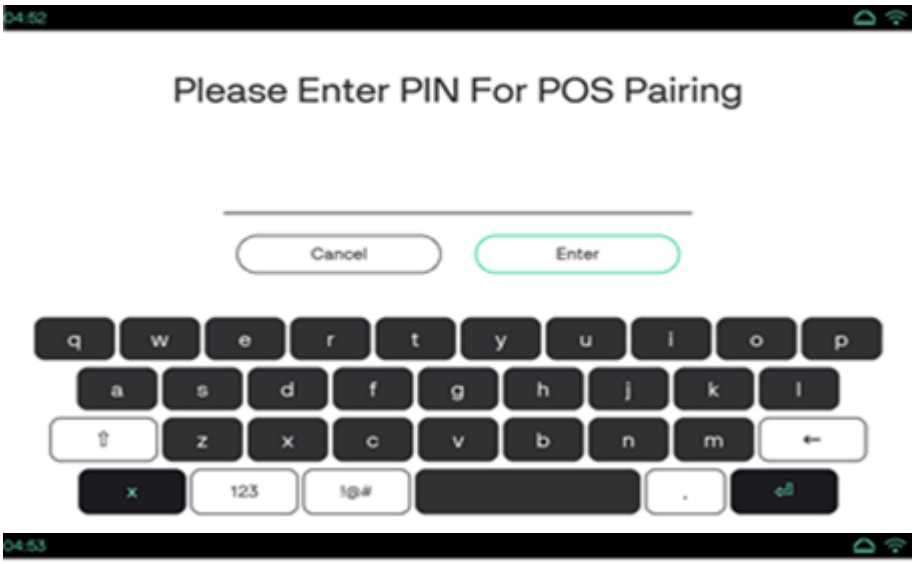
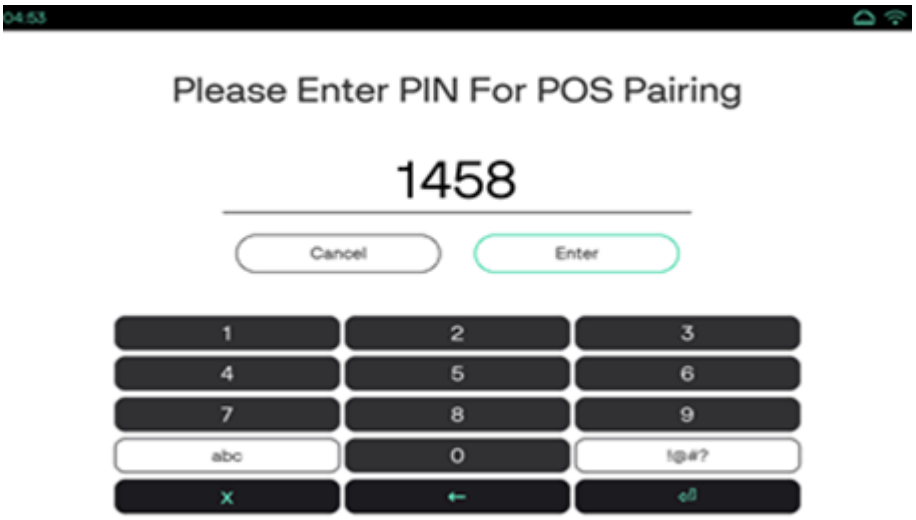
https://verifone.cloud/docs/sca-functional-specification/protocol_spec/administration/register_enc
Updated: 23-Jul-2025

REGISTER_ENCRYPTION

This command is used when full packet encryption is to be implemented.

Device UI Required

Display	User Action	Terminal Action
	Enter the PIN for POS paring.	The device displays POS paring screen.
	Select the Enter button.	

Request Packet

Field	Rule	Type	Minimum	Maximum	Value(s)	Description
FUNCTION_TYPE	Required	Static Value	N/A	N/A	SECURITY	Type of function.
COMMAND	Required	Static Value	N/A	N/A	REGISTER_ENCRYPTION	Command name
KEY	Required	Base64 Encoded Data				RSA 2048-bit public key pair signed by a (possibly) self-signed certificate. This is used to decrypt the MAC_KEY returned in the response. This value is Base64 encoded.
REG_VER	Conditional	List	1	2	Possible values: • 1 - SHA1 with PKCS padding • 2 - SHA2 with OAEP padding • Empty - Uses SHA1 with PKCS padding (by default)	If REG_VER is not sent, then it will be treated as REG_VER 1.

Example

Following is an example of request packet

```
<TRANSACTION>
<FUNCTION_TYPE>SECURITY</FUNCTION_TYPE>
<COMMAND>REGISTER_ENCRYPTION</COMMAND>
<KEY> … </KEY>
<REG_VER>2</REG_VER>
</TRANSACTION>
```

Response Packet

Field	Type	Value	Description
RESPONSE_TEXT	Character	Registered	Processor response text.

Field	Type	Value	Description
RESULT	Character	Valid values: OK or ERROR	This indicates the Result details.
RESULT_CODE	Numeric	Valid values: • -1 • 59001 • 59020 • 59040 • 59051	This indicates the result code. Refer to Result/Error Codes for details.
TERMINATION_STATUS	Character	SUCCESS or FAILURE	This indicates the transaction termination status. This is the overall status of the transaction irrespective of approved or declined. Like, if the output is generated then the status is SUCCESS and if no output is generated then the status will be FAILURE.
TERMINAL_KEY	Base64 Encoded Data	Encrypted AES-128 with Public Key	This is the terminal key used to encrypt and decrypt the transaction data. It will appear as Base64 data. The decoded, decrypted TERMINAL_KEY is to be used as the MAC Key when encryption is enabled.
MAC_LABEL	Character	P_<Random String>	Value to be stored by POS
ENTRY_CODE	Character	Value entered by user	Appears as a Base64 encoded, encrypted value representing 8-character code sent in response for POS to validate.

Example

Following is an example of response packet

```
<RESPONSE>
<RESPONSE_TEXT>Registered</RESPONSE_TEXT>
<RESULT>OK</RESULT>
<RESULT_CODE>-1</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<TERMINAL_KEY> … </TERMINAL_KEY>
<MAC_LABEL>aBC124</MAC_LABEL>
<ENTRY_CODE>…</ENTRY_CODE>
</RESPONSE>
```

Note

The COUNTER resets to 0 with each REGISTER command.

Sample for Register Encryption 1.0:

SCI_REQUEST


```
<TRANSACTION>
<FUNCTION_TYPE>SECURITY</FUNCTION_TYPE>
<COMMAND>REGISTER_ENCRYPTION</COMMAND>
<REG_VER>1</REG_VER>
<KEY>
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsnM6CnGUBdIjvs0Li5S349c1S1pL1n7tQVmQZ
</KEY>
</TRANSACTION>
```


SCI_RESPONSE


```
<RESPONSE>
<RESPONSE_TEXT>Registered P_YI1SWE</RESPONSE_TEXT>
<RESULT>OK</RESULT>
<RESULT_CODE>-1</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<MAC_LABEL>P_YI1SWE</MAC_LABEL>
<ENTRY_CODE>
UuSMf+Rs/fE1RH23YO7R4r/U3OTu153A+2zOSA/NE203WcL9+aP/aAyoYTaeFl4K6yNQdYSZuQWaSHR0
</ENTRY_CODE>
<TERMINAL_KEY>
U7GsMtv3Gd3ZIb2j0/0/9CTdiBrKRJROdoyslluTQ250hGADEMlYD954BD7v8r5WQUcOMYfV+v9GW0lY
</TERMINAL_KEY>
</RESPONSE>
```


Explanation:

```
Private key in base 64
 : MIIEpAIBAAKCAQEAs4gXJt6dfXNi6L3UmGZ0eG/phIZdqfVpQROXyyGULpZ2iI7cllQ4XOLxH1L68V
//2/yPV0MeAH2ypNptlb2Qr2o4IdxDu4nGVHk7CMp8COwKxpljY+RaFqkTYM+hPiwDPLodD3IKhi/ZBcl

Private key in base 64 length : 1592

Public key in base 64
 : MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAs4gXJt6dfXNi6L3UmGZ0eG/phIZdqfVpQR
//2/yPV0MeAH+2ypNptlb2Qr2o4IdxDu4nGVHk7CMp8COwKxpljY+RaFqkTYM+hPiwDPLodD3IKhi/ZBc

Public key in base 64 length : 392
```

Register Request:


```
<TRANSACTION>
<FUNCTION_TYPE>SECURITY</FUNCTION_TYPE>
<COMMAND>REGISTER_ENCRYPTION</COMMAND>  <KEY>
```

MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAs4gXJt6dfXNi6L3UmGZ0eG/phIZdqfVpQROX
</KEY>
</TRANSACTION>

Register Response:

<RESPONSE>
<RESPONSE_TEXT>Registered P_KD1S7Y</RESPONSE_TEXT>
<RESULT>OK</RESULT>
<RESULT_CODE>-1</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<MAC_LABEL>P_KD1S7Y</MAC_LABEL> <ENTRY_CODE>
rkr8DDwPyhWSffdAUs9AG20LCo8DtKnxezV9DPigwBJ8BLojQeixBpZAoj1DwJoJ4jY28T5kK3ueIM8k
</ENTRY_CODE>
<TERMINAL_KEY>
l5hEg80188g1KBQL0lOSxtPQtbvCenMjCMhDhz68QuZZCpml08BHVI+yVUXYC9GrQe/HqYVaC+nCt/Ub
</TERMINAL_KEY>
</RESPONSE>
encryptedMacKey : l5hEg80188g1KBQL0lOSxtPQtbvCenMjCMhDhz68QuZZCpml08BHVI+yVUXYC9
The Requested entry code was: FAAE4612
The Returned entry code was: FAAE4612
The Entry Codes match.
decoded macKey in base64 :JKG+pFJEbWwWNeZtGfae/Q==

Sample for Register Encryption 2.0:

SCI_REQUEST

<TRANSACTION>
<FUNCTION_TYPE>SECURITY</FUNCTION_TYPE>
<COMMAND>REGISTER_ENCRYPTION</COMMAND>
<REG_VER>2</REG_VER>
<KEY>
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsoZ8CoiZl/QBTq71BrW3q5Ay14OQyL5r6/zx
</KEY>
</TRANSACTION>

SCI_RESPONSE

<RESPONSE>
<RESPONSE_TEXT>Registered P_VS2NXH</RESPONSE_TEXT>
<RESULT>OK</RESULT>
<RESULT_CODE>-1</RESULT_CODE>
<TERMINATION_STATUS>SUCCESS</TERMINATION_STATUS>
<MAC_LABEL>P_VS2NXH</MAC_LABEL>
<ENTRY_CODE>
isIfyQwgacj5lAA7ixeRS1Nw1W2leWMvSF4lz7j+Ryu9VAtuglZrnlcuBtUbJx16Lj+TW0A/ju5rxL/C
</ENTRY_CODE>   <TERMINAL_KEY>
BVchapq1WiAnWmD31uPlBrmxNdRlCLiAD+4QM40ALl16KKNyHi/V/EXaZfjXbZ17ysCizrbaPrKwR/0E

</TERMINAL_KEY>
</RESPONSE>