is 56, which translates into 64 ICMP data bytes when combined with the 8 bytes of ICMP header data.
-T *ttl*	Use the specified time-to-live. It represents how many hops the packet can go through before being discarded (when it reaches 0). The default is 255.
-t *tos*	Use the specified hexadecimal type of service.
-v	Verbosity (default none).
-w *maxwait*	Specify a timeout, in seconds, before ping exits regardless of how many packets have been sent or received.

Outputs: Text messages as appropriate.

Example:

```
Offline> PING -I h1 192.168.100.123 <Return>

PING 192.168.100.123 (192.168.100.123): 56 data bytes
64 bytes from 192.168.100.123: icmp_seq=0 ttl=32 time=16 ms
64 bytes from 192.168.100.123: icmp_seq=1 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=2 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=3 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=4 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=5 ttl=32 time=101 ms
64 bytes from 192.168.100.123: icmp_seq=6 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=7 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=8 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=9 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=10 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=11 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=12 ttl=32 time=4 ms
64 bytes from 192.168.100.123: icmp_seq=13 ttl=32 time=4 ms

Offline>
```

Trace TCP/IP route

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **TRACERT**

Function: To view the path taken from the HSM to the specified address.

Authorization: The HSM does not require any authorization to run this command.

Inputs:

Syntax:

tracert [-DdFlInPrvx] [-a | -A *as_server*] [-f *first_ttl*]
[-g *gateway*] [-i *interface*] [-m *max_ttl*] [-p *port*]
[-q *nqueries*] [-s *src_addr*] [-t *tos*] [-w *wait_time*]
host [*packetsize*]

Options:

-A *as_server*	Turn on AS lookups and use the given server instead of the default.
-a	Turn on AS lookups for each hop encountered.
-D	Dump the packet data to standard error before transmitting it.
-d	Turn on socket-level debugging.
-F	Set the "don't fragment" bit.
-f *first_ttl*	Set the initial time-to-live used in the first outgoing probe packet.
-g *gateway*	Specify a loose source route gateway (8 maximum).
-I	Use ICMP ECHO instead of UDP datagrams.
-i *interface*	*interface* Value — HSM Port h1 — Host Port #1 h2 — Host Port #2 m — Management Port (default)
-l ("el")	Display the TTL (time-to-live) value of the returned packet. This is useful for checking for asymmetric routing.
-m *max_ttl*	Set the maximum TTL (maximum number of hops) used in outgoing probe packets. The default is 30 hops (the same default as is used for TCP connections).
-n	Print hop addresses numerically only. By default, addresses are printed both symbolically and numerically. This option saves a nameserver address-to-name lookup for each gateway found on the path.
-P	Set the "don't fragment" bit and use the next hop MTU each time a "need fragmentation" error is received, thus probing the path MTU.
-p *port*	The base UDP port number to be used in probes (default is 33434). The tracert utility hopes that nothing is listening on UDP ports base to base + nhops -1 at the destination host (so an ICMP PORT_UNREACHABLE message is returned to terminate the route tracing). If something is listening on a port in the default range, you can use this option to pick an unused port range.
-q *nqueries*	The number of probes per ttl to *nqueries* (default is three probes).
-r	Bypass the normal routing tables and send directly to a host on an attached network. If the host isn't on a directly attached network, an error is returned. You can use this option to "ping" a local host through an interface that has no route through it (for example, after the interface was

	dropped by routed).
-s *src_addr*	The IP address (must be given as an IP number, not a hostname) to be used as the source address in outgoing probe packets. If the host has more than one IP address, you can use this option to force the source address to be something other than the IP address of the interface that the probe packet is sent on. If the IP address you specify isn't one of this machine's interface addresses, an error is returned and nothing is sent.
-t *tos*	The type-of-service (TOS) to be used in probe packets (default is zero). The value must be a decimal integer in the range 0 to 255. You can use this option to see if different TOSs result in different paths. Not all TOS values are legal or meaningful. You should find the values -t 16 (low delay) and -t 8 (high throughput) useful.
-v	Be verbose. Received ICMP packets other than TIME_EXCEEDED and UNREACHABLEs are listed.
-w *wait_time*	The time (in seconds) to wait for a response to a probe (default is 5).
-x	Toggle checksums. Normally, this prevents tracert from calculating checksums. In some cases, the operating system can overwrite parts of the outgoing packet but not recalculate the checksum (so in some cases the default is to not calculate checksums and using -x causes them to be calculated). Note that checksums are usually required for the last hop when using ICMP ECHO probes (-I).
host	The destination hostname or IP number.
packetsize	The probe datagram length (default is 40 bytes).

Outputs: Text messages as appropriate.

Example:

```
Offline> TRACERT  -I h1 -g 10.10.10.1   10.10.11.2 <Return>

traceroute to 10.10.11.2 (10.10.11.2), 64 hops max, 40 byte packets
 1   10.10.10.1 (10.10.10.1)  5.000 ms  7.000 ms  5.000 ms
 2   10.10.11.2 (10.10.11.2)  5.000 ms  6.000 ms  6.000 ms

Offline>
```

View/Reset Utilization Data

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **UTILSTATS**

Function: To display Utilization Data at the Console. Options to print the data to an HSM-attached printer and to reset accumulated data to zero.

Authorization: The HSM does not require any authorization to run this command.

Notes:
- Utilization statistics are also reset when new software is installed on the HSM.
- The precise meaning of an HSM loading range identified below as, for example, "*10-20%*" is "*from exactly 10% to just under 20%*".
- Statistics are provided irrespective of which host interface the commands are received over.

Inputs:
- Whether to print output to HSM-attached printer
- Whether to Reset data

Outputs: Text messages as in example below.

Note that the number of seconds displayed is not necessarily the number of seconds between the start and end times: rather, it is the number of seconds during this period when data collection was enabled using the UTILENABLE command and the HSM was online.

Example:

```
Online> UTILSTATS <Return>

HSM Serial Number:          A4665271570Q

Report Generation Time:     21-Mar-2011 23:23.05
Report Start Time:          01-JAN-2011 14:25.01
Report End Time:            05-MAR-2011 23:19.59
Total number of secs:       123,456

HSM Loading:
       0-10%:     56,789
       10-20%:    24,109
       20-30%:    21,445
       30-40%:    12,382
       40-50%:    3,288
       50-60%:    2,917
       60-70%:    2,123
       70-80%:    403
       80-90%:    0
       90-100%:   0
       100%:      0

Press "Enter" to continue... <Return>
```

```
Host Command Volumes:
        Cmd Code  Total Transactions  Average TPS
        A0        225                4.79
        A4        99                 2.11
        A6        342                7.28
        A8        408                8.68
        AA        141                3.00
        AC        135                2.87
        AE        84                 1.79
        AG        66                 1.40
        AS        18                 0.38
        AU        94                 2.00
        AW        94                 2.00
        AY        94                 2.00
        B0        50                 1.06
        BA        14                 0.30
        BC        34                 0.72
        BE        42                 0.89
        BG        5                  0.11
        BI        11                 0.23
        BK        128                2.72

Press "Enter" to continue... <Return>

        Cmd Code  Total Transactions  Average TPS
        BM        10                 0.21
        LA        2                  0.04

Instantaneous HSM Load: 17%
Instantaneous Host Command Volumes:
       Cmd Code   Total Transactions  Average TPS
       BM         10                 0.21
       LA         2                  0.04

Send output to printer? [Y/N]: Y <Return>
Reset All Stats? [Y/N]: Y <Return>
All Utilization statistics will be reset to 0. Confirm? [Y/N]: Y
<Return>

Online>
```

View/Reset Health Check Counts

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **May be required** Activity: **diagnostics**		

Command: **HEALTHSTATS**

Function: To display Health Check counts at the Console. Options to print the data to an HSM-attached printer and to reset accumulated data to zero.

Authorization: The HSM does not require any authorization to run this command to view the data.
The HSM must be in Offline/Secure Authorized state (or the activity **diagnostics** must be authorized) for the Management LMK to reset the Health Check Counts

Notes:
- Accumulated health check counts are also reset when new software is installed on the HSM.
- If collection of health check data has been suspended at any time, the counts relating to Fraud Detection (i.e. failed PIN verifications and PIN Attacks) will not represent the values of those counts which will be used by the HSM to trigger return of Error 39 or deletion of LMKs.

Inputs:
- Whether to print output to HSM-attached printer
- Whether to Reset data (requires Offline/Secure Authorized state).

Outputs: Text messages as in example below.

Example:

```
Offline-AUTH> HEALTHSTATS <Return>

HSM Serial Number:            A4665271570Q

Report Generation Time:       21-Dec-2010 23:22.28
Report Start Time:            01-Dec-2010 01:11.21
Report End Time:              21-Dec-2010 23:22.28
Number of reboots:                         3
Number of tampers:                         1
Failed PIN verifies/minute limit exceeded: 57
Failed PIN verifies/hour limit exceeded:   4
PIN Attack Limit exceeded:                 0

Send output to printer? [Y/N]: Y <Return>

Reset All Stats? [Y/N]: Y <Return>
All Utilization statistics will be reset to 0. Confirm? [Y/N]: Y
<Return>

Offline-AUTH>
```

Check the FICON Host Interface

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **FICONTEST**

Function: To check the operation of the FICON Host interface board (if fitted) and optical transceivers.

Authorization: The HSM does not require any authorization to run this command.

Notes:
- This test is appropriate only to payShield 9000 units fitted with the FICON option.
- The test can be run between 2 transceivers or on a single transceiver.
- A suitable FICON optical cable must be used to connect the two transceivers. Where 2 transceivers are being used, a standard FICON cable pair should be used to connect the transceivers.
- Where a single transceiver is being used, the loopback cable provided with the payShield 9000 should be used. Alternatively, one connection out of a standard FICON cable pair can be used.
- The test will send 10 packets and report success/failure on each.
- The test will check that the following components are installed and operational:
  - HSM main board
  - FICON board and connectors
  - Transceivers and connectors
  - Optical cable
  - FICON software

Inputs:
- None

Outputs: Text messages as in example below.

Example:

```
Offline> FICONTEST <Return>

Please connect FICON Port 1 to FICON Port 2 or insert a loopback
cable in FICON port 1 and press enter to continue: <Return>

Packet 1 success
Packet 2 success
Packet 3 success
Packet 4 success
Packet 5 success
Packet 6 success
Packet 7 success
Packet 8 success
Packet 9 success
Packet 10 success

terminating...
10 packets sent, 10 packets received, 0% loss

Offline>
```

Chapter 3 – Local Master Keys

Types of LMKs

A **Variant LMK** is a set of 20 double- or triple-length TDES keys, with different "pairs" and variants of those "pairs" being used to encrypt different types of keys. Note that the term "*pair*" is used regardless of whether the LMK consists of double-length keys, or triple-length keys. The standard LMK format supported in all previous versions of Thales (Racal) HSM firmware consists of 20 double-length TDES keys.

Note: The term "Variant LMK" refers to the fact that variants are applied to the LMK prior to using the LMK; a Variant LMK is not itself a variant of any other key.

A **Key Block LMK** is either a triple-length TDES key, or a 256-bit AES key, and is used to encrypt keys in a key block format. A Key Block LMK is not compatible with a Variant LMK, and it can only be used to encrypt keys in the key block format.

Note: The term "Key Block LMK" refers to the 'key block' method of encrypting keys; a Key Block LMK is not itself stored in the key block format.

Multiple LMKs

It is possible to install multiple LMKs within a single HSM. The precise details of the number and type of installed LMKs are controlled via the HSM's license file:

License	Description
Default – no specific multi-LMK license	Two concurrent LMKs can be installed; however, one must be a Variant LMK, and the other a Key Block LMK.
HSM9-LIC012 LMK x 2 (optional license)	Two concurrent LMKs can be installed; they can be any combination of Variant and Key Block LMKs.
HSM9-LIC013 LMK x 5 (optional license)	Five concurrent LMKs can be installed; they can be any combination of Variant and Key Block LMKs.
HSM9-LIC021 LMK x 10 (optional license)	Ten concurrent LMKs can be installed; they can be any combination of Variant and Key Block LMKs.
HSM9-LIC022 LMK x 20 (optional license)	Twenty concurrent LMKs can be installed; they can be any combination of Variant and Key Block LMKs.

See Chapter 1 of the Host Command Reference Manual for information on how the required LMK can be identified in Host commands.

LMK Table

LMKs are stored in a table within the secure memory of the HSM, with each LMK occupying a different 'slot' within the table. Each slot has the following attributes:

Attribute	Description
LMK ID	A 2-digit number which uniquely indicates the location of each LMK within the table. All references to LMKs are made by specifying the LMK Identifier.
Key Scheme	• "Variant" for traditional Racal/Thales LMK – key encryption performed using the *variant* method. • "Key Block" for enhanced security – key encryption performed using the *key block* method.
Algorithm	• "3DES (2key)" or "3DES (3key)" is used by Variant LMKs. • "3DES (3key)" or "AES (256-bit)" is used by Key Block LMKs. Other algorithm types may be supported in future software releases.
Status	• "Test" indicates that the LMK is used for testing purposes. • "Live" indicates that the LMK is used for live production purposes. When installing LMKs, the HSM will prevent any mixing of Test and Live LMKs within the same slot (i.e. LMK Value and Old/New LMK Value must have the same status).
Comments	User-entered text, which can be used to help identify LMKs.
Authorization	Indicates the authorization status of the HSM for this particular LMK – either a flag (for Authorized State) or a list of authorized activities.
Old/New Status	Flag for each LMK held in Key Change Storage indicating whether they are to be used as an 'old' LMK (loaded via 'LO' command), or a 'new' LMK (loaded via 'LN' command).
LMK Check Value	The check value of the LMK.
Old/New LMK Check Value	The check value of the 'old' or 'new' LMK held in Key Change Storage.

Use the console command VT (View LMK Table) to view the contents of the HSM's LMK table (but not the actual LMK values).

LMK Commands

The HSM provides the following console commands to support LMK operations:

Generate LMK Component(s)

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **GK**

Function: To generate component(s) of an LMK, and store the component(s) on smartcards.
This command may be used to generate components for the following types of LMKs:
- Double-length (2DES) Variant LMK
- Triple-length (3DES) Variant LMK
- Triple-length (3DES) Key Block LMK
- 256-bit AES Key Block LMK.

When creating a Variant LMK or a 3DES Key Block LMK, this command generates the data for a single LMK component card.

When creating an AES Key Block LMK, this command generates the data for all the required number of LMK component cards.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- LMK Scheme (Variant or Key Block).
- LMK Algorithm:
  - Double-length (2DES) or triple-length (3DES) if Variant scheme is selected
  - Triple-length (3DES) or AES if Key Block scheme is selected.
- LMK Status (Test or Live).
- For TDES LMKs (Variant or Key Block):
  - Component set number.
  - Three or four values (A, B, C, D).
    - For a double-length (2DES) Variant LMK, there are 3 secret values: A & B each consist of 16 hex digits, and C is 8 hex digits.
    - For a triple-length (3DES) Variant LMK, there are 4 secret values: A, B & C each consist of 16 hex digits, and D is 8 hex digits.
    - For a triple-length (3DES) Key Block LMK, there are 3 secret values: A, B & C each consist of 16 hex digits.
    - Note: When the secret values A, B, C, D are entered manually, care must be taken to ensure that each (different) LMK component card is generated using a *different* set of values for A, B, C, D.
    - The HSM generates random values if no values are input.
  - In the prompts for the secret values, a 16 hex digit values is referred to as "Secret Value" and an 8 hex

digit value is referred to simply as "Value".
- For an AES Key Block LMK:
  - Number of components.
  - Number of components required to reconstitute the LMK.

Outputs:
- LMK components written to smartcards.
- LMK component check value.

Errors:
- Card not formatted – use the FC command to format the card.
- Not a LMK card –card is not formatted for LMK or key storage.
- Warning – card not blank. Proceed? [Y/N] – LMK card is not blank.
- Overwrite LMK set? [Y/N] – card already contains an LMK component.
- Smartcard error; command/return: 0003 – invalid PIN is entered.
- Invalid PIN; re-enter – a PIN of less than 4 or greater than 8 is entered.

Notes:
- PINs must be entered within 60 seconds of being requested.
- If the CS setting "Card/Password authorization" is set to "Card", then the HSM will write a random password to the 1st and 2nd LMK component cards. These passwords will be required in order to put the HSM into the Authorized State.

Example 1: (Triple-length Variant LMK)

This example generates a triple-length Variant LMK component set, and writes the components to a smartcard.

```
Secure> GK <Return>
Variant scheme or key block scheme? [V/K]: V <Return>
Enter algorithm type [2=2DES, 3=3DES]: 3 <Return>

Key status? [L/T]: L <Return>
LMK component set [1-9]: 1 <Return>
Enter secret value A: AAAA AAAA AAAA AAAA <Return>
Enter secret value B: BBBB BBBB BBBB BBBB <Return>
Enter secret value C: CCCC CCCC CCCC CCCC <Return>
Enter value D:        DDDD DDDD <Return>
Insert blank card and enter PIN: ******** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ


Remove the smartcard and store it securely.


Make another copy? [Y/N]: N <Return>
1 copies made.


Repeat the procedure to generate further component sets.


Secure>
```

Example 2: (Double-length Variant LMK)

This example generates a double-length variant LMK component set, and writes the components to a smartcard.

```
Secure> GK <Return>
Variant scheme or key block scheme? [V/K]: V <Return>
Enter algorithm type [2=2DES, 3=3DES]: 2 <Return>

Key status? [L/T]: L <Return>
LMK component set [1-9]: 1 <Return>
Enter secret value A: AAAA AAAA AAAA AAAA <Return>
Enter secret value B: BBBB BBBB BBBB BBBB <Return>
Enter value C:        CCCC CCCC <Return>
Insert blank card and enter PIN: ******** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ
```

Remove the smartcard and store it securely.

```
Make another copy? [Y/N]: N <Return>
1 copies made.
```

Repeat the procedure to generate further component sets.

```
Secure>
```

Example 3: (Triple-length 3DES Key Block LMK)

This example generates a 3DES key block LMK component, and writes the component to a smartcard.

```
Secure> GK <Return>
Variant scheme or key block scheme? [V/K]: K <Return>
Enter algorithm type [D=DES, A=AES]: D
Key status? [L/T]: L <Return>
LMK component set [1-9]: 1 <Return>
Enter secret value A: <Return>
Enter secret value B: <Return>
Enter secret value C: <Return>
Insert blank card and enter PIN: ******** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ
```

Remove the smartcard and store it securely.

```
Make another copy? [Y/N]: N <Return>
1 copies made.
```

Repeat the procedure to generate further components.

```
Secure>
```

Example 4: (AES Key Block LMK)

This example generates a set of AES key block LMK components, and writes each component to a smartcard.

```
Secure> GK <Return>
Variant scheme or key block scheme? [V/K]: K <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter the number of components to generate: [2-9]: 5 <Return>
Enter the number of components required to reconstitute the
LMK: [2-5]: 2 <Return>
Key status? [L/T]: L <Return>

Check value for the LMK: ZZZZZZ

Insert blank card and enter PIN: ******** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ

Remove the smartcard and store it securely.

Insert blank card and enter PIN: ******** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ

Remove the smartcard and store it securely.

The above sequence is repeated to generate each component card.

Secure>
```

Load LMK

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **LK**

Function: To load LMK components from smartcards.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- LMK Identifier: 2 numeric digits.
- Smartcards (RLMKs are supported) with LMK components.
- PINs for the Smartcards or passwords. The PIN must be entered within 60 seconds.
- Whether to make this LMK the Default/Management LMK - see Notes below.

Outputs:
- Individual LMK component check value(s).
- Final LMK check value.

Notes:
- For PCI HSM compliance, PINs and smartcards must be used to authenticate the Security Officers.
- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.
- If there is not already a Default and/or Management LMK installed (i.e. the LMK IDs identified in the security settings as being the default and management LMKs are empty), you will be asked if you wish to make this new LMK the Default/Management LMK.
- An error is returned if an attempt is made to load an LMK with a single component where:
  - The LMK is not a test LMK, and
  - The security setting to enforce multiple components has been set to YES.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Load failed check comparison - card is blank.
- Not a LMK card - card is not formatted for LMK or key storage.
- Card not formatted - card is not formatted.
- Smartcard error; command/return: 0003 - invalid PIN is entered.
- Invalid PIN; re-enter - a PIN of less than 5 or greater than 8 digits is entered.
- Invalid key – a standard Thales test key cannot be given live status.
- Incompatible key status – the components have different status ("live" or "test").
- Invalid key - Multiple key components required – an attempt has been made to load an LMK (other than a test LMK) using a single component when the security setting to enforce multiple components has been set to YES.

Example 1: (Double-length Variant LMK)

This example loads a double-length Variant LMK from smartcards and installs it in the HSM. There is already Default and Management LMKs installed.

```
Secure> LK <Return>
Enter LMK id: 00 <Return>
Enter comments: Live LMK for ABC Bank <Return>
LMK in selected location must be erased before proceeding
Erase LMK? Y <Return>
Load LMK from components
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure. When all components have been loaded and the HSM displays the LMK Check value, record the check value.

```
LMK Check: ZZZZZZ
LMK id: 00
LMK key scheme: Variant
LMK algorithm: 3DES (2key)
LMK status: Live
Comments: Live LMK for ABC Bank
Confirm details? [Y/N]: Y <Return>
Use the LO command to load LMKs into key change storage.
Secure>
```

Example 2: (Triple-length Variant LMK)

This example loads a triple-length variant LMK from smartcards and installs it in the HSM. There are already Default and Management LMKs installed.

```
Secure> LK <Return>
Enter LMK id: 01 <Return>
Enter comments: Process System One <Return>
LMK in selected location must be erased before proceeding
Erase LMK? Y <Return>
Load LMK from components
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure. When all components have been loaded and the HSM displays the LMK Check value, record the check value.

```
LMK Check: ZZZZZZ
LMK id: 01
LMK key scheme: Variant
LMK algorithm: 3DES (3key)
LMK status: Live
Comments: Process System One
Confirm details? [Y/N]: Y <Return>
Use the LO command to load LMKs into key change storage.
Secure>
```

Example 3: (Any LMK type)

In this example, the PIN is not entered within 60 seconds.

```
Secure> LK <Return>
Enter LMK id [0-9]: 0 <Return>
Enter comments: <Return>
Load LMK from components
Insert card and enter PIN:
Terminated
Secure>
```

Example 4: (Double- or triple-length Variant LMK)

In this example, the security setting requiring use of multiple components has been set to YES, but the user has attempted to load a non-Test LMK using only one component.

```
Secure> LK <Return>
Enter LMK id [0-4]: 0 <Return>
Enter comments: <Return>
Load LMK from components
Insert card and enter PIN: ****<Return>
Check: 562342
Load more components? [Y/N]: n<Return>
LMK Check: 562342
Invalid key - Multiple key components required
Secure>
```

Example 5: (3DES Key Block LMK)

This example loads a 3DES key block LMK from smartcards and installs it in the HSM. There is already Default and Management LMKs installed.

```
Secure> LK <Return>
Enter LMK id: 01 <Return>
Enter comments: Live LMK for XYZ Bank <Return>
LMK in selected location must be erased before proceeding
Erase LMK? Y <Return>
Load LMK from components
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure. When all components have been loaded and the HSM displays the LMK Check value, record the check value.

```
LMK Check: ZZZZZZ
LMK id: 01
LMK key scheme: KeyBlock
LMK algorithm: 3DES (3key)
LMK status: Live
Comments: Live LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Use the LO command to load LMKs into key change storage.
Secure>
```

Example 6: (AES Key Block LMK)

This example loads an AES key block LMK from smartcards and installs it in the HSM. There is already Default and Management LMKs installed.

```
Secure> LK <Return>
Enter LMK id: 02 <Return>
Enter comments: Live LMK for XYZ Bank <Return>
LMK in selected location must be erased before proceeding
Erase LMK? Y <Return>
Load LMK from components
Insert card and enter PIN: ******** <Return>
Check: AAAAAA

Remove the smartcard. Insert the second and subsequent
smartcards and repeat the loading procedure. When all
components have been loaded and the HSM displays the LMK Check
value, record the check value.


LMK Check: ZZZZZZ
LMK id: 02
LMK key scheme: KeyBlock
LMK algorithm: AES-256
LMK status: Live
Comments: Live LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Use the LO command to load LMKs into key change storage.
Secure>
```

Example 7: (AES Key Block LMK - no Default or Management LMK already installed.)

This example loads an AES key block LMK from smartcards and installs it in the HSM. There is no Default or Management LMK already installed.

```
Secure> LK <Return>
Enter LMK id: 02 <Return>
Enter comments: Live LMK for XYZ Bank <Return>
LMK in selected location must be erased before proceeding
Erase LMK? Y <Return>
Load LMK from components
Insert card and enter PIN: ******** <Return>
Check: AAAAAA

Remove the smartcard. Insert the second and subsequent
smartcards and repeat the loading procedure. When all
components have been loaded and the HSM displays the LMK Check
value, record the check value.


LMK Check: ZZZZZZ
LMK id: 02
LMK key scheme: KeyBlock
LMK algorithm: AES-256
LMK status: Live
Comments: Live LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Use the LO command to load LMKs into key change storage.
Do you want to make this LMK the default LMK? [Y/N]: Y <Return>
Do you want to make this LMK the management LMK? [Y/N]: Y
<Return>
Secure>
```

Load 'Old' LMK into Key Change Storage

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Required** Activity: **admin.console**		

Command: **LO**

Function: To load an old LMK component set into Key Change Storage for use in translations from old to new keys. Note that the current LMK must be installed before an "old" LMK can be installed. Also note that it is possible to install a Variant LMK as the "old" LMK, and with a Key Block LMK as the "new" LMK.

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **admin.console** must be authorized, using the Authorizing Officer cards of the specified LMK.

Inputs:
- LMK identifier: 2 numeric digits.
- Smartcards (RLMKs are supported) with old LMK components.
- PINs for the Smartcards or passwords. PINs must be entered within 60 seconds of being requested.

Outputs:
- Individual LMK Component check value(s).
- Final LMK key check value.

Errors:
- No LMK loaded – there is no LMK loaded in main memory.
- Invalid LMK identifier – entered identifier out of range
- Key Block LMK not permitted – it is not permitted to load a Key Block LMK into key change storage if a variant LMK is loaded in main memory.
- Load failed check comparison – card is blank.
- Not a LMK card – card is not formatted for LMK or key storage.
- Card not formatted – card is not formatted.
- Smartcard error; command/return: 0003 – invalid PIN is entered.
- Invalid PIN; re-enter – a PIN of less than 4 or greater than 8 is entered.
- Command only allowed from Secure-Authorized – the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.
- Invalid key – a standard Thales test key cannot be given live status.
- Incompatible cards – the component cards have different formats.
- Incompatible key status – the components have different status ("live" or "test").
- Invalid key - Multiple key components required – an attempt has been made to load an LMK (other than a Test LMK) using a single component when the security setting to enforce multiple components has been set to YES.

Notes:

- For PCI HSM compliance, PINs and smartcards must be used to authenticate the Security Officers.
- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.
- It is not permitted to load a Key Block LMK into the "old" LMK slot of a Variant LMK.
- It is not permitted to load an AES Key Block LMK into the "old" LMK slot of a 3DES Key Block LMK.
- If multiple LMKs are loaded on the HSM, each can have a corresponding old LMK. The ID of the LMK being processed is defined in the command input.

Example 1: (Double-length Variant LMK)

This example loads a double-length Variant LMK from smartcards and installs it as 'old' LMK 00.

```
Secure-AUTH> LO <Return>
Enter LMK id: 00 <Return>
Enter comments: Old LMK for ABC Bank <Return>
Load old LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>

Remove the smartcard. Insert the second and subsequent
smartcards and repeat the loading procedure until all old
component sets have been loaded. When all components have been
loaded and the HSM displays the LMK Check value, ensure that
this is the expected value.


LMK Check: ZZZZZZ
LMK id: 00
LMK key scheme: Variant
LMK algorithm: 3DES (2key)
LMK status: Live
Comments: Old LMK for ABC Bank
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Example 2: (Triple-length Variant LMK)

This example loads a triple-length Variant LMK from smartcards and installs it as 'old' LMK 00.

```
Secure-AUTH> LO <Return>
Enter LMK id: 01 <Return>
Enter comments: Old LMK for Process System One <Return>
Load old LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all old component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 00
LMK key scheme: Variant
LMK algorithm: 3DES (3key)
LMK status: Live
Comments: Old LMK for Process System One
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Example 3: (Double- or triple-length Variant LMK)

This example attempts to load a non-Test LMK using a single component when the security setting to enforce multiple components has been set to YES.

```
Secure-AUTH> LO <Return>
Enter LMK id: 00 <Return>
Enter comments: Old LMK for ABC Bank <Return>
Load old LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: n <Return>
Check: AAAAAA
Invalid key - Multiple key components required
Secure-AUTH>
```

Example 4: (3DES Key Block LMK)

This example loads a 3DES key block LMK from smartcards and installs it as 'old' LMK 01.

```
Secure-AUTH> LO <Return>
Enter LMK id: 01 <Return>
Enter comments: Old LMK for XYZ Bank <Return>
Load old LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all old component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 01
LMK key scheme: Key block
LMK algorithm: 3DES (3key)
LMK status: Live
Comments: Old LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Example 5: (AES Key Block LMK)

This example loads an AES key block LMK from smartcards and installs it as 'old' LMK 02.

```
Secure-AUTH> LO <Return>
Enter LMK id: 02 <Return>
Enter comments: Old LMK for XYZ Bank <Return>
Load old LMK from components.

Insert card and enter PIN: ******** <Return>
Check: AAAAAA
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all old component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 02
LMK key scheme: Key block
LMK algorithm: AES-256
LMK status: Live
Comments: Old LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Load 'New' LMK into Key Change Storage

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Required** Activity: **admin.console**		

Command: **LN**

Function: To load a new LMK component set into Key Change Storage for use in translations from the current LMK to a "new" LMK. Note that the current LMK must be installed before a "new" LMK can be installed. Also note that it is possible to install a Key Block LMK as the "new" LMK, with a Variant LMK as the current LMK.

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **admin.console** must be authorized, using the Authorizing Officer cards of the specified LMK.

Inputs:
- LMK identifier: 2 numeric digits.
- Smartcards (regular HSM or payShield Manager smartcards) with new LMK components.
- PINs for the Smartcards or passwords. PINs must be entered within 60 seconds of being requested.

Outputs:
- Individual LMK Component check value(s).
- Final LMK key check value.

Errors:
- No LMK loaded – there is no LMK loaded in main memory.
- Invalid LMK identifier – entered identifier out of range
- Key Block LMK not permitted – it is not permitted to load a key block LMK into key change storage if a variant LMK is loaded in main memory.
- Load failed check comparison – card is blank.
- Not a LMK card – card is not formatted for LMK or key storage.
- Card not formatted – card is not formatted.
- Smartcard error; command/return: 0003 – invalid PIN is entered.
- Invalid PIN; re-enter – a PIN of less than 4 or greater than 8 is entered.
- Command only allowed from Secure-Authorized – the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.
- Invalid key – a standard Thales test key cannot be given live status.
- Incompatible cards – the component cards have different formats.
- Incompatible key status – the components have different status ("live" or "test").
- Invalid key - Multiple key components required – an attempt has been made to load an LMK (other than a Test LMK) using a single component when the security setting to enforce multiple components has been set to YES.

Notes:

- For PCI HSM compliance, PINs and smartcards must be used to authenticate the Security Officers.
- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.
- It is not permitted to load a Variant LMK into the "new" LMK slot of a Key Block LMK.
- It is not permitted to load a 3DES Key Block LMK into the "new" LMK slot of an AES Key Block LMK.
- If multiple LMKs are loaded on the HSM, each can have a corresponding 'new' LMK. The ID of the LMK being processed is defined in the command input.

Example 1: (Double-length Variant LMK)

This example loads a double-length Variant LMK from smartcards and installs it as 'new' LMK 00.

```
Secure-AUTH> LN <Return>
Enter LMK id: 00 <Return>
Enter comments: New LMK for ABC Bank <Return>
Load new LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all new component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 00
LMK key scheme: Variant
LMK algorithm: 3DES (2key)
LMK status: Live
Comments: New LMK for ABC Bank
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Example 2: (Triple-length Variant LMK)

This example loads a triple-length Variant LMK from smartcards and installs it as 'new' LMK 00.

```
Secure-AUTH> LN <Return>
Enter LMK id: 01 <Return>
Enter comments: New LMK for Process System One <Return>
Load new LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all new component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 00
LMK key scheme: Variant
LMK algorithm: 3DES (3key)
LMK status: Live
Comments: New LMK for Process System One
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Example 3: (Double- or triple-length Variant LMK)

This example attempts to load a non-Test LMK using a single component when the security setting to enforce multiple components has been set to YES.

```
Secure-AUTH> LN <Return>
Enter LMK id: 00 <Return>
Enter comments: New LMK for ABC Bank <Return>
Load new LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: n <Return>
Check: AAAAAA
Invalid key - Multiple key components required
Secure-AUTH>
```

Example 4: (3DES Key Block LMK)

This example loads a 3DES key block LMK from smartcards and installs it as 'new' LMK 01.

```
Secure-AUTH> LN <Return>
Enter LMK id: 01 <Return>
Enter comments: New LMK for XYZ Bank <Return>
Load new LMK from components.
Insert card and enter PIN: ******** <Return>
Check: AAAAAA
Load more components? [Y/N]: Y <Return>
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all new component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 01
LMK key scheme: Key block
LMK algorithm: 3DES (3key)
LMK status: Live
Comments: New LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Example 5: (AES Key Block LMK)

This example loads an AES key block LMK from smartcards and installs it as 'new' LMK 02.

```
Secure-AUTH> LN <Return>
Enter LMK id: 02 <Return>
Enter comments: New LMK for XYZ Bank <Return>
Load new LMK from components.

Insert card and enter PIN: ******** <Return>
Check: AAAAAA
```

Remove the smartcard. Insert the second and subsequent smartcards and repeat the loading procedure until all new component sets have been loaded. When all components have been loaded and the HSM displays the LMK Check value, ensure that this is the expected value.

```
LMK Check: ZZZZZZ
LMK id: 02
LMK key scheme: Key block
LMK algorithm: AES-256
LMK status: Live
Comments: New LMK for XYZ Bank
Confirm details? [Y/N]: Y <Return>
Secure-AUTH>
```

Verify LMK Store

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **V**

Function: To confirm that the check value is identical to the value that was recorded when the LMK set was installed.
For Variant LMKs, the length of the displayed check value is determined by the CS (Configure Security) setting "Restrict Key Check Value to 6 hex chars".
For Key Block LMKs, the length of the displayed check value is always 6 hex digits.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- LMK Identifier: 2 numeric digits.

Outputs:
- Master key check value.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.

Example:

```
Online> V <Return>
Enter LMK id: 03 <Return>
Check: ZZZZZZ
Online>
```

Duplicate LMK Component Sets

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **DC**

Function: To copy an LMK component onto another smartcard.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Smartcard (RLMKs are supported) with LMK component.
- PIN for the smartcard. PINs must be entered within 60 seconds of being requested.

Outputs:
- LMK check value.

Errors:
- Load failed check comparison - card is blank
- Not a LMK card - card is not formatted for LMK or key storage.
- Card not formatted - card is not formatted
- Smartcard error; command/return: 0003 - invalid PIN is entered
- Invalid PIN; re-enter - a PIN of less than 4 or greater than 8 is entered.
- Warning - card not blank. Proceed? [Y/N] - LMK card is not blank
- Overwrite LMK set? [Y/N] - the smartcard already contains an LMK component. It can be overwritten if desired.

Example:

```
Secure> DC <Return>
Insert card to be duplicated and enter PIN: ******** <Return>
Insert blank card and enter PIN: ******** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ
Make another copy? [Y/N]: N <Return>
Secure>
```

Delete LMK

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Required** Activity: **admin.console**		

Command: **DM**

Function: To delete a selected LMK and (if loaded) the LMK in the corresponding location in key change storage.

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **admin.console** must be authorized, using the Authorizing Officer cards of the specified LMK.

Inputs:
- LMK Identifier: 2 numeric digits.

Outputs:
- Display of relevant entry from LMK table and the key change storage table.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Command only allowed from Secure-Authorized - the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.
- LMK id xx is the Default and Management LMK ID – the default and Management LMKs cannot be deleted.

Notes:
- LMKs which are the Default or Management LMK cannot be deleted. They Default and Management LMK must be re-assigned to a new LMK before the desired LMK can be deleted. (The LMK ID of the Management and default LMKs can be viewed by running the QS command.)

Example:

```
Secure-AUTH> DM <Return>
Enter LMK id: 01 <Return>

LMK table entry:
LMK table:
ID Auth       Scheme    Algorithm  Status Check  Comments
01 No         Key Block 3DES(3key) Test   999999 Test LMK for XYZ Bank

Key change storage table:
ID Old/New    Scheme    Algorithm  Status Check  Comments
01 Old        Variant   3DES(2key) Test   876543 Old test LMK for XYZ Bank

Confirm LMK deletion [Y/N]: Y <Return>
LMK deleted from main memory and key change storage

Secure>
```

Delete 'Old' or 'New' LMK from Key Change Storage

Variant ☑	Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**	

Command: **DO**

Function: To delete a selected LMK from key change storage. This command may only be used if an LMK is loaded in the corresponding location in main LMK memory.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- LMK Identifier: 2 numeric digits.

Outputs:
- Display of relevant entry from the key change storage table.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.

Example:

```
Secure> DO <Return>
Enter LMK id: 01 <Return>

Key change storage table entry:
ID  Old/New  Scheme    Algorithm  Status Check  Comments
01  Old      Variant   3DES(2key) Test   876543 Old test LMK for XYZ Bank

Confirm LMK deletion [Y/N]: Y <Return>
LMK deleted from key change storage

Secure>
```

View LMK Table

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **VT**

Function: To display the LMK table and the corresponding table for key change storage.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- Displayed LMK table and key change storage table.
- For each LMK currently installed, the following information is displayed:
  - ID – identifier selected during installation of this LMK.
  - Auth – current authorized status:
    - No – not authorized state/activities not active;
    - Yes – authorized state is active;
    - Yes (nX) – 'n' authorized activities are active (if HSM is configured for multiple authorized activities), with X identifying whether ***H***ost or ***C***onsole commands.
    - (Note that LMKs in key change storage cannot be authorized.)
  - Old/New – Status of key in Key Change Storage
    - Old – key is treated as an 'old' LMK
    - New – key is treated as a 'new' LMK
    - (Note that only LMKs held in Key Change Storage have the Old/New status.)
  - Scheme – The LMK scheme:
    - Variant – indicating a Variant LMK
    - Key Block – indicating a Key Block LMK
  - Algorithm – the LMK algorithm:
    - 3DES (2key) – indicating a double-length TDES Variant LMK
    - 3DES (3key) – indicating a triple-length TDES Variant or triple-length (3DES) Key Block LMK
    - AES-256 – indicating an AES Key Block LMK.
  - Status – the LMK status, selected during generation of the LMK.
    - Live – LMK is a 'live' LMK.
    - Test – LMK is a 'test' LMK.
  - Check – the check value of the LMK.
  - Comments – the comments entered during installation of this LMK.

Errors: None.

Example 1: *The HSM is configured for single authorized state, but has not been authorized*:

```
Secure> VT <Return>

LMK table:

ID Authorized  Scheme   Algorithm  Status Check  Comments
00 No          Variant  3DES(2key) Test   268604 test variant

Key change storage table:
No keys loaded in key change storage


Secure>
```

Example 2: *The HSM is configured for single authorized state, and both host and console commands are authorized for LMK 01*:

```
Secure> VT <Return>

LMK table:

ID Authorized  Scheme   Algorithm  Status Check  Comments
00 No          Variant  3DES(2key) Test   268604 test variant
01 Yes(H,C)    Variant  3DES(2key) Test   268604 test variant
02 Yes(1H,1C)  Variant  3DES(3key) Live   554279 Production 1
Key change storage table:
No keys loaded in key change storage

Secure>
```

Example 3: *The HSM is configured for single authorized state, and only host and commands are authorized for LMK 01 (console command authorization has automatically expired after 12 hours)*:

```
Secure> VT <Return>

LMK table:

ID Authorized  Scheme   Algorithm  Status Check  Comments
00 No          Variant  3DES(2key) Test   268604 test variant
01 Yes(H)      KeyBlock AES-256    Live   963272 Mngmnt LMK

Key change storage table:
No keys loaded in key change storage

Secure>
```

Example 4: *The HSM is configured for multiple authorized activities. Output shows how many host and console commands are authorized for each LMK:*

```
Online-AUTH> VT <Return>

LMK table:
ID Authorized  Scheme   Algorithm  Status Check  Comments
00 Yes(1H,1C)  Variant  3DES(2key) Test   268604 For RST Bank
01 No          KeyBlock 3DES(3key) Test   999999 For XYZ Bank
02 Yes(1H,1C)  Variant  3DES(3key) Live   554279 Production 1
03 Yes(0H,1C)  KeyBlock AES-256    Live   963272 Mngmnt LMK

Key change storage table:
ID Old/New     Scheme   Algorithm  Status Check  Comments
01 Old         Variant  3DES(2key) Test   876543 For XYZ Bank
02 New         Variant  3DES(2key) Live   448796 Old LMK for Production 1

Online-AUTH>
```

Generate Test LMK

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **GT**

Function: To generate one of the standard Thales Test LMKs, and write the component(s) to smartcard(s).
The payShield 9000 supports four different types of LMK:
- 2DES Variant LMK
- 3DES Variant LMK
- 3DES Key Block LMK
- AES Key Block LMK

All three DES-based Test LMKs can be stored on a single smartcard; the AES Test LMK requires two smartcards.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- Type of Test LMK to be generated.
- Prompts for smartcards to be inserted & PINs to be entered.

Outputs:
- Confirmation of Test LMK components being written to smartcards.
- Prompts to make additional copies.

Errors:
- Invalid selection.
- Invalid PIN.

Example 1: *This example writes the standard 2DES Variant Thales Test LMK to a single smartcard:*

```
Online> GT <Return>

Generate Standard Thales Test LMK Set:
  1 - 2DES Variant
  2 - 3DES Variant
  3 - 3DES KeyBlock
  4 - AES KeyBlock
Select Standard Thales Test LMK set to be generated: 1 <Return>
Insert blank card and enter PIN: **** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ

Make another copy? [Y/N]: N <Return>
1 copies made.


Online>
```

Example 2: *This example writes the two components of the standard AES Key Block Thales Test LMK to two separate smartcards:*

```
Online> GT <Return>

Generate Standard Thales Test LMK Set:
  1 - 2DES Variant
  2 - 3DES Variant
  3 - 3DES KeyBlock
  4 - AES KeyBlock
Select Standard Thales Test LMK set to be generated: 4 <Return>
Insert blank card and enter PIN: **** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ
Insert blank card and enter PIN: **** <Return>
Writing keys...
Checking keys...
Device write complete, check: ZZZZZZ

Online>
```

Chapter 4 – Operational Commands

Authorization Commands

The payShield 9000 HSM needs to be authorized for certain commands to be executed - usually those involving clear text data.

There are two methods of authorizing the HSM – using:

- a single Authorized State;
- multiple Authorized Activities.

Note: The console command CS (Configure Security) setting "Enable multiple authorized activities" determines which method is to be used; by default, multiple Authorized Activities are used.

If the HSM needs to be placed in Authorized State using the Authorizing Officer cards (or passwords) corresponding to a particular LMK, then the command will only be authorized for that particular LMK identifier. For example, if the "FK" console command ("Form Key from Components") is authorized using the passwords corresponding to the LMK with identifier "00", then only keys encrypted using LMK "00" may be formed using the command.

It is possible to authorize the HSM using multiple Authorizing Officer cards (or passwords), so that the HSM may be simultaneously authorized for different LMKs.

Note: For PCI HSM compliance, PINs and smartcards must be used to authenticate the Security Officers: passwords must not be used.

The payShield 9000 HSM provides the following console commands to support the authorization of the HSM:

Enter the Authorized State

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **A**

Function: To set the HSM into the Authorized State.
The HSM prompts for either Smartcards or Passwords, as applicable, which must correspond to the LMK being authorized.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- LMK Identifier: 1 or 2 numeric digits.
- PIN (if applicable): 5 to 8 alphanumeric characters. The PIN must be entered within 60 seconds. (4-digit PINs on legacy cards will also be accepted.)
- Either:
  - Smartcards (RLMKs are supported) with authorizing both passwords.
  - Password: 16 alphanumeric characters.

Outputs:
- Text messages as shown in examples.

Notes:
- If the CS setting "Card/Password authorization" is set to "Card", then the passwords required to put the HSM into the Authorized State will be read from smartcards. Note that only the first 2 LMK component cards contain passwords.
- This command is only available when the console command CS (Configure Security) setting "Enable multiple authorized activities [Y/N]" is set to "N".
- For PCI HSM compliance (see Chapter 10 of the *payShield 9000 General Information Manual*), authentication must use smartcards and PINs, not passwords.
- Use of this command will always cause an entry to be made in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.
- Console commands remain authorized for 12 hours (720 minutes) – see Chapter 10 of the *payShield 9000 General Information Manual*.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Card not formatted - card is not formatted.
- Not a LMK card - card is not formatted for LMK or key storage.
- Smartcard error; command/return: 0003 - invalid PIN is entered.
- Invalid PIN; re-enter - a PIN of less than 5 or greater than 8 digits is entered.
- Data invalid; please re-enter - the password is an invalid length.

Example 1: *This example authorizes the HSM using smartcards.*

```
Online> A <Return>
Enter LMK id [0-9]: 00 <Return>
First Officer:
Insert card and enter PIN: ******** <Return>
Second Officer:
Insert card and enter PIN: ******** <Return>
AUTHORIZED
Console authorizations will expire in 720 minutes (12 hours).

Online-AUTH>
```

Example 2: *This example authorizes the HSM using passwords.*

```
Online> A <Return>
Enter LMK id [0-4]: 1 <Return>
First Officer:
Password: ********** <Return>
Second Officer:
Password: ********** <Return>          ← Password too long
Data invalid; please re-enter: ********** <Return>
AUTHORIZED
Console authorizations will expire in 720 minutes (12 hours).

Online-AUTH>
```

Cancel the Authorized State

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **C**

Function: To cancel the Authorized State.
There is an equivalent command available to the host (Host command 'RA')

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- LMK Identifier: 2 numeric digits.

Outputs:
- Text messages as shown in example.

Notes:
- This command is only available when the console command CS (Configure Security) setting "Enable multiple authorized activities [Y/N]" is set to "N".
- Use of this command will always cause an entry to be made in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.

Example 1:

```
Online-AUTH> C <Return>
Enter LMK id [0-9]: 00 <Return>
NOT AUTHORIZED for LMK id 00
Online>
```

Authorize Activity

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **A**

Function: To authorize the HSM to perform certain specified activities.

In command line mode, the operator specifies which activities are to be authorized.

In menu mode, the operator is prompted to enter the activities.

In both cases, the specified activities are authorized by submitting two Security Officer cards or passwords, which must correspond to the LMK being authorized.

Authorized activities can be made persistent, in which case they are retained even if the power to the HSM is cycled.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- LMK Identifer: 2 numeric digits
- Activities to be authorized.
- Timeout value: Number of minutes before HSM will revoke chosen authorized activity. Where the security setting *Enforce Authorization Time Limit* has been set to "YES" (i.e. to the PCI HSM compliant value) then console commands can be authorized for a maximum period of 12 hours (720 minutes).
- PIN (if applicable): 5 to 8 alphanumeric characters. The PIN must be entered within 60 seconds of being requested. (4-digit PINs on legacy cards will also be accepted.)
- Either:
  - Smartcards (RLMKs are supported) with authorizing both passwords.
  - Password: 16 alphanumeric characters.
- Use "-h" to display help.

Outputs:
- Text messages as shown in examples.

Syntax: Syntax: **A** [<*Activity*>] [<*Activity*>] ...

Activity: <*Category*>.[<*Sub-category*>].[<*Interface*>][:<*Timeout*>]

Category = generate|component|genprint|import|export|pin|audit|admin|diag|misc| command

Sub-category (for 'generate|import|export') = key type code, e.g. 001 for ZPK.

Sub-category (for 'pin') = mailer|clear

Interface = host|console

Timeout = value in minutes or 'p' for persistent. (A maximum of 12 hours (720 minutes) is applied to Console commands.}

Names may be shortened but must remain unique.

Errors:

- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Card not formatted - card is not formatted.
- Not a LMK card - card is not formatted for LMK or key storage.
- Smartcard error; command/return: 0003 - invalid PIN is entered.
- Invalid PIN; re-enter - a PIN of less than 4 or greater than 8 is entered.
- Data invalid; please re-enter: the password is an invalid length.

Notes:

- If the CS setting "Card/Password authorization" is set to "Card", then the passwords required to put the HSM into the Authorized State will be read from smartcards. Note that only the first 2 LMK component cards contain passwords.
- This command is only available when the console command CS (Configure Security) setting "Enable multiple authorized activities [Y/N]" is set to "Y".
- For PCI HSM compliance (see Chapter 10 of the *payShield 9000 General Information Manual*), the following security settings must be set:
    - user authentication must be by smartcard and PIN, and not by using passwords.
    - Authorization time limit for Console commands must be enforced.
- Where the security setting *Enforce Authorization Time Limit* has been set to "YES" (i.e. to the PCI HSM compliant value) then console commands can be authorized for a maximum period of 12 hours (720 minutes).
- Use of this command will always cause an entry to be made in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.
- Activities are described in terms of four fields: Category, Sub-Category, Interface and Timeout. If the Timeout field is omitted, the activity remains authorized until cancelled either by the console command "C" or the host command "RA".
- Omitting either the Sub-Category and/or the Interface field is equivalent to authorizing multiple activities consisting of all possible combinations of valid values for the missing fields. For clarification:

  pin.mailer

is equivalent to:

pin.mailer.host

pin.mailer.console

and:

pin

is equivalent to:

pin.clear.console

pin.clear.host

pin.mailer.console

pin.mailer.host

- When authorizing activities, two (or more) activities may overlap, for example:

  pin

  pin.mailer

- There is no requirement to attempt to reduce activities to the minimum set. The list of authorized activities simply consists of all those entered (and authorized) by the user.
- There is one case when it will be necessary to overwrite an existing activity: when only the Timeout field changes. For example, suppose that the following activity is authorized:

  export.001.console:11

  and the user uses the 'A' command to authorize the following activity:

  export.001.console:60

  then this should overwrite the first one (even if the newer activity has a shorter *Timeout* value).
- Note: When omitting the sub-category, but including the interface, there should be two delimiters "." between them:

  Example: export..host allows export of any (valid) key using a host command.
- The option to make an authorization persistent (i.e. to survive across a re-boot of the HSM) is only available for Host commands and where the authorization is also permanent.

Example 1: (Variant or Key Block LMK)

This example authorizes a single activity via the menu.

```
Online> A <Return>
Enter LMK id [0-9]: 0 <Return>
No activities are authorized for LMK id 00.
List of authorizable activities:
generate       genprint      component     import
export         pin           audit         admin
diagnostic     misc          command
Select category: pin <Return>
clear          mailer
Select sub-category, or <RETURN> for all: mailer <Return>
host           console
```

```
Select interface, or <RETURN> for all: <Return>
Enter time limit for pin.mailer, or <RETURN> for permanent:
<Return>
Make activity persistent? [Y/N]: N <Return>
Enter additional activities to authorize? [y/N]: N <Return>

The following activities are pending authorization for LMK id
00:
pin.mailer

First Officer:
Insert Card for Security Officer and enter the PIN: ********
<Return>
Second Officer:
Insert Card for Security Officer and enter the PIN: ********
<Return>

The following activities are authorized for LMK id 00:
pin.mailer

Online-AUTH>
```

Example 2: (Variant or Key Block LMK)

This example authorizes activities via the command line, with no time limits specified.

```
Online> A gene comp genp i e p au ad di m comm<Return>

Enter LMK id [0-4]: 0 <Return>

Console authorizations will expire in 720 minutes (12 hours).

The following activities are pending authorization for LMK id
00:

admin..console:720
admin..host
audit..console:720
audit..host
command..console:720
command..host
component..console:720
component..host
diagnostic..console:720
diagnostic..host
export..console:720
export..host
generate..console:720
generate..host
genprint..console:720
genprint..host
import..console:720
import..host
misc..console:720
misc..host
pin..console:720
pin..host

First officer:
Insert card and enter PIN: ********<Return>

Second officer:
Insert card and enter PIN: ********<Return>

The following activities are authorized for LMK id 00:
```

```
admin..console:720  (720 mins remaining)
admin..host
audit..console:720  (720 mins remaining)
audit..host
command..console:720  (720 mins remaining)
command..host
component..console:720  (720 mins remaining)
component..host
diagnostic..console:720  (720 mins remaining)
diagnostic..host
export..console:720  (720 mins remaining)
export..host
generate..console:720  (720 mins remaining)
generate..host
genprint..console:720  (720 mins remaining)
genprint..host
import..console:720  (720 mins remaining)
import..host
misc..console:720  (720 mins remaining)
misc..host
pin..console:720  (720 mins remaining)
pin..host

Online-AUTH>
```

Example 3: (Variant LMK)

This example authorizes three activities additional Example 1 via the menu.

```
Online-AUTH> A <Return>
Enter LMK id [0-9]: 00 <Return>
The following activities are authorized for LMK id 00:
pin.mailer
List of authorizable activities:
generate        genprint        component       import
export          pin             audit           admin
diagnostic      misc            command
Select category: generate <Return>
000             100             200             001
002             400             003             006
008             009             109             209
309             409             509             709
00a             00b             rsa
Select sub-category, or <RETURN> for all: 000 <Return>
host            console
Select interface, or <RETURN> for all: C <Return>
Enter time limit for generate.000.console, or <RETURN> for
permanent: 60 <Return>

Enter additional activities to authorize? [y/N]: Y <Return>
List of authorizable activities:
generate        genprint        component       import
export          pin             audit           admin
diagnostic      misc            command
Select category: export <Return>
000             100             200             001
002             400             003             006
008             009             109             209
309             409             509             709
00a             00b             rsa
Select sub-category, or <RETURN> for all: 001 <Return>
host            console
Select interface, or <RETURN> for all: H <Return>
Enter time limit for export.001.host, or <RETURN> for
permanent: <Return>
Make activity persistent? [Y/N]: n <Return>

Enter additional activities to authorize? [y/N]: Y <Return>
List of authorizable activities:
generate        genprint        component       import
export          pin             audit           admin
diagnostic      misc            command
Select category: admin <Return>
host            console
Select interface, or <RETURN> for all: c <Return>
Enter time limit for admin, or <RETURN> for permanent: 240
<Return>

Enter additional activities to authorize? [y/N]: n <Return>
The following activities are pending authorization for LMK id
00:
admin..console:240
export.001.host
generate.000.console:60

First Officer
Insert Card for Security Officer and enter the PIN: ****
<Return>
Second Officer
Insert Card for Security Officer and enter the PIN: ****
<Return>

The following activities are authorized for LMK id 00:
admin:240 (240 mins remaining)
export.001.host
generate.000.console:60 (60 mins remaining)
```

```
pin.mailer

Online-AUTH>
```

Example 4: (Variant LMK)

This example authorizes three activities additional to Example 1 via the command line, including time limits.

```
Online-AUTH> A gene.000.con:60 exp.001.host:p admin:240
<Return>
Enter LMK id [0-19]: 00 <Return>

The following activities are pending authorization for LMK id
00:

admin:240
export.001.host:persistent
generate.000.console:60

First Officer:
Insert Card for Security Officer and enter the PIN: ****
<Return>
Second Officer:
Insert Card for Security Officer and enter the PIN: ****
<Return>

The following activities are authorized for LMK id 01:

admin:240  (240 mins remaining)
export.001.host:persistent
generate.000.console:60  (60 mins remaining)

Online-AUTH>
```

Example 5: (Variant or Key Block LMK)

This example authorizes a single activity via the command line.

```
Online> A pin.clear <Return>
Enter LMK id [0-9]: 01 <Return>

Console authorizations will expire in 720 minutes (12 hours).

The following activities are pending authorization for LMK id
01:

pin.clear.console:720
pin.clear.host

First Officer:
Insert Card for Security Officer and enter the PIN: ****
<Return>
Second Officer:
Insert Card for Security Officer and enter the PIN: ****
<Return>

The following activities are authorized for LMK id 01:

pin.clear.console:720  (720 mins remaining)
pin.clear.host

Online-AUTH>
```

Example 6:

This example authorizes an additional three activities via the menu.

(Key Block LMK)

```
Online-AUTH> A <Return>
Enter LMK id [0-9]: 01 <Return>
The following activities are authorized for LMK id 01:
pin.clear
List of authorizable activities:
generate        genprint        component       import
export          pin             audit           admin
diagnostic      misc            command
Select category: export <Return>

01              B0              C0              11
12              13              D0              21
22              E0              E1              E2
E3              E4              E5              E6
31              32              K0              51
52              M0              M1              M2
M3              M4              M5              61
62              63              64              65
P0              71              72              73
V0              V1              V2
Select sub-category, or <RETURN> for all: 72 <Return>
host            console
Select interface, or <RETURN> for all: C <Return>
Enter time limit for export.72.console, or <RETURN> for
permanent: 60 <Return>

Enter additional activities to authorize? [y/N]: Y <Return>
List of authorizable activities:
generate        genprint        component       import
export          pin             audit           admin
diagnostic      misc            command
Select category: admin <Return>
host            console
Select interface, or <RETURN> for all: <Return>
Enter time limit for admin, or <RETURN> for permanent: 240
<Return>

Enter additional activities to authorize? [y/N]: Y <Return>
List of authorizable activities:
generate        genprint        component       import
export          pin             audit           admin
diagnostic      misc            command
Select category: misc <Return>
host            console
Select interface, or <RETURN> for all: c <Return>
Enter time limit for admin, or <RETURN> for permanent:
<Return>
Make activity persistent? [Y/N]: n <Return>

Enter additional activities to authorize? [y/N]: n <Return>
The following activities are pending authorization for LMK id
00:
misc..console
admin:240
export.72.console:60

First Officer
Insert Card for Security Officer and enter the PIN: ****
<Return>
Second Officer
Insert Card for Security Officer and enter the PIN: ****
<Return>

The following activities are authorized for LMK id 01:
misc..console
admin:240 (240 mins remaining)
export.72.console (60 mins remaining)
pin.clear
```

```
Online-AUTH>
```

Example 7: (Key Block LMK)

This example authorizes an additional three activities via the command line.

```
Online-AUTH> a exp.001.con:60 admin:240 misc..console
<Return>
Enter LMK id [0-1]: 01 <Return>

Console authorizations will expire in 720 minutes (12 hours).

The following activities are pending authorization for LMK id
01:

admin:240
export.001.console:60
misc..console:720

First Officer:
Insert Card for Security Officer and enter the PIN: ****
<Return>
Second Officer:
Insert Card for Security Officer and enter the PIN: ****
<Return>

The following activities are authorized for LMK id 01:

admin:240  (228 mins remaining)
export.001.console:60  (60 mins remaining)
export.001.host:persistent
generate.000.console:60  (48 mins remaining)
misc..console:720  (720 mins remaining)
pin.clear.console:720  (712 mins remaining)
pin.clear.host

Online-AUTH>
```

Cancel Authorized Activity

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **C**

Function: To cancel one or more Authorized Activities.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- LMK Identifier: 2 numeric digits.

Outputs:
- Text messages as shown in examples.

Notes:
- This command is only available when the console command CS (Configure Security) setting "Enable multiple authorized activities [Y/N]" is set to "Y".

Syntax: **C** [<*Activity*>] [<*Activity*>] ...

Activity: <*Category*>[.<*Sub-category*>][.<*Interface*>][:<*Timeout*>]

Category = generate|component|genprint|import|export|pin|audit|admin|diag|misc| command

Sub-category (for 'generate|import|export') = key name, e.g. TPK, MK-AC, etc.

Sub-category (for 'pin') = mailer|clear

Interface = host|console

Timeout = value in minutes or 'p' for persistent

Names may be shortened but must remain unique.

When canceling an authorized activity which includes a timeout, the original value of the timeout should be specified.

Note: When omitting the sub-category, but including the interface, there should be two delimiters "." between them:

Example: export..host allows export of any (valid) key using a host command.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Invalid input.

Notes:
- Use of this command will always cause an entry to be made in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Example 1: (Variant or Key Block LMK)

This example cancels an existing activity via the menu.

```
Online-AUTH> C <Return>
Enter LMK id [0-9]: 00 <Return>
Cancel pin.mailer? [y/N] Y <Return>
No activities are authorized for LMK id 00.
Online>
```

Note: This example assumes that the activities in the Authorize Activity command Example 1 (above) are active.

Example 2: (Variant or Key Block LMK)

This example cancels an existing activity via the command line.

```
Online-AUTH> C pin.mailer <Return>
Enter LMK id [0-1]: 00 <Return>
No activities are authorized for LMK id 00.
Online>
```

Note: This example assumes that the activities in the Authorize Activity command Example 2 (above) are active.

Example 3: (Variant LMK)

This example cancels an existing activity via the menu.

```
Online-AUTH> C <Return>
Enter LMK id [0-4]: 00 <Return>
Cancel admin:240 (194 mins remaining) ? [y/N] Y <Return>
Cancel export.001.host? [y/N] N <Return>
Cancel generate.000.console:60 (14 mins remaining)? [y/N] Y
<Return>
Cancel pin.mailer? [y/N] N <Return>
The following activities are authorized for LMK id 00:
export.001.host
pin.mailer
Online-AUTH>
```

Note: This example assumes that the activities in the Authorize Activity command Example 3 (above) are active.

Example 4: (Variant LMK)

This example cancels an existing activity via the command line.

```
Online-AUTH> C gene.000.c admin <Return>
Enter LMK id [0-9]: 00 <Return>
The hollowing activities are authorized for LMK id 00.
export.001.host
pin.mailer
Online-AUTH>
```

Note: This example assumes that the activities in the Authorize Activity command Example 4 (above) are active.

Example 5: (Variant or Key Block LMK)

This example cancels an existing activity via the command line.

```
Online-AUTH> C pin.clear <Return>
Enter LMK id [0-9]: 01 <Return>
No activities are authorized for LMK id 01.
Online>
```

Note: This example assumes that the activities in the Authorize Activity command Example 5 (above) are active.

View Authorized Activities

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **VA**

Function: To view all active authorized activities.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- LMK identifier: 2 numeric digits.

Outputs:
- List of active authorized activities.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.

Example 1: (Multiple authorized activities enabled)

This example applies when multiple authorized activities has been enabled..

```
Online-AUTH> VA <Return>
Enter LMK id: 00 <Return>
The following activities are authorized for LMK id 00:
admin:240 (228 mins remaining)
export.001.host:persistent
generate.000.console:60 (48 mins remaining)

Online-AUTH>
```

Note: This example assumes the activities in the Authorize Activity command Example 4 (above) were authorized 12 minutes ago.

Example 2: (Multiple authorized activities disabled)

This example applies when multiple authorized activities has not been enabled..

```
Online-AUTH> VA <Return>
Enter LMK id [0-9]: 0 <Return>
LMK id 00 is authorized.
Console authorization expires in 716 minute(s).

Online-AUTH>
```

Note: This example assumes that authorized state was enabled 4 minutes ago.

Logging Commands

An Error Log and an Audit Log are provided, each with a command to display the log and a command to clear the log. There is also a command to enable the user to set their time zone, so that the correct time is displayed in audit log reports.

The Error log stores fault information for use by Thales e-Security support personnel. The error log is used to log unexpected software errors, hardware failures and alarm events. Whenever an error occurs, that error code is stored, along with the time, date and severity level. Additional errors that have the same error code cause the time and date of that code to be updated. In this way, each error type remains in the log (with the most recent time and date) and is not lost. The severity levels are:

- Informative (0) Something abnormal happened, but was not important.
- Recoverable (1) Something abnormal happened, but the unit recovered from it without rebooting or losing data.
- Major (2) Something abnormal happened, but the unit recovered from it but may have lost data/information due to restarting a process or re-initializing hardware. The unit may not function in a full capacity.
- Catastrophic (3) Something abnormal happened, and the unit had to reboot to recover.

Only catastrophic errors cause the HSM to reboot. New errors cause the Fault LED on the front panel to flash.

Whenever the HSM state is altered through power-up, key-lock changes or console commands, the Audit log is updated with the action and the time and date. The Audit log can also be configured to record execution of almost any console or host command. The Audit log records state changes until it is 100% full and for each subsequent state change the earliest (i.e. oldest) record in the log is deleted to make room for the new record. A number of host commands are provided which allow the host computer to extract and archive (print) audit records from the HSM.

Management of the Audit journal is performed from the console using the command 'AUDITOPTIONS', whilst 'AUDITLOG' is used to retrieve the log and 'CLEARAUDIT' to clear the log. The HSM must be put into the secure-authorized state in order to execute the 'AUDITOPTIONS' and 'CLEARAUDIT' console commands.

Note: Auditing host or console commands may impact HSM performance.

The payShield 9000 HSM provides the following console commands to support storage and retrieval of HSM settings:

Display the Error Log

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **ERRLOG**

Function: To display the entries in the error log.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- A listing of the errors in the error log, or text message: "Error log is empty".

Errors: None.

Notes: In software versions up to v2.1, power supply errors are added to the error log only when the HSM is restarted. From v2.2 onwards, power supply errors are logged as soon as they are detected.

Example 1: *In this example, there are no entries in the error log.*

```
Offline> ERRLOG <Return>
Error log is empty
Offline>
```

Example 2: *In this example, the Security setting "Allow Error light to be extinguished when viewing Error Log?" is set to NO.*

```
Offline> ERRLOG <Return>
Error Log (3 entries)
-------------------------
1: May 01 09:35:00  ERROR (1): Invalid queue size (Severity: 2, Code =
00000001, Sub-code = 00000002)
2: May 01 09:35:02  ERROR (1): Key3 cannot be specified without key2
(Severity: 0, Code = 00000004, Sub-code = 00000003)
3: May 06 13:55:00  ERROR: [Power Supply:    FAILED (PSU 2 Failed) ]
(Severity: 3, Code = 0x00000001, Sub-Code = 0x0000000E)

Please copy this log to a text file and send it
to your regional Thales E-Security Support center.

Offline>
```

Example 3: *In this example, the Security setting "Allow Error light to be extinguished when viewing Error Log?" is set to YES.*

```
Offline> ERRLOG <Return>
Error Log (3 entries)
-------------------------
1: May 01 09:35:00  ERROR (1): Invalid queue size (Severity: 2, Code =
00000001, Sub-code = 00000002)
2: May 01 09:35:02  ERROR (1): Key3 cannot be specified without key2
(Severity: 0, Code = 00000004, Sub-code = 00000003)
3: May 06 13:55:00  ERROR: [Power Supply:    FAILED (PSU 2 Failed) ]
(Severity: 3, Code = 0x00000001, Sub-Code = 0x0000000E)

Please copy this log to a text file and send it
to your regional Thales E-Security Support center.

Confirm error log has been read and error light should be
extinguished? [Y/N]: Y <Return>

Offline>
```

Clear the Error Log

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **CLEARERR**

Function: To clear the entries in the error log.

Authorization: The HSM must be in the secure state to run this command.

Inputs: None.

Outputs:
- A confirmation message.

Errors: None.

Example:
```
Secure> CLEARERR <Return>
Error log Cleared
Secure>
```

Display the Audit Log

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **AUDITLOG**

Function: To display the entries in the audit log.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- A listing of the entries in the audit log.
  - For authorizations, the period of authorization of Console commands will be indicated by attaching text of the form ":123" (representing 123 minutes) to the identity of the authorized activity.
- The following text messages can be output:
  - Audit Log (in entries)
  - Continue displaying audit log entries? Yes/No/Continuous

Notes:
- Certain items are always recorded in the Audit Log, irrespective of the selections made using AUDITOPTIONS – for further information see Chapter 17 of the *payShield 9000 General Information Manual*. These are:
  - Serial numbers of smartcards used to authenticate users at the HSM or to payShield Manager.
  - Authorization of activities
  - Cancellation of authorization.
  - Key and component entry at the Console or payShield Manager.

  When key and component entry are forcibly logged in this way, the log entry indicates successful completion of the action.
  The user can, as in earlier versions of software, use AUDITOPTIONS to specify that the key and component entry commands are logged: this will normally result in 2 entries in the audit log – one resulting from the AUDITOPTIONS setting indicating that the command was initiated, and the forcible logging indicating the successful completion of the command. If the command does not complete successfully (e.g. because it was cancelled by the user) then there will be no forcible logging, but the entry indicating the command was initiated will still be there if the command was specified in AUDITOPTIONS.
- The Audit Log is now displayed with the most recent entries shown first: up to software version 2.1 the Audit Log was displayed with oldest entries first. This change has been made because, with a maximum length of 50,000 records, it can take a long time to display the complete Audit Log because of the speed limitations of serial connections.

Errors: None.

Example 1:

```
Offline> AUDITLOG <Return>
Audit log is empty
Offline>
```

Example 2:

```
Offline> AUDITLOG <Return>
Audit Log (10 entries)
Counter    Time      Date         Command/Event
------------------------------------------------------------
0000000268  13:55:00  02/Jul/2013  Diagnostic self test failure: Power
0000000267  16:45:07  01/Jul/2013  Authorized activity admin..host was
cancelled for LMK id 0
0000000266  16:45:05  01/Jul/2013  Authorized activity
admin..console:123 was cancelled
0000000265  15:54:02  01/Jul/2013  Key I/O command BK executed
0000000264  15:35:55  01/Jul/2013  Activity component..console:123 was
authorized for LMK id 0
0000000263  15:08:48  01/Jul/2013  Smartcard activated: 20025151
0000000262  15:08:48  01/Jul/2013  Smartcard activated: 20025132
0000000261  10:42:32  01/Jul/2013  Host command CA, response 00
0000000260  10:36:03  01/Jul/2013  Host command CA, response 69
0000000259  10:34:57  01/Jul/2013  System restarted
0000000258  10:32:48  01/Jul/2013  Keylock turned to Online
0000000257  10:32:21  01/Jul/2013  Console command CH
0000000256  09:01:56  01/Jul/2013  Diagnostic self tests passed.

Offline>
```

After 20 entries are displayed continuously, the following text is displayed:

```
Continue displaying audit log entries? [Y/N/C]:
```

Clear the Audit Log

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Required** Activity: **audit.console**		

Command: **CLEARAUDIT**

Function: To clear the entries in the audit log.

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **audit.console** must be authorized, using the Authorizing Officer cards of the Management LMK.

Inputs: None.

Outputs:
- One of the following text messages:
  - Audit Log Cleared
  - Audit Log is empty

Errors:
- Command only allowed from Secure-Authorized - the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.

Example 1:

```
Secure-AUTH> CLEARAUDIT <Return>
Warning! The HSM's audit log contains entries that have not yet
been printed.
Please confirm that you wish to delete the entire audit log.
[Y/N]: Y <Return>
Audit Log Cleared

Secure-AUTH>
```

Audit Options

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Required** Activity: **audit.console**		

Command: **AUDITOPTIONS**

Function: To configure the HSM's auditing functionality.
The HSM can be configured to monitor and record the following events:
- Execution of individual host command
- Execution of individual console command
- User interactions, including:
  - System restart (e.g. power cycle)
  - State transitions (i.e. Offline, Online, Secure)
  - LMK installation / erasure
  - Authorization activation/cancelling
- The running and result of automatic self tests.
- Error responses to Host commands
- Host connection failures resulting from deployment of Access Control Lists.
- Secure Host Communication session negotiation failures resulting from attempted use of out-of-date certificates.

Authorization: The HSM must be in the offline or secure state to use this command to change the items to be audited. Additionally, the HSM must be either in the Authorized State, or the activity **audit.console** must be authorized, using the Authorizing Officer cards of the Management LMK.
The current list of items being audited can be viewed in online state.

Inputs:
- Changes to configuration:
  - Audited console commands:
    - +CXX to enable auditing of console command XX
    - –CXX to disable auditing of console command XX

    The "?" character can be used as a wildcard when specifying the commands.
  - Audited host commands
    - +HXX to enable auditing of host command XX
    - –HXX to disable auditing of host command XX

    The "?" character can be used as a wildcard when specifying the commands.
  - Audit Error responses to Host Commands (Y/N)
  - Audit user actions (Y/N)
  - Audit counter value
  - Audit Utilization Data Resets (Y/N)
  - Audit Automatic Self testing (Y/N)
  - Audit ACL connection failures (Y/N)
  - Audit out-of-date Certificates for Secure Host Sessions (Y/N)

Outputs:
- Current & new configuration details:

- List of audited console commands
- List of audited host commands
- List of user actions
- Results of automatic self tests
- Audit counter value

Notes:

- Certain items are always recorded in the Audit Log, irrespective of the selections made using AUDITOPTIONS – See Chapter 17 of the *payShield 9000 General Information Manual*. These are:
  - Serial numbers of smartcards used to authenticate users at the HSM or to payShield Manager.
  - Authorization of activities
  - Cancellation of authorization.
  - Key and component entry at the Console or Payshield Manager. This relates to the following Console commands (or HSM equivalents):
    - BK Form a Key from Components
    - CV Generate a Card Verification Value
    - D Form a ZMK from Encrypted Components
    - DE Form a ZMK from Clear Components
    - FK Form Key from Components
    - IK Import a Key
    - IV Import a CVK or PVK
    - LK Load LMK
    - LO Move Old LMKs into Key Change Storage
    - PV Generate a Visa PIN Verification Value

  When key and component entry are forcibly logged in this way, the log entry indicates successful completion of the action.
  The user can, as in earlier versions of software, use AUDITOPTIONS to specify that the key and component entry commands are logged: this will normally result in 2 entries in the audit log – one resulting from the AUDITOPTIONS setting indicating that the command was initiated, and the forcible logging indicating the successful completion of the command. If the command does not complete successfully (e.g. because it was cancelled by the user) then there will be no forcible logging, but the entry indicating the command was initiated will still be there if the command was specified in AUDITOPTIONS.
- **Audit Error Responses to Host Commands**: this setting allows any relevant error responses to Host commands to be logged. In this context, "relevant" means error responses which may indicate situations that require investigation by the payShield 9000 Administrators or Security Officers. The use of this setting will therefore not log non-00 error responses which are purely for information or which indicate "business as usual" (e.g. a customer entering an incorrect PIN at a terminal). See *Appendix O* for information on which non-00 error responses are not logged.

- Auditing items (such as heavily used Host commands) which result in a high rate of update to the Audit Log will impact negatively on performance of the HSM.
- After completing the AUDITOPTIONS command, a reboot of the HSM may be required in order to activate the new settings.

Errors:

- Command only allowed from Offline-Authorized - the HSM is not in Offline (or Secure) State, or the HSM is not authorized to perform this operation, or both.
- Invalid Entry - the value entered is invalid.
- Card not formatted to save/retrieve HSM settings - Attempt with another card? [Y/N]

Example:

```
Offline-AUTH> AUDITOPTIONS <Return>
List of Audited Console Commands:
GC, GS, EC, FK
List of Audited Host Commands:
A0, A4, GG, GY
Audit Error Responses to Host Commands:
Disabled
Audit User Actions:
Enabled
Audit Counter Value:
0000000253
Audited utilization data resets:
Enabled
Audited diagnostic self tests:
Disabled

Modify Audited Command List? [Y/N]: Y <Return>
Enter command code (e.g. +CDE) or Q to Quit: +CDE <Return>
Console command DE added to list
Enter command code (e.g. +CDE) or Q to Quit: -HA4 <Return>
Host command A4 removed from list
Enter command code (e.g. +CDE) or Q to Quit: Q <Return>

Audit Error Responses to Host Commands? [Y/N]: Y <Return>

Audit User Actions (Y/N): N <Return>

Audit ACL connection failures? [Y/N]: Y<Return>

Audit out-of-date Certificates for Secure Host sessions? [Y/N]:
Y<Return>

Current Audit Counter value is:    0000000253
Enter new value or <RETURN> for no change: 2000 <Return>

Audit Utilization Data Resets? [Y/N]: Y <Return>

Audit Automatic Self Testing? [Y/N]: Y <Return>

Audit User Actions: YES
Audit Error Responses to Host Commands: YES
Audit utilization data resets: YES
Audit diagnostic self tests: YES
Audit ACL connection failures: YES
Audit out-of-date Certificates for Secure Host Sessions:
YES
Audit Counter Value:
0000002000
```

```
List of Audited Console Commands:
GC, GS, EC, FK, DE
List of Audited Host Commands:
A0, GG, G

Audit Error Responses to Host Commands:
Enabled
Audit User Actions:
Disabled
Audit Counter Value:
00002AAF
Audited utilization data resets:
Enabled
Audited diagnostic self tests:
Enabled

Save Audit Settings to smartcard? [Y/N]: Y <Return>

Insert Card and press Enter: <Return>
Audit Settings written to the smartcard.
Offline-AUTH>
```

Print the Audit Log

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not Required**		

Command: **AUDITPRINT**

Function: To print the HSM's audit log at a printer attached to the HSM.

Authorization: Authorization is not required.

Inputs:
- Whether to print all records, or only unarchived records.

Outputs:
- A list of all the selected records showing the following data:
  - The sequential audit counter
  - The time of the event, in the format HHMMSS
  - The date of the event, in the format DDMMYY
  - The command description, including:
    - The command code type (H=Host, C=Console, F=Fraud Event, A=User Action)
    - The command or action code
    - For Host commands, the response error code.
  - Random MAC key used to generate the MAC
  - MAC calculated over the audit record.

  For more detail, see the Audit Record Format in Appendix B of the *payShield 9000 Host Command Reference Manual*.

  Example of output:

```
Counter  Time   Date   Command MAC Key                          MAC
1001CF33 135209 180511 H M2 06 AA69C75033EA50810209D24F17E93786 ACBC947DA5E06947
1001CF34 135209 180511 H M2 06 5D53F23A43A7AC692C77754FB00EBCA6 E3DFFE68209F4A1E
1001CF35 135209 180511 H M2 06 787C6FC766E544CD4A2EF56DB1DE1C14 D5321C3CF8E36DCB
1001CF36 135209 180511 H M2 06 34D3B4CE59DDC0BA4C128EF88721D50C 86D18019F2E1D717
1001CF37 135209 180511 H M2 06 F893D165B7CADC6DC44A59CF33F895FE C5C14C8D93892004
1001CF38 135209 180511 H M2 06 C364F9C499C89514A3EB6BBA75BC2C87 55BB024854727C41
1001CF39 135209 180511 H M2 06 D229ACB7F9C5EEA7FB55761EEB9947D7 BB6E67CA6DEF2584
1001CF3A 135209 180511 H M2 06 0F5A3BAB8A93FEC30E9C125E585FB005 1D84136FA9162B1B
1001CF3B 135209 180511 H M2 06 7F78D6858D729710477C0CEF18917281 CB6746ADAE4B65AC
1001CF3C 135209 180511 H M2 06 C1EA998068CD989A5383A8EA7B52EB1C F2B5A526C100EAB3
1001CF3D 135209 180511 H M2 06 5BA7D93E19DA1EEA14AAA1BCDB1CB45B 2DCF25D8E0DE381F
1001CF3E 135209 180511 H M2 06 9C019A9DF544F2F31300CCD54DF44DF1 7FA5EA6DA98043C9
1001CF3F 135209 180511 H M2 06 D4AD0D70A5EBFE61B5BAF2DC509FB478 36D504B7E837778B
1001CF40 135209 180511 H M2 06 B29D7E22350640A702255D1A024777AE C8495DF637BA3E6A
1001CF41 135209 180511 H M2 06 4B6C7887A7662663FDD76EEE6FE9BE27 749BD7153ADD5A01
1001CF42 135209 180511 H M2 06 A9048C7578CBE29227FA824AE51B0323 4FB59F661352A05B
1001CF43 135209 180511 H M2 06 27D6C576FE6F1B0537A51175777C5820 B6EE89EF4F65F7BC
```

Notes:
- Printing of an Audit Log record causes its "Archived" flag to be set.

Example:

```
Offline> AUDITPRINT <Return>

Print All records or just Unarchived records? [A/U]: A <Return>

Commence Printing? [Y/N]: Y <Return>

Printing complete: 2000 record(s) printed.

Offline>
```

Time and Date Commands

The SETTIME command is used to set the system time and date used by the payShield 9000 HSM for the audit log entries. The user should use this command to adjust the time for the local timezone. The time and date can be queried using the GETTIME command.

The payShield 9000 HSM provides the following console commands to support storage and retrieval of HSM settings:

Set the Time and Date

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Required** Activity: **admin.console**		

Command: **SETTIME**

Function: To set the system time and date used by the HSM.

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **admin.console** must be authorized, using the Authorizing Officer cards of the Management LMK.

Inputs:
- The time in hours and minutes.
- The date in year, month and day.

Outputs:
- Text messages, as in the example below.

Errors:
- Command only allowed from Secure-Authorized - the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.
- Response invalid. Re-enter - an invalid value has been entered.

Example:

```
Secure-AUTH> SETTIME <Return>
Enter hours [HH](24 hour format): 10 <Return>
Enter minutes [MM]: 08 <Return>
Enter year [YYYY] (2009 or above): 2014 <Return>
Enter month [MM]: 02 <Return>
Enter day [DD]: 12 <Return>
The system time has been modified.
Secure-AUTH>
```

 Setting the date or time back may prevent the payShield Manager from allowing a user to login. Care must be taken when changing the date back such that it is not earlier than the creation date/time of any of the smartcards that will be used to access the HSM.

Query the Time and Date

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **GETTIME**

Function: To query the system time and date.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- The year, month and date.
- The time in hours, minutes and seconds.

Errors: None.

Example:
```
Online> GETTIME <Return>
System date and time: Feb 12 10:08:19 2014
Online>
```

Set Time for Automatic Self-Tests

Variant ☑	Key Block ☑	
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **ST**

Function: Reports the time of day when the daily automatic self-tests required for PCI HSM compliance will be run, and allows this time to be changed.

Authorization: The HSM does not require any authorization to run this command.

Inputs: Time of day.

Outputs: None

Errors: None.

Notes:
- The default time for running the diagnostics is 0900.

Example:

```
Secure> ST <Return>

Self test run time is 09:00.

Change? [Y/N]: Y <Return>

Enter hour [HH] (24 hour format): 13 <Return>
Enter minute [MM]: 55 <Return>

Self test run time changed to 13:55.

Secure>
```

Settings, Storage and Retrieval Commands

Commands are provided to save the payShield 9000 HSM's Alarm, Host and Security settings to a smartcard and to restore the settings to the HSM. Besides the dedicated command to Save HSM Settings to Smartcard, the following individual configuration commands have the option to save settings to smartcard:

- CL (Configure Alarms) to save the Alarm configuration.
- CH (Configure Host) to save the Host port configuration.
- CS (Configure Security) to save the Security configuration.
- AUDITOPTIONS (Audit Options) to save the Audit configuration.

The payShield 9000 HSM provides the following console commands to support storage and retrieval of HSM settings:

Save HSM Settings to a Smartcard

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **admin.console**		

Command: **SS**

Function: To save the Alarm, Host Port, Security, Audit, Command, and PIN Block settings to a smartcard (RACCs are supported).

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **admin.console** must be authorized, using the Authorizing Officer cards of the Management LMK.

Outputs:
- Confirmation messages that Alarm, Host, Security, Audit, Command, and PIN Block settings are saved.

Errors:
- Card not formatted to save/retrieve HSM settings. Attempt with another card? [Y/N] - card is not formatted for storing HSM settings.
- Card not formatted. Attempt with another card? [Y/N] - card is not formatted.
- Command only allowed from Secure-Authorized - the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.

Example:

```
Secure-AUTH> SS <Return>
Insert card and press ENTER: <Return>
ALARM settings saved to the smartcard.
HOST settings saved to the smartcard.
SECURITY settings saved to the smartcard.
AUDIT settings saved to the smartcard.
COMMAND settings saved to the smart card.
PIN BLOCK settings saved to the smart card.
Secure-AUTH>
```

Retrieve HSM Settings from a Smartcard

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Required** Activity: **admin.console**		

Command: **RS**

Function: To read the Alarm, Host Port, Security, Audit, Command, and PIN Block settings from a smartcard. The user is then prompted to use these to overwrite the existing HSM settings. If the settings on the smartcard were saved using a configuration command (CL, CH, CS and AUDITOPTIONS), then only those settings are overwritten.

Authorization: The HSM must be in the secure state to run this command. Additionally, the HSM must be either in the Authorized State, or the activity **admin.console** must be authorized, using the Authorizing Officer cards of the Management LMK.

Inputs:
- Whether to overwrite each of the groups of saved settings.

Outputs:
- The Alarm, Host, Security, Audit, Command, and PIN Block settings stored on the smartcard are listed.

Errors:
- Card not formatted to save/retrieve HSM settings. Attempt with another card? [Y/N] - card is not formatted for storing HSM settings.
- Card not formatted. Attempt with another card? [Y/N] - card is not formatted.
- Command only allowed from Secure-Authorized - the HSM is not in Secure State, or the HSM is not authorized to perform this operation, or both.

Example:

```
Secure-AUTH> RS <Return>
Insert card and press ENTER: <Return>
Temperature Alarm: ON
Motion Alarm: HIGH
Self Test Run Time: 09:00
Overwrite alarm settings with the settings above? [Y/N]: Y <Return>
ALARM settings retrieved from smartcard

Message header length: 4
Protocol: ETHERNET
Character format: ASCII
UDP active: YES
TCP active: YES
TLS active: YES
Number of TCP connections: 1
Well-Known-Port: 1500
Well-Known-TLS-Port: 2500
Number of host interfaces: 1

Overwrite host settings with the settings above? [Y/N]: n <Return>

PIN length: 04
Old encrypted PIN length: 05
Echo: OFF
Atalla ZMK variant support: OFF
Transaction key support: AUSTRALIAN
User storage key length: SINGLE
Select clear PINs: NO
Enable ZMK translate command: NO
Enable X9.17 for import: YES
Enable X9.17 for export: YES
Solicitation batch size: 1024
Single-DES: ENABLED
Prevent single-DES keys from masquerading as double or triple-length
keys: NO
ZMK length: DOUBLE
Decimalization tables: PLAINTEXT
Decimalization table checks enabled: YES
PIN encryption algorithm: A
Authorized state required when importing DES key under RSA key: YES
Minimum HMAC length in bytes: 10
Enable PKCS#11 import and export for HMAC keys: NO
Enable ANSI X9.17 import and export for HMAC keys: NO
Enable ZEK/TEK encryption of ASCII data or Binary data or None: BINARY
Restrict key check values to 6 hex chars : YES
Enable multiple authorized activities: YES
Enable 2DES LMK encryption of 3DES/2048-bit RSA keys: YES
Enable variable length PIN offset: NO
Enable weak PIN checking: NO
Enable PIN block format 34 as output format for PIN translations to
ZPK: NO
Enable PIN block account number translations: NO
Default LMK identifier: 00
Management LMK identifier: 00
Use HSM clock for date/time validation: YES
Additional padding to disguise key length: NO
Key export and import in trusted format only: NO
Protect MULTOS cipher data checksums: YES
Enforce Atalla variant match to Thales key type: NO
Card/password authorization: C
Enable use of Tokens in PIN Translation: NO
Enable use of Tokens in PIN Verification: NO
Restrict PIN block usage for PCI Compliance: NO
Enforce key type separation for PCI Compliance: NO
Enforce Authorization Time Limit: YES
Overwrite security settings with the settings above? [Y/N]: Y <Return>
SECURITY settings retrieved from smartcard.

User Action: ENABLED
```

```
Audit Counter: 00000183
24 Audited Mgmt commands
0 Audited Host commands
Audit Host Errors: DISABLED
0 Audited Console commands
Overwrite auditlog settings with the settings above? [Y/N]: n <Return>

0 Blocked Host commands
0 Blocked Console commands
Overwrite command settings with the settings above? [Y/N]: n <Return>

Pin Block Format 01: ENABLED
Pin Block Format 02: ENABLED
Pin Block Format 03: ENABLED
Pin Block Format 04: ENABLED
Pin Block Format 05: ENABLED
Pin Block Format 34: ENABLED
Pin Block Format 35: ENABLED
Pin Block Format 41: ENABLED
Pin Block Format 42: ENABLED
Pin Block Format 46: ENABLED
Pin Block Format 47: ENABLED
Pin Block Format 48: ENABLED
Overwrite pin block settings with the settings above? [Y/N]: n
<Return>

Secure-AUTH>
```

Key Management Commands

The payShield 9000 HSM provides the following host commands to support generic key management operations:

Generate Key Component

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.{*key*}.console**		

Command: **GC**

Function: To generate a key component and display it in plain and encrypted forms.

	Variant LMK	Key Block LMK
Authorization:	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key type code of the key component being generated.	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key usage code of the key component being generated.
Inputs:	• LMK Identifier: 00-99. • Key Length: 1 (single), 2 (double), 3 (triple). • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme:	• LMK Identifier: 00-99. • Key Algorithm (if AES LMK): 3DES or AES • Key Length: Single/Double/Triple length DES key or (if AES LMK) 128/192/256-bit AES key. • Key Scheme: • Key Usage: See the Key Usage Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Component Number: 1-9. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data.
Outputs:	• Clear text key component. • Key component encrypted under an appropriate variant of the selected LMK. • Component check value.	• Clear text key component. • Key Block containing the component encrypted under the selected LMK. • Component check value.

Notes:

- When generating key components encrypted by a Key Block LMK, the "Component Number" field stored within the component's key block header can be used to help identify individual components. Note, however, that this field is not examined or used by the HSM's FK command when forming a key from these components.

Errors:

- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Invalid key scheme for key length - the Key Scheme is inappropriate for Key length.
- Invalid key scheme - an invalid key scheme is entered.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example generates a double length DES key component in plaintext & encrypted form.

```
Online-AUTH> GC <Return>
Enter LMK id: 00 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key type: 001 <Return>
Enter key scheme: U <Return>

Clear Component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
Encrypted Component: UYYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 2: (3DES Key Block LMK)

This example generates a double length DES key component in plaintext & encrypted form.

```
Online-AUTH> GC <Return>
Enter LMK id: 01 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter component number [1-9]: 2 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: N <Return>

Clear component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
Encrypted component: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 3: (AES Key Block LMK)

This example generates a double length DES key component in plaintext & encrypted form.

```
Online-AUTH> GC <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: D <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Enter key usage: P0 <Return>
```

```
Enter mode of use: N <Return>
Enter component number [1-9]: 2 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: N <Return>

Clear component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
Encrypted component: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 4: (AES Key Block LMK)

This example generates a 128-bit AES key component in plaintext & encrypted form.

```
Online-AUTH> GC <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter key length [128,192,256]: 128 <Return>
Enter key scheme: S <Return>
Enter key usage: K0 <Return>
Enter mode of use: N <Return>
Enter component number [1-9]: 2 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: N <Return>

Clear component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
Encrypted component: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Generate Key and Write Components to Smartcard

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.{*key*}.console**		

Command: **GS**

Function: To generate a key in 2 or 3 components, write the plaintext components to smartcards, and display the key encrypted under the LMK.

	Variant LMK	Key Block LMK
Authorization:	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key type code of the key being generated.	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key usage code of the key being generated.
Inputs:	• LMK Identifier: 00-99. • Key Length: 1 (single), 2 (double), 3 (triple). • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme. • Number of components: 2-3. • Smartcard PINs. PINs must be entered within 60 seconds of being requested.	• LMK Identifier: 00-99. • Key Algorithm (if AES LMK): 3DES or AES • Key Length: Single/Double/Triple length DES key or (if AES LMK) 128/192/256-bit AES key. • Key Scheme. • Number of components: 2-3. • Key Usage: See the Key Usage Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Key Version Number: 00-99. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data. • Smartcard PINs. PINs must be entered within 60 seconds of being requested.
Outputs:	• Key encrypted under an appropriate variant of the selected LMK. • Key check value.	• Key Block containing the key encrypted under the selected LMK. • Key check value.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Invalid PIN; re-enter - a PIN of less than 4 or greater than 8 is entered.

- Smartcard error; command/return: 0003 - invalid PIN is entered.
- Warning - card not blank. Proceed? [Y/N] - the smartcard entered is not blank.
- Overwrite key component? [Y/N] - the smartcard already contains a key component. It can be overwritten if desired.
- Device write failed - the component could not be verified.
- Invalid key scheme for key length - the Key scheme is inappropriate for Key length.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Invalid key scheme - an invalid key scheme is entered.
- Invalid entry - an invalid number of components has been entered.
- Not a LMK card - card is not formatted for LMK or key storage.
- Card not formatted - card is not formatted.
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example writes two double length DES key components to two smartcards, and encrypts the formed key.

```
Online-AUTH> GS <Return>
Enter LMK id: 00 <Return>
Enter key length [1,2,3]: 1 <Return>
Enter key type: 001 <Return>
Enter key scheme: 0 <Return>
Enter number of components [2-3]: 2 <Return>
Insert card 1 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Insert card 2 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Encrypted key: YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 2: (3DES Key Block LMK)

This example generates and writes two double length 3DES key components to two smartcards, and encrypts the formed key.

```
Online-AUTH> GS <Return>
Enter LMK id: 01 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Enter number of components [2-3]: 2 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 00 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 00 <Return>
Enter optional block data: L <Return>
Enter more optional blocks? [Y/N]: N <Return>
Insert card 1 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Insert card 2 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 3: (AES Key Block LMK)

This example generates and writes two double length 3DES key components to two smartcards, and encrypts the formed key.

```
Online-AUTH> GS <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: D <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Enter number of components [2-3]: 2 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 00 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 00 <Return>
Enter optional block data: L <Return>
Enter more optional blocks? [Y/N]: N <Return>
Insert card 1 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Insert card 2 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 4: (AES Key Block LMK)

This example generates and writes two128-bit AES key components to two smartcards, and encrypts the formed key.

```
Online-AUTH> GS <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter key length [128,192,256]: 128 <Return>
Enter key scheme: S <Return>
Enter number of components [2-3]: 2 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 00 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 00 <Return>
Enter optional block data: L <Return>
Enter more optional blocks? [Y/N]: N <Return>
Insert card 1 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
```

```
Insert card 2 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Encrypt Clear Component

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.{*key*}.console**		

Command: **EC**

Function: To encrypt a clear text component and display the result at the console.
If the component does not have odd parity, odd parity will be forced before encryption by the selected LMK.

	Variant LMK	Key Block LMK
Authorization:	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key type code of the component being encrypted.	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key usage code of the component being encrypted.
Inputs:	• LMK Identifier: 00-99. • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme. • Clear Component: 16/32/48 hex digits.	• LMK Identifier: 00-99. • Component Algorithm (if AES LMK): 3DES or AES • Component Length: Single/Double/Triple length DES key or (if AES LMK) 128/192/256-bit AES key. • Key Scheme. • Key Usage: See the Key Usage Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Component Number: 1-9. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data. • Clear Component: 16/32/48 hex digits.
Outputs:	• Component encrypted under an appropriate variant of the selected LMK. • Component check value.	• Key Block containing the component encrypted under the selected LMK. • Component check value.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Data invalid; please re-enter - the input data does not contain 16 or 32 or 48 hexadecimal characters. Re-enter the

correct number of hexadecimal characters.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Invalid key scheme - an invalid key scheme is entered.
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example encrypts a plaintext double length DES key component.

```
Online-AUTH> EC <Return>
Enter LMK id: 00 <Return>
Enter key type: 001 <Return>
Enter key Scheme: U <Return>
Enter component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Encrypted component: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 2: (3DES Key Block LMK)

This example encrypts a plaintext double length DES key component.

```
Online-AUTH> EC <Return>
Enter LMK id: 01 <Return>
Enter component length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter component number [1-9]: 2 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 00 <Return>
Enter optional block data: L <Return>
Enter more optional blocks? [Y/N]: N <Return>
Enter component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Encrypted component: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 3: (AES Key Block LMK)

This example encrypts a plaintext double length DES key component.

```
Online-AUTH> EC <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: D <Return>
Enter component length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Enter key usage: D0 <Return>
Enter mode of use: N <Return>
Enter component number [1-9]: 2 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 00 <Return>
Enter optional block data: L <Return>
Enter more optional blocks? [Y/N]: N <Return>
Enter component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Encrypted component: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
```

```
Online-AUTH>
```

Example 4: (AES Key Block LMK)

This example encrypts a plaintext 128-bit AES key component.

```
Online-AUTH> EC <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter component length [128,192,256]: 128 <Return>
Enter key scheme: S <Return>
Enter key usage: K0 <Return>
Enter mode of use: N <Return>
Enter component number [1-9]: 2 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 00 <Return>
Enter optional block data: L <Return>
Enter more optional blocks? [Y/N]: N <Return>
Enter component: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Encrypted component: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Form Key from Components

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.{*key*}.console**		

Command: **FK**

Function: To build a key from components. If clear components are used, they will not be checked for parity, but odd parity will be forced on the final key before encryption under the selected LMK.

	Variant LMK	Key Block LMK
Authorization:	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key type code of the key being formed.	The HSM must be in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key usage code of the key being formed.
Inputs:	• LMK Identifier: 00-99. • Key Length: 1 (single), 2 (double), 3 (triple). • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme. Must be U, T, or None/Z. • Component Type: X (xor), H (half), E (encrypted), S (smartcard), T (third). • Number of Components: 1-9 if the security setting "Enforce Multiple Key Components" has been set to "NO", otherwise 2-9. • Clear Components: 16/32/48 hex digits.	• LMK Identifier: 00-99. • Key Algorithm (if AES LMK): 3DES or AES • Key Length: Single/Double/Triple length DES key or (if AES LMK) 128/192/256-bit AES key. • Key Scheme. • Component Type (for AES keys): X (xor), E (encrypted), S (smartcard), • Component Type (for DES keys): X (xor), E (encrypted), S (smartcard), H (half), T (third). • Number of Components: 1-9 if the security setting "Enforce Multiple Key Components" has been set to "NO", otherwise 2-9. • Key Usage: See the Key Usage Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Key Version Number: 00-99. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data. • Clear Components: 16/32/48 hex digits.
Outputs:	• Key encrypted under an appropriate variant of the selected LMK. • Key Check Value.	• Key Block containing the component encrypted under the selected LMK. • Key Check Value.

Notes:
- PINs must be entered within 60 seconds of being requested.
- When using key components encrypted by a Key Block LMK, the FK command ignores the "Component Number" field stored within each component key block.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Incompatible header values - the field values are incompatible between components.
- Incompatible key status optional blocks - there is a mismatch between the values contained in one or more key status optional blocks.
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Invalid key scheme - an invalid key scheme is entered.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Key all zero - the key is invalid.
- Invalid entry - an invalid number of components has been entered.
- Data invalid; please re-enter - the amount of input data is incorrect. Re-enter the correct number of hexadecimal characters.
- Invalid PIN; re-enter - a PIN of less than 4 or greater than 8 is entered.
- Smartcard error; command/return: 0003 - invalid PIN is entered.
- No component card - no key component on the provided smartcard.
- Not a LMK card - card is not formatted for LMK or key storage.
- Card not formatted - card is not formatted.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Notes:
- Component type H is not permitted for Triple – DES keys.
- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Example 1: (Variant LMK)

This example forms a key from plaintext component.

```
Online-AUTH> FK <Return>
Enter LMK id: 00 <Return>
Enter key length[1,2,3]: 2 <Return>
Enter key type: 002 <Return>
Enter key scheme: U <Return>
Component type [X,H,E,S,T]: X <Return>
Enter number of components [1-9]: 2 <Return>

Enter component 1: **** **** **** **** **** **** **** **** 
<Return>
```

```
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Enter component 2: **** **** **** **** **** **** **** ****
<Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 2: (Variant LMK)

This example forms a key from components on a smartcard.

```
Online-AUTH> FK <Return>
Enter LMK id: 00 <Return>
Enter key length[1,2,3]: 2 <Return>
Enter key type: 002 <Return>
Enter key scheme: U <Return>
Component type [X,H,E,S,T]: S <Return>
Enter number of components (1-9): 2 <Return>

Insert card 1 and enter PIN: ******** <Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Insert card 2 and enter PIN: ******** <Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 3: (Variant LMK)

This example forms a key from encrypted components.

```
Online-AUTH> FK <Return>
Enter LMK id: 00 <Return>
Enter key length[1,2,3]: 2 <Return>
Enter key type: 002 <Return>
Enter key scheme: U <Return>
Component type [X,H,E,S,T]: E <Return>
Enter number of components (1-9): 2 <Return>

Enter component 1: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Enter component 2: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 4: (Variant LMK)

The security settings require that multiple components are used to form keys, but the user attempts to form a key from one component.

```
Online-AUTH> FK <Return>
Enter LMK id: 00 <Return>
Enter key length[1,2,3]: 2 <Return>
Enter key type: 002 <Return>
Enter key scheme: U <Return>
Component type [X,H,E,S,T]: E <Return>
Enter number of components (2-9): 1 <Return>

Invalid Entry
Enter number of components (2-9): 2 <Return>

Enter component 1: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Enter component 2: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 5: (3DES Key Block LMK)

This example forms a single length DES key from plaintext components.

```
Online-AUTH> FK <Return>
Enter LMK id: 01 <Return>
Enter key length [1,2,3]: 1 <Return>
Enter key scheme: S <Return>
Component type [X,H,E,S,T]: X <Return>
Enter number of components [1-9]: 2 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 99 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: N <Return>

Enter component 1: **** **** **** **** <Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Enter component 2: **** **** **** **** <Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 6: (3DES Key Block LMK)

This example forms a double length 3DES key from components on a smartcard.

```
Online-AUTH> FK <Return>
Enter LMK id: 01 <Return>
Enter Key Length[1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Component type [X,H,E,S,T]: S <Return>
Enter number of components (1-9): 2 <Return>

Insert card 1 and enter PIN: ******** <Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Insert card 2 and enter PIN: ******** <Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 7: (AES Key Block LMK)

This example forms a double length 3DES key from plaintext components.

```
Online-AUTH> FK <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: D <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme: S <Return>
Component type [X,H,E,S,T]: X <Return>
Enter number of components [1-9]: 2 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 99 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: N <Return>

Enter component 1: **** **** **** **** **** **** **** ****
<Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>
```

```
Enter component 2: **** **** **** **** **** **** **** ****
<Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 8: (AES Key Block LMK)

This example forms a 128-bit AES key from components on a smartcard.

```
Online-AUTH> FK <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter key length [128,192,256]: 128 <Return>
Enter key scheme: S <Return>
Component type [X,E,S]: S <Return>
Enter number of components [1-9]: 2 <Return>
Enter key version number: 00 <Return>
Enter optional blocks? [Y/N]: N <Return>

Insert card 1 and enter PIN: ******** <Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Insert card 2 and enter PIN: ******** <Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 8: (AES Key Block LMK)

This example forms a 128-bit AES key from encrypted components.

```
Online-AUTH> FK <Return>
Enter LMK id: 02 <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter key length [128,192,256]: 128 <Return>
Enter key scheme: S <Return>
Component type [X,E,S]: E <Return>
Enter number of components [1-9]: 3 <Return>
Enter key version number: 00 <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 03 <Return>
Enter optional block data: 2005:12:21:00 <Return>
Enter more optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 04 <Return>
Enter optional block data: 2007:12:21:00 <Return>
Enter more optional blocks? [Y/N]: N <Return>

Enter component 1: S XXXXXXXX……XXXXXX <Return>
Component 1 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Enter component 2: S XXXXXXXX……XXXXXX <Return>
Component 2 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Enter component 3: S XXXXXXXX……XXXXXX <Return>
Component 3 check value: XXXXXX
Continue? [Y/N]: Y <Return>

Encrypted key: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Generate Key

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Variant LMK	Authorization: **Determined by KTT(G&E)** Activity: **generate.{*key*}.console** and **export.{*key*}.console**	
Key Block LMK	Authorization: **If export to non-KB.** Activity: **export.{key}.console**	

Command: **KG**

Function: To generate a random key and return it encrypted under the LMK and optionally under a ZMK (for transmission to another party).

	Variant LMK	Key Block LMK
Authorization:	This command examines the 'Generate' flag of the given key type within the Key Type Table to determine the authorization requirement. If the flag is 'A', the HSM must either be in the Authorized State, or the activity **generate.{*key*}.console** must be authorized, where 'key' is the key type code of the key being generated. If the generated key is required to be exported under the ZMK, this command also examines the 'Export' flag of the given key type within the Key Type Table. If the flag is 'A', the HSM must either be in the Authorized State, or the activity **export.{*key*}.console** must be authorized, where 'key' is the key type code of the key being exported.	The authorization requirement for this command depends solely on the type of export being requested: Exported key scheme / Authorization: No export: None 'S' (*Thales Key Block*): None 'R' (*TR-31 Key Block*): None 'U', 'T' (*Variant*): Required 'Z', 'X', 'Y' (*X9.17*): Required If authorization is required, the HSM must either be in the Authorized State, or the activity **export.{*key*}.console** must be authorized, where 'key' is the key usage code of the key being exported.
Inputs:	• LMK Identifier: 00-99. • Key Length: 1 (single), 2 (double), 3 (triple). • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme (LMK). • Key Scheme (ZMK) (if exporting). • ZMK (if exporting). • Key Block values if exporting to TR-31 format	• LMK Identifier: 00-99. • Key Algorithm (if AES LMK): 3DES or AES • Key Length: Single/Double/Triple length DES key or (if AES LMK) 128/192/256-bit AES key. • Key Scheme (LMK). • Key Scheme (ZMK) (if exporting). • ZMK (if exporting). • Key Usage: See the Key Usage Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*.

		• Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Key Version Number: 00-99. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data. • Exportability of exported key (if exporting).
Outputs:	• Key encrypted under an appropriate variant of the selected LMK. • Key/Key Block encrypted under the ZMK (if exporting). • Key Check Value.	• Key Block containing the key encrypted under the selected LMK. • Key/Key Block encrypted under the ZMK (if exporting). • Key Check Value.

Notes: For legacy reasons, the export of a ZMK, ZEK or DEK from encryption under a key block LMK to encryption under a ZMK (in variant/X9.17 format) will not be permitted. Specifically, such export of keys with key usage = "K0", "52", "D0", "21" or "22" will be prohibited.

Errors:

- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Must be in Authorized State or Activity not authorized - the key type provided requires the HSM to be in Authorized State.
- Data invalid; please re-enter - the encrypted ZMK does not contain the correct characters, or the key check value does not contain 6 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Key parity error; please re-enter - the ZMK does not have odd parity on each byte. Re-enter the encrypted ZMK and check for typographic errors.
- Invalid key scheme for key length - the Key scheme is inappropriate for Key length.
- Invalid key scheme - the key scheme is invalid.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example generates a new double length DES key.

```
Online> KG <Return>
Enter LMK id [0-4]: 0 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key type: 002 <Return>
Enter key scheme (LMK): U <Return>
Enter key scheme (ZMK): <Return>
Enter ZMK: <Return>
Key under LMK: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key Check value: ZZZZZZ
Online>
```

Example 2: (Variant LMK)

This example generates a new double length DES key, and exports it to X9.17 format.

```
Online-AUTH> KG <Return>
Enter LMK id [0-4]: 00 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key type: 002 <Return>
Enter key scheme (LMK): U <Return>
Enter key scheme (ZMK): X <Return>
Enter ZMK: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter XMK variant: 0 <Return>
Key under LMK: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key under ZMK: X YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 3: (Variant LMK)

This example generates a new double length DES key, and exports it to TR-31 format.

```
Online-AUTH> KG <Return>
Enter LMK id [0-4]: 00 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key type: 001 <Return>
Enter key scheme (LMK): U <Return>
Enter key scheme (ZMK): R <Return>
Enter ZMK: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter XMK variant: 0 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 44 <Return>
Enter exportability: N <Return>
Enter optional blocks? [Y/N]: N <Return>
Key under LMK: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key under ZMK: R YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 4: (3DES Key Block LMK)

This example generates a new double length DES key, and exports it to X9.17 format.

```
Online-AUTH> KG <Return>
Enter LMK id [0-4]: 01 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme (LMK): S <Return>
Enter key scheme (ZMK): X <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter XMK variant: 0 <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 22 <Return>
Enter exportability: N <Return>
Enter optional blocks? [Y/N]: N <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key under ZMK: X YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
```

```
                    Online-AUTH>
```

Example 5: (3DES Key Block LMK)

This example generates a new double length DES key, and exports it to TR-31 format.

```
Online> KG <Return>
Enter LMK id [0-4]: 01 <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme (LMK): S <Return>
Enter key scheme (ZMK): R <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter XMK variant: 0 <Return>
Enter key usage: 72 <Return>
Enter mode of use: N <Return>
Enter key version number: 33 <Return>
Enter exportability: E <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 03 <Return>
Enter optional block data: 2005:12:21:00 <Return>
Enter more optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 04 <Return>
Enter optional block data: 2007:12:21:00 <Return>
Enter more optional blocks? [Y/N]: N <Return>
Enter exportability field for exported key block: <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key under ZMK: R YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online>
```

Example 6: (AES Key Block LMK)

This example generates a new double length DES key.

```
Online-AUTH> KG <Return>
Enter LMK id [0-4]: 02 <Return>
Enter algorithm type [D=DES, A=AES]: D <Return>
Enter key length [1,2,3]: 2 <Return>
Enter key scheme (LMK): S <Return>
Enter key scheme (ZMK): <Return>
Enter ZMK: <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 00 <Return>
Enter exportability: N <Return>
Enter optional blocks? [Y/N]: N <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 7: (AES Key Block LMK)

This example generates a new 128-bit AES key.

```
Online-AUTH> KG <Return>
Enter LMK id [0-4]: 02 <Return>
Enter algorithm type [D=DES, A=AES]: A <Return>
Enter key length [128,192,256]: 128 <Return>
Enter key scheme (LMK): S <Return>
Enter key scheme (ZMK): <Return>
Enter ZMK: <Return>
Enter key usage: K0 <Return>
Enter mode of use: N <Return>
Enter key version number: 00 <Return>
Enter exportability: N <Return>
Enter optional blocks? [Y/N]: N <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Import Key

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **command.ik.console**		

Command: **IK**

Function: To import a key from encryption under a ZMK to encryption under an LMK. If the key imported does not have odd parity a warning will be issued and odd parity will be forced on the key before encryption under the specified LMK.

Authorization: The HSM must either be in the Authorized State, or the activity **command.ik.console** must be authorized.
For AES LMKs, keys can only be exported in Thales Key Block format.

	Variant LMK	Key Block LMK
Inputs:	• LMK Identifier: 00-99. • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme (LMK). • ZMK to be used to decrypt the key. • Key/Key Block to be imported.	• LMK Identifier: 00-99. • Key Scheme (LMK). • ZMK to be used to decrypt the key. • Key/Key Block to be imported. For import from Variant/X9.17: • Key Usage: See the Key Usage Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Key Version Number: 00-99. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data. For import from a key block format: • Modified Key Usage • Optional Block data.
Outputs:	• Key encrypted under an appropriate variant of the selected LMK. • Key Check Value.	• Key Block containing the key encrypted under the selected LMK. • Key Check Value.

Notes:
- For legacy reasons, the import of a ZMK or DEK from encryption under a ZMK (in variant/X9.17 format) to encryption under a key block LMK will not be permitted. Specifically, such import of keys with key usage = "K0", "52", "D0", "21" or "22" will be prohibited.

- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.
- If the option "Enforce Atalla variant match to Thales key type" is set to YES in the CS console command, the following matchings between Atalla variant and Thales variant key types will be enforced:

Key Type	Atalla Variant	Thales Variant (*)	Thales Variant (Ø)
TPK ZPK	1 or 01	002 LMK 14-15 001 LMK 06-07	70D LMK 36-37/7 001 LMK 06-07
ZEK	2 or 02	00B LMK 32-33 00A LMK 30-31	00B LMK 32-33 00A LMK 30-31
TAK ZAK CVK	3 or 03	003 LMK 16-17 008 LMK 26-27 402 LMK 14-15/4	003 LMK 16-17 008 LMK 26-27 402 LMK 14-15/4
TMK TPK PVK	4 or 04	002 LMK 14-15 002 LMK 14-15 002 LMK 14-15	80D LMK 36-37/8 70D LMK 36-37/7 002 LMK 14-15
TMK	5 or 05	002 LMK 14-15	80D LMK 36-37/8
BDK type-1	8 or 08	009 LMK 28-29	009 LMK 28-29
MK-AC	9 or 09	109 LMK 28-29/1	109 LMK 28-29/1
MK-SMI	9 or 09	209 LMK 28-29/2	209 LMK 28-29/2
MK-SMC	9 or 09	309 LMK 28-29/3	309 LMK 28-29/3
TEK	26	30B LMK 32-33/3	30B LMK 32-33/3
BDK type-2	30	609 LMK 28-29/6	609 LMK 28-29/6
BDK type-3	8 or 08	809 LMK 28-29/8	809 LMK 28-29/8

* Applies if the security setting "Enforce key type 002 separation for PCI HSM compliance" has the value "N"

Ø Applies if the security setting "Enforce key type 002 separation for PCI HSM compliance" has the value "Y"

Errors:

- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Must be in Authorized State or Activity not authorized - the key type provided requires the HSM to be in Authorized State.
- Data invalid; please re-enter - the encrypted ZMK does not contain the correct characters, or the key check value does not contain 6 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Key parity error; re-enter key - the parity of the ZMK is not odd.
- Warning: key parity corrected - the parity of the key encrypted under the ZMK is not odd.
- Invalid key scheme - the key scheme is invalid.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example imports a key from X9.17 format.

```
Online> IK <Return>
Enter LMK id: 00 <Return>
Enter Key type: 002 <Return>
Enter Key Scheme: U <Return>
Enter ZMK: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter key: X XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Encrypted key: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
```

Example 2: (Variant LMK)

This example imports a key from TR-31 format.

```
Online> IK <Return>
Enter LMK id: 00 <Return>
Enter key type: 009 <Return>
Enter key scheme (LMK): U <Return>
Enter ZMK: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter key: R XXXXXXXX……XXXXXX <Return>
Key under LMK: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online>
```

Example 3: (3DES Key Block LMK)

This example imports a key from X9.17 format.

```
Online-AUTH> IK <Return>
Enter LMK id: 01 <Return>
Enter key scheme (LMK): S <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter key: X XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 27 <Return>
Enter exportability: N <Return>
Enter optional blocks? [Y/N]: N <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 4: (3DES Key Block LMK)

This example imports a key from TR-31 format. Note that a new (more restrictive) value for the imported key block's Key Usage field is entered during the import process.

```
Online> IK <Return>
Enter LMK id: 01 <Return>
Enter key scheme (LMK): S <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter key: R XXXXXXXX……XXXXXX <Return>
Enter modified key usage: 72 <Return>
Enter optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 03 <Return>
Enter optional block data: 2005:12:21:00 <Return>
Enter more optional blocks? [Y/N]: Y <Return>
Enter optional block identifier: 04 <Return>
Enter optional block data: 2007:12:21:00 <Return>
Enter more optional blocks? [Y/N]: N <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online>
```

Example 5: (3DES or AES Key Block LMK)

This example imports a key from Thales Key Block format.

```
Online> IK <Return>
Enter LMK id: 01 <Return>
Enter key scheme (LMK): S <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter key: S XXXXXXXX……XXXXXX <Return>
Key under LMK: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online>
```

Export Key

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Variant LMK	Authorization: **Determined by KTT(E)** Activity: **export.{*key*}.console**	
Key Block LMK	Authorization: **If export to non-KB.** Activity: **export.{key}.console**	

Command: **KE**

Function: To translate a key from encryption under the specified LMK to encryption under a ZMK.

	Variant LMK	Key Block LMK
Authorization:	This command examines the 'Export' flag of the given key type within the **Key Type Table** to determine whether authorization is required. If required, the HSM must either be in the Authorized State, or the activity **export.{*key*}.console** must be authorized, where 'key' is the key type code of the key being exported.	The authorization requirement for this command depends on the type of export being requested: Exported key scheme / Authorization: 'S' (*Thales Key Block*) – None 'R' (*TR-31 Key Block*) – None 'U', 'T' (*Variant*) – Required 'Z', 'X', 'Y' (*X9.17*) – Required If authorization is required, the HSM must either be in the Authorized State, or the activity **export.{*key*}.console** must be authorized, where 'key' is the key usage code of the key being exported. For AES LMKs, keys can only be exported in Thales Key Block format.
Inputs:	• LMK Identifier: 00-99. • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Scheme (ZMK). • ZMK to be used to encrypt the key. • Key to be exported. For export to Thales Key Block & TR-31: • Key Usage: See the Key Usage Table in Chapter 8 of	• LMK Identifier: 00-99. • Key Scheme (ZMK). • ZMK to be used to encrypt the key. • Key to be exported. For export to key block format: • Exportability of exported key.

	the *payShield 9000 Host Programmer's Manual*. • Mode of Use: See the Mode of Use Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Key Version Number: 00-99. • Exportability: See the Exportability Table in Chapter 8 of the *payShield 9000 Host Programmer's Manual*. • Optional Block data. *Note export from a Variant LMK to Thales Key Block is not permitted.*	
Outputs:	• Key/Key Block encrypted under the ZMK. • Key Check Value.	• Key/Key Block encrypted under the ZMK. • Key Check Value.

Notes: For legacy reasons, the export of a ZMK, ZEK or DEK from encryption under a key block LMK to encryption under a ZMK (in variant/X9.17 format) will not be permitted. Specifically, such export of keys with key usage = "K0", "52", "D0", "21" or "22" will be prohibited.

Errors:

- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Must be in Authorized State or Activity not authorized - the key type provided requires the HSM to be in Authorized State.
- Data invalid; please re-enter - the encrypted ZMK or key does not contain 16 or 32 hex or 1 alpha + 32 hex or 1 alpha + 48 hex. Re-enter the correct number of hexadecimal characters.
- Key parity error; re-enter key - the ZMK or key does not have odd parity on each byte. Re-enter the key and check for typographic errors.
- Invalid key scheme - the key scheme is invalid.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example exports a key to X9.17 format.

```
Online-AUTH> KE <Return>
Enter Key type: 002 <Return>
Enter Key Scheme: X <Return>
Enter ZMK: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter key: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Key under ZMK: X YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 2: (Variant LMK)

This example exports a key to TR-31 format.

```
Online-AUTH> KE <Return>
Enter LMK id: 00 <Return>
Enter key type: 001 <Return>
Enter key scheme (ZMK): R <Return>
Enter ZMK: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter key: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter key usage: P0 <Return>
Enter mode of use: N <Return>
Enter key version number: 44 <Return>
Enter exportability: N <Return>
Enter optional blocks? [Y/N]: N <Return>
Key under ZMK: R YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 3: (3DES Key Block LMK)

This example exports a key to X9.17 format.

```
Online-AUTH> KE <Return>
Enter LMK id: 01 <Return>
Enter key scheme (ZMK): X <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter key: S XXXXXXXX……XXXXXX <Return>
Key under ZMK: X YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online-AUTH>
```

Example 4: (3DES Key Block LMK)

This example exports a key to TR-31 format.

```
Online> KE <Return>
Enter LMK id: 01 <Return>
Enter key scheme (ZMK): R <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter key: S XXXXXXXX……XXXXXX <Return>
Enter exportability field for exported key block: <Return>
Key under ZMK: R YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online>
```

Example 5: (3DES or AES Key Block LMK)

This example exports a key to Thales Key Block format.

```
Online> KE <Return>
Enter LMK id: 01 <Return>
Enter key scheme (ZMK): S <Return>
Enter ZMK: S XXXXXXXX……XXXXXX <Return>
Enter key: S XXXXXXXX……XXXXXX <Return>
Enter exportability field for exported key block: <Return>
Key under ZMK: S YYYYYYYY……YYYYYY
Key check value: ZZZZZZ
Online>
```

Generate a Check Value

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Variant LMK	Authorization: **Required if ≠ 6 digits** Activity: **generate.{*key*}.console**	
Key Block LMK	Authorization: **Not required.**	

Command: **CK**

Function: To generate a key check value (KCV) for a key encrypted under a specified LMK.

	Variant LMK	Key Block LMK
Authorization:	This command only requires authorization when calculating either 8 or 16 digit Key Check Values. If required, the HSM must either be in the Authorized State, or the activity **generate.{*key*}.console** must be authorized, where 'key' is the key type of the key being used. Regardless of the authorization requirement, this command examines the 'Generate' flag of the given key type within the **Key Type Table** to determine whether the check value can be calculated.	The HSM does not require any authorization to run this command. Note: Key Check Values of key blocks are always 6-digits in length.
Inputs:	• LMK Identifier: 00-99. • Key Type: See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*. • Key Length: 1 (single), 2 (double), 3 (triple). • Key.	• LMK Identifier: 00-99. • Key.
Outputs:	• Key Check Value.	• Key Check Value.

Errors:

- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Incompatible LMK schemes - the LMK schemes are different.
- Data invalid; please re-enter - incorrect number of characters.
- Key parity error; re-enter key - the entered key does not have odd parity on each byte. Re-enter the complete line (key and Key-Type code) and check for typographic errors.
- Invalid key type; re-enter - the key type is invalid. See the Key Type Table in Chapter 7 of the *payShield 9000 Host Programmer's Manual*.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Example 1: (Variant LMK)

This example generates a check value of a key.

```
Online-AUTH> CK <Return>
Enter LMK id: 00 <Return>
Enter key type code: 001 <Return>
Enter key length flag [S/D/T]: D <Return>
Enter encrypted key: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
<Return>
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Example 2: (Key Block LMK)

This example generates a check value of a key.

```
Online> CK <Return>
Enter LMK id: 01 <Return>
Enter key block: S XXXXXXXXXXXXXXX……XXXXXXXXX <Return>
Key check value: ZZZZZZ
Online>
```

Payment System Commands

The payShield 9000 HSM provides the following console commands to support some of the card payment systems host commands.

Generate a Card Verification Value

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **misc.console**		

Command: **CV**

Function: To generate a VISA CVV or MasterCard CVC.

Authorization: The HSM must be either in the Authorized State, or the activity **misc.console** must be authorized, using the Authorizing Officer cards of the relevant LMK.

Inputs:
- LMK identifier: indicates the LMK to use when decrypting the supplied CVK(s).
- Encrypted CVK
- Primary account number (PAN) for the card: up to 19 decimal digits.
- Card Expiry date: 4 decimal digits.
- Service code: 3 decimal digits.

Outputs:
- Card Verification Value: 3 decimal digits.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Data invalid; please re-enter - possibly incorrect key length. Could also be incorrect PAN, card expiry date, or service code length or non-decimal PAN, card expiry date or service code.
- Key parity error; please re-enter - the parity of the key entered is not odd.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Notes: Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Example 1: (Variant LMK)

This example generates a CVV using a CVK pair encrypted in variant format.

```
Online-AUTH> CV <Return>
Enter LMK id: 00 <Return>
Enter key A: XXXX XXXX XXXX XXXX <Return>
Enter key B: XXXX XXXX XXXX XXXX <Return>
Enter PAN: 1234567812345678 <Return>
Enter expiry date: 0694 <Return>
Enter service code: 123 <Return>
CVV: 321
Online-AUTH>
```

Example 2: (Variant LMK)

This example generates a CVV using a double length CVK in variant format.

```
Online-AUTH> CV <Return>
Enter LMK id: 00 <Return>
Enter key A: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter PAN: 1234567812345678 <Return>
Enter expiry date: 0694 <Return>
Enter service code: 123 <Return>
CVV: 321
Online-AUTH>
```

Example 3: (Key Block LMK)

This example generates a CVV using a CVK in key block format.

```
Online-AUTH> CV <Return>
Enter LMK id: 01 <Return>
Enter key block: S XXXXXXXX……XXXXXX <Return>
Enter PAN: 1234567812345678 <Return>
Enter expiry date: 0694 <Return>
Enter service code: 123 <Return>
CVV: 321
Online-AUTH>
```

Generate a VISA PIN Verification Value

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **misc.console**		

Command: **PV**

Function: To generate a VISA PIN Verification Value (PVV).

Authorization: The HSM must be either in the Authorized State, or the activity **misc.console** must be authorized, using the Authorizing Officer cards of the relevant LMK.

Inputs:
- LMK identifier: indicates the LMK to use when decrypting the supplied PVK(s).
- Encrypted PVK.
- The PVV data block comprising:
  - The 11 right-most digits of the account number (excluding check digit): 11 decimal digits.
  - The PIN verification key indicator (PVKI): 1 decimal digit.
  - The 4 left-most digits of the clear PIN: 4 decimal digits.

Outputs:
- The PIN Verification Value (PVV): 4 decimal digits.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Data invalid; please re-enter - the PVK A, PVK B or the PVV data block field is not 16 characters long. Re-enter the correct number of characters.
- Key parity error; please re-enter - the PVK A or PVK B does not have odd parity on each byte. Re-enter the encrypted PVK A or PVK B and check for typographic errors.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.
- Various key block field errors – the value entered is invalid, or incompatible with previously entered values.

Notes:
- The completion of this activity will always be entered in the audit log irrespective of the AUDITOPTIONS settings,

Example 1: (Variant LMK)

This example generates a PVV using a PVK pair in variant format.

```
Online-AUTH> PV <Return>
Enter LMK id: 00 <Return>
Enter key A: XXXX XXXX XXXX XXXX <Return>
Enter key B: XXXX XXXX XXXX XXXX <Return>
Enter PVV data block: XXXXXXXXXXX N NNNN <Return>
PVV: NNNN
Online-AUTH>
```

Example 2: (Variant LMK)

This example generates a PVV using a double length PVK in variant format.

```
Online-AUTH> PV <Return>
Enter LMK id: 00 <Return>
Enter key A: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter PVV data block: XXXXXXXXXXX N NNNN <Return>
PVV: NNNN
Online-AUTH>
```

Example 3: (Key Block LMK)

This example generates a PVV using a PVK in key block format.

```
Online-AUTH> PV <Return>
Enter LMK id: 01 <Return>
Enter key block: S XXXXXXXX……XXXXXX <Return>
Enter PVV data block: XXXXXXXXXXX N NNNN <Return>
PVV: NNNN
Online-AUTH>
```

Load the Diebold Table

Variant ☑	Key Block ☑	
Online ☑	Offline ☒	Secure ☒
Authorization: **Required** Activity: **misc.console**		

Command: **R**

Function: To load the Diebold table into user storage in the HSM.

Authorization: The HSM must be online and must be either in the Authorized State, or the activity **misc.console** must be authorized, using the Authorizing Officer cards of the relevant LMK.

Inputs:
- LMK identifier: indicates the LMK to use when encrypting the supplied values.
- Location in user storage at which to store the Diebold table. See notes below.

Outputs:
- The 512-character encrypted table: 16 lines of 32 hexadecimal characters each.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Command only allowed from Online-Authorized - the HSM is not online, or the HSM is not authorized to perform this operation, or both.
- Invalid index - the specified location in user storage is out of range. Enter a valid value.
- Data invalid; please re-enter - the entered index is not 3 hexadecimal characters long, or a table entry is not 16 hexadecimal characters long. Re-enter the correct number of hexadecimal characters.
- Invalid table: duplicate or missing values - some of the data entered is not a valid entry for a Diebold table. Check the table and re-enter the data, checking for typographic errors.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes:
- Encryption of the Diebold Table:
  - If the security setting "Enforce key type 002 separation for PCI HSM compliance" has the value "N", the Diebold table is encrypted using LMK pair 14-15 variant 0.
  - If the security setting "Enforce key type 002 separation for PCI HSM compliance" has the value "Y", the Diebold table is encrypted using LMK pair 36-37 variant 6.
- User Storage is structured in different ways depending on whether the security setting "User storage key length" has a fixed length value (setting = S(ingle), D(ouble), T(riple)) or is variable (setting = V(ariable)).
  - If the length is fixed, the Diebold table is stored as 32 contiguous blocks of 16 characters. The index for the first block must be in the range 000-FE0.

- If the length is variable, the Diebold table is stored as a single block of 512 characters. Because this needs to use one of the larger slots capable of handling blocks larger than 100 bytes, the index must be in the range 000-07F.

See Chapter 15 of the *payShield 9000 Host Programmer's Manual* for further information.

- If the security setting "Enforce key type 002 separation for PCI HSM compliance" is changed, the Diebold Table must be re-entered by using this command. Therefore it is important that the cleartext version of the table is retained.

Example: *The security setting "User storage key length" has a fixed length value.*

```
Online-AUTH> R <Return>
Enter LMK id: 00 <Return>
Enter index (000 - FE0): XXX <Return>
Now enter table, 16 hex digits/line
Line 01: XXXX XXXX XXXX XXXX <Return>
XXXX XXXX XXXX XXXX OK? [Y/N] Y <Return>
Line 02:
…
…
Line 32: XXXX XXXX XXXX XXXX <Return>
XXXX XXXX XXXX XXXX OK? [Y/N] Y <Return>

XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
…
…
XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX (16 lines of encrypted
table are displayed)
Online-AUTH>
```

Note: *The result of the "R" command gives no indication as to the LMK scheme or LMK identifier used in the command. When this value is used with other (host) commands, the user must ensure that the correct LMK is specified in the command.*

Encrypt Decimalization Table

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **misc.console**		

Command: **ED**

Function: To encrypt a 16 digit decimalization table for use with host commands using IBM 3624 PIN Generation & Verification.

Authorization: The HSM must be either in the Authorized State, or the activity **misc.console** must be authorized, using the Authorizing Officer cards of the relevant LMK.

Inputs:
- LMK identifier: indicates the LMK to use when encrypting the decimalization table.
- Decimalization table. 16 decimal digits that specify the mapping between hexadecimal & decimal numbers.
- The HSM by default checks that the decimalization table contains at least 8 different digits, with no digit repeated more than 4 times. This feature may be disabled using the Configure Security parameter "Enable decimalization table check". Disabling of this feature is not recommended.

Outputs:
- Encrypted decimalization table:
  - 16 Hex characters when using a Variant LMK or a 3DES Key Block LMK.
  - 32 Hex characters when using an AES LMK.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Not Authorized - the HSM is not authorized to perform this operation.
- Decimalization table invalid - the decimalization table is not all decimal or does not contain at least 8 different digits with no digit repeated more than 4 times.
- Master Key Parity Error - the contents of the HSM storage have been corrupted or erased. Do not continue. Inform the security department.

Example: (Variant or 3DES Key Block LMK)

This example encrypts a decimalization table using a Variant LMK (same applies with 3DES Key Block LMK).

```
Online-AUTH> ED <Return>
Enter LMK id: 00 <Return>
Enter decimalization table: 0123456789012345 <Return>
Encrypted decimalization table: XXXX XXXX XXXX XXXX
Online-AUTH>
```

Example: (AES Key Block LMK)

This example encrypts a decimalization table using an AES LMK.

```
Online-AUTH> ED <Return>
Enter LMK id: 00 <Return>
Enter decimalization table: 0123456789012345 <Return>
Encrypted decimalization table: XXXX XXXX XXXX XXXX XXXX XXXX
XXXX XXXX
Online-AUTH>
```

Note:

- The result of the "ED" command gives no indication as to the LMK scheme or LMK identifier used in the command. When this value is used with other (host) commands, the user must ensure that the correct LMK is specified in the command.

Translate Decimalization Table

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **misc.console**		

Command: **TD**

Function: To translate an encrypted decimalization table from Encryption under an old LMK to encryption under the corresponding new LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **misc.console** must be authorized, using the Authorizing Officer cards of the relevant LMK.

Inputs:
- LMK identifier: indicates the LMK to use when translating the decimalization table.
- Encrypted Decimalization table. This is the result of encrypting a decimalization table using the ED command. The size of the encrypted decimalization table depends on the LMK used to encrypt it: for DES-based Variant and 3DES Key Block LMKs, the size is 16 hex digits. For AES Key Block LMKs, the size is 32 hex digits.
- The HSM by default checks that the decimalization table contains at least 8 different digits, with no digit repeated more than 4 times. This feature may be disabled using the Configure Security parameter "Enable decimalization table check". Disabling of this feature is not recommended.

Outputs:
- Encrypted decimalization table:
  - 16 Hex characters when using a Variant LMK or a 3DES Key Block LMK.
  - 32 Hex characters when using an AES LMK.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Not Authorized - the HSM is not authorized to perform this operation.
- Decimalization Table Invalid - decimalization table not all decimal or does not contain at least 8 different digits with no digit repeated more than 4 times.
- Master Key Parity Error - the contents of the HSM storage have been corrupted or erased. Do not continue. Inform the security department.
- No LMK in Key Change Storage - Key Change storage is empty.

Example: (Variant or 3DES Key Block LMK)

```
Online-AUTH> TD <Return>
Enter LMK id: 00 <Return>
Enter decimalization table encrypted under old LMK : xxxxxxxxxxxxxxxx
<Return>
Decimalization table encrypted under new LMK       : YYYYYYYYYYYYYYYY
Online-AUTH>
```

Example: (AES Key Block LMK)

```
Online-AUTH> TD <Return>
Enter LMK id: 00 <Return>
Enter decimalization table encrypted under old LMK : xxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxx <Return>
Decimalization table encrypted under new LMK       : YYYYYYYYYYYYYYYY
YYYYYYYYYYYYYYYY
Online-AUTH>
```

Note:

- The result of the "TD" command gives no indication as to the LMK scheme or LMK identifier used in the command. When this value is used with other (host) commands, the user must ensure that the correct LMK is specified in the command.

Generate a MAC on an IPB

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **misc.console**		

Command: **MI**

Function: To generate a MAC on the Cryptogram component of a CAP IPB.

Authorization: The HSM must be either in the Authorized State, or the activity **misc.console** must be authorized, using the Authorizing Officer cards of the relevant LMK.

Inputs:
- LMK identifier: indicates the LMK to use when generating the MAC.
- 8 byte IPB represented as 16 hex ASCII characters.

Outputs:
- 4 byte MAC over the plaintext IPB input data.

Errors:
- Invalid LMK identifier - no LMK loaded or entered identifier out of range.
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- IPB is not 8 bytes. Please re-enter - the validation of the IPB failed.
- Warning: Less than 16 '1'bits in IPB - the IPB contains less than 16 '1' bits.

Example:

```
Online-AUTH> MI <Return>
Enter LMK id: 00 <Return>
Enter IPB: FFFFFFFF00000000 <Return>
      MAC: FB1A 3C1A
Online-AUTH>
```

Note:

- The result of the "MI" command gives no indication as to the LMK scheme or LMK identifier used in the command. When this value is used with other (host) commands, the user must ensure that the correct LMK is specified in the command.

Smartcard Commands

The payShield 9000 HSM provides the following console commands to support HSM smartcards. Please note that some of these commands are designed to operate only with the legacy HSM smartcards while other may support both the legacy and new smartcards used in the payShield Manager.

NOTE: DO NOT REPEATEDLY ENTER INVALID PINS. A LEGACY SMARTCARD "LOCKS" AFTER EIGHT SUCCESSIVE INVALID PINS HAVE BEEN ENTERED. LEGACY SMARTCARDS CAN BE "UNLOCKED" BY REFORMATTING, WHICH DELETES THE ENTIRE CONTENTS OF THE CARD. NEW SMARTCARDS USED BY THE PAYSHIELD MANAGER LOCK AFTER FIVE SUCCESSIVE INVALID PINS HAVE BEEN ENTERED. THEY MAY BE UNLOCKED BY RECOMMISSIOING THEM.

Format an HSM Smartcard

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **FC**

Function: To format an HSM smartcard for use by the HSM. Different formats are used for LMK storage and saving HSM settings. payShield Manager cards do not need to be formatted.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- (LMK cards): Smartcard PIN: 5 to 8 alphanumeric characters.
- Date: 6 numeric character format DDMMYY.
- Time: 6 numeric characters; format hhmmss.
- Issuer ID: maximum 35 alphanumeric characters.
- User ID: maximum 35 alphanumeric characters.

Outputs:
- Text messages:
  - Insert card and press ENTER.
  - Format card for HSM settings/LMKs? [H/L]
  - Enter new PIN for smartcard.
  - Re-enter new PIN.
  - Enter format code.
  - Enter date.
  - Enter time.
  - Enter Issuer ID.
  - Enter User ID.
  - Format complete.
  - Card already formatted, continue? [Y/N].

Note:
- This command only operates with legacy HSM smartcards.

Errors:
- Invalid PIN; re-enter - the PIN entered is fewer than 5 or greater than 8 digits.
- PINs did not agree - the new PINs entered for the card did not match each other.
- Invalid input. Entry must be in numeric format - non numeric value is entered for time or date.

Example 1:

```
Online> FC <Return>
Insert card and press ENTER: <Return>
Card already formatted, continue? [Y/N]: Y <Return>
Format card for HSM settings/LMKs? [H/L]: L <Return>
Erasing card
Formatting card . . .
Enter new PIN for Smartcard: ******* <Return>
Re-enter new PIN: ******* <Return>
Enter time [hhmmss]: 153540 <Return>
Enter date [ddmmyy]: 261093 <Return>
Enter User ID: Joe Small <Return>
Enter Issuer ID: Big Bank plc <Return>
Format complete
Online>
```

Example 2:

```
Online> FC <Return>
Insert card and press ENTER: <Return>
Card already formatted, continue? [Y/N]: Y <Return>
Format card for HSM settings/LMKs? [H/L]: H <Return>
      Erasing card
      Formatting card . . .
Format complete
Online>
```

Create an Authorizing Officer Smartcard

Variant ☑		Key Block ☑
Online ☒	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **CO**

Function: To copy the Password for an Authorizing Officer to another smartcard (RLMKs are supported) so that it can be used to set the HSM into the Authorized State. Note that only LMK component cards 1 and 2 contain the Password.

Authorization: The HSM must be in the offline or secure state to run this command.

Inputs:
- Smartcard PIN: 5 to 8 alphanumeric characters. PINs must be entered within 60 seconds of being requested.

Outputs:
- Text messages:

Insert Card for Component Set 1 or 2 and enter the PIN.
Insert Card for Authorizing Officer and enter the PIN.
Copy Complete.

Errors:
- Card not formatted - card not formatted
- Not a LMK card - card is not formatted for LMK or key storage.
- Smartcard error; command/return: 0003 - an invalid PIN was entered.
- Invalid PIN; re-enter - PIN is fewer than 5 or greater than 8 digits.
- Card not blank - copy failed.

Example:

```
Offline> CO <Return>
Insert Card for Component Set 1 or 2 and enter PIN: ********
<Return>
Insert Card for Authorizing Officer and enter PIN: ********
<Return>
Copy complete.
Offline>
```

Verify the Contents of a Smartcard

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **VC**

Function: To verify the key component or share held on a smartcard. The HSM reads the key component from the smartcard, computes the check value, compares this with the check value stored on the card and displays the result.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- Smartcard PIN: 5 to 8 alphanumeric characters. PINs must be entered within 60 seconds of being requested.

Outputs:
- Component Set check value:
  - For Variant LMKs, the length of the displayed check value is determined by the CS (Configure Security) setting "Restrict Key Check Value to 6 hex chars".
  - For Key Block LMKs, the length of the displayed check value is always 6 hex digits.
- Comparison: Pass or Fail.
- Text messages:
  - Check:
  - Compare with card:

Errors:
- Card not formatted - card not formatted
- Not a LMK card - card is not formatted for LMK or key storage.
- Smartcard error; command/return: 0003 - an invalid PIN was entered.
- Invalid PIN; re-enter - PIN is fewer than 5 or greater than 8 digits.

Example:

```
Online> VC <Return>
Insert card and enter PIN: ******** <Return>

Scheme: Variant
Check: 012345.
Compare with card: Pass.
Online>
```

If a smartcard is defective or cannot be successfully verified, replace it. Copy a verified smartcard (from the same set of components) onto a replacement.

NOTE: DISPOSE OF THE FAULTY SMARTCARD IN A SECURE MANNER.

Change a Smartcard PIN

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **NP**

Function: To select a new PIN for a smartcard (RACCs and RLMKs are supported) without changing any of the other details stored on the card.
The old PIN must be submitted before a change is effected and the new PIN must be supplied correctly at two consecutive prompts.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- Smartcard PIN: 5 to 8 alphanumeric characters. PINs must be entered within 60 seconds of being requested.

Outputs:
- Text messages:
  - Insert Card and press ENTER.
  - Enter current PIN.
  - Enter new PIN for smartcard.
  - Re-enter new PIN.
  - PIN change completed.

Errors:
- Card not formatted - card not formatted
- Not a LMK card - card is not formatted for LMK or key storage.
- Smartcard error; command/return: 0003 - an invalid PIN was entered.
- Invalid PIN; re-enter - PIN is fewer than 5 or greater than 8 digits.
- PINs did not agree - the new PINs entered for the smartcard did not match.

Example:

```
Online> NP <Return>
Insert card and press ENTER: <Return>
Enter current PIN: **** <Return>
Enter new PIN for smartcard: **** <Return>
Re-enter new PIN: **** <Return>
PINs did not agree
Enter new PIN for smartcard: **** <Return>
Re-enter new PIN: **** <Return>
PIN change completed
Online>
```

Read Unidentifiable Smartcard Details

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **RC**

Function: To read otherwise unidentifiable smartcards (RACCs and RLMKs supported).

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- Text messages:
  - Insert Card and press ENTER when ready.
  - This card is formatted for saving and retrieving HSM settings.
  - Version, as stored on card: decimal integer.
  - Date, as stored on card; format: YY/MM/DD.
  - Time, as stored on card; format: hh:mm:ss.
  - User ID, as stored on card; free format alphanumeric.
  - Issuer ID, as stored on card; free format alphanumeric.
  - Data Zone Size, as stored on card: decimal integer.
  - Max Data Free, as stored on card: decimal integer.

Errors:
- Card not formatted - card not formatted
- Not a LMK card - card is not formatted for LMK or key storage.

Example 1:

```
Online> RC <Return>
Insert card and press ENTER: <Return>
Format version: 0001
Issue time: 11:53:00
Issue date: 93/10/25
User ID: Bill Weasel
Issuer ID: Big Bank plc
User-data zone size: 0000
Free: 0392
Online>
```

Example 2:

```
Online> RC <Return>
Insert card and press ENTER: <Return>
This card is formatted for saving and retrieving HSM settings.
Online>
```

Eject a Smartcard

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **EJECT**

Function: To eject the smartcard from the smartcard reader.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs: None.

Errors: None.

Example:
```
Online> EJECT <Return>
Online>
```

DES Calculator Commands

The payShield 9000 HSM provides the following console commands to support the encryption and decryption of data with a given plaintext single, double or triple-length DES key:

Single-Length Key Calculator

Variant ☑	Key Block ☑	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **N**

Function: To encrypt and decrypt the given data block with the given single-length key.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- Key (no parity required): 16 hexadecimal characters.
- Data block: 16 hexadecimal characters.

Outputs:
- The data encrypted with the key.
- The data decrypted with the key.

Errors:
- Data invalid; please re-enter - the entered data does not comprise 16 hexadecimal characters. Re-enter the correct number of hexadecimal characters.

Example:

```
Online> N <Return>
Enter key: XXXX XXXX XXXX XXXX <Return>
Enter data: XXXX XXXX XXXX XXXX <Return>
Encrypted: YYYY YYYY YYYY YYYY
Decrypted: YYYY YYYY YYYY YYYY
Online>
```

Double-Length Key Calculator

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **$**

Function: To encrypt and decrypt the given data block with the given double-length key.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- The double-length key (odd parity is required): 32 hexadecimal characters.
- Data block: 16 hexadecimal characters.

Outputs:
- The data encrypted with the key.
- The data decrypted with the key.

Errors:
- Data invalid; please re-enter - the entered data does not comprise 32 hexadecimal characters. Re-enter the correct number of hexadecimal characters.

Example:

```
Offline> $ <Return>
Enter key: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter data: XXXX XXXX XXXX XXXX <Return>
Encrypted: YYYY YYYY YYYY YYYY
Decrypted: YYYY YYYY YYYY YYYY
Offline>
```

Triple-Length Key Calculator

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **T**

Function: To encrypt and decrypt the given data block with the given triple-length key.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- The triple-length key (odd parity is required): 48 hexadecimal characters.
- Data block: 16 hexadecimal characters.

Outputs:
- The data encrypted with the key.
- The data decrypted with the key.

Errors:
- Data invalid; please re-enter - Re-enter the correct number of hexadecimal characters.

Example:

```
Offline> T <Return>
Enter key: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
XXXX XXXX <Return>
Single, Double, or Triple length data? (S,D,T): S <Return>
Enter data: XXXX XXXX XXXX XXXX <Return>
Encrypted: YYYY YYYY YYYY YYYY
Decrypted: YYYY YYYY YYYY YYYY
Offline>
```

Legacy Commands

The following console commands are redundant, but are retained for backwards compatibility. They have been superseded by newer (usually more generic) commands – refer to the individual commands for details.

Note: The following commands always use the default LMK, which must be a variant LMK.

Generate a ZMK Component

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **F** (superseded by GC)

Function: To generate a ZMK component and display it in plain and encrypted forms.
Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- Clear text ZMK component: 16 or 32 hexadecimal characters.
- ZMK component encrypted under a variant of LMK pair 04-05: 16 or 32 hexadecimal characters.
- Component check value; formed by encrypting 64 binary zeros with the component and returning the left-most 24 bits: 6 hexadecimal characters.

Errors:
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes:
- The F, Z and D console commands create and manipulate ZMK components encrypted under LMK 04-05 variant 1. This is to maintain backward compatibility with previous releases of firmware, in which ZMK components were the only types of components supported.
- The recommended method of using ZMK components is to use the GC, EC and FK console commands, which process components of any key type.
- The length of the generated ZMK component will be dictated by the console Configure Security setting "ZMK Length: Single/Double".

Example:

```
Online> F <Return>
Clear ZMK Component: XXXX XXXX XXXX XXXX
Encrypted ZMK Component: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online>
```

Generate a ZMK & Write to Smartcards

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **generate.000.console**		

Command: **GZ** (superseded by GS)

Function: To generate a ZMK in 2 to 9 component and write the components to Smartcards.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **generate.000.console** must be authorized.

Inputs:
- Number of components, 1 numeric digit.

Outputs:
- ZMK encrypted under LMK pair 04-05: 16 or 32 hexadecimal characters.
- ZMK Check value; formed by encrypting 64 binary zeros with the ZMK; 16 hexadecimal characters, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Invalid PIN; re-enter - the entered PIN is not 4 – 8 digits.
- Smartcard error; command/return: 0003 - invalid PIN is entered.
- Not a LMK card - card is not formatted for LMK or key storage.
- Card not formatted - card is not formatted.
- Warning - card not blank. Proceed? [Y/N] - the smartcard entered is not blank.
- Overwrite ZMK component? [Y/N] - a ZMK component already exists on the card.
- Invalid entry - invalid number of components entered.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes:

- The length of the generated ZMK component will be dictated by the console Configure Security setting "ZMK Length: Single/Double".
- PINs must be entered within 60 seconds of being requested.

Example:

```
Online-AUTH> GZ <Return>
Enter number of components [2-3]: 2 <Return>
Insert card 1 and enter PIN: ******** <Return>
Make additional copies? [Y/N]: N <Return>
Insert card 2 and enter PIN: ******** <Return>
Make additional copies? [Y/N] N <Return>
Encrypted ZMK: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Encrypt a Clear ZMK Component

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.000.console**		

Command: **Z** (superseded by EC)

Function: To encrypt a clear text component and display the result at the console.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **component.000.console** must be authorized.

Inputs:
- Clear text ZMK component: 16 or 32 hexadecimal characters.

Outputs:
- The ZMK component encrypted under a variant of LMK pair 04-05: 16 or 32 hexadecimal characters.
- Component check value; formed by encrypting 64 binary zeros with the component and returning the left-most 24 bits: 6 hexadecimal characters.

Errors:
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Data invalid; please re-enter - the input data does not contain 16 or 32 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Component parity error; re-enter component - the entered component does not have odd parity on each byte. Ensure the component has odd parity and re-enter.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes:
- The F, Z and D console commands create and manipulate ZMK components encrypted under LMK 04-05 variant 1. This is to maintain backward compatibility with previous releases of firmware, in which ZMK components were the only types of components supported.
- The recommended method of using ZMK components is to use the GC, EC and FK console commands, which process components of any key type.
- The length of the generated ZMK component will be dictated by the console Configure Security setting "ZMK Length: Single/Double".

Example:

```
Online-AUTH> Z <Return>
Enter ZMK Component: ********** <Return>
Encrypted ZMK Component: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Form a ZMK from Encrypted Components

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.000.console**		

Command: **D** (superseded by FK)

Function: To form a ZMK from encrypted components. The components may either be entered from the console or read from Smartcards.
The manually entered components must have been encrypted using the Z command, or generated using the F command.
Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **component.000.console** must be authorized.

Inputs:
- Type of input, smartcard or keyboard.
- The number of key components to be entered: 2 to 9.
- The ZMK components, each encrypted under a variant of LMK pair 04-05: 16 or 32 hexadecimal characters.

Outputs:
- The ZMK encrypted under LMK 04-05: 16 or 32 hexadecimal characters.
- The key check value, formed by encrypting 64 binary zeros with the ZMK, and returning all 64 bits: 16 or 32 hexadecimal characters, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:
- Command only allowed from Authorized - The HSM is not authorized to perform this operation.
- Invalid entry - invalid number of components entered.
- Data invalid; please re-enter - the input data does not contain 16 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Component parity error; re-enter component - the entered component does not have odd parity on each byte. Re-enter the encrypted component and check for typographic errors.
- Invalid PIN; re-enter - the entered PIN is not 4 to 8 digits or the PIN does not match the PIN of the card.
- Card checksum mismatch - the components on the cards do not match.
- Smartcard error; command/return: 0003 – invalid PIN is entered.
- Not a LMK card – card is not formatted for LMK or key storage.
- Card not formatted – card is not formatted.
- No component card – there are no ZMK components on the card.
- Internal failure 12: function aborted – the contents of LMK storage have been corrupted or erased. Do not continue.

Inform the Security Department.

Notes:

- The F, Z and D console commands create and manipulate ZMK components encrypted under LMK 04-05 variant 1. This is to maintain backward compatibility with previous releases of firmware, in which ZMK components were the only types of components supported.
- The recommended method of using ZMK components is to use the GC, EC and FK console commands, which process components of any key type.
- The length of the generated ZMK component will be dictated by the console Configure Security setting "ZMK Length: Single/Double".
- PINs must be entered within 60 seconds of being requested.
- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Example 1: *This example forms a ZMK from plaintext components.*

```
Online-AUTH> D <Return>
Input components from smartcards? [Y/N]: N <Return>
Enter number of components (2-9): 2 <Return>
Enter encrypted component 1: ********** <Return>
Enter encrypted component 2: ********** <Return>
Encrypted ZMK: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Example 2: *This example forms a ZMK from components on smartcards.*

```
Online-AUTH> D <Return>
Input components from smartcards? [Y/N]: Y <Return>
Enter number of components (2-9): 2 <Return>
Insert card 1 and enter PIN: ******** <Return>
Insert card 2 and enter PIN: ******** <Return>
Encrypted ZMK: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Form a Key from Components

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.{*key*}.console**		

Command: **BK** (superseded by FK)

Function: To build a key from clear components. The components are not checked for parity, but odd parity is forced on the final key before encryption under the LMK.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **component.{*key*}.console** must be authorized, where 'key' is the key type code of the key being formed.

Inputs:
- Key Type; 1 numeric digit:
  "0" - Base Derivation Key (BDK)
  "1" - Card Verification Key (CVK)
  "2" - Zone PIN Key (ZPK)
- The number of key components to be entered: 2 to 9.
- The clear key component. Each BDK component must contain 32 hexadecimal characters and each CVK or ZPK component must contain 16 hexadecimal characters.

Outputs:
- The key formed by exclusive-ORing the entered components, forcing odd parity and encrypting under the appropriate LMK pair:
  - Key type "0" - LMK pair 28 - 29, 32 hexadecimal digits.
  - Key type "1" - LMK pair 14 - 15 variant 4, 16 hexadecimal digits.
  - Key type "2" - LMK pair 06 - 07, 16 hexadecimal digits.
- The key check value, formed by encrypting a block of zeros with the key, and returning all 64 bits: 16 hexadecimal characters, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Invalid entry - invalid number of components has been entered.
- Data invalid; please re-enter - the amount of input data is incorrect or non-hexadecimal characters have been entered. Re-enter the correct number of hexadecimal characters.
- Internal failure 12: function aborted – the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes: Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Example 1: *This example forms a BDK from components.*

```
Online-AUTH> BK <Return>
Enter key type [0=BDK, 1=CVK, 2=ZPK]: 0 <Return>
Enter number of components (2-9): 2 < Return>
Enter component 1: ********************************* <Return>
Enter component 2: ********************************* <Return>

Encrypted key: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Example 2: *This example forms a CVK from components.*

```
Online-AUTH> BK <Return>
Enter key type [0=BDK, 1=CVK, 2=ZPK]: 1 <Return>
Enter number of components (2-9): 3 <Return>
Enter component 1: ***************** <Return>
Enter component 2: ***************** <Return>
Enter component 3: ***************** <Return>

Encrypted key: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Example 3: *This example forms a ZPK from components.*

```
Online-AUTH> BK <Return>
Enter key type [0=BDK, 1=CVK, 2=ZPK]: 2 <Return>
Enter number of components (2-9): 2 <Return>
Enter component 1: ***************** <Return>
Enter component 2: ***************** <Return>

Encrypted key: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Import a CVK or PVK

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **import.{*key*}.console**		

Command: **IV** (superseded by IK)

Function: To import VISA PVK or CVK from encryption under ZMK to encryption under LMK.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **import.{*key*}.console** must be authorized, where 'key' is either '402' (CVK) or '002' (PVK).

Inputs:
- ZMK encrypted under LMK pair 04-05: 16 or 32 hexadecimal characters.
- Key type: C or P (for CVK or PVK respectively).
- Key A and B encrypted under the ZMK: 16 hexadecimal characters.
- ZMK variant: 1 or 2 digit, value 0-99 (or <Enter> to ignore). Used only when interworking with Atalla systems. Refer to the CS command. Note that this input is not requested when the ZMK variant support is set to off.

Outputs:
- Key A and B encrypted under LMK 14-15 or variant: 16 hexadecimal characters.
- Key check value: 16 hexadecimal characters, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Data invalid; please re-enter - incorrect input data length or invalid ZMK variant.
- Key parity error; re-enter - the ZMK or key entered does not have odd parity.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes:
- The completion of this activity will always be entered in the audit log irrespective of the AUDITOPTIONS settings,

Example:

```
Online-AUTH> IV <Return>
Key type [Pvk/Cvk]: C <Return>
Enter ZMK: xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx <Return>
(Enter ZMK variant: x <Return>, if enabled by CS command)
Enter key A: xxxxxxxxxxxxxxxx <Return>
Enter key B: xxxxxxxxxxxxxxxx <Return>
Key A under LMK: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Key B under LMK: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Generate a Zone PIN Key

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **B** (superseded by KG)

Function: To generate a random ZPK and return it encrypted under the LMK and under a ZMK (for transmission to another party). The ZPK can be a VISA Acquirer or Issuer Working key.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- The ZMK (VISA Zone Control Master Key, ZCMK) encrypted under LMK pair 04-05 (as generated using the D command): 16 or 32 hexadecimal characters.
- The ZMK key check value (as generated using the D command or by extracting the first 6 digits generated using the CK command): 6 hexadecimal characters.
- The ZMK variant: 1 or 2 digit, value 0-99 (or <Enter> to ignore). Used only when interworking with Atalla systems. Refer to the CS command. Note that this input is not requested when the ZMK variant support is set to off.

Outputs:
- The ZPK encrypted under the ZMK: 16 hexadecimal characters.
- The ZPK encrypted under LMK pair 06-07: 16 hexadecimal characters.
- The ZPK check value, formed by encrypting 64 binary zeros with the ZPK and returning the left-most 48 bits: 12 hexadecimal characters, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:

- Data invalid; please re-enter - the encrypted ZMK does not contain 16 or 32 hexadecimal characters, or the key check value is not 6 characters or the ZMK variant is invalid. Re-enter the correct number of hexadecimal characters.
- Key parity error; re-enter - the ZMK does not have odd parity on each byte. Re-enter the encrypted ZMK and check for typographic errors.
- Check failed, re-enter check value or abort - invalid 6 character check value has been entered.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example:

```
Online> B <Return>
Enter encrypted ZMK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter ZMK check value: XXXXXX <Return>
(Enter ZMK variant: X <Return>, if enabled by CS command)
ZPK encrypted for transmission: YYYY YYYY YYYY YYYY
ZPK encrypted for bank: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online>
```

Translate a Zone PIN Key

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **export.001.console**		

Command: **WK** (superseded by KE)

Function: To translate a ZPK from encryption under the LMK to encryption under a ZMK.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **export.001.console** must be authorized.

Inputs:
- ZMK encrypted under LMK pair 04-05: 16 or 32 hexadecimal characters.
- The ZPK encrypted under LMK pair 06-07: 16 hexadecimal characters.
- The ZMK variant: 1 or 2 digit, value 0-99 (or <Enter> to ignore). Used only when interworking with Atalla systems. Refer to the CS command. Note that this input is not requested when the ZMK variant support is set to Off.

Outputs:
- The ZPK encrypted under the ZMK: 16 hexadecimal characters.
- The key check value for the ZPK; generated by encrypting 64 binary zeros with the key: 16 hexadecimal characters, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:
- Command only allowed from Authorized - the HSM is not authorized to perform this operation.
- Data invalid; please re-enter - the encrypted ZMK does not contain 16 or 32 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Key parity error; re-enter key - the ZMK does not have odd parity on each byte. Re-enter the key and check for typographic errors.
- Key parity error - the ZPK does not have odd parity on each byte. Re-enter the key and check for typographic errors.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example:

```
Online-AUTH> WK <Return>
Enter encrypted ZMK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
(Enter ZMK variant: X <Return>, if enabled by CS command)
Enter encrypted ZPK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
ZPK encrypted under ZMK: YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ ZZZZ ZZZZ
Online-AUTH>
```

Generate a CVK Pair

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **KA** (superseded by KG)

Function: To generate a CVK pair and output the key encrypted under a variant of LMK pair 14-15.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- CVK A encrypted under a variant of LMK pair 14-15: 16 hexadecimal characters.
- The key check value for CVK A; formed by encrypting 64 binary zeros with the key and returning the left-most 24 bits: 6 hexadecimal characters.
- CVK B encrypted under a variant of LMK pair 14-15: 16 hexadecimal characters.
- The key check value for CVK B; formed by encrypting 64 binary zeros with the key and returning the left-most 24 bits: 6 hexadecimal characters.

Errors:
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example:

```
Online> KA <Return>
Encrypted CVK A: YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Encrypted CVK B: YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online>
```

Translate a CVK Pair from LMK to ZMK

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **KB** (superseded by KE)

Function: To translate a CVK pair from encryption under a variant of LMK pair 14-15 to encryption under a ZMK.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- CVK A encrypted under a variant of LMK pair 14-15: 16 hexadecimal characters.
- CVK B encrypted under a variant of LMK pair 14-15: 16 hexadecimal characters.
- ZMK encrypted under LMK pair 04-05: 16 or 32 hexadecimal characters.
- The ZMK variant: 1 or 2 digit, value 0-99 (or <Enter> to ignore). Used only when interworking with Atalla systems. Refer to the CS command. Note that this input is not requested when the ZMK variant support is set to off.

Outputs:
- CVK A encrypted under the ZMK.
- The key check value for CVK A, formed by encrypting 64 binary zeros with the key and returning the left-most 24 bits: 6 hexadecimal characters.
- CVK B encrypted under the ZMK.
- The key check value for CVK B, formed by encrypting 64 binary zeros with the key and returning the left-most 24 bits: 6 hexadecimal characters.

Errors:
- Data invalid; please re-enter - the encrypted key does not contain the correct number of hexadecimal characters or an invalid ZMK variant was entered.
- Key parity error - the key does not have odd parity on each byte. Re-enter the key and check for typographic errors.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example:

```
Online> KB <Return>
Enter encrypted CVK A: XXXX XXXX XXXX XXXX <Return>
Enter encrypted CVK B: XXXX XXXX XXXX XXXX <Return>
Enter encrypted ZMK: XXXX XXXX XXXX XXXX <Return>
(Enter ZMK variant: X <Return>, if enabled by CS command)
Encrypted CVK A: YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Encrypted CVK B: YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online>
```

Generate a Double-Length ZMK Component

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **DD** (superseded by GC)

Function: To generate a double-length random ZMK component and display the value at the console screen.
The command ignores the S/D (single/double length) parameter set by the CS (Configure Security) command.
Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs: None.

Outputs:
- The clear ZMK component.

Errors: None.

Example:
```
Online> DD <Return>
Clear ZMK component: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Online>
```

Form a ZMK from Clear Components

Variant ☑		Key Block ☒
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **component.000.console**		

Command: **DE** (superseded by FK)

Function: To enter a ZMK as either two single-length components (halves) or as two to nine double-length components.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM must be either in the Authorized State, or the activity **component.000.console** must be authorized.

Inputs:
- A half-length or full-length flag.
- The number of components.
- The clear components: each 16 or 32 hexadecimal characters.

Outputs:
- The ZMK encrypted under LMK pair 04-05.
- The key check value (KCV) for the ZMK, if restrict KCV is enabled in the CS command the output will be restricted to the 6 most significant digits with padding zeros for the remainder.

Errors:
- Command only allowed from Authorized - the HSM must be in Authorized State.
- Data invalid; please re-enter - the input data does not contain 16 or 32 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Notes:

- The DE command differs from the D command as follows:
  - It uses clear components (not encrypted components).
  - It forms the ZMK from two 16-character halves, or from two to nine 32-character components.
- When H/F is set to H, two 16-character halves are used: the user is prompted to enter 16 left characters, then 16 right characters. (The unit concatenates the left and right halves).
- When H/F is set to F, two to nine 32-character components are used: the user is prompted to enter the first component, then the second component, then the third, etc., according to the number of components to be entered. (The unit exclusive-OR combines the 32-character components).
- The parity of the components is not checked, but the resulting ZMK has odd parity forced before encryption.
- If the Echo parameter entered in the CS (Configure Security) command has been set to N (on), the clear components are echoed onto the screen as they are entered. If this is not required, either:
- Use the CS command to set the Echo parameter to F (off);

or

- Enter ∧ (i.e. press the Shift and 6 keys) before entering each component.
- Use of this command will always create an entry in the Audit Log – see Chapter 17 of the *payShield 9000 General Information Manual*.

Example 1:

```
Online-AUTH> DE <Return>
Half or full-length components? (H/F): H <Return>
Enter clear left half: XXXX XXXX XXXX XXXX <Return>
Enter clear right half: XXXX XXXX XXXX XXXX <Return>
Encrypted ZMK: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ
Online-AUTH>
```

Example 2:

```
Online-AUTH> DE <Return>
Half or full-length components? (H/F): F <Return>
Enter number of clear components (2-9): 3 <Return>
Enter component 1: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter component 2: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter component 3: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Encrypted ZMK: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ
Online-AUTH>
```

Generate a BDK

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **DG** (superseded by KG)

Function: To generate a random BDK, displaying it encrypted under the LMK pair and under a ZMK, and a BDK check value.
Equivalent to Host BI command

Notes: The command also prompts for a variant. If the recipient requires a variant to the ZMK, enter the appropriate variant number.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- ZMK encrypted under LMK pair 04-05 (generated by the DE command): 32 hexadecimal characters.
- ZMK variant (or <Re ignore). (The command ignores the setting of the Atalla ZMK variant support parameter entered in the CS (Configure Security) command).
- ZMK key check value (generated by the DE command) or the value generated by the console CK command or Host BU command.

Outputs:
- BDK encrypted under the ZMK: 32 hexadecimal characters.
- BDK encrypted under LMK pair 28-29: 32 hexadecimal characters.
- BDK check value.

Errors:
- Data invalid; please re-enter - the encrypted ZMK does not contain 32 hexadecimal characters or the key check value does not contain 8 hexadecimal characters. Re-enter the correct number of hexadecimal characters.
- Key parity error; please re-enter - the entered ZMK does not have odd parity on each byte. Re-enter the encrypted ZMK and check for typographic errors.
- Check failed; re-enter check value or abort - the ZMK check key value is not correct. Re-enter the correct check value.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example:

```
Online> DG <Return>
Enter encrypted ZMK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
Enter ZMK variant: X <Return>
Enter ZMK check value: XXXX XXXX <Return>
BDK encrypted for transmission: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
BDK encrypted under LMK: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZ ZZZZ
Online>
```

Generate & Export a KML

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **DA** (superseded by KG)

Function: To generate a double-length Master Load Key (KML) and return it encrypted under Variant 2 of LMK pair 04-05, and under a Zone Control Master Key (ZCMK). A check value for the KML is also returned.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- ZCMK, encrypted under LMK pair 04-05: 32 hexadecimal characters.
- (Optional) Atalla Variant – 1 or 2 numeric digit; this value is required only if support for Atalla variants is set using the "CS" console command (see Ref.2)

Outputs:
- KML, encrypted under the ZCMK: 32 hexadecimal characters.
- KML, encrypted under Variant 2 of LMK pair 04-05.
- KML check value, formed by encrypting a block of binary zeros with the key and returning the left-most 24 bits of the result: 6 hexadecimal characters.

Errors:
- Data invalid; please re-enter - the entered value does not contain 32 hexadecimal characters or invalid ZMK variant was entered. Re-enter the correct number of characters.
- Key parity error - the plaintext key does not have odd parity on each byte. Re-enter the correct value.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example:

```
Online> DA <Return>
Enter ZMK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
(Enter ZMK variant: V <Return>, if enabled by CS command.)
KML encrypted for transmission: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
KML encrypted under LMK: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online>
```

Generate a CSCK

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **YA** (superseded by KG)

Function: Generates a new CSCK and displays it encrypted under the LMK.
Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- A CSCK length flag.

Outputs:
- The new CSCK, encrypted under LMK 14-15 variant 4.

Errors:
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example 1:
```
Online> YA <Return>
Enter CSCK length [S/D]: D <Return>
CSCK encrypted under LMK: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Online>
```

Example 2:
```
Online> YA <Return>
Enter CSCK length [S/D]: S <Return>
CSCK encrypted under LMK: YYYY YYYY YYYY YYYY
Online>
```

Export a CSCK

Variant ☑	Key Block ☒	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **YB** (superseded by KE)

Function: This command accepts a Zone Master Key (ZMK) and a CSCK encrypted under the LMK. It decrypts and checks parity on both keys, and if correct encrypts the CSCK under the ZMK and displays it.

Note: This command will only operate using a variant default LMK.

Authorization: The HSM does not require any authorization to run this command.

Inputs:
- A flag to indicate the length of the ZMK.
- A ZMK encrypted under LMK 04-05 (generated by the "DE" command), 16/32 hexadecimal characters.
- A ZMK variant (or <Return> to ignore).
- Note: the Atalla variant support parameter (set with the "CS" command) is ignored. CSCK encrypted under LMK 14-15 variant 4, 16/32 hexadecimal characters.

Outputs:
- The CSCK encrypted under the ZMK.
- A Key Check Value (KCV) for the CSCK.

Errors:
- Data invalid; please re-enter - the keys are not 16 or 32 hexadecimal digits in length or invalid ZMK variant was entered.
- Key parity error - the key just entered did not have odd parity; check for typographical errors and re-enter.
- Internal failure 12: function aborted - the contents of LMK storage have been corrupted or erased. Do not continue. Inform the Security Department.

Example 1:

```
Online> YB <Return>
Enter ZMK length [S/D]: D <Return>
Enter ZMK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
(Enter ZMK variant: V <Return>, if enabled by CS command.)
Enter CSCK: XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>
CSCK encrypted for transmission: YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online>
```

Example 2:

```
Online> YB <Return>
Enter ZMK length [S/D]: S <Return>
Enter ZMK: XXXX XXXX XXXX XXXX <Return>
(Enter ZMK variant: V <Return>, if enabled by CS command.)
Enter CSCK: XXXX XXXX XXXX XXXX <Return>
CSCK encrypted for transmission: YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ
Online>
```

Chapter 5 – payShield Manager

Introduction

This chapter describes the commands used to configure the HSM for use with the payShield Manager.

Note: payShield 9000 HSMs must contain an appropriate license (HSM9-LIC037) before they can be remotely managed.

The payShield 9000 HSM provides the following console commands to support the payShield Manager:

From version 2.2a software, the HSM's private key, the certified public key and the Domain Authority self-signed public key certificate are recovered by use of the HSM Master Key (HRK) if a tamper attempt has occurred. Console commands to manage the HRK are included in *Chapter 6 – Certificate Management*.

Initialize Domain Authority

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **RI**

Function: To configure the Domain Authority parameter table and generate a Domain Authority RSA key pair and write the results to smart cards.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Domain Authority parameters (if modified from the default)
- Number of Domain Authority share cards (parameter "n", $3 \leq n \leq 9$)
- Number of shares to recover the Domain Authority private key (parameter "m", $3 \leq m \leq n$)

Outputs:
- Prompts, as above
- Key generation message
- Prompt to enter smart card and PIN
- Continuation message
- Message giving number of copies of public key card
- Summary of Domain Authority information

Errors:
- Invalid value
- Smart card warning – smart card already contains a Domain Authority private key share or a public key certificate

Notes:
- Legacy HSM smartcards are used to store the Domain Authority.
- The Domain Authority private key is broken into a number of shares to be used by the threshold scheme and each share is written to a separate smart card.
- In addition to the Domain Authority private key share, the self-certified Domain Authority public key and the Domain Authority parameter table will be written to each smart card.
- The length of the RSA modulus and the public exponent are determined by the values held in the Domain Authority parameter table.
- The Domain Authority information will be summarized at the end of the operation to ensure that no errors were made.

Example:

*This example demonstrates the use of the **RI** console command to generate a Domain Authority consisting of 5 (previously formatted) Domain Authority cards, any 3 of which are required to recover (and therefore use) the Domain Authority's private key.*

```
Secure> RI <Return>
Issuer name: [default = DomAuth]: <Return>
Signature algorithm [RSA]: (press enter) <Return>
Hash Algorithm: [SHA-1, SHA-256 (default = SHA-256)]: <Return>
Domain Authority RSA key length: [1024-2048 (default = 2048)]: <Return>
HSM RSA key length: [1024-2048 (default = 2048)]: 1536 <Return>
Card RSA key length: [1024-2048 (default = 2048)]: 1024 <Return>
Public exponent: [3, 65537 (default = 65537)]: <Return>
Enter number of Domain Authority private key shares: [3-9]: 5 <Return>
Enter number of shares to recover the Domain Authority private key: [3-
5]:3 <Return>
Enter 9 character alpha-numeric Domain Authority serial number : DA0000001
<Return>
Generating Domain Authority key pair ...

Insert first Domain Authority private key card and enter PIN: ********
<Return>
Insert second Domain Authority private key card and enter PIN: ********
<Return>
Insert third Domain Authority private key card and enter PIN: ********
<Return>
Insert fourth Domain Authority private key card and enter PIN: ********
<Return>
Insert fifth Domain Authority private key card and enter PIN: ********
<Return>

Domain Authority generation complete as follows:

Issuer name: DomAuth
Signature algorithm: RSA
Hash Algorithm: SHA-256
Domain Authority RSA key length: 2048
HSM RSA key length: 1536
Card RSA key length: 1024
Public exponent: 65537

Number of Domain Authority private key shares: 5
Number of shares to recover private key: 3

Secure>
```

Generate an HSM Certificate

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **RH**

Function: To generate the HSM's public/private key pair for use with remote management, and produce the HSM's public key certificate (signed by the Domain Authority), and store it inside the HSM.
The HSM's private key, the certified public key and the Domain Authority self-signed public key certificate are stored in secure memory. They are backed up internally when an HSM Master Key (HRK) is installed – see commands SK/SL for details.

Authorization: The HSM can be in any state to run this command.

Inputs:
- None.

Outputs:
- Prompt to enter smart card and PIN. (NOTE: the PIN must be entered within 60 seconds.)
- Key generation message
- Confirmation message
- Domain Authority Parameter Table (as retrieved from the Domain Authority share cards)

Errors:
- Public key error
- Private key error
- Invalid serial number

Notes:
- The Domain Authority private key is recovered from "m" share cards. The self-signed Domain Authority public key certificate, the Domain Authority parameter table, and the threshold scheme parameters are read from each card.
- The processing ensures that all "m" Domain Authority share cards contain identical copies of the Domain Authority parameter table and the threshold scheme parameters.
- After the Domain Authority private key is recovered, the Domain Authority Parameter Table is displayed to the user to ensure that the information is correct.
- The HSM generates an RSA key pair and uses the Domain Authority private key to create the HSM's public key certificate. The length of the RSA modulus and the public exponent for the generated key are determined by the values held in the Domain Authority parameter table.

Example**:**

This example shows the use of the RH command to generate an HSM's certificate. In this example, 3 shares are required to recover the Domain Authority private key.

```
Online> RH <Return>

Insert Domain Authority private key card and enter PIN: ******** <Return>
Insert another Domain Authority private key card and enter PIN: ********
<Return>
Insert another Domain Authority private key card and enter PIN: ********
<Return>

Domain Authority parameters as follows:

Issuer name: CertAuth
Signature algorithm: RSA
Hash Algorithm: SHA-256
Domain Authority RSA key length: 2048
HSM RSA key length: 1536
Card RSA key length: 1024
Public exponent: 65537

Continue generating HSM Certificate using the above Domain Authority
parameters [Y/N]: Y <Return>

Generating HSM key pair ...

HSM certificate generated and stored.

Online>
```

Backup Domain Authority Card

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **RZ**

Function: To backup an existing Domain Authority card that was previously created using the RI console command.

Authorization: The HSM must be in the Secure state to run this command.

Inputs:
- None

Outputs:
- Prompts to enter smart cards
- Prompt to enter number of backup cards

Errors:
- Not in Secure state
- Invalid PIN; re-enter
- Card not formatted or card inserted incorrectly

Example**:** *This example shows the use of the RZ command to create a backup of an existing Domain Authority card.*

```
Secure> RZ <Return>

Insert Domain Authority component card to be copied and enter PIN: ****
<Return>
Enter number of back-up cards required: 1 <Return>
Insert Domain Authority component card to be written to and enter PIN: ****
<Return>

Secure>
```

Add a RACC to the whitelist

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XA**

Function: To add a RACC to the whitelist on the HSM.

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- None

Outputs:
- None

Example 1**:**

```
Secure> XA <Return>

Insert payShield Manager Smartcard and press ENTER: <Return>
Enter PIN:  ****** <Return>

Do you want to add card XYZ123 to the whitelist? Y <Return>

Card XYZ123 added to whitelist.

Secure>
```

Decommission the HSM

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XD**

Function: To decommission the HSM by deleting the payShield Managers keys and groups.

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- None

Outputs:
- None

Example 1**:**

```
Secure> XD <Return>

Do you want to erase the payShield Manager's keys and groups? [Y/N]: Y
<Return>

Secure>
```

Remove RACC from the whitelist

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XE**

Function: To remove an RACC from the whitelist.

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- None

Outputs:
- None

Example 1**:**

```
Secure> XE <Return>

Choice    ID            Type
   1      ABC321        restricted
   2      XYZ123        restricted
Which RACC do you want to remove? 1 <Return>

Card ABC321 removed from whitelist

Secure>
```

Commission the HSM

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XH**

Function: To commission the HSM

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- None

Outputs:
- None

Example 1**:**

```
Secure> XH <Return>

Please have all Customer Trust Anchor (CTA) payShield Manager
smartcards available
Insert first CTA payShield Manager Smartcard and press ENTER: <Return>
Enter PIN:  ****** <Return>
Insert CTA payShield Manager Smartcard 2 of 3 and press ENTER: <Return>
Enter PIN:  ****** <Return>
Insert CTA payShield Manager Smartcard 3 of 3 and press ENTER: <Return>
Enter PIN:  ****** <Return>

Starting the commissioning of the HSM process...
Please insert left key card and press ENTER: <Return>
Enter PIN:  ****** <Return>
Please insert right key card and press ENTER: <Return>
Enter PIN:  ****** <Return>

Successfully commissioned HSM

Secure>
```

Generate Customer Trust Anchor

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XI**

Function: Generates the Customer Trust Anchor and stores them on smartcards.

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- Country
- State
- Locality
- Organization
- Organizational Unit
- Common Name
- Email
- Number of private shares
- Number of shares needed to recover private key

Outputs:
- None

Example 1**:**

```
Secure> XI <Return>

Please enter the certificate Subject information:
     Country Name (2 letter code) [US]: US <Return>
     State or Province Name (full name) []: Florida <Return>
     Locality Name (eg, city) []: Plantation <Return>
     Organization Name (eg, company) []: Thales <Return>
     Organizational Unit Name (eg, section) []: Production <Return>
     Common Name (e.g. server FQDN or YOUR name) [CTA]: CTA <Return>
     Email Address []: info@thalesesec.com <Return>

Enter number of Customer Trust Authority private key shares [3-9]: 3
<Return>
Enter number of shares to recover the Customer Trust Authority private
key [3-3]: 3 <Return>

   Issued to: CTA, Issued by: CTA
   Validity : Jan  9 10:28:49 2015 GMT to Jan  3 10:28:49 2040 GMT
   Unique ID: EE3CB7CE8343B464CC04278188CF7EB3 - 3DE05514 (Root)

Insert payShield Manager Smartcard 1 of 3 and press ENTER: <Return>
Enter new PIN for smartcard:  ****** <Return>
Re-enter new PIN:  ****** <Return>
Working....
CTA share written to smartcard.

Insert payShield Manager Smartcard 2 of 3 and press ENTER: <Return>
Enter new PIN for smartcard:  ****** <Return>
Re-enter new PIN:  ****** <Return>
Working....
CTA share written to smartcard.

Insert payShield Manager Smartcard 3 of 3 and press ENTER: <Return>
Enter new PIN for smartcard:  ****** <Return>
Re-enter new PIN:  ****** <Return>
Working....
CTA share written to smartcard.

Successfully generated a Customer Trust Anchor
Secure>
```

Make an RACC left or right key

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XK**

Function: Defines a RACC as either a left or right key in the whitelist on the HSM.

Authorization: The HSM must be in Secure state to run this command.

Inputs: Left or Right (card type)

Outputs:
- None

Example 1**:**

```
Secure> XK <Return>

Insert payShield Manager Smartcard and press ENTER: <Return>
Enter PIN:  ****** <Return>
Do you want to make ABC321 a [L]eft or [R]ight key? L <Return>

Card ABC321 is now a left key.

Secure>
```

Commission a smartcard

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XR**

Function: To commission a smartcard.

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- None

Outputs:
- None

Example 1**:**

```
Secure> XR <Return>

Please have all Customer Trust Anchor (CTA) payShield Manager
smartcards available
Insert first CTA payShield Manager Smartcard and press ENTER: <Return>
Enter PIN:  ******
Insert CTA payShield Manager Smartcard 2 of 3 and press ENTER: <Return>
Enter PIN:  ******
Insert CTA payShield Manager Smartcard 3 of 3 and press ENTER: <Return>
Enter PIN:  ******
Enforce a PIN change on first use? [Y/N]: N <Return>
Insert a payShield Manager Smartcard to be commissioned and press
ENTER: <Return>
Enter new PIN for smartcard:  ****** <Return>
Re-enter new PIN:  ****** <Return>
Do you wish to add the smartcard A3 to the HSM whitelist [Y/N]: Y
<Return>
Assign smartcard as a Left or Right Key RACC? [L/R/N]: N <Return>
Would you like to commission another card? [Y/N]: N <Return>

Secure>
```

Transfer existing LMK to RLMK

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XT**

Function: To transfer an existing HSM LMK stored on legacy smartcards to payShield Manager RLMK cards for use through the payShield Manager.

In order to transfer a Variant LMK you will be required to fully reassemble the LMK (bring all the components together). Then, the fully formed Variant LMK is split amoung shares onto the pre-comissioned payShield Manager RLMK cards.

For Key Block LMKs, they are not stored as components on non-payShield Manager smart cards, but as shares. However, you must bring a quarom of share holders together, reconstitute the LMK, and then split it amoung shares onto the pre-comissioned payShield Manager RLMK cards.

Authorization: The HSM must be in Secure state to run this command.

Inputs:
- Number of shares to split LMK into
- Number of Components required to reconstitute LMK

Outputs:
- None

Example 1**:**

```
Secure> XT <Return>

Please have all the local LMK components and enough commissioned RACCs
to receive the LMK ready.

Insert card and press ENTER: <Return>
Enter PIN: ***** <Return>

Check: 268604
Load more components? [Y/N]: N <Return>

LMK Check:      268604
LMK key scheme: Variant
LMK algorithm:  3DES(2key)
LMK status:     Test

Is this the LMK you wish to transfer? [Y/N]: Y <Return>

Enter the number of shares to split the LMK into: [2-9]: 2 <Return>
Enter the number of shares required to reconstitute the LMK: [2-2]: 2
<Return>

Insert a commissioned card 1 of 2 and press ENTER: <Return>
Enter PIN:  ****** <Return>

Card Check: E0CBF4
LMK share written to smartcard.

Insert a commissioned card 2 of 2 and press ENTER: <Return>
Enter PIN:  ****** <Return>

Card Check: E0CBF4
LMK share written to smartcard.
Want to test the reassembly of the LMK? Y <Return>

Please have all the RLMK shares ready
Insert RLMK card and press ENTER: <Return>
Enter PIN:  ****** <Return>
```

```
LMK share 1 read (1 of 2) Card Check: E0CBF4
Insert RLMK card and press ENTER: <Return>
Enter PIN:  ****** <Return>
LMK share 2 read (2 of 2) Card Check: E0CBF4

LMK Check 268604

Secure>
```

Decommission a smartcard

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **XX**

Function: To decommission a payShield Manager smartcard.

Authorization: The HSM may be in any state to run this command.

Inputs: •None

Outputs: •None

Example 1**:**

```
Secure> XX <Return>

Please insert card to decommission and press ENTER: <Return>
Warning: Resetting a payShield Manager Smartcard to its original state
will erase all key material from the card.

Are you sure? [Y/N]: Y <Return>

payShield Manager Smartcard successfully decommissioned
Would you like to decommission another card? [Y/N]: N <Return>

Secure>
```

HSM commissioning status

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **XY**

Function: To show the state of the HSM Management commissioning and whitelist.

Authorization: The HSM may be in any state to run this command.

Inputs:
- None

Outputs:
- Customer Trust Anchor installed
- HSM Public Key installed
- Is HRK password user defined
- Is HRK available for use
- Authorized RACCs

Example 1**:**

```
Secure> XY <Return>

Customer Trust Anchor Installed      : Yes
   1 - Issued to: CTA, Issued by: CTA
   Validity : Dec 11 16:20:17 2014 GMT to Dec  5 16:20:17 2039 GMT
   Unique ID: A86AF14A28253F313B00516875E69C9B - 21722E26 (Root)


HSM Public Key Certificate Installed : Yes
   2 - Issued to: A4665275330S, Issued by: CTA
   Validity : Jan  9 10:44:20 2015 GMT to Jan  3 10:44:20 2040 GMT
   Unique ID: 99734BD96B59EFF036B8218FD3DA2EDD - 21722E26

Is HRK passphrase user defined       : No
Is HRK available for use             : Yes

Authorized RACCs                     : 4
ID                    RACC Type
ABC321                left key
SCA00000001           left key
SCB00000001           right key
XYZ123                restricted

Secure>
```

Duplicate CTA share

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **XZ**

Function: To duplicate a CTA share smartcard.

Authorization: The HSM must be in Secure state to run this command.

Inputs: •None

Outputs: •None

Notes: The CTA must be installed prior to running this command.

Example 1**:**

```
Secure> XZ <Return>

Insert a CTA share payShield Manager Smartcard to be duplicated:
Enter PIN:  ****** <Return>
Working...
Please insert a commissioned payShield Manager smartcard and press
ENTER: <Return>
Enter PIN:  ****** <Return>
Working...
CTA share written to smartcard.

Secure>
```

Chapter 6 – Certificate Management

Introduction

This chapter describes the commands used to configure a payShield 9000 HSM such that the host and/or management connection is protected using TLS. For full details of implementing a secure host TLS connection, refer to the section "Secure Host Communications" in Chapter 14 of the *payShield 9000 General Information Manual*.

The Certificate Requests and Certificates may be stored on / loaded from a regular USB memory stick.

The required format for the USB memory stick is FAT32. The Operating System used in the payShield 9000 supports most types of USB memory stick, but may not have the drivers for some of the newer types. If difficulties are experienced when trying to read from or write to a USB device, an alternative memory stick should be used.

The HSM's certificate signing request (CSR) structure is compliant with PKCS#10. The client must use the same key type as is included in the HSM's CSR.

The HSM uses certificate formats compliant with X.509.

Note: payShield 9000 HSMs must contain an appropriate license (HSM9-LIC036) before the host connection can use TLS.

The payShield 9000 HSM provides the following console commands to manage the HSM's private key, the certified public key and the CA self-signed public key certificate to support host and management TLS:

The HRK is also required to allow recovery of the HSM's private key, the certified public key and the CA self-signed public key certificate used for payShield Manager - see *Chapter 5 – payShield Manager*.

Generate Certificate Signing Request

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SG**

Function: To generate the HSM's public/private key pair for use with host or management TLS, and output the public key in the form of a Certificate Signing Request (CSR).
The private key is stored in tamper protected memory. It is backed up internally using the HSM Master Key (HRK) – see commands SK for details.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Certificate fields (Country, State, Locality, Org Name, Org Unit Name, Common Name, E-mail Address).
- Filename when saving to USB memory stick

Outputs:
- Prompts, as above
- Prompt to save to USB memory stick
- Certificate Signing Request

Errors:
- File exists – replace?

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.
- The HRK must be installed (using the SK console command) prior to using this command.
- The exported file will automatically have the extension ".CSR".
- A maximum certificate chain length of 6 is supported.
- The required format for the USB memory stick is FAT32. The Operating System used in the payShield 9000 supports most types of USB memory stick, but may not have the drivers for some of the newer types. If difficulties are experienced when trying to read from or write to a USB device, an alternative memory stick should be used.

Example 1: *This example demonstrates the use of the **SG** console command to generate a management TLS key pair, and output the certificate request (CSR) to a USB storage device for signing by an external CA.*

```
Secure> SG <Return>

What type of certificate do you want to generate (using specific key
type)?
  1 - Host TLS
  2 - Management TLS
Type: 2 <Return>

Select a method for generating a certificate:
  1 - Externally signed (CSR)
  2 - Internally signed (signed by Customer Security Domain)
Selection: 1 <Return>

Please enter the certificate Subject information:

   Country Name (2 letter code) []: UK <Return>
   State or Province Name (full name) []: London <Return>
   Locality Name (eg, city) []: London <Return>
   Organization Name (eg, company) []: Thales eSecurity <Return>
   Organizational Unit Name (eg, section) []: Support
   Common Name (e.g. server FQDN or YOUR name) [B4665309394G-mgmt]:
<Return>
   Email Address []: support@thalesesecurity.com <Return>

Do you wish to save the CSR to a file [Y/N]: Y <Return>
Enter filename: CSR-CH <Return>
-----BEGIN CERTIFICATE REQUEST-----
MIIB6zCCAXECAQIwgaQxCzAJBgNVBAYTAlVLMQ8wDQYDVQQIDAZMb25kb24xDzAN
BgNVBAcMBkxvbmRvbjEZMBcGA1UECgwQVGhhbGVzIGVTZWN1cml0eTEQMA4GA1UE
CwwHU3VwcG9ydDEaMBgGA1UEAwwRQjQ2NjUzMDkzOTRHLW1nbXQxKjAoBgkqhkiG
9w0BCQEWG3N1cHBvcnRAdGhhbGVzZXNlY3VyaXR5LmNvbTB2MBAGByqGSM49AgEG
BSuBBAAiA2IABBYSgNDvK03fi+XG8Dj9kGFS3QZu33kHsvVolybyCTbJBmbcDCdg
3lw2SnuP9XGfrc4ajs9/+SKDlWOtXu4s70YXXZFGHJyYrgdH8s/uANHrPe+TQsEK
ox50OdO8bcxFy6BNMEsGCSqGSIb3DQEJDjE+MDwwDAYDVR0TAQH/BAIwADAOBgNV
HQ8BAf8EBAMCB4AwHAYDVR0RBBUwE4IRQjQ2NjUzMDkzOTRHLW1nbXQwCQYHKoZI
zj0EAQNpADBmAjEAqjADvRmOU9SdRpAplfK4rixWcqa6X7KrrcQE5NpFFXnsML3J
2xJC63ONP6Far2pVAjEAzdEr+dZ/JqYriXrhALja+Gij4pvERwf4iJX3n0xo/Z/R
Ng+nlyURCZMQ/YZd+iHs
-----END CERTIFICATE REQUEST-----

Successfully generated TLS management certificate

Secure>
```

Import Certificate

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SI**

Function: To import a certificate for storage inside the HSM for use with or host or management TLS.
The certificate may be one of the following:
- HSM certificate
- Client certificate
- Sub-CA certificate (for either HSM or client)
- Root-CA certificate (for either HSM or client)

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- File selection
- Prompt for import of additional certificates

Outputs:
- Prompts, as above
- Filenames of certificates on USB memory stick
- Summary of imported certificate (Issued to/by, Validity, ID)
- Chain of Trust statement (for an HSM certificate)

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.
- The HSM's public/private key pair must be installed (using the SG console command) prior to using this command.
- The file(s) to be imported must have the extension ".CRT".
- A maximum certificate chain length of 6 is supported.
- The required format for the USB memory stick is FAT32. The Operating System used in the payShield 9000 supports most types of USB memory stick, but may not have the drivers for some of the newer types. If difficulties are experienced when trying to read from or write to a USB device, an alternative memory stick should be used.

Example 1: *This example demonstrates the use of the **SI** console command to import the root CA certificate (that signed the HSM's certificate) into the HSM.*

```
Secure> SI <Return>

Please select the function that the key was created for:
  1 - Host TLS
  2 - Management TLS
Type: 2 <Return>

Select File
   1 - HSM-0001.crt
   2 - BankXYZRootCA.crt
   3 - Client.crt
   4 - ClientRootCA.crt
File: 2 <Return>

Imported Trusted CA Certificate
         Issued to: RootCA, Issued by: RootCA
         Validity : Oct  4 10:58:16 2013 GMT to Oct  2 10:58:16 2023 GMT
         Unique ID: 00 - D06AA1E4 (Root)

Do you wish to import another certificate? N <Return>

Secure>
```

Export HSM Certificate's Chain of Trust

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SE**

Function: To export the HSM certificate's chain of trust (i.e. the chain of certificates required to authenticate the HSM's certificate, up to and including the root CA certificate).

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Filename when saving to USB memory stick

Outputs:
- Prompts, as above
- Prompt to save to USB memory stick
- Certificate Chain of Trust is displayed at the console, and (if requested) saved to the USB memory stick

Errors:
- File exists – replace?

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.
- The HSM's public/private key pair must be installed (using the SG console command) prior to using this command.
- The exported file will automatically have the extension ".CRT".
- A maximum certificate chain length of 6 is supported.
- The required format for the USB memory stick is FAT32. The Operating System used in the payShield 9000 supports most types of USB memory stick, but may not have the drivers for some of the newer types. If difficulties are experienced when trying to read from or write to a USB device, an alternative memory stick should be used.

Example 1: *This example demonstrates the use of the **SE** console command to export the HSM certificate's chain of trust (in this case, just the root CA certificate) to a USB memory stick.*

```
Secure> SE <Return>

Please select the function that the key was created for:
  1 -  Host TLS
  2 -  Management TLS
Type: 2 <Return>
Do you wish to save to a file [Y/N]: Y <Return>
Enter filename: BankXYZRootCA <Return>

Bank XYZ

-----BEGIN CERTIFICATE-----
MIID+TCCAuGgAwIBAgIJAJyPxxP6oxAQMA0GCSqGSIb3DQEBBQUAMIGyMQswCQYD
VQQGEwJVSzEYMBYGA1UECBMPQnVja2luZ2hhbXNoaXJlMRUwEwYDVQQHEwxMb25n
IENyZW5kb24xDzANBgNVBAoTBlRoYWxlczEMMAoGA1UECxMDUE1HMR4wHAYDVQQD
ExVwYXlTaGllbGQgQ2VydGlmaWNhdGUxMzAxBgkqhkiG9w0BCQEWJGphbWVzLnRv
cmp1c3NlbkB0aGFsZXMtZXNlY3VyaXR5LmNvbTAeFw0xMzA1MDkxMDU5MjJaFw0y
MzA1MDcxMDU5MjJaMIGyMQswCQYDVQQGEwJVSzEYMBYGA1UECBMPQnVja2luZ2hh
bXNoaXJlMRUwEwYDVQQHEwxMb25nIENyZW5kb24xDzANBgNVBAoTBlRoYWxlczEM
MAoGA1UECxMDUE1HMR4wHAYDVQQDExVwYXlTaGllbGQgQ2VydGlmaWNhdGUxMzAx
BgkqhkiG9w0BCQEWJGphbWVzLnRvcmp1c3NlbkB0aGFsZXMtZXNlY3VyaXR5LmNv
bTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANTFR+dFeafMZsMwgeOK
vWxjmaUOP6z5mK+qeD4wYvNP5cv1GVqKoMFTNkJL+jeBSyo39IR0T4AoalroUb6F
yi76nmv0VVqFgPWIS92bRBozGp8dZU09aJQGCuOIjEvKuUtddWrpp0ClFEnTXXsx
LpfjTal5vSl+D9lazkMiFxdi7OUQyf6CiVuoch7bq0A4nmcjSlPyE/b3FpJn6zul
S+/DvRo4N4wJBHkZftAyPHZUYaV84perRG4CRbirFUfpRH1kVC+P6Gal/KMKWlzE
kKJOIxZqtaU973/AD4CV2QZtMurFC9m9p84uOW2SinMeKEdolVTFgVo+h3KjFHM/
yVsCAwEAAaMQMA4wDAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQUFAAOCAQEAoHEN
1QyqWSTXkhtAnu+F3gy/Qs/wYLszaYYlBUSQasjN866SzRC/jVtYT6UYabvOke5B
9Z4KNsICkRtmgdYpic0kjK40RjUdw4QZu4jC+EM4eY8HTa7fSaH1nxrkPAEUwNKZ
o3Re+3jQeIx6gi5rnLf/FZ1cEP1fySh0hzuSo2xSIY/hwUWhlZJYZKBu3wzfHG1d
GB7D4xU4jUTvkKJQDuCHUdSrf+cMstN9dkrhYNNw49L9tYrD0ZzlPM3rVXD28uAL
Wt+CPOtsjIixNRl8vZmEVJDWJaRibCcfrTeDBs4O3hmAgx/Mdv5FX/NSjhZZO15m
X4FkYiQv2CJb7J/vAw==
-----END CERTIFICATE-----

Secure>
```

View Installed Certificate(s)

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **SV**

Function: To view the list of currently installed certificates (for use with host and management TLS). Individual certificates can be displayed in full.

Authorization: The HSM can be in any state to run this command.

Inputs:
- Certificate to be displayed in full.

Outputs:
- List of currently installed TLS certificates.
- Prompts, as above
- Status of HSM's TLS private key – installed or not installed
- HSM TLS Certificate installed – maximum of 1 certificate
- Client TLS Certificate(s) installed – maximum of 10 certificates
- CA TLS Certificate(s) installed – maximum of 10 certificates
- Chain of trust validity – for the HSM's TLS certificate chain
- Contents of a selected certificate.
- A maximum certificate chain length of 6 is supported.

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.

Example 1: *This example demonstrates the use of the **SV** console command to view the list of currently installed management TLS certificates, and to display the contents of one of the CA certificates.*

```
Online> SV <Return>
Please select the function that the key was created for:
  1 - Host TLS
  2 - Management TLS
Type: 2 <Return>
TLS Management Private Key installed: Yes

TLS Management Certificate installed:

     1 - Issued to: B4665309394G-mgmt, Issued by: CSDH
         Validity : Aug  1 16:49:41 2017 GMT to Jul 26 16:49:41 2042 GMT
         Unique ID: ECADBCC35326FC40A1F71C613ECC64C9 - 3999D1DA

TLS Management Client certificate(s) installed: No

TLS Management CA Certificate(s) installed:

     2 - Issued to: ch, Issued by: ch
         Validity : Jun  8 12:06:38 2017 GMT to Jun  2 12:06:38 2042 GMT
         Unique ID: 921E0A9891EE8F4092B8FD8C304A08AC - 3A057A4B (Root)

     3 - Issued to: RootCA, Issued by: RootCA
         Validity : Oct  4 10:58:16 2013 GMT to Oct  2 10:58:16 2023 GMT
         Unique ID: 00 - D06AA1E4 (Root)

Chain of Trust validated: No

Select an item to view: 2 <Return>

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            92:1e:0a:98:91:ee:8f:40:92:b8:fd:8c:30:4a:08:ac
    Signature Algorithm: ecdsa-with-SHA256
        Issuer: CN=ch
        Validity
            Not Before: Jun  8 12:06:38 2017 GMT
            Not After : Jun  2 12:06:38 2042 GMT
        Subject: CN=ch
        Subject Public Key Info:
            Public Key Algorithm: id-ecPublicKey
                Public-Key: (521 bit)
                pub:
                    04:00:e9:52:c5:d0:0c:5b:d1:1b:eb:39:8c:53:e4:
                    ca:7e:25:fa:02:da:73:44:ce:ce:22:09:f0:88:c2:
                    85:2e:ca:f1:4b:85:f1:ba:61:b4:49:a8:d2:8f:0a:
                    ba:12:00:ec:ff:d7:6b:6a:b2:4e:0e:d0:cf:45:20:
                    5d:d5:fc:f6:bc:47:bf:01:0c:06:23:98:1b:a5:f4:
                    70:dd:30:17:fc:3b:1c:52:12:b0:6f:0d:06:37:89:
                    34:05:91:69:94:de:47:c4:d2:69:8b:22:4d:8a:23:
                    9c:37:5e:d3:b8:fa:83:9d:26:95:7b:3a:0a:80:b2:
                    1e:c7:aa:92:6a:14:bc:bf:0c:93:49:b7:8b
                ASN1 OID: secp521r1
                NIST CURVE: P-521
        X509v3 extensions:
            X509v3 Basic Constraints: critical
                CA:TRUE, pathlen:1
            X509v3 Key Usage: critical
                Digital Signature, Certificate Sign
    Signature Algorithm: ecdsa-with-SHA256
         30:81:87:02:41:20:99:1e:c3:58:64:88:38:79:f2:20:07:91:
         3c:1e:38:40:62:74:52:f2:24:b5:f2:0c:67:23:77:d5:b8:8d:
         9a:e2:e0:17:7c:09:ab:87:5c:9d:11:59:04:96:a6:86:dc:ed:
         ba:ab:b1:7b:45:c7:cc:8d:38:8a:f9:8d:82:e0:52:23:02:42:
         01:7a:c7:72:97:71:be:ff:1a:76:ce:fe:c5:67:ad:f6:a8:be:
         62:87:b1:de:26:76:84:59:30:cf:dd:e7:2f:0c:dc:95:5d:b0:
         10:64:78:ba:08:c2:09:f7:38:a7:a7:ff:80:f1:9a:d3:74:f3:
         e4:55:88:c4:00:5f:8c:27:8c:d2:26:9f

Do you wish to view another item? N <Return>
Online>
```

Delete Installed Certificate(s)

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SD**

Function: To delete a currently installed host or management TLS certificate.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Certificate to be deleted.

Outputs:
- Prompts, as above
- List of currently installed certificates.
- Status of HSM's private key – installed or not installed
- HSM Certificate installed
- Client Certificate(s) installed
- CA Certificate(s) installed
- Chain of trust validity – for the HSM's certificate chain
- Prompt to delete another certificate

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.

Example 1: *This example demonstrates the use of the **SD** console command to remove a client certificate from the HSM.*

```
Secure> SD <Return>

Please select the function that the key was created for:
  1 -  Host TLS
  2 -  Management TLS
Type: 2 <Return>
TLS Management Private Key installed: Yes

TLS Management Certificate installed:

     1 - Issued to: B4665309394G-mgmt, Issued by: CSDH
         Validity : Aug  1 16:49:41 2017 GMT to Jul 26 16:49:41 2042 GMT
         Unique ID: ECADBCC35326FC40A1F71C613ECC64C9 - 3999D1DA

TLS Management Client certificate(s) installed: No

TLS Management CA Certificate(s) installed:

     2 - Issued to: RootCA, Issued by: RootCA
         Validity : Oct  4 10:58:16 2013 GMT to Oct  2 10:58:16 2023 GMT
         Unique ID: 00 - D06AA1E4 (Root)

     3 - Issued to: ch, Issued by: ch
         Validity : Jun  8 12:06:38 2017 GMT to Jun  2 12:06:38 2042 GMT
         Unique ID: 921E0A9891EE8F4092B8FD8C304A08AC - 3A057A4B (Root)

     4 - Issued to: CSDH, Issued by: ch
         Validity : Aug  1 16:49:37 2017 GMT to Jul 26 16:49:37 2042 GMT
         Unique ID: CE6ECE100148124B61B0B1E2C593DA5D - 3A057A4B

Chain of Trust validated:

         ch (Root)
            CSDH

     5 - TLS Management Private Key

Select an item to delete (6 for ALL): 4 <Return>
```

```
You have selected to delete #4.
Are you sure you wish to proceed? [Y/N]: Y <Return>

Do you wish to delete another item? N <Return>

Secure>
```

Generate HRK

Variant ☑		Key Block ☑
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SK**

Function: To generate a new HSM Recovery Key (HRK). Once installed, the HRK will be used to back-up secret key material inside the HSM into persistent memory (a process known as key synchronization).
The following secret key material is backed-up in this process:

- Secure Host Communications TLS key material:
  - HSM's private key
- payShield Manager TLS key material:
  - HSM's private key
  - HSM's public key certificate
  - CA public key certificate

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Passphrases 1 & 2 (each entered twice for verification).

Outputs:
- Prompts, as above.
- Passphrase rules.
- Creating HRK message.
- Key synchronization message.

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.
- The HRK replaces the RMK (used in previous versions of software).

Example 1: *This example demonstrates the use of the **SK** console command to generate an HRK.*

```
Secure> SK <Return>

**** NOTE ****
Passphrase rules as follows:
1 - Must be between 8 and 30 characters long.
2 - Can contain spaces
3 - Must be comprised of (at a minimum):
   2 digits
   2 uppercase characters
   2 lowercase characters
   2 symbols (e.g. !/?.#:')

   Enter administrator 1 passphrase: ********************
Re-enter administrator 1 passphrase: ********************

   Enter administrator 2 passphrase: **************
Re-enter administrator 2 passphrase: **************

Creating HRK. Please, wait ... DONE

HRK generated successfully

Key synchronization complete

Secure>
```

Change HRK Passphrase

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SP**

Function: To change one of the passphrases associated with the HRK.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Existing passphrase 1 or 2.
- New passphrase 1 or 2 (entered twice for verification).

Outputs:
- Prompts, as above.
- Passphrase rules.
- Creating HRK message.
- Key synchronization message.

Notes:
- The HRK replaces the RMK (used in previous versions of software).
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.

Example 1: *This example demonstrates the use of the **SP** console command change administrator #1's HRK passphrase.*

```
Secure> SP <Return>

**** NOTE ****
Passphrase rules as follows:
1 - Must be between 8 and 30 characters long.
2 - Can contain spaces
3 - Must be comprised of (at a minimum):
   2 digits
   2 uppercase characters
   2 lowercase characters
   2 symbols (e.g. !/?.#:')
4 - Cannot use the same passphrase that was used within the past 10 previous
attempts

Select administrator password to change [1,2]: 1
   Enter administrator 1 current passphrase: ********************
   Enter administrator 1 new passphrase: ************
Re-enter administrator 1 new passphrase: ************


Changing passphrases. Please, wait ... DONE

HRK generated successfully

Secure>
```

Restore HRK

Variant ☑	Key Block ☑	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **SL**

Function: To restore the HRK (and also the secret key material backed-up by the HRK) in the event of erasure of tamper protected memory.

Authorization: The HSM must be in the secure state to run this command.

Inputs:
- Passphrases 1 & 2.

Outputs:
- Prompts, as above.
- Restoring HRK message.
- Key synchronization message.

Errors:
- HRK already loaded.

Notes:
- See Chapter 14 of the *payShield 9000 General Information Manual* for a description of how Secure Host Communications works on the payShield 9000.
- The HRK replaces the RMK (used in previous versions of software).

Example 1: *This example demonstrates the use of the **SL** console command to generate an HRK.*

```
Secure> SL <Return>

Enter administrator 1 passphrase: ********************
Enter administrator 2 passphrase: **************

Recovering HRK. Please, wait ... DONE

HRK recovered successfully

Key synchronization complete

Secure>
```

Chapter 7 – KMD Support Commands

Introduction

This section describes the set of console commands that facilitate the operation of the Thales Key Management Device (KMD) in a PCI PIN compliant manner.

Generate KTK Components

Variant ☐		Key Block ☐
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **KM**

Function: To generate the components of a KMD Transport Key (KTK), and store the components on smartcards.

Authorization: None

Inputs:
- Number of components to generate
- Prompt for smartcards & PINs to be entered

Outputs:
- Check Value of new KTK

Example 1: *This example demonstrates the use of the **KM** console command to generate two KTK components on smartcards.*

```
Secure> KM <Return>

Enter number of components [2-3]: 2 <Return>
Insert card 1 and enter PIN: ****** <Return>
KTK Component Check: ZZZZZZ
Make additional copies? [Y/N]: N <Return>

Insert card 2 and enter PIN: ****** <Return>
KTK Component Check: ZZZZZZ
Make additional copies? [Y/N]: N <Return>

KTK Check Value: ZZZZZZ

Secure>
```

Install KTK

Variant ☐	Key Block ☐	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **KN**

Function: To install a KMD Transport Key (KTK) into the HSM.

Authorization: None

Inputs:
- Number of components to use
- Prompt for smartcards & PINs to be entered

Outputs:
- Check value of new KTK

Example 1: *This example demonstrates the use of the **KN** console command to install a KTK in KTK Id 01, using two smartcards.*

```
Secure> KN <Return>
Enter KTK id [00-19]: 01 <Return>
Enter comments: KTK for KMD in secure room <Return>
KTK in selected location must be erased before proceeding.
Erase KTK? [Y/N]: Y <Return>

Load KTK in components
Insert card and enter PIN: ****** <Return>
Component Check: ZZZZZZ
Load more components? [Y/N]: Y <Return>

Insert card and enter PIN: ****** <Return>
Component Check: ZZZZZZ
Load more components? [Y/N]: N <Return>

KTK id: 01
KTK key scheme: Variant
KTK algorithm: 3DES (2key)
Comments: KTK for KMD in secure room
Confirm details? [Y/N]: Y <Return>

Secure>
```

View KTK Table

Variant ☐	Key Block ☐	
Online ☑	Offline ☑	Secure ☑
Authorization: **Not required**		

Command: **KT**

Function: To display the KTK table.

Authorization: None

Inputs:
- None

Outputs:
- List of installed KTKs

Example 1: *This example demonstrates the use of the **KT** console command to display the list of all KTKs currently installed in the HSM.*

```
Online> KT <Return>

KTK table:
ID Scheme   Algorithm   Check   Comments
01 Variant  3DES(2key)  292489  KTK for KMD in secure room
03 Variant  3DES(2key)  549235  KTK for 2nd KMD

Online>
```

Import Key encrypted under KTK

Variant ☑		Key Block ☑
Online ☑	Offline ☑	Secure ☑
Authorization: **Required** Activity: **command.kk.console**		

Command: **KK**

Function: To translate a key from encryption under a KTK to encryption under an LMK.

Authorization: The HSM must either be in the Authorized State, or the activity **command.kk.console** must be authorized.

Inputs:
- LMK Identifier
- Key Type Code
- Key Scheme (LMK)
- KTK Identifier
- Key encrypted under KTK

Outputs:
- Key encrypted under LMK

Example 1: *This example demonstrates the use of the **KK** console command to import a double-length DES ZMK (key type 000) from encryption under KTK Id 01 to encryption under LMK Id 02.*

```
Online-AUTH> KK <Return>

Enter LMK id: 02 <Return>
Enter Key type: 000 <Return>
Enter Key Scheme (LMK): U <Return>

Enter KTK id: 01 <Return>
Enter key: U XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX <Return>

LMK encrypted key: U YYYY YYYY YYYY YYYY YYYY YYYY YYYY YYYY
Key check value: ZZZZZZ

Online-AUTH>
```

Delete KTK

Variant ☐	Key Block ☐	
Online ☒	Offline ☒	Secure ☑
Authorization: **Not required**		

Command: **KD**

Function: To delete a selected KTK from the HSM.

Authorization: None

Inputs:
- KTK Identifier

Outputs:
- Display of relevant entry from KTK table.

Example 1: *This example demonstrates the use of the **KD** console command to delete a previously installed KTK (KTK Id 01) from the HSM.*

```
Secure> KD <Return>
Enter KTK id: 01 <Return>

KTK table entry:
ID Scheme   Algorithm   Check   Comments
01 Variant  3DES(2key)  292489  KTK for KMD in secure room

Confirm KTK deletion [Y/N]: Y <Return>
KTK deleted from main memory

Secure>
```

Appendix A – Error Codes

The information from this Appendix has been moved to Appendix A of the *payShield 9000 General Information Manual*.

Appendix B – Core HSM Commands

The information from this Appendix has been moved to Appendix B of the *payShield 9000 General Information Manual*.

Appendix C – PIN Block Formats

The information from this Appendix has been moved to Chapter 14 of the *payShield 9000 Host Programmer's Manual*.

Appendix D – Key Scheme Table

The information from this Appendix has been moved to Appendix A of the *payShield 9000 Host Programmer's Manual*.

Appendix E – Variant LMKs

The information from this Appendix has been moved to Chapter 7 of the *payShield 9000 Host Programmer's Manual*.

Appendix F – Key Block LMKs

The information from this Appendix has been moved to Chapter 8 of the *payShield 9000 Host Programmer's Manual*.

Appendix G – List of Authorizable Activities

The information from this Appendix has been moved to Appendix D of the *payShield 9000 General Information Manual*.

Appendix H – Reduced Character Sets

The information from this Appendix has been moved to Appendix B of the *payShield 9000 Host Programmer's Manual*.

Appendix I – Configure Security Settings

For a description of the security parameters referenced in the CS and QS Console commands, see the section "Configure Security" in Chapter 2 of the *payShield 9000 Security Operations Manual*.

Appendix J – Fraud Detection Functions

The information from this Appendix has been moved to Chapter 7 of the *payShield 9000 General Information Manual*.

Appendix K – Thales Key Block / TR-31 Key Usage Conversion

The information from this Appendix has been moved to Appendix C of the *payShield 9000 Host Programmer's Manual*.

Appendix L – Utilization Data

The information from this Appendix has been moved to Chapter 8 of the *payShield 9000 General Information Manual*.

Appendix M – Health Check Data

The information from this Appendix has been moved to Chapter 9 of the *payShield 9000 General Information Manual*.

Appendix N – PCI HSM Compliance

The information from this Appendix has been moved to Chapter 10 of the *payShield 9000 General Information Manual*.

Appendix O – Error Responses Excluded from Audit Log

If the option to Audit Error Responses to Host Commands has been selected using AUDITOPTIONS, those errors which may require attention by the HSM Administrators or Security Officers are logged.

The following non-00 error responses will not be included in the Audit Log:

	Not Audited if error response is:		
Cmnd	**01**	**02**	**43**
A6	X		
BC	X		
BE	X		
BK		X	
BY	X		
CG	X		
CK	X	X	
CM	X		
CO	X		
CQ	X		
CU	X		
DA	X	X	
DC	X		
DE		X	
DU	X	X	
EA	X	X	
EC	X		
EE		X	
EG	X		
EI			X
F0	X		
F2	X		
FA	X		
FU	X		
G2	X		
G4	X		
GO	X		
GQ	X		
GS	X		
GU	X		
J0			X
K2	X		
KE			X
KO			X
P0	X		
PG	X		
PY	X		
QQ	X		
QS	X		
QU	X		
QW	X		
XM	X		
XK	X		
ZU	X		

Glossary

The information from this Appendix has been moved to Appendix G of the *payShield 9000 General Information Manual*.

General Abbreviations

The information from this Appendix has been moved to Appendix H of the *payShield 9000 General Information Manual*.

Americas
Arboretum Plaza II, 9442 Capital of Texas Highway North,
Suite 100, Austin, TX 78759 USA
Tel: +1 888 343 5773 or +1 512 257 3900
Fax: +1 954 888 6211
E-mail: CPL_Sales_AMS_TG@thalesgroup.com

Asia Pacific
Unit 904-906A, Core D
Cyberport 3, 100 Cyberport Road
Pokfulam, Hong Kong | Tel: +852 3157 7111
Fax: +852 2815 8141 | E-mail: apacsales.cpl@thalesgroup.com

Europe, Middle East, Africa
350 Longwater Ave, Green Park,
Reading, Berkshire, UK RG2 6GF
Tel: +44 (0)1844 201800 | Fax: +44 (0)1844 208550
E-mail: CPL_Sales_EMEA_TG@thalesgroup.com

> cpl.thalesgroup.com <