ICSF

Enhanced PIN Security Mode

Contents

Chapter 1. Introduction

Overview

This document describes changes to ICSF callable services based on firmware changes to the 4764 (PCIXCC or CEX2C) cryptographic coprocessors.

Summary of Changes

The following firmware changes were made to the PCIXCC and CEX2C cryptographic coprocessors:

- Clear PIN Generate Alternate (CPA) was modified so that the PVV output value is always 4 digits in length, rather than being padded with zeroes to the length of the PIN.
- Enhanced PIN Security

  Note: Since the enhancements made to PIN security changed the way some PIN functions work, an access control point has been defined (via ICSF APAR OA13764) so that the enhancements can be enabled or disabled, at the discretion of the customer, using a TKE workstation. This PTR Enhanced PIN Security access control point is disabled by default.

  When disabled, the PTR Enhanced PIN Security access control point allows the cryptographic coprocessor to continue processing in the old manner, without the security enhancement, and to continue to be completely compatible with existing applications. If the access control point is enabled via a TKE workstation, the card switches to the enhanced, more secure mode of operation.

  – Processing changes when the access control point is enabled, if the PIN block format is 3624 or 3621, for Clear PIN Generate Alternate (CPA), PIN Change/Unblock (PCU), Encrypted PIN Translate (PTR), or Encrypted PIN Verify (PVR).

    The changes are as follows:

    - Extraction method PADDIGIT, which is used by CPA, PTR, and PVR to determine the PIN length by scanning from right to left until a digit, not equal to the pad digit, is found. The minimum PIN length is set at 4 digits as it is with the CCF, so scanning ceases one digit after the position of the 4th PIN digit in the block. No changes are made for any extraction method other than PADDIGIT.
    - Do not examine the PIN to see if it contains the pad digit.

  – Processing changes to not examine the PIN, in the output PIN block, to see if it contains the pad digit, if the PIN block format is 3624 or 3621, and the access control point is enabled for Clear PIN Encrypt (CPE), Encrypted PIN Generate (EPG), or Encrypted PIN Translate (PTR).

Chapter 2. Introducing Symmetric Key Cryptography and Using Symmetric Key Callable Services

Managing Personal Authentication

The process of validating personal identities in a financial transaction system is called *personal authentication*. The personal identification number (PIN) is the basis for verifying the identity of a customer across the financial industry networks. ICSF checks a customer-supplied PIN by verifying it using an algorithm. The financial industry needs functions to generate, translate, and verify PINs. These functions prevent unauthorized disclosures when organizations handle personal identification numbers.

ICSF supports the following algorithms for generating and verifying personal identification numbers:

- IBM 3624
- IBM 3624 PIN offset
- IBM German Bank Pool
- IBM German Bank Pool PIN Offset (GBP-PINO)
- VISA PIN validation value
- Interbank

With ICSF, you can translate PIN blocks from one format to another. ICSF supports the following formats:

- ANSI X9.8
- ISO formats 0, 1, 2
- VISA formats 1, 2, 3, 4
- IBM 4704 Encrypting PINPAD format
- IBM 3624 formats
- IBM 3621 formats
- ECI formats 1, 2, 3

With the capability to translate personal identification numbers into different PIN block formats, you can use personal identification numbers on different systems.

Verifying Credit Card Data

The Visa International Service Association (VISA) and MasterCard International, Incorporated have specified a cryptographic method to calculate a value that relates to the personal account number (PAN), the card expiration date, and the service code. The VISA card-verification value (CVV) and the MasterCard card-verification code (CVC) can be encoded on either track 1 or track 2 of a magnetic striped card and are used to detect forged cards. Because most online transactions use track-2, the ICSF callable services generate and verify the CVV[1] by the track-2 method.

The VISA CVV generate callable service calculates a 1- to 5-byte value through the DES-encryption of the PAN, the card expiration date, and the service code using two data-encrypting keys or two MAC keys. The VISA CVV verify callable service calculates the CVV by the same method, compares it to the CVV supplied by the application (which reads the credit card's magnetic stripe) in the *CVV_value*, and issues a reason code that indicates whether the card is authentic.

1. The VISA CVV and the MasterCard CVC refer to the same value. CVV is used here to mean both CVV and CVC.

Clear PIN Encrypt Callable Service

To format a PIN into a PIN block format and encrypt the results, use the Clear PIN Encrypt callable service. You can also use this service to create an encrypted PIN block for transmission. With the **RANDOM** keyword, you can have the service generate random PIN numbers. Use of this service requires the optional PCICC or PCIXCC/CEX2C. An enhanced PIN security mode, on PCIXCC/CEX2C, is available for formatting an encrypted PIN block into IBM 3621 format or IBM 3624 format. See “Clear PIN Encrypt (CSNBCPE)” on page 7 for more information.

Clear PIN Generate Alternate Callable Service

To generate a clear VISA PIN validation value from an encrypted PIN block, call the Clear PIN Generate Alternate callable service. This service also supports the IBM-PINO algorithm to produce a 3624 offset from a customer selected encrypted PIN.

An enhanced PIN security mode is available for extracting PINs from encrypted PIN blocks. This mode only applies on PCIXCC/CEX2C when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. See “Clear PIN Generate Alternate (CSNBCPA)” on page 10 for more information.

Note: The PIN block must be encrypted under either an input PIN-encrypting key (IPINENC) or output PIN-encrypting key (OPINENC). Using an IPINENC key requires NOCV keys to be enabled in the CKDS. Functions other than VISA PIN validation value generation require the optional PCICC or PCIXCC/CEX2C.

Clear PIN Generate Callable Service

To generate personal identification numbers, call the Clear PIN Generate callable service. Using a PIN generation algorithm, data used in the algorithm, and the PIN generation key, the callable service generates a clear PIN, a PIN verification value, or an offset. The callable service can only execute in special secure mode.

Encrypted PIN Generate Callable Service

To generate personal identification numbers, call the Encrypted PIN Generate callable service. Using a PIN generation algorithm, data used in the algorithm, and the PIN generation key, the callable service generates a PIN and using a PIN block format and the PIN encrypting key, formats and encrypts the PIN. Use of this service requires the optional PCICC or PCIXCC/CEX2C. An enhanced PIN security mode, on PCIXCC/CEX2C, is available for formatting an encrypted PIN block into IBM 3621 format or IBM 3624 format. See “Encrypted PIN Generate (CSNBEPG)” on page 16 for more information.

Encrypted PIN Translate Callable Service

To translate a PIN from one PIN-encrypting key to another or from one PIN block format to another or both, call the Encrypted PIN Translate callable service. You must identify the input PIN-encrypting key that originally enciphers the PIN. You also need to specify the output PIN-encrypting key that you want the callable service to use to encipher the PIN. If you want to change the PIN block format, specify a different output PIN block format from the input PIN block format. An enhanced PIN security mode, on PCIXCC/CEX2C, is available for formatting an encrypted PIN block into IBM 3621 format or IBM 3624 format. The enhanced security mode is also available for extracting PINs from encrypted PIN blocks. This

mode only applies when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. See “Encrypted PIN Translate (CSNBPTR)” on page 20 for more information.

Encrypted PIN Verify Callable Service

To verify a supplied PIN, call the Encrypted PIN Verify callable service. You need to specify the supplied enciphered PIN, the PIN-encrypting key that enciphers it, and other relevant data. You must also specify the PIN verification key and PIN verification algorithm. It compares the two personal identification numbers; if they are the same, it verifies the supplied PIN. See Chapter 3, “Financial Services,” on page 7 for additional information.

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for extracting PINs from encrypted PIN blocks. This mode only applies when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. See “Encrypted PIN Verify (CSNBPVR)” on page 26 for more information.

PIN Change/Unblock Callable Service

To support PIN change algorithms specified in the VISA Integrated Circuit Card Specification, call the PIN Change/Unblock callable service. The callable service can only execute on an z890 or Requires May 2004 or later version of Licensed Internal Code (LIC).

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for extracting PINs from encrypted PIN blocks. This mode only applies when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. See “PIN Change/Unblock (CSNBPCU)” on page 32 for more information.

Transaction Validation Callable Service

To support generation and validation of American Express card security codes, call the Transaction Validation callable service. The callable service can only execute on an z890 or Requires May 2004 or later version of Licensed Internal Code (LIC).

Chapter 3. Financial Services

The process of validating personal identities in a financial transaction system is called *personal authentication*. The personal identification number (PIN) is the basis for verifying the identity of a customer across financial industry networks. ICSF provides callable services to translate, verify, and generate PINs. You can use these callable services to prevent unauthorized disclosures when organizations handle PINs.

The following callable services are described in the following topics:

PIN Callable Services

You use the PIN callable services to generate, verify, and translate PINs. This section discusses the PIN callable services, as well as the various PIN algorithms and PIN block formats supported by ICSF. It also explains the use of PIN-encrypting keys.

For more information about PIN-encrypting keys, see *z/OS Cryptographic Services ICSF Administrator's Guide*.

Clear PIN Encrypt (CSNBCPE)

The Clear PIN Encrypt callable service formats a PIN into one of the following PIN block formats and encrypts the results. You can use this service to create an encrypted PIN block for transmission. With the **RANDOM** keyword, you can have the service generate random PIN numbers.

- IBM 3621 format
- IBM 3624 format
- ISO-0 format (same as the ANSI X9.8, VISA-1, and ECI formats)
- ISO-1 format (same as the ECI-4 format)
- ISO-2 format
- IBM 4704 encrypting PINPAD (4704-EPP) format
- VISA 2 format
- VISA 3 format
- VISA 4 format
- ECI2 format
- ECI3 format

Note: A clear PIN is a sensitive piece of information. Ensure that your application program and system design provide adequate protection for any clear PIN value.

An enhanced PIN security mode is available for formatting an encrypted PIN block into IBM 3621 format or IBM 3624 format. To do this, you must enable the PTR Enhanced PIN Security access control point (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits. No other PIN block consistency checking will occur.

Format

```
CALL CSNBCPE(
             return_code,
             reason_code,
             exit_data_length,
             exit_data,
             PIN_encrypting_key_identifier,
             rule_array_count,
             rule_array,
             clear_PIN,
             PIN_profile,
             PAN_data,
             sequence_number,
             encrypted_PIN_block )
```

Parameters

return_code

Direction: Output Type: Integer

The return code specifies the general result of the callable service. Appendix A, “Return and Reason Codes” lists the return codes.

reason_code

Direction: Output Type: Integer

The reason code specifies the result of the callable service that is returned to the application program. Each return code has different reason codes that indicate specific processing problems. Appendix A, “Return and Reason Codes” lists the reason codes.

exit_data_length

Direction: Input/Output Type: Integer

The length of the data, in bytes, that is passed to the installation exit. The length can be from X'00000000' to X'7FFFFFFFFF' (2 gigabytes). The data is defined in the *exit_data* parameter.

exit_data

Direction: Input/Output Type: String

The data that is passed to the installation exit.

PIN_encrypting_key_identifier

Direction: Input/Output Type: String

The 64-byte string containing an internal key token or a key label of an internal key token. The internal key token contains the key that encrypts the PIN block. The control vector in the internal key token must specify an OPINENC key type and have the CPINENC usage bit set to 1.

rule_array_count

Direction: Input Type: Integer

The number of 8 byte keywords you are supplying in the *rule_array* parameter. Valid values are 0, 1, and 2.

rule_array

Direction: Input Type: Character string

Keywords that provide control information to the callable service. The keyword is left-justified in an 8-byte field, and padded on the right with space characters. All keywords must be in contiguous storage. The rule-array keywords are shown as follows:

Table 1. Process Rules for the Clear PIN Encrypt Callable Service

Process Rule	Description
ENCRYPT	This is the default. Use of this keyword is optional.
RANDOM	Causes the service to generate a random PIN value. The length of the PIN is based on the value in the *clear_PIN* variable. Set the value of the clear PIN to as many zero digits as the desired random PIN; pad the remainder of the clear PIN variable with space characters.

clear_PIN

Direction: Input Type: String

A 16-character string with the clear PIN. The value in this variable must be left-justified and padded on the right with space characters.

PIN_profile

Direction: Input Type: String

A 24-byte string containing three 8-byte elements with a PIN block format keyword, the format control keyword NONE, and a pad digit as required by certain formats.

PAN_data

Direction: Input Type: String

A 12-byte Personal Account Number (PAN) in character format. The service uses this parameter if the PIN profile specifies the ISO-0 or VISA-4 keyword for the PIN block format. Otherwise, ensure that this parameter is a 12-byte variable in application storage. The information in this variable will be ignored, but the variable must be specified.

Note: When using the ISO-0 keyword, use the 12 rightmost digits of the PAN data, excluding the check digit. When using the VISA-4 keyword, use the 12 leftmost digits of the PAN data, excluding the check digit.

sequence_number

Direction: Input Type: Integer

The 4-byte integer. The service currently ignores the value in this variable. For future compatibility, the suggested value is 99999.

encrypted_PIN_block

Direction: Output Type: String

The field that receives the 8-byte encrypted PIN block.

Restrictions

The caller must be in task mode, not in SRB mode.

The format control specified in the PIN profile must be NONE. If PBVC is specified as the format control, the service will fail.

Usage Notes

SAF will be invoked to check authorization to use the Clear PIN Encrypt callable service and the label of the *PIN_encrypting_key_identifier*.

The following table lists the required cryptographic hardware for each server type and describes restrictions for this callable service.

Table 2. Clear PIN Encrypt Required Hardware

Server	Required cryptographic hardware	Restrictions
IBM @server zSeries 800 IBM @server zSeries 900	PCI Cryptographic Coprocessor	
IBM @server zSeries 990 IBM @server zSeries 890	PCI X Cryptographic Coprocessor Crypto Express2 Coprocessor	
IBM System z9 109	Crypto Express2 Coprocessor	

Clear PIN Generate Alternate (CSNBCPA)

Use the clear PIN generate alternate service to generate a clear VISA PVV (PIN validation value) from an input encrypted PIN block, or to produce a 3624 offset from a customer-selected encrypted PIN. The PIN block can be encrypted under either an input PIN-encrypting key (IPINENC) or an output PIN-encrypting key (OPINENC).

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for extracting PINs from encrypted PIN blocks. This mode only applies when specifying a

PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. To do this, you must enable the PTR Enhanced PIN Security access control point (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits and a PIN length minimum of 4 is enforced. No other PIN-block consistency checking will occur.

Format

```
CALL CSNBCPA(
             return_code,
             reason_code,
             exit_data_length,
             exit_data,
             PIN_encryption_key_identifier,
             PIN_generation_key_identifier,
             PIN_profile,
             PAN_data,
             encrypted_PIN_block,
             rule_array_count,
             rule_array,
             PIN_check_length,
             data_array,
             returned_PVV )
```

Parameters

return_code

Direction: Output Type: Integer

The return code specifies the general result of the callable service. Appendix A, “Return and Reason Codes” lists the return codes.

reason_code

Direction: Output Type: Integer

The reason code specifies the result of the callable service that is returned to the application program. Each return code has different reason codes that are assigned to it that indicate specific processing problems. Appendix A, “Return and Reason Codes” lists the reason codes.

exit_data_length

Direction: Input/Output Type: Integer

The length of the data, in bytes, that is passed to the installation exit. The length can be from X'00000000' to X'7FFFFFFF' (2 gigabytes). The data is identified in the *exit_data* parameter.

exit_data

Direction: Input/Output Type: String

The data that is passed to the installation exit.

PIN_encryption_key_identifier

Direction: Input/Output Type: String

A 64-byte string consisting of an internal key token that contains an IPINENC or OPINENC key or the label of an IPINENC or OPINENC key that is used to encrypt the PIN block. If you specify a label, it must resolve uniquely to either an IPINENC or OPINENC key. If the *PIN_encryption_key_identifier* identifies a key which does not have the default PIN encrypting control vector (either IPINENC or OPINENC), the request will be routed to the PCI Cryptographic Coprocessor for processing.

PIN_generation_key_identifier

Direction: Input/Output Type: String

A 64-byte string that consists of an internal key token that contains a PIN generation (PINGEN) key or the label of a PINGEN key. If the *PIN_generation_key_identifier* identifies a key that does not have the default PIN generating control vector, the request will be routed to the PCI Cryptographic Coprocessor for processing.

PIN_profile

Direction: Input Type: Character string

The three 8-byte character elements that contain information necessary to extract a PIN from a formatted PIN block. The pad digit is needed to extract the PIN from a 3624 or 3621 PIN block in the Clear PIN Generate Alternate callable service.

PAN_data

Direction: Input Type: String

A 12-byte field that contains 12 characters of PAN data. The personal account number recovers the PIN from the PIN block if the PIN profile specifies ISO-0 or VISA-4 block formats. Otherwise it is ignored, but you must specify this parameter.

For ISO-0, use the rightmost 12 digits of the PAN, excluding the check digit. For VISA-4, use the leftmost 12 digits of the PAN, excluding the check digit.

encrypted_PIN_block

Direction: Input Type: String

An 8-byte field that contains the encrypted PIN that is input to the VISA PVV generation algorithm. The service uses the IPINENC or OPINENC key that is specified by the *PIN_encryption_key_identifier* parameter to encrypt the block.

rule_array_count

Direction: Input Type: Integer

The number of 8 byte keywords specified in the *rule_array* parameter. The value may be 1 or 2. If the default extraction method for a PIN block format is desired, you may code the rule array count value as 1.

rule_array

Direction: Input Type: Character string

The process rule for the PIN generation algorithm. Specify IBM-PINO or VISA-PVV (the VISA PIN verification value) in an 8-byte field, left-justified, and padded on the right with space characters. The *rule_array* points to an array of one or two 8-byte elements as follows:

Table 3. Rule Array Elements for the Clear PIN Generate Alternate Callable Service

Rule Array Element	Function of Rule Array keyword
1	PIN calculation method
2	PIN extraction method

The first element in the rule array must specify one of the keywords that indicate the PIN calculation method as shown below:

Table 4. Rule Array Keywords (First Element) for the Clear PIN Generate Alternate Service

PIN Calculation Method Keyword	Meaning
IBM-PINO	This keyword specifies use of the IBM 3624 PIN Offset calculation method.
VISA-PVV	This keyword specifies use of the VISA PVV calculation method.

If the second element in the rule array is provided, one of the PIN extraction method keywords can be specified for the given PIN block format. If the default extraction method for a PIN block format is desired, you can code the rule array count value as 1.

The PIN extraction methods operate as follows:

PINBLOCK
Specifies that the service use one of the following:

- the PIN length, if the PIN block contains a PIN length field
- the PIN delimiter character, if the PIN block contains a PIN delimiter character.

PADDIGIT
Specifies that the service use the pad value in the PIN profile to identify the end of the PIN.

HEXDIGIT
Specifies that the service use the first occurrence of a digit in the range from X'A' to X'F' as the pad value to determine the PIN length.

PINLENxx
Specifies that the service use the length specified in the keyword, where xx can range from 4 (PINLEN04) to 16 (PINLEN16) digits, to identify the PIN.

PADEXIST
Specifies that the service use the character in the 16th position from the left of the PIN block as the value of the pad value.

PIN_check_length

Direction: Input Type: Integer

The length of the PIN offset used for the IBM-PINO process rule only. Otherwise, this parameter is ignored. Specify an integer from 4 through 16.

Note: The PIN_check_length should be less than or equal to the PIN length. If the PIN_check_length is greater than the PIN length, the *PIN_check_length* parameter will be set to the PIN length.

The length of the PIN offset, in the returned result, will be determined by the value that the *PIN_check_length* parameter identifies.

data_array

Direction: Input Type: String

Three 16-byte elements. Table 5 describes the format when IBM-PINO is specified. Table 6 describes the format when VISA-PVV is specified.

Table 5. Data Array Elements for the Clear PIN Generate Alternate Service (IBM-PINO)

Array Element	Description
decimalization_table	This element contains the decimalization table of 16 characters (0 to 9) that are used to convert hexadecimal digits (X'0' to X'F') of the enciphered validation data to the decimal digits X'0' to X'9').
validation_data	This element contains 1 to 16 characters of account data. The data must be left justified and padded on the right with space characters.
Reserved-3	This field is ignored, but you must specify it.

Table 6. Data Array Elements for the Clear PIN Generate Alternate Callable Service (VISA-PVV)

Array Element	Description
trans_sec_parm	For VISA-PVV only, the leftmost 12 digits. Eleven digits of the personal account number (PAN). One digit key index. The rest of the field is ignored.
reserved-2	This field is ignored, but you must specify it.
reserved-3	This field is ignored, but you must specify it.

returned_PVV

Direction: Output Type: Character

A 16-byte area that contains the 4-byte PVV left-justified and padded on the right with space characters.

Restrictions

The IBM-PINO PIN calculation method requires the optional PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor.

On CCF systems, to use an IPINENC key, you must install the NOCV-enablement keys in the CKDS.

Usage Notes

On CCF systems, to use an IPINENC key, you must install the NOCV-enablement keys in the CKDS.

The following table lists the PIN block variant constants (PBVC) to use.

Note: PBVC is supported for compatibility with prior releases of OS/390 ICSF and existing ICSF applications. If PBVC is specified in the format control parameter of the PIN profile, the Clear PIN Generate Alternate service will not be routed to a PCI or PCIX Cryptographic Coprocessor for processing. This means that only control vectors and extraction methods valid for the Cryptographic Coprocessor Feature may be used if PBVC formatting is desired. It is recommended that a format control of NONE be used for maximum flexibility.

Restriction: PBVC is not supported on an IBM @server zSeries 990.

Table 7. PIN Block Variant Constants (PBVCs)

PIN Format Name	PIN Block Variant Constant (PBVC)
ECI-2	X'00000000000093000000000000009300'
ECI-3	X'00000000000095000000000000009500'
ISO-0	X'00000000000088000000000000008800'
ISO-1	X'0000000000008B000000000000008B00'
VISA-2	X'0000000000008D000000000000008D00'
VISA-3	X'0000000000008E000000000000008E00'
VISA-4	X'00000000000090000000000000009000'
3621	X'00000000000084000000000000008400'
3624	X'00000000000082000000000000008200'
4704-EPP	X'00000000000087000000000000008700'

The following table lists the required cryptographic hardware for each server type and describes restrictions for this callable service.

Table 8. Clear pin generate alternate required hardware

Server	Required cryptographic hardware	Restrictions
IBM @server zSeries 800 IBM @server zSeries 900	Cryptographic Coprocessor Feature	If PBVC is specified for format control, the request will be routed to a Cryptographic Coprocessor Feature. ICSF routes the request to a PCI Cryptographic Coprocessor if: • The *PIN_encryption_key_identifier* identifies a key which does not have the default PIN encrypting control vector (either IPINENC or OPINENC). • IBM-PINO PIN calculation method is specified. • Anything is specified other than the default in the PIN extraction method keyword for the given PIN block format in *rule_array*.
IBM @server zSeries 990 IBM @server zSeries 890	PCI X Cryptographic Coprocessor Crypto Express2 Coprocessor	Format control in the PIN profile parameter must specify NONE.
IBM System z9 109	Crypto Express2 Coprocessor	Format control in the PIN profile parameter must specify NONE.

Encrypted PIN Generate (CSNBEPG)

The Encrypted PIN Generate callable service formats a PIN and encrypts the PIN block. To generate the PIN, the service uses one of the following PIN calculation methods:

- IBM 3624 PIN
- IBM German Bank Pool Institution PIN
- Interbank PIN

To format the PIN, the service uses one of the following PIN block formats:

- IBM 3621 format
- IBM 3624 format
- ISO-0 format (same as the ANSI X9.8, VISA-1, and ECI-1 formats)
- ISO-1 format (same as the ECI-4 format)
- ISO-2 format
- IBM 4704 encrypting PINPAD (4704-EPP) format
- VISA 2 format
- VISA 3 format
- VISA 4 format
- ECI-2 format
- ECI-3 format

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for formatting an encrypted PIN block into IBM 3621 format or IBM 3624 format. To do this, you must enable the PTR Enhanced PIN Security access control point (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits. No other PIN block consistency checking will occur.

Format

```
CALL CSNBEPG(
             return_code,
             reason_code,
             exit_data_length,
             exit_data,
             PIN_generating_key_identifier,
             outbound_PIN_encrypting_key_identifier
             rule_array_count,
             rule_array,
             PIN_length,
             data_array,
             PIN_profile,
             PAN_data,
             sequence_number,
             encrypted_PIN_block )
```

Parameters

return_code

Direction: Output Type: Integer

The return code specifies the general result of the callable service. Appendix A, “Return and Reason Codes” lists the return codes.

reason_code

Direction: Output Type: Integer

The reason code specifies the result of the callable service that is returned to the application program. Each return code has different reason codes that indicate specific processing problems. Appendix A, “Return and Reason Codes” lists the reason codes.

exit_data_length

Direction: Input/Output Type: Integer

The length of the data, in bytes, that is passed to the installation exit. The length can be from X'00000000' to X'7FFFFFFFF' (2 gigabytes). The data is defined in the *exit_data* parameter.

exit_data

Direction: Input/Output Type: String

The data that is passed to the installation exit.

PIN_generating_key_identifier

Direction: Input/Output Type: String

The 64-byte internal key token or a key label of an internal key token in the CKDS. The internal key token contains the PIN-generating key. The control vector must specify the PINGEN key type and have the EPINGEN usage bit set to 1.

outbound_PIN_encrypting_key_identifier

Direction: Input Type: String

A 64-byte internal key token or a key label of an internal key token in the CKDS. The internal key token contains the key to be used to encrypt the formatted PIN and must contain a control vector that specifies the OPINENC key type and has the EPINGEN usage bit set to 1.

rule_array_count

Direction: Input Type: Integer

The number of 8 byte keywords you are supplying in the *rule_array* parameter. The value must be 1.

rule_array

Direction: Input Type: Character string

Keywords that provide control information to the callable service. Each keyword is left-justified in an 8-byte field, and padded on the right with blanks. All keywords must be in contiguous storage. The rule array keywords are shown as follows:

Table 9. Process Rules for the Encrypted PIN Generate Callable Service

Process Rule	Description
GBP-PIN	This keyword specifies the IBM German Bank Pool Institution PIN calculation method is to be used to generate a PIN.
IBM-PIN	This keyword specifies the IBM 3624 PIN calculation method is to be used to generate a PIN.
INBK-PIN	This keyword specifies the Interbank PIN calculation method is to be used to generate a PIN.

PIN_length

Direction: Input Type: Integer

An integer defining the PIN length for those PIN calculation methods with variable length PINs; otherwise, the variable should be set to zero.

data_array

Direction: Input Type: String

Three 16-byte character strings, which are equivalent to a single 48-byte string. The values in the data array depend on the keyword for the PIN calculation method. Each element is not always used, but you must always declare a complete data array. The numeric characters in each 16-byte string must be from 1 to 16 bytes in length, uppercase, left-justified, and padded on the right with space characters. Table 10 on page 19 describes the array elements.

Table 10. Array Elements for the Encrypted PIN Generate Callable Service

Array Element	Description
Decimalization_table	Decimalization table for IBM and GBP only. Sixteen characters that are used to map the hexadecimal digits (X'0' to X'F') of the encrypted validation data to decimal digits (X'0' to X'9').
Trans_sec_parm	For Interbank only, 16 digits. Eleven right-most digits of the personal account number (PAN). A constant of 6. One digit key selector index. Three digits of PIN validation data.
Validation_data	Validation data for IBM and IBM German Bank Pool padded to 16 bytes. One to sixteen characters of hexadecimal account data left-justified and padded on the right with blanks.

lists the data array elements required by the process rule (*rule_array* parameter). The numbers refer to the process rule's position within the array.

PIN_profile

Direction: Input Type: String array

A 24-byte string containing the PIN profile including the PIN block format.

PAN_data

Direction: Input Type: String

A 12-byte string that contains 12 digits of Personal Account Number (PAN) data. The service uses this parameter if the PIN profile specifies the ISO-0 or VISA-4 keyword for the PIN block format. Otherwise, ensure that this parameter is a 4-byte variable in application storage. The information in this variable will be ignored, but the variable must be specified.

Note: When using the ISO-0 keyword, use the 12 rightmost digits of the PAN data, excluding the check digit. When using the VISA-4 keyword, use the 12 leftmost digits of the PAN data, excluding the check digit.

sequence_number

Direction: Input Type: Integer

The 4-byte string that contains the sequence number used by certain PIN block formats. The service uses this parameter if the PIN profile specifies the 3621 or 4704-EPP keyword for the PIN block format. Otherwise, ensure that this parameter is a 4-byte variable in application data storage. The information in the variable will be ignored, but the variable must be declared. To enter a sequence number, do the following:

- Enter 99999 to use a random sequence number that the service generates.
- For the 3621 PIN block format, enter a value in the range from 0 to 65535.
- For the 4704-EPP PIN block format, enter a value in the range from 0 to 255.

encrypted_PIN_block

Direction: Output Type: String

The field where the service returns the 8-byte encrypted PIN.

Restrictions

The caller must be in task mode, not in SRB mode.

The format control specified in the PIN profile must be NONE. If PBVC is specified as the format control, the service will fail.

Usage Notes

SAF will be invoked to check authorization to use the Encrypted PIN Generate callable service and any key labels specified as input.

The following table lists the required cryptographic hardware for each server type and describes restrictions for this callable service.

Table 11. Encrypted Pin Generate Required Hardware

Server	Required cryptographic hardware	Restrictions
IBM @server zSeries 800 IBM @server zSeries 900	PCI Cryptographic Coprocessor	
IBM @server zSeries 990 IBM @server zSeries 890	PCI X Cryptographic Coprocessor Crypto Express2 Coprocessor	
IBM System z9 109	Crypto Express2 Coprocessor	

Encrypted PIN Translate (CSNBPTR)

Use the Encrypted PIN Translate callable service to reencipher a PIN block from one PIN-encrypting key to another and, optionally, to change the PIN block format, such as the pad digit or sequence number.

The unique-key-per-transaction (UKPT) key derivation for single and double-length keys is available for the Encrypted PIN Translate service. This support is available for the *input_PIN_encrypting_key_identifier* and the *output_PIN_encrypting_key_identifier* parameters for both REFORMAT and TRANSLAT process rules. The rule-array keyword determines which PIN key(s) are derived key(s).

The Encrypted PIN Translate service can be used for unique-key-per-transaction key derivation.

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for formatting an encrypted PIN block into IBM 3621 format or IBM 3624 format. To do this, you must enable the PTR Enhanced PIN Security access control point (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits. No other PIN block consistency checking will occur.

The enhanced PIN security mode also extracts PINs from encrypted PIN blocks. This mode only applies when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. You must enable the Enhanced PIN Security Mode command (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits and a PIN length minimum of 4 is enforced. As with formatting an encrypted PIN block, no other PIN-block consistency checking will occur.

Format

```
CALL CSNBPTR(
             return_code,
             reason_code,
             exit_data_length,
             exit_data,
             input_PIN_encrypting_key_identifier,
             output_PIN_encrypting_key_identifier,
             input_PIN_profile,
             PAN_data_in,
             PIN_block_in,
             rule_array_count,
             rule_array,
             output_PIN_profile,
             PAN_data_out,
             sequence_number,
             PIN_block_out )
```

Parameters

return_code

Direction: Output Type: Integer

The return code specifies the general result of the callable service. Appendix A, “Return and Reason Codes” lists the return codes.

reason_code

Direction: Output Type: Integer

The reason code specifies the result of the callable service that is returned to the application program. Each return code has different reason codes that indicate specific processing problems. Appendix A, “Return and Reason Codes” lists the reason codes.

exit_data_length

Direction: Input/Output Type: Integer

The length of the data, in bytes, that is passed to the installation exit. The length can be from X'00000000' to X'7FFFFFFF' (2 gigabytes). The data is identified in the *exit_data* parameter.

exit_data

Direction: Input/Output Type: String

The data that is passed to the installation exit.

input_PIN_encrypting_key_identifier

Direction: Input/Output Type: String

The input PIN-encrypting key (IPINENC) for the *PIN_block_in* parameter specified as a 64-byte internal key token or a key label. If keyword UKPTOPIN, UKPTBOTH, DUKPT-IP or DUKPT-BH is specified in the *rule_array*, then the *input_PIN_encrypting_key_identifier* must specify a key token or key label of a KEYGENKY key with the UKPT usage bit enabled.

output_PIN_encrypting_key_identifier

Direction: Input/Output Type: String

The output PIN-encrypting key (OPINENC) for the *PIN_block_out* parameter specified as a 64-byte internal key token or a key label. If keyword UKPTOPIN, UKPTBOTH, DUKPT-IP or DUKPT-BH is specified in the *rule_array*, then the *output_PIN_encrypting_key_identifier* must specify a key token or key label of a KEYGENKY with the UKPT usage bit enabled.

input_PIN_profile

Direction: Input Type: Character string

The three 8-byte character elements that contain information necessary to either create a formatted PIN block or extract a PIN from a formatted PIN block. A particular PIN profile can be either an input PIN profile or an output PIN profile depending on whether the PIN block is being enciphered or deciphered by the callable service.

If you choose the TRANSLAT processing rule (this is not enforced on the PCIXCC/CEX2C) in the *rule_array* parameter, the *input_PIN_profile* and the *output_PIN_profile* must specify the same PIN block format. If you choose the REFORMAT processing rule in the *rule_array* parameter, the input PIN profile and output PIN profile can have different PIN block formats. If you specify UKPTIPIN/DUKPT-IP or UKPTBOTH/DUKPT-BH in the *rule_array* parameter, then the *input_PIN_profile* is extended to a 48-byte field and must contain the current key serial number.

The pad digit is needed to extract the PIN from a 3624 or 3621 PIN block in the Encrypted PIN Translate callable service with a process rule (*rule_array* parameter) of REFORMAT. If the process rule is TRANSLAT, the pad digit is ignored.

PAN_data_in

Direction: Input Type: Character string

The personal account number (PAN) if the process rule (*rule_array* parameter) is REFORMAT and the input PIN format is ISO-0 or VISA-4 only. Otherwise, this parameter is ignored. Specify 12 digits of account data in character format.

For ISO-0, use the rightmost 12 digits of the PAN, excluding the check digit.

For VISA-4, use the leftmost 12 digits of the PAN, excluding the check digit.

PIN_block_in

Direction: Input Type: String

The 8-byte enciphered PIN block that contains the PIN to be translated.

rule_array_count

Direction: Input Type: Integer

The number of process rules specified in the *rule_array* parameter. The value can be 1, 2 or 3.

rule_array

Direction: Input Type: Character string

The process rule for the callable service.

Table 12. Keywords for Encrypted PIN Translate Callable Service

Keyword	Meaning
Processing Rules (required)	
REFORMAT	Changes the PIN format, the contents of the PIN block, and the PIN-encrypting key.
TRANSLAT	Changes the PIN-encrypting key only. It does not change the PIN format and the contents of the PIN block.
PIN Block Format and PIN Extraction Method (optional)	See PIN block formats and PIN extraction method keywords. **Note:** If a PIN extraction method is not specified, the first one listed for the PIN block format will be the default.
DUKPT Keywords - Single-length key derivation (optional)	
UKPTIPIN	The *input_PIN_encrypting_key_identifier* is derived as a single-length key. The *input_PIN_encrypting_key_identifier* must be a KEYGENKY key with the UKPT usage bit enabled. The *input_PIN_profile* must be 48 bytes and contain the key serial number.
UKPTOPIN	The *output_PIN_encrypting_key_identifier* is derived as a single-length key. The *output_PIN_encrypting_key_identifier* must be a KEYGENKY key with the UKPT usage bit enabled. The *output_PIN_profile* must be 48 bytes and contain the key serial number.
UKPTBOTH	Both the *input_PIN_encrypting_key_identifier* and the *output_PIN_encrypting_key_identifier* are derived as a single-length key. Both the *input_PIN_encrypting_key_identifier* and the *output_PIN_encrypting_key_identifier* must be KEYGENKY keys with the UKPT usage bit enabled. Both the *input_PIN_profile* and the *output_PIN_profile* must be 48 bytes and contain the respective key serial number.
DUKPT Keywords - Double-length key derivation (optional) - Requires May 2004 or later version of Licensed Internal Code (LIC)	
DUKPT-IP	The *input_PIN_encrypting_key_identifier* is derived as a double-length key. The *input_PIN_encrypting_key_identifier* must be a KEYGENKY key with the UKPT usage bit enabled. The *input_PIN_profile* must be 48 bytes and contain the key serial number.

Table 12. Keywords for Encrypted PIN Translate Callable Service (continued)

Keyword	Meaning
DUKPT-OP	The *output_PIN_encrypting_key_identifier* is derived as a double-length key. The *output_PIN_encrypting_key_identifier* must be a KEYGENKY key with the UKPT usage bit enabled. The *output_PIN_profile* must be 48 bytes and contain the key serial number.
DUKPT-BH	Both the *input_PIN_encrypting_key_identifier* and the *output_PIN_encrypting_key_identifier* are derived as a double-length key. Both the *input_PIN_encrypting_key_identifier* and the *output_PIN_encrypting_key_identifier* must be KEYGENKY keys with the UKPT usage bit enabled. Both the *input_PIN_profile* and the *output_PIN_profile* must be 48 bytes and contain the respective key serial number.

output_PIN_profile

Direction: Input Type: Character string

The three 8-byte character elements that contain information necessary to either create a formatted PIN block or extract a PIN from a formatted PIN block. A particular PIN profile can be either an input PIN profile or an output PIN profile, depending on whether the PIN block is being enciphered or deciphered by the callable service.

- If you choose the TRANSLAT processing rule in the *rule_array* parameter, the *input_PIN_profile* and the *output_PIN_profile* must specify the same PIN block format.
- If you choose the REFORMAT processing rule in the *rule_array* parameter, the input PIN profile and output PIN profile can have different PIN block formats.
- If you specify UKPTOPIN or UKPTBOTH in the *rule_array* parameter, then the *output_PIN_profile* is extended to a 48-byte field and must contain the current key serial number.
- If you specify DUKPT-OP or DUKPT-BH in the *rule_array* parameter, then the *output_PIN_profile* is extended to a 48-byte field and must contain the current key serial number.

PAN_data_out

Direction: Input Type: Character string

The personal account number (PAN) if the process rule (*rule_array* parameter) is REFORMAT and the output PIN format is ISO-0 or VISA-4 only. Otherwise, this parameter is ignored. Specify 12 digits of account data in character format.

For ISO-0, use the rightmost 12 digits of the PAN, excluding the check digit.

For VISA-4, use the leftmost 12 digits of the PAN, excluding the check digit.

sequence_number

Direction: Input Type: Integer

The sequence number if the process rule (*rule_array* parameter) is REFORMAT and the output PIN block format is 3621 or 4704-EPP only. Specify the integer value 99999. Otherwise, this parameter is ignored.

PIN_block_out

Direction: Output Type: String

The 8-byte output PIN block that is reenciphered.

Restriction

Use of the ISO-2 PIN block format requires the optional PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor.

Use of the UKPT keywords require the optional PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor. Use of the DUKPT keyword requires a PCIXCC/CEX2C.

Usage Notes

PIN block formats are more rigorously validated on the IBM @server zSeries 990 than on CCF systems.

Some PIN block formats are known by several names. The following table shows the additional names.

Table 13. Additional Names for PIN Block Formats

PIN Format	Additional Name
ISO-0	ANSI X9.8, VISA format 1, ECI format 1
ISO-1	ECI format 4

The following table lists the PIN block variant constants (PBVC) to be used.

Note: PBVC is NOT supported on the IBM @server zSeries 990. If PBVC is specified in the format control parameter of the PIN profile, the Encrypted PIN Translate service will not be routed to a PCI or PCIX Cryptographic Coprocessor for processing. This means that only control vectors and extraction methods valid for the Cryptographic Coprocessor Feature may be used if PBVC formatting is desired. It is recommended that a format control of NONE be used for maximum flexibility.

Table 14. PIN Block Variant Constants (PBVCs)

PIN Format Name	PIN Block Variant Constant (PBVC)
ECI-2	X'00000000000093000000000000009300'
ECI-3	X'00000000000095000000000000009500'
ISO-0	X'00000000000088000000000000008800'
ISO-1	X'0000000000008B000000000000008B00'
VISA-2	X'0000000000008D000000000000008D00'
VISA-3	X'0000000000008E000000000000008E00'
VISA-4	X'00000000000090000000000000009000'
3621	X'00000000000084000000000000008400'
3624	X'00000000000082000000000000008200'

Table 14. PIN Block Variant Constants (PBVCs) (continued)

PIN Format Name	PIN Block Variant Constant (PBVC)
4704-EPP	X'00000000000087000000000000008700'

The following table lists the required cryptographic hardware for each server type and describes restrictions for this callable service.

Table 15. Encrypted pin translate required hardware

Server	Required cryptographic hardware	Restrictions
IBM @server zSeries 800 IBM @server zSeries 900	Cryptographic Coprocessor Feature	If PBVC is specified for format control, the request will be routed to the Cryptographic Coprocessor Feature.
	PCI Cryptographic Coprocessor	ICSF routes this service to a PCI Cryptographic Coprocessor if: • The control vector in a supplied PIN encrypting key cannot be processed on the Cryptographic Coprocessor Feature. • UKPT support is requested. • The PIN profile specifies the ISO-2 PIN block format. • if the *input_PIN_encrypting_key_identifier* identifies a key which does not have the default input PIN encrypting key control vector (IPINENC) • if the *output_PIN_encrypting_key_identifier* identifies a key which does not have the default output PIN encrypting key control vector (OPINENC) • if anything is specified other than the default in the PIN extraction method keyword for the given PIN block format in *rule_array* DUKPT-IP, DUKPT-OP and DUKPT-BH keywords are not supported.
IBM @server zSeries 990 IBM @server zSeries 890	PCI X Cryptographic Coprocessor Crypto Express2 Coprocessor	Format control in the PIN profile parameter must specify NONE. Use of DUPKT requires Requires May 2004 or later version of Licensed Internal Code (LIC).
IBM System z9 109	Crypto Express2 Coprocessor	Format control in the PIN profile parameter must specify NONE. Use of DUPKT requires Requires May 2004 or later version of Licensed Internal Code (LIC).

Encrypted PIN Verify (CSNBPVR)

Use the Encrypted PIN Verify callable service to verify that one of the following customer-selected trial PINs is valid:

- IBM 3624 (IBM-PIN)
- IBM 3624 PIN offset (IBM-PINO)

- IBM German Bank Pool (GBP-PIN)
- IBM German Bank Pool PIN offset (GBP-PINO) - not supported on the IBM @server zSeries 990
- VISA PIN validation value (VISA-PVV)
- VISA PIN validation value (VISAPVV4)
- Interbank PIN (INBK-PIN)

The unique-key-par-transaction key derivation for single and double-length keys is available for the *input_PIN_encrypting_key_identifier* parameter.

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for extracting PINs from encrypted PIN blocks. This mode only applies when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. To do this, you must enable the PTR Enhanced PIN Security access control point (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits and a PIN length minimum of 4 is enforced. No other PIN-block consistency checking will occur.

Format

```
CALL CSNBPVR(
              return_code,
              reason_code,
              exit_data_length,
              exit_data,
              input_PIN_encrypting_key_identifier,
              PIN_verifying_key_identifier,
              input_PIN_profile,
              PAN_data,
              encrypted_PIN_block,
              rule_array_count,
              rule_array,
              PIN_check_length,
              data_array )
```

Parameters

return_code

Direction: Output Type: Integer

The return code specifies the general result of the callable service. Appendix A, “Return and Reason Codes” lists the return codes.

reason_code

Direction: Output Type: Integer

The reason code specifies the result of the callable service that is returned to the application program. Each return code has different reason codes that indicate specific processing problems. Appendix A, “Return and Reason Codes” lists the reason codes.

exit_data_length

Direction: Input/Output Type: Integer

The length of the data, in bytes, that is passed to the installation exit. The length can be from X'00000000' to X'7FFFFFFF' (2 gigabytes). The data is identified in the *exit_data* parameter.

exit_data

Direction: Input/Output Type: String

The data that is passed to the installation exit.

input_PIN_encrypting_key_identifier

Direction: Input/Output Type: String

The 64-byte key label or internal key token containing the PIN-encrypting key (IPINENC) that enciphers the PIN block. If keyword UKPTIPIN or DUKPT-IP is specified in the *rule_array*, then the *input_PIN_encrypting_key_identifier* must specify a key token or key label of a KEYGENKY with the UKPT usage bit enabled.

PIN_verifying_key_identifier

Direction: Input/Output Type: String

The 64-byte key label or internal key token that identifies the PIN verify (PINVER) key.

input_PIN_profile

Direction: Input Type: Character string

The three 8-byte character elements that contain information necessary to either create a formatted PIN block or extract a PIN from a formatted PIN block. A particular PIN profile can be either an input PIN profile or an output PIN profile depending on whether the PIN block is being enciphered or deciphered by the callable service. If you specify UKPTIPIN in the *rule_array* parameter, then the *input_PIN_profile* is extended to a 48-byte field and must contain the current key serial number.

If you specify DUKPT-IP in the *rule_array* parameter, then the *input_PIN_profile* is extended to a 48-byte field and must contain the current key serial number.

The pad digit is needed to extract the PIN from a 3624 or 3621 PIN block in the encrypted PIN Verify callable service.

PAN_data

Direction: Input Type: Character string

The personal account number (PAN) is required for ISO-0 and VISA-4 only. Otherwise, this parameter is ignored. Specify 12 digits of account data in character format.

For ISO-0, use the rightmost 12 digits of the PAN, excluding the check digit.

For VISA-4, use the leftmost 12 digits of the PAN, excluding the check digit.

encrypted_PIN_block

Direction: Input Type: String

The 8-byte enciphered PIN block that contains the PIN to be verified.

rule_array_count

Direction: Input Type: Integer

The number of process rules specified in the *rule_array* parameter. The value can be 1, 2 or 3.

rule_array

Direction: Input Type: Character string

The process rule for the PIN Verify algorithm.

Table 16. Keywords for Encrypted PIN Verify

Keyword	Meaning
Algorithm Value (required)	
GBP-PIN	The IBM German Bank Pool PIN. It verifies the PIN entered by the customer and compares that PIN with the institution generated PIN by using an institution key.
GBP-PINO	The IBM German Bank Pool PIN offset. It verifies the PIN entered by the customer by comparing with the calculated institution PIN (IPIN) and adding the specified offset to the pool PIN (PPIN) generated by using a pool key. GBP-PINO is not supported on the IBM @server zSeries 990.
IBM-PIN	The IBM 3624 PIN, which is an institution-assigned PIN. It does not calculate the PIN offset.
IBM-PINO	The IBM 3624 PIN offset, which is a customer-selected PIN and calculates the PIN offset.
INBK-PIN	The Interbank PIN verify algorithm.
VISA-PVV	The VISA PIN verify value.
VISAPVV4	The VISA PIN verify value. If the length is 4 digits, normal processing for VISA-PVV will occur. The VISAPVV4 requires a PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor. If one is not available, the service will fail. If the length is greater than 4 digits, the service will fail.
PIN Block Format and PIN Extraction Method (optional)	See PIN block formats and PIN extraction method keywords. **Note:** If a PIN extraction method is not specified, the first one listed for the PIN block format will be the default.
DUKPT Keyword - Single-length key derivation (optional)	
UKPTIPIN	The *input_PIN_encrypting_key_identifier* is derived as a single-length key. The *input_PIN_encrypting_key_identifier* must be a KEYGENKY key with the UKPT usage bit enabled. The *input_PIN_profile* must be 48 bytes and contain the key serial number.
DUKPT Keyword - Double-length key derivation (optional) - Requires May 2004 or later version of Licensed Internal Code (LIC)	

Table 16. Keywords for Encrypted PIN Verify (continued)

Keyword	Meaning
DUKPT-IP	The *input_PIN_encrypting_key_identifier* is to be derived using the DUKPT algorithm. The *input_PIN_encrypting_key_identifier* must be a KEYGENKY key with the DUKPT usage bit enabled. The *input_PIN_profile* must be 48 bytes and contain the key serial number.

PIN_check_length

Direction: Input Type: Integer

The PIN check length for the IBM-PIN or IBM-PINO process rules only. Otherwise, it is ignored. Specify the rightmost digits, 4 through 16, for the PIN to be verified.

data_array

Direction: Input Type: String

Three 16-byte elements required by the corresponding *rule_array* parameter. The data array consists of three 16-byte fields whose specification depend on the process rule. If a process rule only requires one or two 16-byte fields, then the rest of the data array is ignored by the callable service. Table 17 describes the array elements.

Table 17. Array Elements for the Encrypted PIN Verify Callable Service

Array Element	Description
decimalization_table	Decimalization table for IBM and GBP only. Sixteen decimal digits of 0 through 9.
PIN_offset	Offset data for IBM-PINO and GBP-PINO. One to twelve numeric characters, 0 through 9, left-justified and padded on the right with space characters. For IBM-PINO, the PIN offset length is specified in the *PIN_check_length* parameter. For GBP-PINO, the PIN offset is always 4 digits. For IBM-PIN and GBP-PIN, the field is ignored.
trans_sec_parm	For VISA, only the leftmost 12 digits of the 16-byte field are used. These consist of the rightmost 11 digits of the personal account number (PAN) and a one-digit key index. The remaining four characters are ignored. For Interbank only, all 16 bytes are used. These consist of the rightmost 11 digits of the PAN, a constant of X'6', a one-digit key index, and three numeric digits of PIN validation data.
RPVV	For VISA-PVV only, referenced PVV (4 bytes) that is left-justified. The rest of the field is ignored.
validation_data	Validation data for IBM and GBP padded to 16 bytes. One to sixteen characters of hexadecimal account data left-justified and padded on the right with blanks.

lists the data array elements required by the process rule (*rule_array* parameter). The numbers refer to the process rule's position within the array.

Restrictions

GBP-PINO is only supported if the CSNBPVR callable service is processed on the Cryptographic Coprocessor Feature. If the service is routed to a PCI or PCIX Cryptographic Coprocessor, the service request will fail if the GBP-PINO calculation method is specified. GBP-PINO is not supported on the IBM @server zSeries 990 or IBM @server zSeries 890.

Use of the ISO-2 PIN block format requires the optional PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor.

Use of the UKPTIPIN keyword requires the optional PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor.

Use of the VISAPVV4 keyword requires the optional PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor.

Use of the DUKPT-IP keyword requires a PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor.

Usage Notes

PIN block formats are more rigorously validated on the IBM @server zSeries 990 than on CCF systems.

The following table lists the PIN block variant constants (PBVC) to be used.

Note: **Restriction**: PBVC is not supported on an IBM @server zSeries 990. If PBVC is specified in the format control parameter of the PIN profile, the Encrypted PIN Verify service will not be routed to a PCI or PCIX Cryptographic Coprocessor for processing. This means that only control vectors and extraction methods valid for the Cryptographic Coprocessor Feature may be used if PBVC formatting is desired. It is recommended that a format control of NONE be used for maximum flexibility.

Table 18. PIN Block Variant Constants (PBVCs)

PIN Format Name	PIN Block Variant Constant (PBVC)
ECI-2	X'00000000000093000000000000009300'
ECI-3	X'00000000000095000000000000009500'
ISO-0	X'00000000000088000000000000008800'
ISO-1	X'0000000000008B000000000000008B00'
VISA-2	X'0000000000008D000000000000008D00'
VISA-3	X'0000000000008E000000000000008E00'
VISA-4	X'00000000000090000000000000009000'
3621	X'00000000000084000000000000008400'
3624	X'00000000000082000000000000008200'
4704-EPP	X'00000000000087000000000000008700'

The following table lists the required cryptographic hardware for each server type and describes restrictions for this callable service.

Table 19. Encrypted pin verify required hardware

Server	Required cryptographic hardware	Restrictions
IBM @server zSeries 800 IBM @server zSeries 900	Cryptographic Coprocessor Feature	If PBVC is specified for format control, the request will be routed to the Cryptographic Coprocessor Feature.
	PCI Cryptographic Coprocessor	ICSF routes the request to a PCI Cryptographic Coprocessor if: • The PIN profile specifies the ISO-2 PIN block format. • Anything is specified other than the default in the PIN extraction method keyword for the given PIN block format in *rule_array*. • The *input_PIN_encrypting_key_identifier* identifies a key which does not have the default PIN encrypting key control vector (IPINENC). • The *PIN_verifying_key_identifier* identifies a key which does not have the default PIN verify key control vector. • The VISAPVV4 rule array keyword is specified. • You request UKPT support. The DUKPT-IP keyword is not supported.
IBM @server zSeries 990 IBM @server zSeries 890	PCI X Cryptographic Coprocessor Crypto Express2 Coprocessor	Format control in the PIN profile parameter must specify NONE. GBP-PINO rule array parameter is not supported. DUKPT keyword requires May 2004 or later version of Licensed Internal Code (LIC).
IBM System z9 109	Crypto Express2 Coprocessor	Format control in the PIN profile parameter must specify NONE. GBP-PINO rule array parameter is not supported. DUKPT keyword requires May 2004 or later version of Licensed Internal Code (LIC).

Related Information

None.

PIN Change/Unblock (CSNBPCU)

The PIN Change/Unblock callable service is used to generate a special PIN block to change the PIN accepted by an integrated circuit card (smart card). The special PIN block is based on the new PIN and the card-specific diversified key and, optionally, on the current PIN of the smart card. The new PIN block is encrypted with a session key. The session key is derived in a two-step process. First, the card-specific diversified key (ICC Master Key) is derived using the TDES-ENC algorithm of the Diversified Key Generate callable service. The session key is then generated according to the rule array algorithm:

- TDES-XOR - XOR ICC Master Key with the Application Transaction Counter (ATC)
- TDESEMV2 - use the EMV2000 algorithm with a branch factor of 2
- TDESEMV4 - use the EMV2000 algorithm with a branch factor of 4

The generating DKYGENKY key cannot have replicated halves. The *encryption_issuer_master_key_identifier* is a DKYGENKY key that permits generation of a SMPIN key. The *authentication_ issuer_master_key_identifier* is also a DKYGENKY key that permits generation of a double-length MAC key.

The PIN block format is specified by the VISA ICC Card specification: two mutually exclusive rule array keywords, VISAPCU1 and VISAPCU2. They refer to whether the current PIN is used in the generation of the new PIN. For VISAPCU1, it is not used; for VISAPCU2, it is used.

An enhanced PIN security mode, on PCIXCC/CEX2C, is available for extracting PINs from encrypted PIN blocks. This mode only applies when specifying a PIN-extraction method for an IBM 3621 or an IBM 3624 PIN-block. To do this, you must enable the PTR Enhanced PIN Security access control point (offset X'0313') to the active role. When activated this mode limits checking of the PIN to decimal digits and a PIN length minimum of 4 is enforced. No other PIN-block consistency checking will occur.

Format

```
CALL CSNBPCU(
             return_code,
             reason_code,
             exit_data_length,
             exit_data,
             rule_array_count,
             rule_array,
             authentication_issuer_master_key_length,
             authentication_issuer_master_key_identifier,
             encryption_issuer_master_key_length,
             encryption_issuer_master_key_identifier,
             key_generation_data_length,
             key_generation_data,
             new_reference_PIN_key_length,
             new_reference_PIN_key_identifier,
             new_reference_PIN_block,
             new_reference_PIN_profile,
             new_reference_PIN_PAN__data,
             current_reference_PIN_key_length,
             current_reference_PIN_key_identifier,
             current_reference_PIN_block,
             current_reference_PIN_profile,
             current_reference_PIN_PAN__data,
             output_PIN_data_length,
             output_PIN_data,
             output_PIN_profile,
             output_PIN_message_length,
             output_PIN_message )
```

Parameters

return_code

Direction: Output Type: Integer

The return code specifies the general result of the callable service. Appendix A, “Return and Reason Codes” lists the return codes.

reason_code

Direction: Output Type: Integer

The reason code specifies the result of the callable service that is returned to the application program. Each return code has different reason codes assigned to it that indicates specific processing problems. Appendix A, “Return and Reason Codes” lists the reason codes.

exit_data_length

Direction: Input/Output Type: Integer

The length of the data, in bytes, that is passed to the installation exit. The length can be from X'00000000' to X'7FFFFFFF' (2 gigabytes). The data is identified in the *exit_data* parameter.

exit_data

Direction: Input/Output Type: String

The data that is passed to the installation exit.

rule_array_count

Direction: Input Type: Integer

The number of 8 byte keywords you are supplying in the *rule_array* parameter. The valid values are 1 and 2.

rule_array

Direction: Input Type: String

Keywords that provides control information to the callable service. The keywords are left-justified in an 8-byte field and padded on the right with space characters. The keywords must be in contiguous storage. Specify 1 or 2 of the options below:

Table 20. Rule Array Keywords for PIN Change/Unblock

Keyword	Meaning
Algorithm (optional)	
TDES-XOR	TDES encipher clear data to generate the intermediate (card-unique) key, followed by XOR of the final 2 bytes of each key with the ATC counter. This is the default.
TDESEMV2	Same processing as in the Diversified Key Generate service.
TDESEMV4	Same processing as in the Diversified Key Generate service.
PIN processing method (required)	
VISAPCU1	Form the new PIN from the new reference PIN and the intermediate (card-unique) key only.
VISAPCU2	Form the new PIN from the new reference PIN, the intermediate (card-unique) key and the current reference PIN.

authentication_issuer_master_key_length

Direction: Input Type: Integer

The length of the *authentication_issuer_master_key_identifier* parameter. Currently, the value must be 64.

authentication_issuer_master_key_identifier

Direction: Input/Output Type: String

The label name or internal token of a DKYGENKY key type that is to be used to generate the card-unique diversified key. The control vector of this key must be a DKYL0 key that permits the generation of a double-length MAC key (DMAC). This DKYGENKY may not have replicated key halves.

encryption_issuer_master_key_length

Direction: Input Type: Integer

The length of the *encryption_issuer_master_key_identifier* parameter. Currently, the value must be 64.

encryption_issuer_master_key_identifier

Direction: Input/Output Type: String

The label name or internal token of a DKYGENKY key type that is to be used to generate the card-unique diversified key and the secure messaging session key for the protection of the output PIN block. The control vector of this key must be a DKYL0 key that permits the generation of a DMPIN key type. This DKYGENKY key may not have replicated key halves.

key_generation_data_length

Direction: Input Type: Integer

The length of the *key_generation_data* parameter. This value must be 10, 18, 26, or 34 bytes.

key_generation_data

Direction: Input Type: String

The data provided to generate the card-unique session key. For TDES-XOR, this consists of 8 or 16 bytes of data to be processed by TDES to generate the card-unique diversified key followed by a 16 bit ATC counter to offset the card-unique diversified key to form the session key. For TDESEMV2 and TDESEMV4, this may be 10, 18, 26 or 34 bytes.

new_reference_PIN_key_length

Direction: Input Type: Integer

The length of the *new_reference_PIN_key_identifier* parameter. Currently, the value must be 64.

new_reference_PIN_key_identifier

Direction: Input/Output Type: String

The 64-byte label name or internal key token of a PIN encrypting key that is to be used to decrypt the *new_reference_PIN_block*. This must be an IPINENC or OPINENC key. If the label name is supplied, the name must be unique in the CKDS.

new_reference_PIN_block

Direction: Input Type: String

This is an 8-byte field that contains the enciphered PIN block of the new PIN.

new_reference_PIN_profile

Direction: Input Type: String

This is a 24-byte field that contains three 8-byte elements with a PIN block format keyword, a format control keyword (NONE) and a pad digit as required by certain formats.

new_reference_PIN_PAN_data

Direction: Input Type: String

This is a 12-byte field containing a PAN in character format. This data might be needed to recover the new reference PIN if the format is ISO-0 or VISA-4. If neither is used, this parameter may be space characters.

current_reference_PIN_key_length

Direction: Input Type: Integer

The length of the *current_reference_PIN_key_identifier* parameter. For the current implementation, the value must be 64. If the *rule_array* contains VISAPCU1, this value must be 0.

current_reference_PIN_key_identifier

Direction: Input/Output Type: String

The 64-byte label name or internal token of a PIN encrypting key that is to be used to decrypt the *current_reference_PIN_block*. This must be an IPINENC or OPINENC key. If the label name is supplied, the name must be unique on the CKDS. If the *rule_array* contains VISAPCU1, this value is ignored.

current_reference_PIN_block

Direction: Input Type: String

This is an 8-byte field that contains the enciphered PIN block of the new PIN. If the *rule_array* contains VISAPCU1, this value is ignored.

current_reference_PIN_profile

Direction: Input Type: String

This is a 24-byte field that contains three 8-byte elements with a PIN block format keyword, a format control keyword (NONE) and a pad digit as required by certain formats. If the *rule_array* contains VISAPCU1, this value is ignored.

current_reference_PIN_PAN_data

Direction: Input Type: String

This is a 12-byte field containing PAN in character format. This data may be needed to recover the new reference PIN if the format is ISO-0 or VISA-4. If neither is used, this parameter may be blanks. If the *rule_array* contains VISAPCU1, this value is ignored.

output_PIN_data_length

Direction: Input Type: Integer

Currently this field is reserved. The value of this parameter should be 0.

output_PIN_data

Direction: Input Type: String

Currently this field is reserved.

output_PIN_profile

Direction: Input Type: String

This is a 24-byte field that contains three 8-byte elements with a PIN block format keyword (VISAPCU1 or VISCPU2), a format control keyword (NONE) and 8 space characters.

output_PIN_message_length

Direction: Input/Output Type: Integer

The length of the *output_PIN_message* field. Currently the value must be at least 16.

output_PIN_message

Direction: Output Type: String

The reformatted PIN block with the new reference PIN enciphered under the SMPIN session key.

Usage Notes

There are additional access points for this service.

RACF will be invoked to check authorization to use the PIN uhange/unblock service and any label name specified.

The following table lists the required cryptographic hardware for each server type and describes restrictions for this callable service.

PIN Change/Unblock (CSNBPCU)

Table 21. PIN Change/Unblock hardware

Server	Required cryptographic hardware	Restrictions
IBM @server zSeries 800 IBM @server zSeries 900		Not supported
IBM @server zSeries 990 IBM @server zSeries 890	PCI X Cryptographic Coprocessor Crypto Express2 Coprocessor	Requires May 2004 or later version of Licensed Internal Code (LIC)
IBM System z9 109	Crypto Express2 Coprocessor	Requires May 2004 or later version of Licensed Internal Code (LIC)

Appendix A. Return and Reason Codes

Return Codes and Reason Codes

Return Codes

Table 22 has been updated to add return code 12 for the ICSF callable services.

Table 22. Return Codes

Return Code Hex (Decimal)	Description
C (12)	The call to the service could not be processed because ICSF found something wrong in its environment, a cryptographic internal device driver component has determined that data passed, as part of the crypto request from either the CCA application or the user application, was not valid for the request. **User action**: Review the reason code and contact your system programmer or the IBM Support Center.

Reason Codes for Return Code C (12)

Table 23 lists reason codes returned from callable services that give return code 12.

Table 23. Reason Codes for Return Code 12 (12)

Reason Code Hex (Decimal)	Description
4 (4)	ICSF: Your call to an ICSF callable service resulted in an abnormal ending. The request parameter block failed consistency checking. **User action**: Contact your system programmer or the IBM Support Center.
301 (769)	A cryptographic internal device driver component has determined that data passed as part of the crypto request from either the CCA application or the user application was not valid for the request. **User action**: Contact your system programmer or the IBM Support Center. **REASONCODES**: ICSF 4 (4)

Appendix B. Access Control Points and Callable Services for TKE Version 4.0 and Higher

Note: Access control point PTR Enhanced PIN Security is always disabled in the DEFAULT role for all customers (TKE and non-TKE).

A TKE Workstation is required to enable this access control point.

The PTR Enhanced PIN Security access control point will not be enabled in the DEFAULT role. This access control point must be explicitly enabled using the TKE workstation. To activate, you must enable the ACTIVE role for the PTR Enhanced PIN Security access control point (offset X'0313'), which is not the default. If you do not enable this access control point, the previous PIN logic will be used.