NAUTILUS HYOSUNG

MoniPlus2 and MoniPlus2S Encryption and Remote Key Transfer

All information described in this manual is a licensed product of Nautilus Hyosung Corporation.

It is the policy of Nautilus Hyosung Corporation to improve products as new technology, components, software, and firmware become available. Therefore Nautilus Hyosung reserves the right to change specifications without notice.

Important Note:

This document is intended for the use of the individual or entity to which it is addressed only, and contains information that is privileged, confidential and that may not be made public by law or agreement. If the recipient of this document is not the intended recipient or entity, you are hereby notified that any further dissemination, distribution or copying of this information is strictly prohibited.

Table of Contents

Revision History

Version	Date	Change Summary
1.0	7/5/2011	Initial Issue

1 Introduction

This document contains encryption and remote key transfer (RKT) information for the Nautilus Hyosung (NH) MoniPlus2 and MoniPlus2S applications and NH ATMs. This document is organized as follows:

- Section 2, *Security and Encryption*, presents a brief description of ATM security and encryption.
- Section 3, *Base 94 Encoding and Decoding*, describes the encoding and decoding of encryption keys.
- Section 4, *Message Authentication*, presents a brief description of the use of MACing.
- Section 5, *Encryption and RKT Messages*, contains the details of the encryption and RKT messages exchanged between the ATM and the host.
- Section 6, *Encryption and RKT Supervisor Functions*, describes the encryption and RKT functions available through the Supervisor menus.
- Section 7, *Obtaining a Public Key Exchange Signature*, explains how to obtain a signature from Nautilus Hyosung and contains the key and signature formats.

1.1 Audience

This document is written for ATM host handler programmers who need to write the encryption-related interface to NH ATMs. A secondary audience for this document includes NH Professional Services personnel who are responsible for supporting these programmers.

However, anyone interested in learning about MoniPlus2 and MoniPlus2S encryption and RKT would benefit from reading selected sections of this document.

1.2 Additional References

The following documents complement this one:

- MoniPlus2 NDC Terminal Programmer’s Reference manual – also applies to the MoniPlus2S NDC messages
- MoniMax *XXXX* Operator Manual (hardware dependent) (MoniPlus2)

1.3 Required MoniPlus2 and MoniPlus2S Software Versions

Your ATMs must be running MoniPlus2 **version 01.04.05** or higher, or MoniPlus2S **version 2.0.0** or higher, to support the encryption and RKT features described in this document. Use the version function in Supervisor as described below to verify the version of your application software. The application version will be the first entry displayed in the versions list and will be listed on the screen as shown below, depending upon the version of MoniPlus you are running.

- MoniPlus2 – use the Supervisor>Version Info function - AP : V 01.04.05
- MoniPlus2S – use the Supervisor>Configuration>Version Information function - AP: V 2.0.0

2 Security and Encryption

Since VISA and MasterCard issued their secure key encryption mandates, Encrypting PIN Pads (EPPs) and Triple DES have been the industry standard. All Nautilus Hyosung ATMs that run MoniPlus2 and MoniPlus2S have Encrypting PIN Pads (EPP) that support Triple DES. Further, the ISO-0 and ISO-3 Pin Block formats, which are the only PCI approved PIN block formats, are fully supported by NH MoniPlus2 and MoniPlus2S ATMs and are the only formats recommended for use.

The additional mandates of having a unique key per device and frequent key changes have made it almost mandatory to use Remote Key Transfer (RKT). All MoniPlus2 and MoniPlus2S ATMs also support RKT, including the initial loading of the encryption keys into the ATM. The details of initial key loading and RKT in general are included in the next several sections of this document.

2.1 Remote Initial Key Loading

The initial encryption keys for the ATM, including the Master key and PIN key can be downloaded using the RSA encryption and signature schemes.

The remote initial key loading process consists of the following three parts:

- EPP Authentication
- Changing the ATM Encryption Key Entry Mode
- Loading the Encryption Keys

These topics are discussed in detail the remainder of this section.

2.1.1 EPP Authentication

A strictly secure method must be used to transfer an encryption key from the Host Security Module (HMS) to the terminal’s EPP module. To achieve this strict security, as defined within the industry, the following three conditions must be met:

1. The encryption key must remain secret. Only the HSM, which is the source of the key, and the EPP, which is the destination of the key, can know the key.
2. The HSM can only send a key to an authorized EPP.
3. The EPP can only accept a key from an authorized HSM.

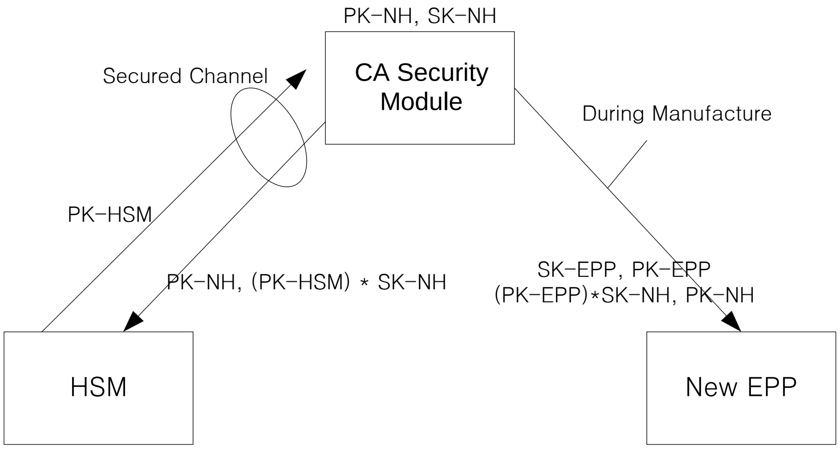
To satisfy these conditions, RSA encryption is used. This is an asymmetric encryption scheme that uses a pair of keys. The followings show a brief description of the RSA scheme.

- One of the key pair is a secret key (SK) and the other is a public key (PK).
- By using the RSA scheme, anyone can encrypt data with the public key, but only someone who owns the secret key can decrypt the key data.

- Signatures are used to prevent messages from being changed during transmission. A Secure Hashing Algorithm is used to sign the message and the result is encrypted.
- The secret key is used to generate the signature, so only someone who holds the secret key can create the signature, but anyone who has the public key can verify the signature.

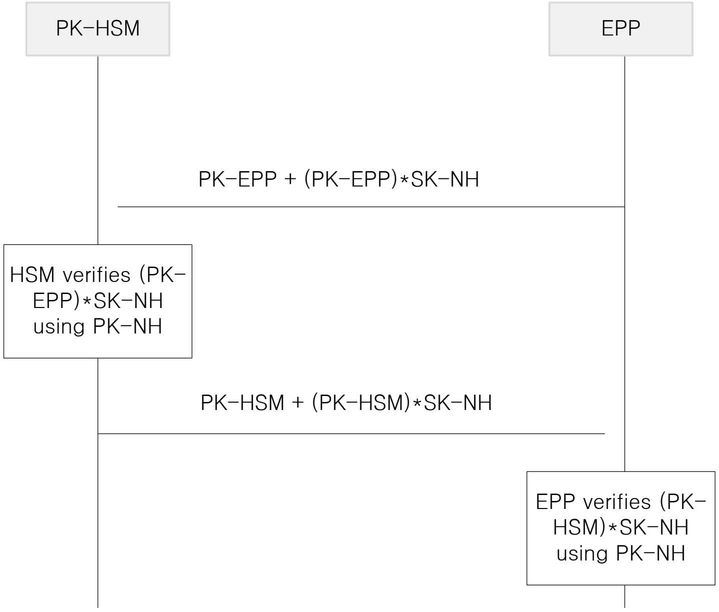
The actual encoding used with RSA encrypted keys and signatures is discussed briefly in the *Base 94 Encoding and Decoding* section on page 8 in this document.

The following figure demonstrates the EPP authentication process used by NH.



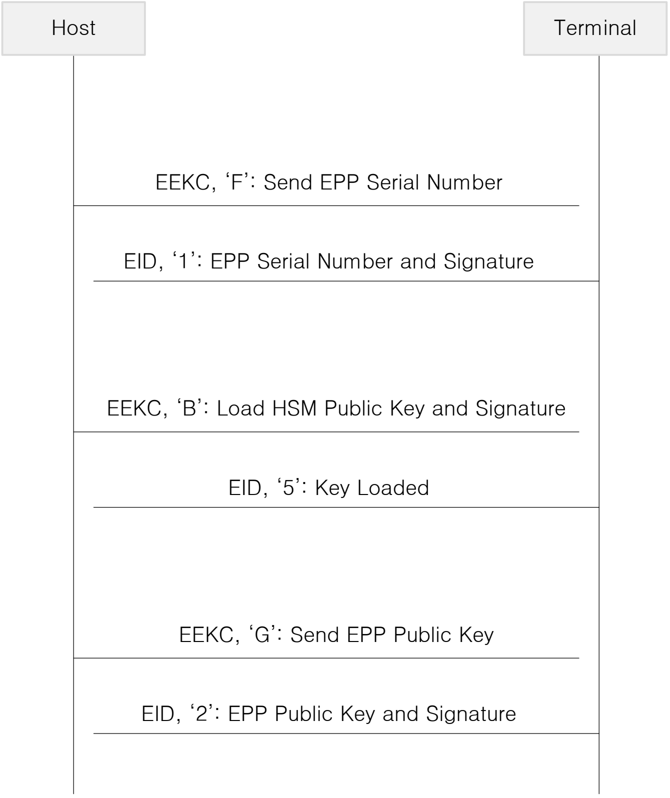
- Nautilus Hyosung’s secret key (SK-NH) is kept in a CA security module. The CA security module generates the public key belonging to the EPP (PK-EPP) and the secret key belonging to the EPP (SK-EPP) and downloads them to the new EPP during the EPP manufacturing process. During this process, the SK-NH is used to sign the PK-EPP to give (PK-EPP)*SK-NH. This signature and the PK-NH are written to the EPP.
- The HSM sends its public key (PK-HSM) to the CA security module using some secure channel, and the CA security module will sign the received PK-HSM with the SK-NH to give (PK-HSM)*SK-NH.
- This signature (PK-HSM)*SK-NH and the PK-NH are written to the HSM.
- Signing the HSM public key, which happens once between the HSM and the CA Security Module, is an off-line process, so the transfer of the key and signature can be performed manually.
- The authentication between the EPP and the HSM can be established by exchanging public keys and verifying the signatures using the PK-NH.

The exchange of public keys between the HSM and the EPP is shown in the following figure.


PK–HSM
EPP
PK–EPP + (PK–EPP)*SK–NH
HSM verifies (PK–EPP)*SK–NH using PK–NH
PK–HSM + (PK–HSM)*SK–NH
EPP verifies (PK–HSM)*SK–NH using PK–NH

- In addition to the EPP’s RSA key pair, every EPP has a unique serial number (SN-EPP) which is also written to the EPP during the manufacturing process. The EPP serial number is also signed with the SK-NH and can be transmitted to the host and HSM.
- When the newly manufactured EPP is installed (usually when the new terminal is installed), the above messages must be exchanged between the host and terminal in order to authenticate the EPP to the HSM and the HSM to the EPP.
- Once the authentication process is complete, the RSA encryption scheme can be used to download the initial keys (A, B, and V key) into the EPP.
- The Load Extended Encryption Key message is used to send RSA information to the terminal, and the terminal responds with an Encryptor Initialization Data message. Refer to the *Encryption and RKT Messages* section on page 15 for more information on the messages.

The following diagram shows the specific MoniPlus2 and MoniPlus2S message sequence and message modifiers used for EPP authentication.

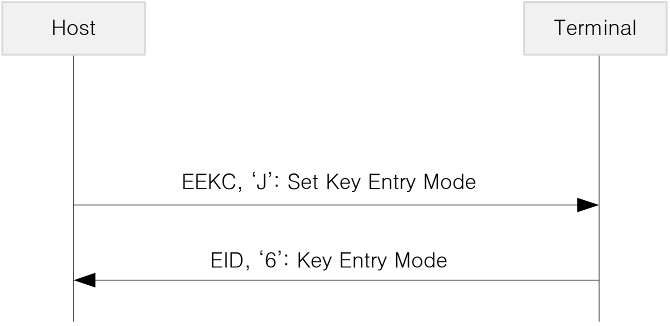

Host
Terminal
EEKC, 'F': Send EPP Serial Number
EID, '1': EPP Serial Number and Signature
EEKC, 'B': Load HSM Public Key and Signature
EID, '5': Key Loaded
EEKC, 'G': Send EPP Public Key
EID, '2': EPP Public Key and Signature

Note: EEKC=Extended Encryption Key Change and EID=Encryptor Initialization Data

2.1.2 Changing the Encryption Key Entry Mode

Once the EPP authentication process has been completed, the host must change the EPP to the key entry mode before it can download the initial keys.

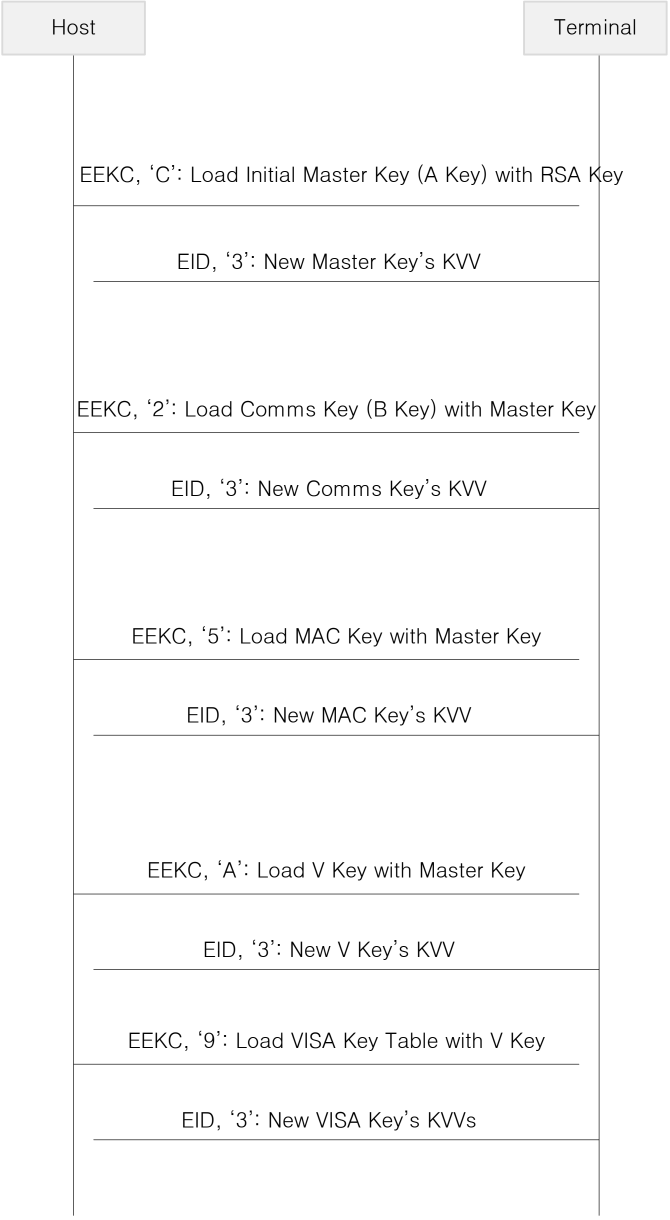
Changing the key entry mode of the EPP can be performed remotely using the message sequence shown in the following diagram. Note that changing the key entry mode deletes all the encryption keys in the EPP.



Note: EEKC=Extended Encryption Key Change and EID=Encryptor Initialization Data

2.1.3 Loading Encryption Keys

After changing the EPP into key entry mode, the host can download the encryption keys to the terminal using the following message sequence.


Host
Terminal
EEKC, 'C': Load Initial Master Key (A Key) with RSA Key
EID, '3': New Master Key's KVV
EEKC, '2': Load Comms Key (B Key) with Master Key
EID, '3': New Comms Key's KVV
EEKC, '5': Load MAC Key with Master Key
EID, '3': New MAC Key's KVV
EEKC, 'A': Load V Key with Master Key
EID, '3': New V Key's KVV
EEKC, '9': Load VISA Key Table with V Key
EID, '3': New VISA Key's KVVs

Note: EEKC=Extended Encryption Key Change, EID=Encryptor Initialization Data and KVV=Key Verification Value

3 Base 94 Encoding and Decoding

RSA encrypted keys and signatures are encoded in Base 94 for transmission to the terminal. The terminal and the host use the Base 94 encoding and decoding scheme as follows.

1. Public keys, RSA encrypted data and signatures consist of blocks of 256 bytes.
2. During the EPP authentication process, both encrypted keys and signatures must be exchanged with the HSM in the host.
3. Control characters are used for the communications protocol and message formatting, thus only graphic characters can be transmitted.
    - The Base 94 encoding scheme includes 94 graphic characters that can be included in the NDC message fields. These characters contain ASCII codes in the range of 20 to 7E hex. These characters make it possible to perform Base 94 encoding, which is more efficient than Base 16 or 10 encoding.
    - The Base 94 encoding scheme gives a ratio for encoded bytes to binary bytes of 5 to 4. Therefore, a 256-byte RSA data block can be transmitted in 320 bytes while hexadecimal encoding results in 512 bytes and decimal encoding results in 768 bytes.

3.1 Encoding

Using Base 94 encoding every four bytes of binary data become five characters of encoded data. The Base 94 encoding algorithm is described below.

1. Extract each four bytes from the source data (src_data) and form a 32-bit word (acc) such that the first byte (src_data[0]) becomes the least significant byte of acc.

```
acc = 0
for i=0 to 3
{
        acc = acc * 256 + src_data[3-i]
 }
```

2. Convert acc into five Base 94 digits by dividing by 94 and taking the modulus. Then add 32 (ASCII space) in order to shift each digit into the usable character range. Save the encode data (enc_data).

```
for i=0 to 4
{
             enc_data[i] = acc % 94 + 32
             acc = acc /94
}
```

If EBCDIC is used for message transmission, Base 94 encoding is still performed using the ASCII character set and then the result is converted to EBCDIC.

3.2 Decoding

The Base 94 decoding process is described below.

1. Extract each 5 bytes of encoded data (enc_data) and convert it from Base 94 to create a 32-bit word (acc). Each digit is shifted into the range of 0 to 93 by subtracting 32. The first digit is the least significant digit.

```
acc = 0
for i=0 to 4
{
        acc = acc * 94 + enc_data[4-i] - 32
}
```

2. Split acc up into four bytes of destination data (dst_data). The least significant byte of acc becomes dst_data[0].

```
for i=0 to 3
{
        dst_data[i] = acc % 256
        acc = acc / 256
}
```

4 Message Authentication

Message Authentication is used by some hosts to secure the messages sent between the host and terminal. When the message authentication feature is configured, either the terminal or the host will append a Message Authentication Code (MAC) to the end of the message. The MAC is generated by using the message content, which ensures that the MAC will be unique for each message. Both full and selective MACing are supported in MoniPlus2 and MoniPlus2S.

The following messages can include a MAC field. The MAC field will be included at the end of the message, preceded by a field separator.

- Transaction request messages (Terminal to Host)
- Solicited status messages including both device fault and terminal state (Terminal to Host)
- Transaction reply messages (Host to Terminal)
- Load state table commands (Host to Terminal)
- Load FIT commands (Host to Terminal)
- Load MAC field selection command (Host to Terminal)
- Load dispenser currency cassette mapping table (Host to Terminal)
- EMV configuration message (Host to Terminal)

Full message authentication is generated using the entire message, starting from the first field following the protocol-dependent message header up to the field separator preceding the MAC field.

With selective message authentication, selected fields in the message, which are defined in the MAC field selection table, are used for calculating the MAC. When selected fields are used for MACing, the selected fields are combined to produce a data string, and this data string is used to calculate the MAC. If the data string is null, the MAC will be set to ‘00000000’.

4.1 MAC Calculation Process

Triple DES MACing conforms to ANSI standard X9.19. The MAC calculation process is described in the following steps.

1. Extract the first 8 bytes of data and encrypt it using the MAC key.
2. XOR the encrypted 8 bytes with the next 8 bytes extracted from the message data. If the rest of the data is less than 8 bytes, then 0 pad to the right in order to make 8 bytes of input data.
3. Encrypt the XOR’d value using the MAC key.
4. Repeat step 2 and 3 until the end of message data is reached.
5. The first four bytes of the final encryption data make the MAC field which is eight hexadecimal digits, each of which is converted to a character in the range ‘0’-‘9’, ‘A’-‘F’.

If the MAC key is double-length, steps 1 to 4 are performed using the first half of the key. The final calculated code is then decrypted by the second half of the key and encrypted again using the first half of the key. Step 5 is used to extract the final MAC.

4.1.1 MAC field from the Terminal to the Host

The terminal generates the MAC according to the process described above and attaches the MAC to the end of message. When the message with the MAC field is sent to the host, the same MAC calculation is performed by the host and an attempt is made to match them. If the transmitted MAC and the MAC calculated by the host are identical, the host regards the transmitted message as valid and continues with processing. If the MACs are not identical, the host regards the transmitted message as invalid.

4.1.2 MAC field from the Host to the Terminal

The host generates the MAC and attaches the MAC field at the end of message. When the terminal receives the message, the terminal calculates the MAC using the message received. If the transmitted MAC and the MAC calculated by the terminal are identical, the terminal regards the transmitted message as valid and continues with processing. If the MACs are not identical, the terminal regards the transmitted message as invalid and transmits a specific command reject message to the host.

4.1.3 Time Variant Number

The Time Variant Number (TVN) is an additional security feature that is available when MACing is used. A TVN is used with transaction request messages and solicited status messages.

When this feature is configured, the terminal generates a TVN based on the terminal time and includes it in the message. When the host receives a message containing a TVN, the host will send back the same TVN in the next message, such as in the transaction reply. After the MAC is verified as correct, the terminal compares the received TVN to the TVN that the terminal created. If the TVNs match, the terminal accepts the message and processes it. If the TVNs do not match, the terminal sends a specific command reject message to the host.

4.2 Full Message Authentication

Full Message Authentication uses all fields in the message. The configuration parameters for full MAC are set through a MAC Supervisor function, where you enter 10 one-digit flag values. The following table shows the flags related to this Message Authentication configuration. Only flags 1, 2, 8, 9 and 10 are used.

The following are the NH recommended flag settings for MACing depending upon your requirements:

- 0000000000 - for no MACing
- 0100000001 - for Full MACing on transaction requests and solicited status messages
- 1100000001 for Full MACing on transaction requests, solicited status messages and transaction reply messages

Flag	*Value*	*Meaning*
Flag 1	‘0’	Do not check the time variant number in the Transaction Reply message or the MAC field in the Transaction Reply, State Table Load, FIT Load or Dispenser Currency Cassette Mapping Table messages.
	‘1’	Check the time variant number in the Transaction Reply message and the MAC field in the Transaction Reply, State Table Load, FIT Load and Dispenser Currency Cassette Mapping Table messages.
Flag 2	‘0’	Do not send time variant number (TVN) and the MAC field in the transaction request message.

Flag	***Value***	***Meaning***
	‘1’	Send time variant number (TVN) and the MAC field in the transaction request message.
	Valid combinations of flag 1 and flag 2 are 00, 01, and 11. If the combination is ‘01’, the MAC and TVN fields are still expected in the specified messages from the host, but they are ignored.	
Flag 8	‘0’	Do not check the Security Terminal Number in the transaction reply message.
	‘1’	Check the Security Terminal Number in the transaction reply message.
Flag 9	‘0’	The MAC calculation is performed over the whole message.
	‘1’	The MAC calculation is performed on the fields that are selected by the MAC field selection table.
Flag 10	‘0’	Do not include TVN and MAC fields in the solicited device status message.
	‘1’	Include TVN and MAC fields in the solicited device status message if flag 2 is set.

- Flags 3-7 are not used and must be set to zero when flags 9 and 10 are used.
- Flags 1, 2 and 10 determine whether or not the MAC is to be performed.
- Flag 8 is used to determine whether or not the received message is for this terminal.
- Flag 9 determines the MAC type.

4.3 Selective Message Authentication

Selective Message Authentication is designed for long messages, such as a transaction reply containing printer data. For these messages, it takes several seconds to calculate the MAC, so a time delay can occur in transaction processing. Selective Message Authentication solves this problem by selecting shorter fields to be MAC’d.

The MAC field selection table consists of four fields that correspond to the transaction request, transaction reply, solicited status messages, and one combined field for other types of message.

The following table shows the MAC field selection for each message.

Solicited Status Message

Offset	**Meaning**
0	‘0’: Use Full Message Authentication. Ignore the following digits. ‘1’: Use Selective Message Authentication by using the following digits. If the relevant offset bit is set to 1, it means that the field is selected for the MAC calculation.
1	Fields 2 and 3: Message class and Message sub-class
2	Field 5: Logical Unit Number
3	Reserved
4	Field 8: Time Variant Number
5	Field 9: Status Descriptor
6	Field : Status Information Sub field 1
7	Field: Status Information Sub field 2. If this field is selected, any group separators within this field are excluded for the MAC calculation.
8	Field: Status Information Sub field 3

Solicited Status Message

Offset	Meaning
9	Field: Status Information Sub field 4
10	Field: Status Information Sub field 5

Other Messages

Offset	Meaning
0	‘0’: Do not MAC FIT load messages
	‘1’: MAC FIT load messages
1	‘0’: Do not MAC state table load message
	‘1’: MAC state table load messages
2	‘0’: Do not MAC terminal state messages
	‘1’: MAC terminal state messages
3	‘0’: Do not MAC dispenser currency cassette mapping table messages
	‘1’: MAC dispenser currency cassette mapping table messages

Track 1, Track 2, Track 3

Offset	Meaning
0	‘0’: MAC full track data. Ignore the following offset data.
	‘1’: Selectively MAC the following fields. If the relevant offset bit is set to 1, it means that the field is selected for the MAC calculation.
1	Sub-field 1 (including the start sentinel)
2-n	Sub-fields 2-n

- A maximum of five fields is possible on Track 1 and 2, and ten on Track 3.
- If the sub-fields on the card are less than the maximum number of sub-fields, the excess bytes are set to zero. The last sub-field contains the end sentinel. If the sub-fields including start and end sentinels are specified to be included for the MAC, then the sentinels will be included.

EMV ICC (Smart Card) Configuration Messages

Offset	Meaning
0	‘0’: Do not MAC ICC currency data objects table messages
	‘1’: MAC ICC currency data objects table messages
1	‘0’: Do not MAC ICC transaction data objects table messages
	‘1’: MAC ICC transaction data objects table messages
2	‘0’: Do not MAC ICC language support table messages
	‘1’: MAC ICC language support table messages
3	‘0’: Do not MAC ICC terminal data objects table messages

EMV ICC (Smart Card) Configuration Messages

Offset	Meaning
	‘1’: MAC ICC terminal data objects table messages
4	‘0’: Do not MAC ICC terminal acceptable AIDs table messages ‘1’: MAC ICC terminal acceptable AIDs table messages

- The above information is only available with an EMV Integrated Circuit Card (ICC or Smart Card).

Selective MAC Default Field Values

Message Type	Default Field Values
Transaction Request	Selective MAC on: TVN, Track 2 data (sub-field l), Operation Code data, Amount Entry Field, PIN buffer, Smart Card Data ID ‘5’ and the following field for smart card data used by EMV/CAM 2 Exits.
Transaction Reply	Selective MAC on: Message Sequence/TVN, Number of Notes to Dispense, Number of Coins to Dispense, Transaction Serial Number, Function Identifier, EMV ICC data ID ‘5’ and the following EMV ICC data.
Solicited Status	Full MAC
FIT/State Tables	Full MAC
Terminal State	No MAC
Track 1	Full MAC
Track 2	Selective MAC on first sub-field
Track 3	Full MAC
Dispenser Currency Cassette Mapping Table	Full MAC
EMV ICC Configuration	Full MAC

- These defaults can be changed with a download configuration message.
- When selective fields are used, the selected fields are used in sequence to build the data string; field separators and group separators are excluded. Empty fields are omitted.
- Once the single data string is built from the selected fields, the MAC calculation process applies in the same way as for full MACing.

5 Encryption and RKT Messages

This section explains the messages that are exchanged between the host and the terminal for loading, deleting and querying key information and Encrypting PIN Pad (EPP) data at the terminal. These messages follow the well-know NDC message format used by many ATM vendors.

5.1 Configuration Data Load Commands and Terminal Responses

The Configuration Data Load commands are used to exchange encryption key-related information. All of these messages start with message class number '3.' The following table summarizes the encryption-related message numbers.

Command	***Message Class***	***Message Sub-Class***	***Identifier***
Load Encryption Key	3	3	1, 2, 5
Load Extended Encryption Key	3	4	1, 2, 5 and B - R selective
Load MAC field Selection	3	1	B

Refer to the **MoniPlus2 NDC Terminal Programmer's Reference** manual for details on all of the configuration data load-related messages. The next several sections explain the details of the most common commands related to encryption and RKT.

5.2 Load Encryption Key Message

The host can replace the encryption keys, including the Master Key (A key) and Communication Key (B key) that were initially installed in the terminal using the Load Encryption Key message. This message can be used for single-length or double-length keys. The new keys included in the message are encrypted using the same algorithm that is used in the terminal.

In addition to including the new, encrypted key data, the Load Encryption Key message will also specify the following:

- The current encryption keys to replace.
- The current encryption key that the terminal must use to decrypt the new key.

When a power failure occurs at the terminal, the keys will remain unchanged.

The Load Encryption Key message does not reset the configuration ID to 0000.

The Load Encryption Key message format is included in the following table.

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘3’**
10	Message Modifier	1	M	**Var**
11	Field Separator	1	M	**FS**
12	New Key Data	24 or 288	M	**Var**
13	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal

8	Field Separator		Mandatory FS
9	Message Sub-Class	‘3’	Set to ‘3’ for encryption key information
10	Modifier		This single-character modifier specifies which new encrypted key is being downloaded and which key the terminal must use to decrypt it. The modifiers that are not supported with the Secure Key mandate are noted in the table and are not addressed in this document.
		‘1’	A new master key is being loaded; decipher it with the current master key.
		‘2’	A new communications key is being loaded; decipher it with the current master key.
		‘3’	A new communications key is being loaded; decipher it with the current communications key – **this is no longer supported with Secure Key.**
		‘4’	Use the locally-entered communications key as the current communications key – **this is no longer supported with Secure Key.**
		‘5’	A new MAC key is being loaded; decipher it with the current master key.
		‘6’	A new MAC key is being loaded; decipher it with the current communications key – **this is no longer supported with Secure Key.**
		‘7’	Use the locally-entered communications key as the current MAC key – **this is no longer supported with Secure Key.**
11	Field Separator		Optional FS, this FS is present if the modifier (field 10) is set to 1, 2 or 5.
12	New Key Data		This field contains the encrypted key data. The data consists of entries of 3-character decimal numbers in the range of 000 to 255. Each 3-characters entry defines one byte of key data is represented by two hexadecimal digits. The key lengths are: • Single DES – 24 bytes • Tripe DES – 48 bytes
13	Trailer		Network protocol dependent

The terminal will perform the following actions after receiving this command:

- Load the new key data into the encryptor indicating which key must be used to decrypt the new key data being loaded.
- Respond to the host with a Ready status message after the terminal successfully downloads a new key to the encryptor; otherwise, the terminal will return a command reject.

The Ready message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘2’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Field Separator	1	O	**FS**
9	Status Descriptor	1	M	**‘9’**
10	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘2’	Set to ‘2’ for a status message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Field Separator		Optional FS, this field is sent only if MACing is used.
9	Status Descriptor	‘9’	Ready - the host instruction was completed successfully.
10	Trailer		Network protocol dependent

5.3 Load Extended Encryption Key Message

This message is an extension of the Load Encryption Key message and supports double-length keys with Triple DES and the public key structure. If the terminal uses single-length keys and it receives double-length keys, or vice versa, the terminal will respond to the host with a command reject status message.

The host can replace the Master Key (‘A’ key) and Communication Key (‘B’ key) that were initially installed in the terminal with new ones. The host can change the communications key when the terminal is in the ‘out-of-service’ mode or ‘in-service’ mode, but not during a transaction. The new keys in the message are encrypted with the same algorithm used in the terminal.

In addition to including the new, encrypted key data, the Load Extended Encryption Key message will also specify the following:

- The current encryption keys to replace.
- The current encryption key that the terminal must use to decrypt the new key.

When a power failure occurs at the terminal, the key will remain unchanged.

The Load Extended Encryption Key message does not reset the configuration ID to 0000.

The Load Extended Encryption Key command can be divided into several sub-commands according to the modifier and the terminal response can be different for each sub-command.

The following table briefly lists the types of key load commands, classified by sub-modifier, that are supported with Triple DES/Secure Key. Each of the messages is described in the next several sections by sub-modifier. Note that any sub-modifiers that are not supported with Secure Key have been omitted.

Message Class	*Message Class*	*Sub-Modifier*	*Key Load Command*
3	4	1	New Master Key with the Current Master Key
3	4	2	New Communications Key with the Current Master Key
3	4	5	New MAC Key with the Current Master Key
3	4	B	Load HSM Public Key and Signature
3	4	C	Load Initial Master Key with the RSA Key
3	4	D	Load Initial Communications Key with the RSA Key
3	4	F	Send EPP Serial Number
3	4	G	Send EPP Public Key
3	4	H	Send All KVVs
3	4	J	Set Key Entry Mode
3	4	N	Send ATM Random Number
3	4	Q	Send Encryptor Capabilities and State
3	4	R	Load Sub Public Key and Signature

5.4 New Master Key with Current Master Key (3 4…1)

This message is specified with modifier ‘1.’

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘1’**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	0 – 640	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘1’	Modifier ‘1’ means that a new master key is being downloaded; decipher it with current master key. The old key will be overwritten.
11	Field Separator		Mandatory FS
12	Key Data Size		3-digit hexadecimal number that defines the size of the following ‘new key data’ field. For a single-length DES key, the size is ‘018’ hex (24 decimal).

		For a double-length DES key, the size is '030' hex (48 decimal).
13	New Key Data	This field contains the encrypted key data whose length is defined in the previous field. It contains only one key. The data consists of entries of 3-character decimal numbers in the range of 000 to 255. Each entry (3-characters) represents one byte of data that can be represented as a 2-digit hexadecimal number. For example, the data will be shown as below (spaces have been inserted for clarity): 255 254 253 252 251 250 249 248… (equivalent to FF FE FD FC FB FA F9 FA…)
14	Trailer	Network protocol dependent

When the terminal finishes downloading the new key data to the EPP, the terminal responds to the host with an Encryptor Initialization message containing the terminal's key verification value. This assumes a reject condition does not occur.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	New Key KVV	6 or 72	M	**Var**
11	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS

8	Information Identifier	‘3’	Set to ‘3’ to indicate the KVV for the key just loaded
9	Field Separator		Mandatory FS
10	New Key KVV		This field contains the 6-digit (or 72-digit) Key Verification Value of the key that was just loaded.
11	Trailer		Network protocol dependent

5.5 New Communications Key with Current Master Key (3 4…2)

This message is specified with modifier ‘2.’

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘2’**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	0 – 640	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘2’	Modifier ‘2’ means that a new communications key is being downloaded, decipher it with the current master key. The old key will be overwritten.
11	Field Separator		Mandatory FS
12	Key Data Size		3-digit hexadecimal number that defines the size of the following ‘new key data’ field

		• For a single-length DES key, the size is ‘018’ hex (24 decimal). • For a double-length DES key, the size is ‘030’ hex (48 decimal).
13	New Key Data	This field contains the encrypted key data whose length is defined in the previous field. It contains only one key. The data consists of entries of 3-character decimal numbers in the range of 000 to 255. Each entry (3-characters) represents one byte of data that can be represented as a 2-digit hexadecimal number. For example, this field will be shown as follows (spaces have been inserted for clarity): 255 254 253 252 251 250 249 248 … (equivalent to FF FE FD FC FB FA F9 FA…)
14	Trailer	Network protocol dependent

When the terminal completes downloading the new key data, the terminal responds to the host with an Encryptor Initialization message containing the terminal’s key verification value. This assumes a reject condition does not occur.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘3’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	New Key KVV	6 or 72	M	**Var**
11	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘3’	Set to ‘3’ for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO

6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	‘3’	Set to ‘3’ for sending the KVV for the key just loaded
9	Field Separator		Mandatory FS
10	New Key KVV		This field contains the 6-digit (or 72-digit) Key Verification Value of the key that was just loaded
11	Trailer		Network protocol dependent

5.7 New MAC Key with Current Master Key (3 4…5)

This message is specified with modifier ‘5’and indicates that a new MAC key is being downloaded. The MAC key will be deciphered with the current master key.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘5’**
11	Field Separator	1	O	**FS**
12	Key Data Size	3	O	**Var**
13	New Key Data	0 – 640	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘5’	Modifier ‘5’ means that a new MAC key is being downloaded, decipher it with the current master key. The old key will be overwritten.
11	Field Separator		Mandatory FS
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following ‘new key data’ field.

		• For a single-length DES key, the size is '018' hex (24 decimal). • For a double-length DES key, the size is '030' hex (48 decimal).
13	New Key Data	This field contains the encrypted key data whose length is defined in field 12. It contains only one key. The data consists of entries of 3-character decimal numbers in the range of 000 to 255. Each entry (3-characters) represents one byte of data that can be represented as a 2-digit hexadecimal number. For example, the data in this field will be as follows (spaces have been inserted for clarity): 255 254 253 252 251 250 249 248 … (equivalent to FF FE FD FC FB FA F9 FA…)
14	Trailer	Network protocol dependent

When the terminal completes downloading the new key data to the EPP, the terminal responds to the host with an Encryptor Initialization message containing the terminal Key Verification Value. This assumes a reject condition does not occur.

The Encryptor Initialization message format is shown in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	New Key KVV	6 or 72	M	**Var**
11	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO

6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	‘3’	Set to ‘3’ for sending the KVV for the key just loaded
9	Field Separator		Mandatory FS
10	New Key KVV		This field contains the 6-digit (or 72-digit) Key Verification Value of the key that was just loaded
11	Trailer		Network protocol dependent

5.9 Load Host Security Module (HSM) Public Key (3 4…B)

This message is specified with the modifier 'B' and indicates that the HSM public key and signature are being downloaded.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**'3'**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**'4'**
10	Message Modifier	1	M	**'B'**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	0 – 640	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'3'	Set to '3' for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	'4'	Set to '4' for extended encryption key information
10	Modifier	'B'	Modifier 'B' means that the HSM public key and signature are being downloaded
11	Field Separator		Mandatory FS
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following 'new key data' field. For the HSM public key and signature, the total data size is 640

		bytes.
13	New Key Data	This field contains the HSM public key (PK-HSM) and signature block; the data is encoded with base 94.
14	Trailer	Network protocol dependent

When the terminal completes downloading and verifying the HSM public key and signature, the terminal responds to the host with an Encryptor Initialization message. This assumes a reject condition does not occur.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘3’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**‘5’**
9	Field Separator	1	M	**FS**
10	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘3’	Set to ‘3’ for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	‘5’	Set to ‘5’ for ‘key loaded’
9	Field Separator		Mandatory FS
10	Trailer		Network protocol dependent

5.10 Load Initial Master Key with RSA Key (3 4…C)

This message is specified with modifier ‘C’ and indicates that the initial master key (A Key) is being downloaded. The terminal will use the EPP public key (PK-EPP) and signature block to decipher this new initial master key.

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘C’**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	0 – 640	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘C’	Modifier ‘C’ means that a new initial master key is being downloaded; use the EPP public key (PK-EPP) and signature block to decipher it.
11	Field Separator		Mandatory FS
12	Key Data Size		A 3-digit hexadecimal number which defines the size of the

		following ‘new key data’ field. For the initial master key, the total data size is 640 bytes.
13	New Key Data	This field contains the initial master key and is encrypted with the EPP public key (PK-EPP) and signature block created using the SK-HSM. The key data is encoded with base 94 as has a length of 640-bytes. The signature is generated from a random number concatenated with the encrypted double length DES key when the enhanced signature remote key protocol is used. The random number is returned in response to an EEKC request with modifier ‘N,’ and is not included in the message from the host to the EPP.
14	Trailer	Network protocol dependent

When the terminal completes downloading and verifying the initial master key (A-Key), the terminal responds to the host with an Encryptor Initialization message. This assumes a reject condition does not occur.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘3’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**‘5’**
9	Field Separator	1	M	**FS**
10	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘3’	Set to ‘3’ for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS

8	Information Identifier	‘5’	Set to ‘5’ meaning ‘key loaded’
9	Field Separator		Mandatory FS
10	Trailer		Network protocol dependent

5.11 Load New Initial Communications Key with RSA Key (3 4…D)

This message is specified with modifier ‘D’ and indicates that a new initial communications key (B Key) is being downloaded. The terminal will use the EPP public key (PK-EPP) and signature block to decipher the new key.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘D’**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	0 – 640	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘D’	Modifier ‘D’ means that a new initial communications key is being downloaded; use the EPP public key (PK-EPP) and signature block to decipher it.
11	Field Separator		Mandatory FS
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following

		‘new key data’ field. For the new initial communications key, the total data size is 640 bytes.
13	New Key Data	This field contains the new initial communications key, encrypted with the EPP public key (PK-EPP) and signature block created using the SK-HSM. The key data is encoded with base 94 and is 640-bytes in length. The signature is generated from a random number concatenated with the encrypted double length DES key when the enhanced signature remote key protocol is used. The random number is returned in response to an EEKC request with modifier ‘N,’ and is not included in the message from the host to the EPP.
14	Trailer	Network protocol dependent

When the terminal completes downloading and verifying the new initial communications key (B-Key), the terminal responds to the host with an Encryptor Initialization message. This assumes a reject condition does not occur.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘3’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**‘5’**
9	Field Separator	1	M	**FS**
10	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘3’	Set to ‘3’ for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS

7	Field Separator		Mandatory FS
8	Information Identifier	‘5’	Set to ‘5’ for ‘key loaded’
9	Field Separator		Mandatory FS
10	Trailer		Network protocol dependent

5.13 Send EPP Serial Number (3 4…F)

This message is specified with modifier ‘F’ and indicates that the host is requesting that the terminal send its EPP serial number to the host.

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘F’**
11	Field Separator	1	O	**FS**
12	Key Data Size	3	O	**Var**
13	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘F’	Send EPP serial number. When the terminal receives this message with modifier ‘F,’ the terminal will respond to the host with the EPP’s signed serial number in the defined message format. The response message format is described below. If the terminal is not in the key entry mode, this command will be rejected. To change the key mode to the EPP key entry mode, the host should send modifier ‘J’ (Set key entry mode) or the ‘key entry mode’ could be entered through supervisor.

11	Field Separator	Optional FS – this FS will not typically be present in this message.
12	Key Data Size	A 3-digit hexadecimal number that defines the size of the following 'new key data' field. Since modifier 'F' does not require 'new key data,' this field will typically not be present.
13	Trailer	Network protocol dependent

The terminal responds to the host with an Encryptor Initialization message containing its signed EPP serial number.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	EPP Serial Number	8	M	**Var**
11	EPP Serial Number Signature	320	M	**Var**
12	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	'1'	Set to '1 for sending the EPP serial number
9	Field Separator		Mandatory FS
10	EPP Serial Number		8-digit EPP serial number
11	EPP Serial Number Signature		This field contains the 320-digit EPP serial number signature that

		was created using the RSA NH key. The data is Base 94 encoded.
12	Trailer	Network protocol dependent

5.14 Send EPP Public Key (3 4…G)

This message is specified with modifier ‘G’ and indicates that the host is requesting that the terminal send its EPP public key to the host.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘G’**
11	Field Separator	1	O	**FS**
12	Key Data Size	3	O	**Var**
13	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘G’	Send EPP public key. When the terminal receives this message, the terminal will respond to the host with its signed EPP’s public key in the defined message format. The response message format is described below.
11	Field Separator		Optional FS – this FS will not typically be included in this message.
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following ‘new key data’ field. Since the modifier ‘G’ does not require a ‘new

		key data' field, this field will not typically be present.
13	Trailer	Network protocol dependent

The terminal responds to the host with an Encryptor Initialization message containing its signed EPP public key. The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	EPP Public Key	320	M	**Var**
11	EPP Public Key Signature	320	M	**Var**
12	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	'2'	Set to '2' for sending the EPP public key
9	Field Separator		Mandatory FS
10	EPP Public Key		This field contains the 320-digit EPP public key that was encoded with base 94. The exponent of the EPP public key is always 65537, so it is not sent to the host.
11	EPP Public Key Signature		This field contains the 320-digit EPP public key signature that was created using the RSA NH key. The data is Base 94 encoded.
12	Trailer		Network protocol dependent

5.15 Send All Key Verification Values (3 4…H)

This message is specified with modifier ‘H’ and indicates that the host is requesting that the terminal send all if its key verification values to the host.

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘H’**
11	Field Separator	1	O	**FS**
12	Key Data Size	3	O	**Var**
13	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘H’	Send all key verification values (KVVs). When the terminal receives this message, the terminal will respond to the host with all KVVs in the defined message format. The response message format is described below.
11	Field Separator		Optional FS - this FS will typically not be present in this message.
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following ‘new key data’ field. Since the modifier ‘H’ does not require a ‘new

		key data,' field is typically not present.
13	Trailer	Network protocol dependent

The terminal responds to the host with an Encryptor Initialization message containing all of its KVVs. The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	Master Key KVV	6	M	**Var**
11	Communications Key KVV	6	M	**Var**
12	MAC Key KVV	6	M	**Var**
13	B Key KVV	6	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	'4'	Set to '4' for sending all KVVs
9	Field Separator		Mandatory FS
10	Master Key KVV		This field contains the 6-digit Key Verification Value of the master key. If the master key has not been loaded, six zeros will be sent.
11	Communications Key KVV		This field contains the 6-digit Key Verification Value of the communications key. If the communications key has not been

		loaded, all six zeros will be sent.
12	MAC Key KVV	This field contains the 6-digit Key Verification Value of the MAC key. If the MAC key has not been loaded, six zeros will be sent.
13	B Key KVV	This field contains the 6-digit Key Verification Value of the B key. If the B key has not been loaded, six zeros will be sent.
14	Trailer	Network protocol dependent

5.16 Set Key Entry Mode (3 4…J)

This message is specified with the modifier 'J' and indicates that the host is requesting that the terminal set its EPP mode to 'key entry' mode.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**'3'**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**'4'**
10	Message Modifier	1	M	**'J'**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	1	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'3'	Set to '3' for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	'4'	Set to '4' for extended encryption key information
10	Modifier	'J'	Set to 'J' for 'set key entry' mode. This message will be only accepted after the host has determined that the terminal has the ability to download RSA keys. This verification can be done by exchanging public keys and reading the EPP serial number. If this message is sent to the terminal after a power up, but before the exchange of public keys, the terminal will reject this

			command with the specific reject reason code of 'C18.'
11	Field Separator		Mandatory FS
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following 'new key data' field. This field will always be '001' because the following field contains a single character of key mode information.
13	New Key Data		This field contains the single-character value of the key entry mode into which the EPP should enter. The possible values are '1' to '4' as follows:
		'1'	Set key mode to single length without XOR
		'2'	Set key mode to single length with XOR
		'3'	Set key mode to double length with XOR
		'4'	Set key mode to double length restricted
14	Trailer		Network protocol dependent

The terminal responds to the host with an Encryptor Initialization message containing information for the current key entry mode.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	Key Entry Mode	1	M	**Var**
11	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO

6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	‘6’	Set to ‘6’ to indicate that the key entry mode was changed
9	Field Separator		Mandatory FS
10	Key Entry Mode		This field contains the terminal’s current EPP key entry mode. Possible values are ‘1’ to ‘4’ as follows:
		‘1’	Single length without XOR
		‘2’	Single length with XOR
		‘3’	Double length with XOR
		‘4’	Double length, restricted
11	Trailer		Network protocol dependent

5.18 Send Random ATM Number (3 4…N)

This message is specified with modifier ‘N’ and indicates the host is requesting that the terminal send its ATM random number to the host.

Note: a registry key change is required to enable this feature on the ATM. The registry key to change and the value to use are:

HKey_Local_Machine\Software\ATM\DevInfo\PINPAD
RKT_RandomNumber = 3

You must reboot the ATM after you make this change.

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘N’**
11	Field Separator	1	O	**FS**
12	Key Data Size	3	O	**Var**
13	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘N’	Set to ‘N’ for the terminal to send its ATM random number. If the terminal receives this message and the proper registry key is not set

		as described above, the ATM will return a specific command reject.
11	Field Separator	Optional FS – this field will not be present with this message
12	Key Data Size	A 3-digit hexadecimal number which defines the size of the following 'new key data' field. Since the modifier 'N' does not require key data, this field will not be present.
13	Trailer	Network protocol dependent

The terminal responds to the host with an Encryptor Initialization message containing its random ATM number.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'2'**
3	Message Sub-Class	1	M	**'3'**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	ATM Random Number	Var	M	**Var**
11	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'2'	Set to '2' for a solicited response to the host
3	Message Sub-Class	'3'	Set to '3' for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	'9'	Set to '9' for sending the ATM random number
9	Field Separator		Mandatory FS
10	ATM Random Number		This field contains the ATM random number in ASCII hexadecimal format.
11	Trailer		Network protocol dependent

5.19 Send Encryptor Capabilities and State (3 4…Q)

This message is specified with modifier 'Q' and requests that the terminal send the capabilities and state of its encryptor to the host.

Host to Terminal:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**'3'**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**'4'**
10	Message Modifier	1	M	**'Q'**
11	Field Separator	1	O	**FS**
12	Key Data Size	3	O	**Var**
13	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	'3'	Set to '3' for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	'4'	Set to '4' for extended encryption key information
10	Modifier	'Q'	Set to 'Q' to request that the terminal send the capabilities and state of its encryptor
11	Field Separator		Optional FS – this FS will not be present in this message
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following 'new key data' field. Since modifier 'N' does not require 'new key data,' this field will not be present.
13	Trailer		Network protocol dependent

The terminal responds to the host with an Encryptor Initialization message containing the capabilities and state of its encryptor.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘3’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**Var**
9	Field Separator	1	M	**FS**
10	Remote Key Protocol	2	M	**Var**
11	Certificate State	2	M	**Var**
12	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘3’	Set to ‘3’ for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	‘B’	Set ‘B’ to send the capabilities and state of the encryptor
9	Field Separator		Mandatory FS
10	Remote Key Protocol		This field contains the capabilities of the encryptor as a 2-digit decimal value as follows:
		‘00’	None
		‘01’	Basic signature
		‘02’	Basic certificate
		‘03’	Basic signature and certificate
		‘04’	Enhanced signature

		‘05’	Enhanced signature and certificate
11	Certificate State		This field contains the certificate state as a 2-digit decimal value as follows:
		‘00’	Not ready or not supported
		‘01’	Certificate primary
		‘02’	Certificate secondary
12	Trailer		Network protocol dependent

5.20 Load Sub Public Key and Signature (3 4…R)

This message is specified with modifier ‘R’ and indicates that the sub public key and signature are being downloaded.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘4’**
10	Message Modifier	1	M	**‘R’**
11	Field Separator	1	M	**FS**
12	Key Data Size	3	M	**Var**
13	New Key Data	Var	M	**Var**
14	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by the terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by the terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘4’	Set to ‘4’ for extended encryption key information
10	Modifier	‘R’	Set to ‘R’ for downloading a sub public key. Modifier ‘R’ is only supported in the enhanced signature key mode. If the terminal is not in the enhanced signature mode, a specific command reject will be sent to the host.
11	Field Separator		Mandatory FS
12	Key Data Size		A 3-digit hexadecimal number that defines the size of the following

		‘new key data’ field.
13	New Key Data	This field contains the actual key data of the sub public key.
14	Trailer	Network protocol dependent

When the terminal completes downloading and verifying the sub public key, it responds to the host with an Encryptor Initialization message. This assumes a reject condition does not occur.

The Encryptor Initialization message format is included in the following table.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘3’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Information Identifier	1	M	**‘5’**
9	Field Separator	1	M	**FS**
10	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘3’	Set to ‘3’ for an encryptor initialization message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Information Identifier	‘5’	Set to ‘5’ meaning ‘key loaded’
9	Field Separator		Mandatory FS
10	Trailer		Network protocol dependent

5.22 Load MAC Field Selection (3 1…B)

This command selects the messages and fields in the message that are to be applied for the calculation of the Message Authentication Code (MAC). The fields will be included in the MAC if the relevant offset byte is set to 1.

Host to Terminal:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘3’**
3	Response Flag	1	O	**Var**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3	O	**Var**
6	Field Separator	1	M	**FS**
7	Message Sequence Number	3	O	**Var**
8	Field Separator	1	M	**FS**
9	Message Sub-Class	1	M	**‘1’**
10	Message Identifier	1	M	**‘B’**
11	Field Separator	1	M	**FS**
12	Transaction Request Field	Var (47)	M	**Var**
13	Field Separator	1	O	**FS**
14	Transaction Reply Field	Var (36)	O	**Var**
15	Field Separator	1	O	**FS**
16	Solicited Status Field	11	O	**Var**
17	Field Separator	1	O	**FS**
18	Other Messages Field	4	O	**Var**
19	Field Separator	1	O	**FS**
20	Track 1 Field	6	O	**Var**
21	Field Separator	1	O	**FS**
22	Track 2 Field	6	O	**Var**
23	Field Separator	1	O	**FS**
24	Track 3 Field	11	O	**Var**
25	Field Separator	1	O	**FS**
26	EMV Smart Card Configuration	5	O	**Var**
27	Field Separator	1	O	**FS**

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
28	Message Authentication Code	8	O	**Var**
29	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘3’	Set to ‘3’ for a data load command
3	Response Flag		Not used, ignored by the terminal
4	Field Separator		Mandatory FS
5	Logical Unit Number		Reserved for future use, ignored by terminal
6	Field Separator		Mandatory FS
7	Message Sequence Number		Reserved for future use, ignored by terminal
8	Field Separator		Mandatory FS
9	Message Sub-Class	‘1’	Set to ‘1’ for the customization data load command
10	Message Identifier	‘B’	Set to ‘C’ for the message authentication field selection
11	Field Separator		Mandatory FS
12	Transaction Request Field		This field defines the fields selected for MAC generation in the transaction request message. This field can contain up to 47 characters (maximum) where ‘1’= selected for the MAC and ‘0’= not selected are repeated. The fields in the transaction request message are selected for the MAC generation if the relevant offset byte is set to 1. The offset and its value for the fields in the transaction request message are described below.

Byte Offset	Fields to be selected for MACing
0	0 – Full message authentication. Ignore the following digits in the field. 1 – Selective message authentication, select the fields to be included in the MAC if the relevant byte is set to ‘1’
1	Field 2 and 3 (Message class and message sub-class)
2	Field 5 (LUNO)
3	Reserved
4	Field 8 (Time Variant Number)
5	Field 10 (Top of Receipt Flag)

6	Field 11 (Message Co-Ordination Number)
7	Field 13 (Track 2 Data)
8	Field 15 (Track 3 Data)
9	Field 17 (Operation Code Data)
10	Field 19 (Amount Entry Field)
11	Field 21 (PIN Buffer)
12	Field 23 (General Purpose Buffer B)
13	Field 25 (General Purpose Buffer C)
14	Field 27 (Track 1 Identifier)
15	Field 28 (Track 1 Data)
16	Field 30 and 31 (Transaction Status Data Identifier and Last Transaction Status Data), optionally buffer ‘f’. Buffer ‘f’ will be included if more than 4 hopper types are supported by the coin dispenser.
17	Reserved
18	Reserved
19	Reserved
20	Reserved
21	Reserved
22	Reserved
23	Reserved
24	Reserved
25	Reserved
26	Reserved
27	Reserved
28	Reserved
29	Reserved
30	Reserved
31	Reserved
32	Reserved
33	Reserved
34	Reserved
35	Reserved
36	Reserved

37	Reserved
38	Field group 32 (CSP Data ID ‘U’ group)
39	Field group 34 (Confirmation CSP Data ID ‘V’ group)
40	Field 36-1 and 36-2 (VC Data ID ‘W’ group)
41	Field 36-4 and 36-5 (VC Data ID ‘X’ group)
42	Field 36-7 and 36-8 (VC Data ID ‘Y’ group)
43	Field 36-10 and 36-11 (VC Data ID ‘Z’ group)
44	Field 36-13 and 36-14 (VC Data ID ‘[’ group)
45	Field 36-16 and 36-17 (VC Data ID ‘\’ group)
46	Field group 37 (Smart Card Data ID ‘5’ group)
51	Field group 43 (Field ID ‘e’ barcode reader group)

13	Field Separator	Optional Field Separator
14	Transaction Reply Field	This field defines the fields to be selected for the MAC in the transaction reply message. This field can contain up to 47 characters (maximum) where ‘1’= selected for the MAC and ‘0’= not selected are repeated. The fields are selected if the relevant offset byte is set to 1. The offsets for the fields in the transaction reply message are described in the below table.

Byte Offset	Fields to be Selected for MACing
0	0 – Full message authentication. Ignore the following digits in the field. 1 – Selective message authentication, select the fields to be included in the MAC if the relevant byte is set to ‘1’
1	Field 2 and 3 (Message class and message sub-class)
2	Field 5 (LUNO)
3	Field 7 (Message Sequence/Time Variant Number)
4	Field 9 (Next State Number)
5	Field group 11 and field group 13 (Number of Type 1, 2, 3, 4, 5, 6, and 7 Notes to Dispense, Number of Hopper Type 1, 2, 3, …, and n Coins to Dispense)
6	Field 15 (Transaction Serial Number)

		7	Field 16 (Function Identifier)
		8	Field 17 (Screen Number)
		9	Field 18 (Screen Display Update)
		10	Field 20 (Message Co-Ordination Number)
		11	Field 21 (Card Return/Retain Flag)
		12	Field group 22 (Printer Flag and Printer Data)
		13	Field group 22 (Printer Flag and Printer Data) Valid, if there is second type of printer data.
		14	Field group 22 (Printer Flag and Printer Data) Valid, if there is third type of printer data.
		15	Field 24-1 (Buffer Identifier ‘4’)
		16	Field 24-1 (Track 3 Data)
		17	Reserved
		18	Reserved
		19	Reserved
		20	Reserved
		21	Reserved
		22	Reserved
		23	Reserved
		24	Reserved
		25	Reserved
		26	Reserved
		27	Field group 26 (Track 1 Data Group)
		28	Field group 28 (Track 2 Data Group)
		29	Field group 30 (VC Data ‘M’ Group)
		30	Field group 32 (VC Data ‘N’ Group)
		31	Field group 34 (VC Data ‘O’ Group)
		32	Field group 36 (VC Data ‘P’ Group)
		33	Field group 38 (VC Data ‘Q’ Group)
		34	Field group 40 (VC Data ‘R’ Group)
		35	Field group 46 (Check Destination Data Buffer ID ‘a’ group)
		36	Field group 44 (EMV ICC Data ID ‘5’ group)
15	Field Separator	Optional FS	
16	Solicited Status Field	This field contains the MAC selection data for the solicited	

		status message.
17	Field Separator	Optional FS
18	Other Messages Field	This field contains the MAC selection data for the FIT load, state tables load, terminal state status, and dispenser currency cassette mapping table messages.
19	Field Separator	Optional FS
20	Track 1 Field	This field contains the MAC selection data for card track 1.
21	Field Separator	Optional FS
22	Track 2 Field	This field contains the MAC selection data for card track 2.
23	Field Separator	Optional FS
24	Track 3 Field	This field contains the MAC selection data for card track 3.
25	Field Separator	Optional FS.
26	EMV Smart Card Configuration	This field contains the MAC selection data for the EMV smart card configuration message.
27	Field Separator	Optional FS – this FS is present only when MAC is in use.
28	Message Authentication Code	This field contains the value for authentication of this message, and this field and previous FS are present only if MACing is selected and the flags are set correctly.
29	Trailer	Network protocol dependent

The terminal responds to the host with a Ready Status message after processing the MAC selection message. The Ready Status message format follows.

Terminal to Host:

No.	***Field Name***	***Number of Characters***	***Mandatory or Optional***	***Value***
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘2’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Field Separator	1	O	**FS**
9	Status Descriptor	1	M	**‘9’**
10	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited response to the host
3	Message Sub-Class	‘2’	Set to ‘2’ for status message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Field Separator		Optional FS – this field is sent only if MACing is used
9	Status Descriptor	‘9’	The host instruction was completed successfully
10	Trailer		Network protocol dependent

5.23 Command Rejects

When the terminal receives a host command or a transaction reply command, it will reject the command if a condition has occurred that prevents the terminal from performing the command.

There are two types of command rejects: command reject and specific command reject.

1. Command Reject: the terminal can send a command reject by setting the status descriptor to 'A' without a status information field.
2. Specific Command Reject: if the configuration parameters downloaded with the Load Enhanced Configuration Parameters command designate to send a specific command reject, the terminal can send a specific command reject by setting the status descriptor to 'C' with a status information field including a reject reason.

5.23.1 Typical Reject Reasons

The following are typical scenarios that will cause a command reject to be sent to the host.

- The terminal receives a message with an illegal message class – legal message classes are 1, 2, 3, 4, 6, 7, or 8.
- The terminal receives a message with an illegal sub-class.
- The terminal receives a host command message with an illegal command code.
- The terminal receives a host command message with an illegal command modifier.
- The terminal receives a message with a field separator in an illegal position.
- The terminal receives a message that has insufficient fields.
- The terminal has insufficient memory to hold the FIT entry.
- The dispense amount requested in a transaction reply message is larger than the number of notes/or coins reported in the Hardware Configuration.
- The message co-ordination number in a transaction reply message does not match the number in the transaction request message and is not '0.'
- The terminal receives a transaction reply message that contains an illegal Function ID.
- An Exchange Encryption Key message is been sent to the terminal before the initial keys are entered.
- There are more than 13 print fields in a transaction reply message.
- The terminal receives a 'Load Date and Time' command that contains an invalid date and time.

5.24 Specific Command Reject Message

The following table describes the format of the specific command reject message including detailed descriptions of the reject reasons.

Terminal to Host:

No.	*Field Name*	*Number of Characters*	*Mandatory or Optional*	*Value*
1	Header	Var	M	**Var**
2	Message Class	1	M	**‘2’**
3	Message Sub-Class	1	M	**‘2’**
4	Field Separator	1	M	**FS**
5	Logical Unit Number	3 or 9	M	**Var**
6	Field Separator	1	M	**FS**
7	Field Separator	1	M	**FS**
8	Time Variant Number	8	O	**Var**
9	Field Separator	1	O	**FS**
10	Status Descriptor	1	M	**Var**
11	Field Separator	1	M	**FS**
12-1	Status Value	1	M	**Var**
12-2	Status Qualifier	2	O	**Var**
13	Field Separator	1	O	**FS**
14	Message Authentication Code	8	O	**Var**
15	Trailer	Var	M	**Var**

1	Header		Network Protocol dependent
2	Message Class	‘2’	Set to ‘2’ for a solicited message
3	Message Sub-Class	‘2’	Set to ‘2’ for a status message
4	Field Separator		Mandatory FS
5	Logical Unit Number		LUNO
6	Field Separator		Mandatory FS
7	Field Separator		Mandatory FS
8	Time Variant Number		TVN
9	Field Separator		Optional FS – this field is included when MACing is in use
10	Status Descriptor	‘C’	This field is set to ‘C’ for a Specific Command Reject
11	Field Separator		Mandatory FS

12-1	Status Value		This field contains the reason for the command reject as follows:
		‘1’	MAC failure
		‘2’	TVN failure
		‘3’	Security terminal number mismatch
		‘A’	Message format error
		‘B’	Field value error
		‘C’	Illegal message type for the current mode
		‘D’	Hardware failure
		‘E’	The command is not supported
12-2	Status Qualifier		The value of this field depends on the status value field as follows.
			If the status value field is set to ‘A’ (Message Format Error), this field can contain one of the following values:
		01	Message length error
		02	Field Separator missing/unexpectedly found
		03	Transaction Reply message has too many print groups
		04	Group Separator missing/unexpectedly found
		07	Malformed XML
		08	XML does not conform to the XML schema
			If the status value field is set to ‘B’ (Field Value Error), this field can contain one of the following values:
		01	Illegal Message Class
		02	Illegal Message Sub-Class or Identifier
		03	Illegal Encryption Key Change or Extended Encryption Key Change Message Modifier
		04	Illegal Terminal Command Code
		05	Illegal Terminal Command Modifier
		06	Illegal Transaction Reply Function Identifier
		07	Data field contains non-decimal digit
		08	Data field value out of range
		09	Invalid Message Co-ordination number
		10	Illegal FIT number
		11	Too many notes in a dispense function
		13	Unrecognized Document Destination
		15	Unrecognized Buffer Identifier

17	Document Name Error
18	Screen identifier out of range
20	No data supplied to endorse check
22	Invalid Encryption Key Size
23	RSA Signature Verification Failed
24	Signature or Encryption Key PKCS#1 Packing Failed
25	Signature or Encryption Key PKCS#1 Unpacking Failed
26	Invalid Signature or Encryption Key PKCS#1 Pad Block Type
27	Fixed Header Decryption Failed
28	Null Byte After Padding Missing
29	Invalid Pad Byte Count
34	Invalid/Incomplete Check Identifier(s)
35	Passbook update not supported in specified Transaction Reply Function
	If the status value field is set to 'C' (Illegal Message Type for Current Mode), this field can contain one of the following values:
01	Message type only accepted while terminal is In-Service and expecting a transaction reply message
02	Message cannot be accepted while diagnostics are in progress
03	Message cannot be accepted while in Out-of-Service or Supply mode
04	Message cannot be accepted while in In-Service mode
10	Message cannot be accepted while processing a transaction reply
11	Check not present in check processor transport while processing a transaction reply
15	Encryption Key Change or Extended Encryption Key Change message cannot be accepted during a transaction process, while the terminal is in Suspend mode, or while the operator is initiating the supervisor settlement transaction
17	Key change operation cannot be accepted in restricted encryption mode
18	Key entry mode is not authorized
	If the status value field is set to 'D' (Hardware Failure), this field can contain one of the following values:
01	Encryption failure during Encryption Key Change or Extended Encryption Key Change message

		02	Time-of-Day Clock failure or invalid data sent during Date/Time Set command
		06	Insufficient disk space
		07	File IO error
		08	File not found
			If the status value field is set to 'E' (Not Supported), this field can contain one of the following values:
		01	A DLL required to complete the transaction reply processing is missing
		02	Required device not configured, or sideways print on the receipt is requested, but either the printer does not have the capability or has not been configured for sideways printing
		05	Journal printer backup is inactive
13	Field Separator		Optional FS – this field is only sent if MACing is being used
14	Message Authentication Code		MAC
15	Trailer		Network protocol dependent

6 Encryption and RKT Supervisor Functions

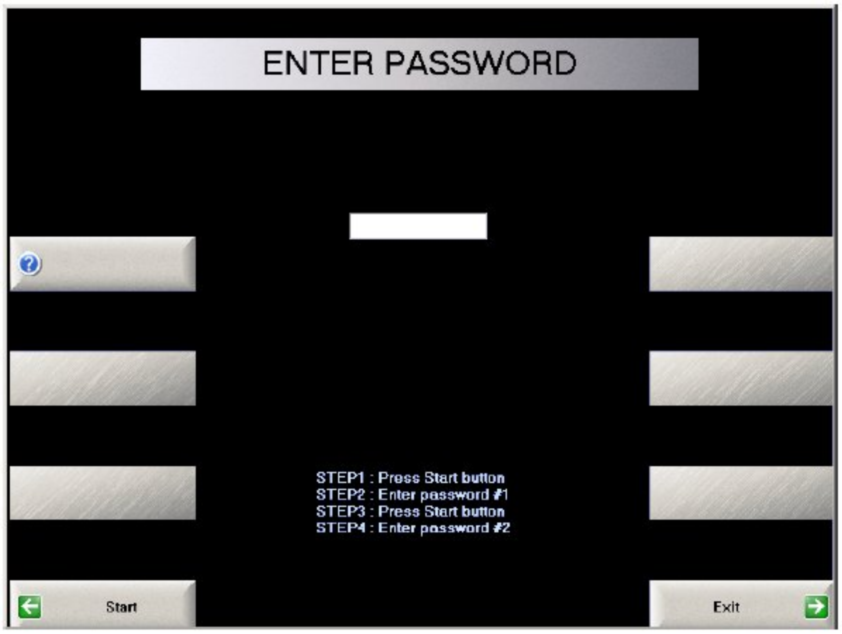
The Supervisor functions related to encryption and RKT are accessed from the Technician menus in supervisor mode. Different screens display for MoniPlus2 vs. MoniPlus2S. However, the same functions are available for both and include:

- Change one or both of the key management passwords
- View the current remote key information including the EPP serial number and the status of the keys loaded in the ATM through RKT
- Clear all keys in the EPP
- Test the EPP device
- Enter the master encryption keys
- View the status of locally entered keys

6.1 Key Management Instructions for MoniPlus2

To display the MoniPlus2 Key Management menu:

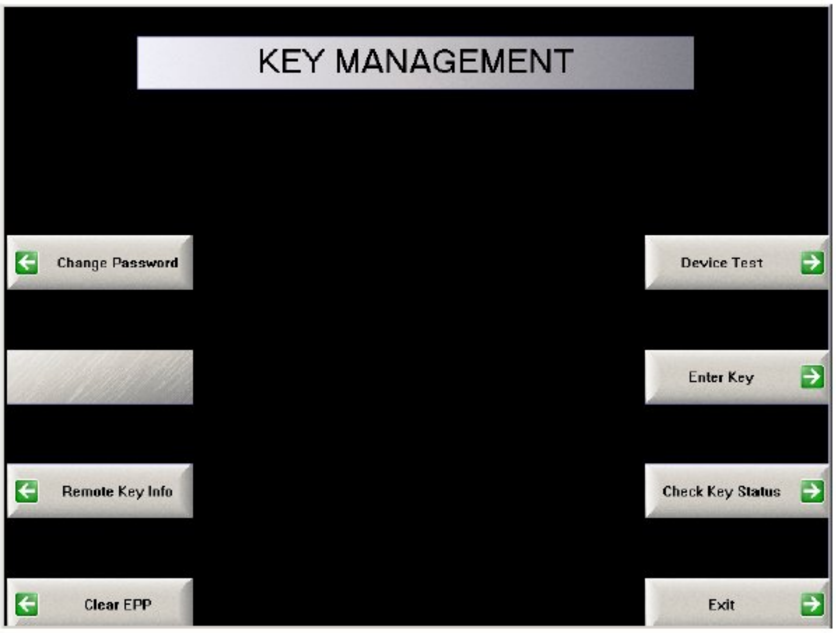
1. Move the mode switch to Supervisor.
2. When the Supervisor main menu displays, select **3 Technician** and enter the Technician password.
3. When the Technician menu displays, select **1 Diagnostics**.
4. On the Diagnostics menu, select **Pinpad**. The password screen shown below displays to you. Follow the instructions on the screen to login.


ENTER PASSWORD

STEP1 : Press Start button
STEP2 : Enter password #1
STEP3 : Press Start button
STEP4 : Enter password #2

5. After you successfully login, the Key Management menu shown below displays to allow you to perform your encryption and key management functions. Refer to your ATM operator manual for further details on these functions. **Note:** the ATM will reboot after you exit MoniPlus2 Diagnostic mode.

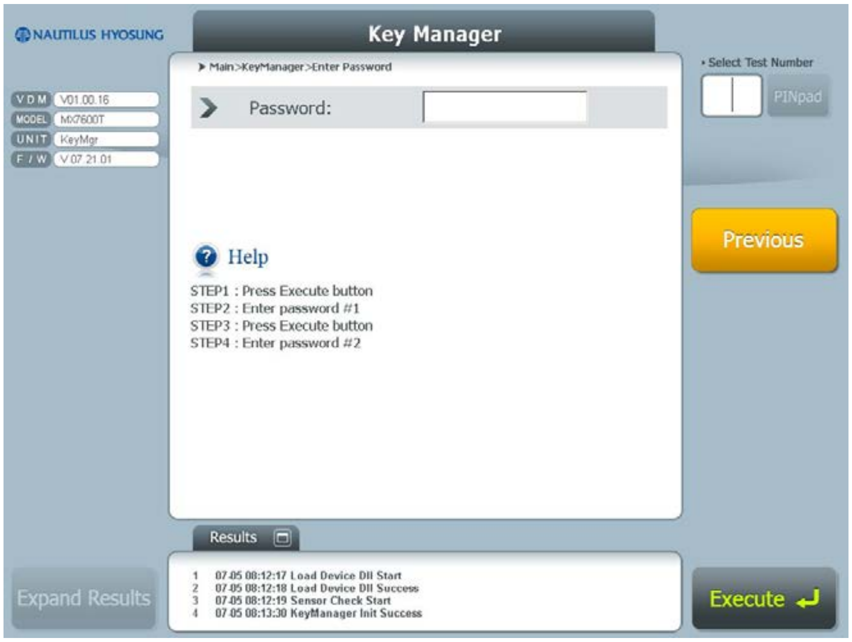
   Although NH supervisor functions exist to enter the Comms and MAC keys, the Secure Key mandates from VISA and MasterCard imply that only the Master Key can be manually entered at the ATM. Therefore, **NH recommends that any keys other than the Master Key always be downloaded from the host, encrypted under the Master Key.**



6.2 Key Manager Instructions for MoniPlus2S

To display the MoniPlus2S Key Manager menu:

1. Select Supervisor.
2. Select Technician mode and enter the Technician password to login.
3. When the Technician menu displays, select **8 Diagnostics**.
4. On the Diagnostics menu, select **4 Key Manager**. The Key Manager password screen shown on the top of the next page displays to you. Follow the instructions on the screen to login.



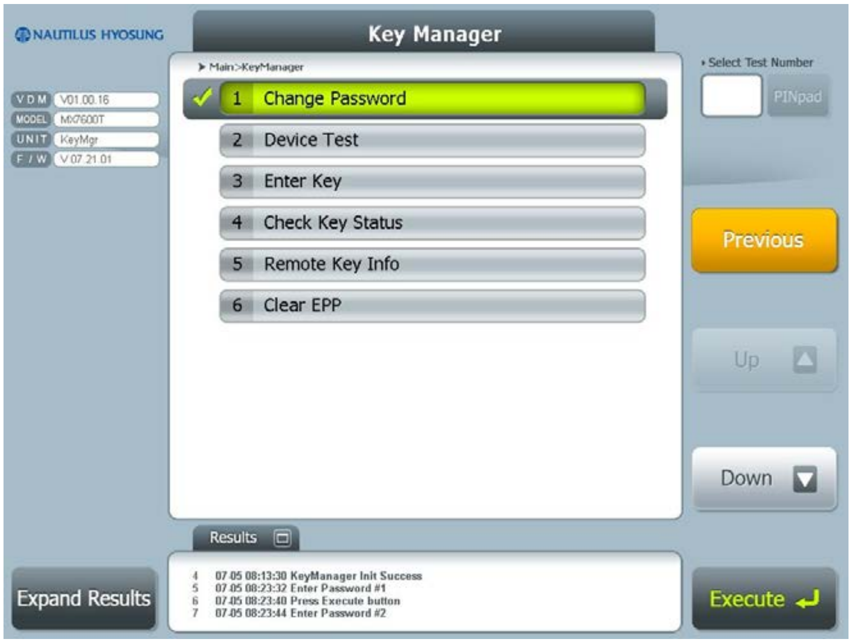
V D M: V01.00.16
MODEL: MX7600T
UNIT: KeyMgr
F / W: V 07.21.01

STEP1 : Press Execute button
STEP2 : Enter password #1
STEP3 : Press Execute button
STEP4 : Enter password #2

```
1    07-05 08:12:17 Load Device Dll Start
2    07-05 08:12:18 Load Device Dll Success
3    07-05 08:12:19 Sensor Check Start
4    07-05 08:13:30 KeyManager Init Success
```

5. After you successfully login, the Key Manager menu shown below displays to allow you to perform your encryption and key management functions. Refer to the Help menus available on the screens for further details on these functions. **Note:** Unlike MoniPlus2, the ATM will NOT reboot after you exit MoniPlus2S Diagnostic mode.

   Although NH supervisor functions exist to enter the Comms and MAC keys, the Secure Key mandates from VISA and MasterCard imply that only the Master Key can be manually entered at the ATM. Therefore, **NH recommends that any keys other than the Master Key always be downloaded from the host, encrypted under the Master Key.**



V D M: V01.00.16
MODEL: MX7600T
UNIT: KeyMgr
F / W: V 07.21.01

```
4    07-05 08:13:30 KeyManager Init Success
5    07-05 08:23:32 Enter Password #1
6    07-05 08:23:40 Press Execute button
7    07-05 08:23:44 Enter Password #2
```

7 Obtaining a Public Key Exchange Signature

The following describes the steps for obtaining a Signature for a Host Public Key with Nautilus Hyosung. File formats and samples are included following the exchange process.

1. The customer sends his Public Key (PK) to Nautilus Hyosung (NH).
   - The key file format must be an ASCII hex file as described in the *Host Public Key Format* table on the next page.
   - The key file should use the following naming convention:

     HostPK.txt
   - The key should be sent by the customer's authorized security expert who should provide to NH the following information:

     Company Name
     Company Address
     Contact Name
     Contact Phone Number
     Contact email address
   - The key file should be sent in an email using PGP or as a Zip file using password protection to the specified NH representative. The customer representative should contact the NH representative to exchange PGP authentication information or to provide Zip file password details as required.

2. The NH Certificate Authority (CA) will create the following files:
   - Digest of the customer's PK as **HostPK_Hash.txt**
   - Signature of the customer's PK as **HostPK_Sign.txt**
   - Digest for the signature of the customer's PK as **HostPK_Sign_Hash.txt**
   - NH CA Public Key as **CAPK.txt**
   - Digest for the CA Public Key as **CAPK_Hash.txt**

3. The NH contact will send the files generated by the CA to the customer using the same security method established during the initial key exchange.

7.1 Host Public Key Format

The customer should provide his Public Key as an ASCII hex string file, named **HostPK.txt**, using the format shown in the table below.

Field	Description	Length	Value
1	Tag	8	30 82 01 0A
2	Modulus Tag	8	02 82 01 01
3	Public Key Modulus	514	00 (prefix) … 2048 bit modulus
4	Exponent Tag	4	02 03
5	Public Key Exponent	6	Exponent 01 00 01
6	Signature	512	Signature of the Host Public Key signed with the Host Private Key (SK-HOST)

Example of a Host Public Key File

Field	Description	Value
1	Tag	3082010A
2	Modulus Tag	02820101
3	Public Key Modulus	00 (prefix) **A5B0B0C51522E389CD0AEE581AEDF1F4A7FAAF34CA7F5EBE 5C2C9BABBF3DD2E2B903571B468BEFBAAF9CCECE4E9ADCE A1F9AAB49FD569C5CA9CB66E84B23BABB5554A1831AF1301CB B93C739D3D87FC4261A2C5FAC58F62A8A1B5896C384DF7742A63 A3BA8D8941E9730BE98AB82A9A24659179B93994B43D327D6257A 62F1D269586CD857CB623177D22E0577D8CD019DE420B472B18DF F7A26E7881A105511E8303CEE305F8F08F3CCD9790B3D4E36110B 5278F145D2B670996797490024E63AC6D36E58AD45D2D8FB91E4C 6971B3106085934B186968A12ED5CD091AF79A3EE8A13FF5B409A 24491D8D4DC4471572556615261 4BB2080DED7B8BC60FCE7BB**
4	Exponent Tag	0203
5	Public Key Exponent	010001
6	Signature	**9896D9ADD3663A60C2FD023362D3EDC5E799219D34ACEFC4 3C6B4F03679039F636BFC66BA37FE39F5CDCA43A9E5F90F0D 6B3A696F7CF9980AF0C014F5A42F8935C72EB6C0D848BAB63 C73D9A18D6B425FC33FB98CE371D8400EED5D3625A650D92 99B6081F2BFDD1AFFD7ECA01D359EF69F851FD9ECB752A7B A9B140AC54A469DDDD0870EFA01855AA8A524C0C64529608 BC2A8CE34FB61925D1523FE29929792ADC1ECB615F092D48**

Field	Description	Value
		C0F5DA8A0086B5EC611D1C470772D97E6C68D46044383996 6730605B2686588056A1F18CA2966C0B5746BC4F8D45615D4 6334989CCE7215DA0777970C3542603851FD1E572C892C9FA 34934751DA9C59F5717A3BE49A58

7.2 Digest for the Host Public Key

Nautilus Hyosung will provide a Digest for the Host Public Key as an ASCII hex string file, named **HostPK_Hash.txt**, in the following format.

Field	Description	Length	Value
1	Digest	40	SHA-1 Calculation of fields 1-6 of the Host Public Key defined above.

Example of a Host PK Digest File

Field	Description	Value
1	Digest	**C9984F7D5AC08D4A5B991E67335F4F498FBBE3AF**

7.3 Host Public Key Signature

Nautilus Hyosung will provide a Signature of the Host PK as an ASCII hex string file, named **HostPK_Sign.txt**, in the following format.

Field	Description	Length	Value
1	Signature	512	Signature for Fields 1–5 of the Host PK using the NH CA Secret Key (SK-CA)

Example of a Host Public Key Signature File

Field	Description	Value
1	Signature	**29241659244606E07E86FA2F09727F3DE7CEB0DD16792B30C 658794A9835CE444AC238AFDE660EB776F7EF9595C245F0B5 362E912B7FD11013401DBED9EA50AA253E1D3769DAA1BBB5 42D665B12CDA7B6D11CF752D45D96A676E8434C248F57E41 BB63F0065BA3951C08EB2AC89D07724D3BC43D4EB8CB189B DFB076F9C489BAA0F4DEA62305FB299C695855678A36C66E2 B72959F07BA962D9823A51A63CF0738DC3FA9CEE3D79ED79 511995A50F9BD933EFAA71E96F8F56D4ABA34149C23D268 9BDD32DB23E117F9D579DD0A6AE3AB346D4C9C0267A0A4F E564A69938E1F7FBC6E8A087B611937CF8E2C20E1CE463327 3669B4B2FF658843DF0BA8854B966**

7.4 Digest for the Signature of the Host Public Key

Nautilus Hyosung will provide a Digest for the Signature of the Host PK as an ASCII Hex string file, named **CAPK_Sign_Hash.txt**, in the following format.

Field	Description	Length	Value
1	Digest	40	SHA-1 Calculation of the Host PK signature.

Example of the Digest for a Signature File

Field	Description	Value
1	Digest	**43F42CBF514331090456DE9818C0356E7AFA131C**

7.5 NH CA Public Key (PK CA) Format

Nautilus Hyosung will provide its CA PK as an ASCII hex string file, named **CAPK.txt**, in the following format.

Field	Description	Length	Value
1	Tag	8	30 82 01 0A
2	Modulus Tag	8	02 82 01 01
3	NH CA Modulus	514	00 (prefix) … 2048 bit modulus
4	Exponent Tag	4	02 03
5	Public Key Exponent	6	Exponent 01 00 01
6	Signature	512	Signature of the CA PK signed with the NH CA Private Key.

7.6 Digest for NH CA Public Key:

Nautilus Hyosung will provide a Digest for the NH CA Public Key as an ASCII hex string file, named **CAPK_Hash.txt**, in the following format.

Field	Description	Length	Value
1	Digest	40	SHA-1 Calculation of fields 1-6 of the NH CA Public Key.

8 Index