Dataset Encryption

Changing the AES master key in a sysplex: procedure and auditing

This document can be found on the web, www.ibm.com/support/techdocs
Search for document number WP##### under the category of "White Papers.

Version Date: November 25, 2017

IBM Systems LBS France

Philippe RICHARD
Consulting IT specialist
Philippe_richard@fr.ibm.com

Introduction, Notes about this document and/or Acknowledgments

This whitepaper is based on a presentation written by Eysha Powers "Load AES MK" at https://www.ibm.com/developerworks/community/wikis/form/api/wiki/be0cb4c9-e5c5-4588-8d23-c896b7ec8ba3/page/a0439101-53d0-41a0-a933-7b32a84f9023/attachment/0393b6bb-7dfc-4ef5-8334-76abb570c410/media/Step4_LoadAESMK.pdf
It was complemented with the description of the work and experiments which were done during a Redbook residency about Dataset Encryption.
We describe our step by step procedure using the ICSF panels in the context of Changing symmetric AES master keys in a sysplex environment.
We also cover the Auditing aspects of a Master key change.
This whitepaper was written by Philippe RICHARD, with the help of all the members who participated in this residency, and reviewed by Eysha S. Powers, Enterprise Cryptography,IBM Systems.
Special thanks to the team who worked on this topic during the residency:

- Thomas LIU (Thomas.Liu@anz.com)
- Brad Habbershaw (habber@ca.ibm.com)
- Andy Coulson (acoulson@uk.ibm.com)

This document is presented "As-Is" and IBM does not assume responsibility for the statements expressed herein. It reflects the opinions of the authors of this whitepaper.
If you have questions about the contents of this document, please direct them to Philippe Richard LBS France (philippe_richard@fr.ibm.com).

Trademarks

Table of Contents

Changing the AES master key in a sysplex: procedure and auditing

In this paper, we outline the steps necessary to change the AES master key. In our environment both the PLEX75 (SC74/SC75) and the standalone system SC60 were originally using the same master key which had been initialized using PPINIT.

For our scenario of transporting and exchanging keys, we had the requirement to have different master keys, so we decided to change the master key of PLEX75. It was also an opportunity for us to document the process of changing master keys, and document our experience of doing this change.

We will describe the procedure we followed, and some of the questions we asked ourselves during this process. We will also describe some problems we encountered and how we solved them and finally discuss the operational considerations for doing a master key change. In addition, we will cover auditing and review the data and information you can collect for auditing purposes.

This document covers the following:

1. Enter the master key parts by using the ICSF Master Key Entry.
2. Initiate Coordinated CKDS Master Key Change
3. Load the New Master Key Registers
4. Reencipher the key data sets under the new master keys. This fills an empty VSAM data set with the reenciphered keys and makes the data set the new key data set. This new reenciphered key data set is a disk copy.
5. Change the new master keys and activate the reenciphered key data sets.
6. Verify the Master Keys are Active
7. Auditing the master key change:

Master Keys are used to protect sensitive cryptographic keys that are active on your system.

Master Keys are stored in secure hardware in the cryptographic feature.

Master Keys are used only to encipher and decipher keys.

Master Keys should be changed periodically.

Changing symmetric master keys in a sysplex environment

Changing the master keys consists of:

1. Loading the new master key registers on all coprocessors on all member of the sysplex sharing the active CKDS.
2. Allocating a new CKDS.
3. Reenciphering the CKDS.
4. Setting the master keys and making the reenciphered CKDS the active CKDS.

Changing the master keys in a sysplex can be done by using the coordinated CKDS change master key utility. After loading the new master keys and allocating the new CKDS, the utility is initiated on one LPAR and all members of the sysplex sharing the CKDS will participate. The CKDS will be reenciphered on the initiating LPAR, all members will refresh to the reenciphered CKDS, and set the master keys. See Symmetric master keys and the CKDS for additional information.

To change the symmetric master keys, you need to:

a. Enter the master key parts into the new master key registers.
b. Reencipher the CKDS under the new master key.
c. Set the symmetric master keys and make the reenciphered CKDS the active CKDS.

Because this procedure branches into different instructions based on whether ICSF is running in noncompatibility, compatibility, or co-existence mode, you should first understand the following background information on these modes before referring to and performing the procedure.

ICSF runs in noncompatibility, compatibility, or co-existence mode with the IBM cryptographic products, and Programmed Cryptographic Facility (PCF). You specify which mode ICSF runs in by using an installation option. For a description of the modes and how to specify an installation option, see z/OS Cryptographic Services ICSF System Programmer's Guide.

In noncompatibility mode, ICSF allows you to change the master key with continuous operations. Therefore, applications can continue to run without disruption. However, when ICSF is in compatibility mode or co-existence mode, you should use a different procedure to activate the changed master key. This is to ensure that no application is holding an internal token with the wrong master key.

In all three modes, you enter the new master key and reencipher the disk copy of the CKDS under the new master key using the master key panels. In noncompatibility mode, you then set the new master key and refresh the in-storage copy of the CKDS with the disk copy using the master key panels or a utility program.

In our environment we run in non compatibility mode

```
IEF695I START CSF      WITH JOBNAME CSF      IS ASSIGNED TO U
$HASP373 CSF      STARTED
CSFO0230 CKDSN(PLEX75.SHARED.SCSFCKDS)
CSFO0230 PKDSN(PLEX75.SHARED.SCSFPKDS)
CSFO0230 TKDSN(PLEX75.SHARED.SCSFTKDS)
CSFO0230 SYSPLEXCKDS(YES,FAIL(YES))
CSFO0230 SYSPLEXTKDS(YES,FAIL(YES))
CSFO0230 SYSPLEXPKDS(YES,FAIL(YES))
CSFO0230 COMPAT(NO)
CSFO0230 SSM(YES)
CSFO0230 CHECKAUTH(NO)
CSFO0230 USERPARM(USERPARM)
CSFO0230 REASONCODES(ICSF)
CSFO0166 DEFAULT CICS WAIT LIST WILL BE USED.
IEE252I MEMBER CTICSF00 FOUND IN SYS1.IBM.PARMLIB
CSFM608I A CKDS KEY STORE POLICY IS DEFINED.
CSFM608I A PKDS KEY STORE POLICY IS DEFINED.
CSFM610I GRANULAR KEYLABEL ACCESS CONTROL IS ENABLED.
CSFM611I XCSFKEY EXPORT CONTROL FOR AES IS ENABLED.
CSFM611I XCSFKEY EXPORT CONTROL FOR DES IS ENABLED.
CSFM612I PKA KEY EXTENSIONS CONTROL IS DISABLED.
CSFM654I KEY ARCHIVING USE CONTROL IS DISABLED.
CSFM600I CONNECTION ESTABLISHED TO ICSF SYSPLEX GROUP SYSICSF
CSFM639I ICSF COMMUNICATION LEVEL FOR CKDS CHANGED FROM 0 TO
CSFM653I CKDS LOADED 9 RECORDS WITH AVERAGE SIZE 253
CSFM600I CONNECTION ESTABLISHED TO ICSF SYSPLEX GROUP SYSICSF
CSFM639I ICSF COMMUNICATION LEVEL FOR PKDS CHANGED FROM 0 TO
CSFM653I PKDS LOADED 6 RECORDS WITH AVERAGE SIZE 425
CSFC0322 DUPLICATE TOKENS FOUND IN DATASET PLEX75.SHARED.SCSF
CSFM600I CONNECTION ESTABLISHED TO ICSF SYSPLEX GROUP SYSICSF
CSFM639I ICSF COMMUNICATION LEVEL FOR TKDS CHANGED FROM 0 TO
CSFM129I MASTER KEY DES ON CRYPTO EXPRESS6 COPROCESSOR 6C00,
CSFM129I MASTER KEY AES ON CRYPTO EXPRESS6 COPROCESSOR 6C00,
CSFM129I MASTER KEY RSA ON CRYPTO EXPRESS6 COPROCESSOR 6C00,
CSFM129I MASTER KEY ECC ON CRYPTO EXPRESS6 COPROCESSOR 6C00,
CSFM111I CRYPTOGRAPHIC FEATURE IS ACTIVE. CRYPTO EXPRESS6 COP
CSFM111I CRYPTOGRAPHIC FEATURE IS ACTIVE. CRYPTO EXPRESS6 ACC
CSFM130I CRYPTOGRAPHY - RSA SERVICES ARE AVAILABLE.
CSFM130I CRYPTOGRAPHY - ECC SERVICES ARE AVAILABLE.
CSFM133I THERE ARE NO ACTIVE PKCS11 COPROCESSORS.
CSFM015I FIPS 140 SELF CHECKS FOR PKCS11 SERVICES SUCCESSFUL.
CSFM400I CRYPTOGRAPHY - SERVICES ARE NOW AVAILABLE.
CSFM130I CRYPTOGRAPHY - DES SERVICES ARE AVAILABLE.
CSFM127I CRYPTOGRAPHY - AES SERVICES ARE AVAILABLE.
```

```
CSFM126I CRYPTOGRAPHY - FULL CPU-BASED SERVICES ARE AVAILABLE
CSFM001I ICSF INITIALIZATION COMPLETE
CSFM640I ICSF RELEASE FMID=HCR77C1.
```

As you can see above we have set COMPAT(NO)

Our ICSF parameter member CSFPRM03 contains:

```
BROWSE    SYS1.PARMLIB(CSFPRM03) -
 Command ===>
*******************************
CKDSN(PLEX75.SHARED.SCSFCKDS)
PKDSN(PLEX75.SHARED.SCSFPKDS)
TKDSN(PLEX75.SHARED.SCSFTKDS)
SYSPLEXCKDS(YES,FAIL(YES))
SYSPLEXTKDS(YES,FAIL(YES))
SYSPLEXPKDS(YES,FAIL(YES))
COMPAT(NO)
SSM(YES)
CHECKAUTH(NO)
USERPARM(USERPARM)
REASONCODES(ICSF)
```

Sysplex communication level

There are several communication protocols for sysplex support for the key data set. ICSF uses the level protocol of the lowest release of ICSF of the systems that are sharing a particular key data set. The newer protocols provide performance enhancements for update processing in a sysplex environment.

For the coordinated administration utilities, the minimum level is 2. This is the level that was introduced in ICSF FMID HCR7790 for the CKDS and in ICSF FMID HCR77A0 for the PKDS and TKDS. All systems in the sysplex, regardless of their active key data set, must be at the minimum level or higher in order for ICSF to use this new protocol.

To determine what protocol ICSF is using, check the ICSF job log for the following message:

```
CSFM639I ICSF COMMUNICATION LEVEL FOR CKDS CHANGED FROM 0 TO 3.
```

If you are using ICSF FMID HCR77B1 or later, the DISPLAY ICSF operator command can be used to display current communication level.

```
D ICSF,KDS

RESPONSE=SC74

 CSFM668I 13.56.23 ICSF KDS
917
   CKDS  PLEX75.SHARED.SCSFCKDS
```

```
     FORMAT=KDSR      COMM LVL=3  SYSPLEX=Y  MKVPs=DES
AES
   PKDS  PLEX75.SHARED.SCSFPKDS

     FORMAT=KDSR      COMM LVL=3  SYSPLEX=Y  MKVPs=RSA
ECC
   TKDS  PLEX75.SHARED.SCSFTKDS

     FORMAT=VARIABLE  COMM
LVL=3  SYSPLEX=Y  MKVPs=None
```

Coordinated change master key and coordinated refresh utilities

There are utilities that coordinate key data set refreshes and master key changes across sysplex members sharing the same active key data set. The coordinated administration functions simplify key data set management by automating the manual process for performing local refreshes and local master key changes. Although a sysplex environment is not required to use these functions, sysplex environments gain the maximum benefit from them when the changes are coordinated across all LPARs sharing the same active key data set.

The utilities are initiated from a single ICSF LPAR. This LPAR drives the operation across the sysplex by using sysplex messaging to other members sharing the same active key data set. Only one coordinated administration function may be performed at a time.

Coordinated KDS refresh

The coordinated KDS refresh utility, for the CKDS and PKDS only, drives the initiating system to send sysplex messages to all sysplex members sharing the same active key data set, instructing them to either refresh their in-store key data set copy of the active key data set, or refresh their in-store key data set copy to a new key data set. Performing a coordinated refresh to a new key data set results in the new key data set becoming the active key data set for all sysplex members in this key data set sysplex cluster.

Coordinated KDS change master key

> The master keys are loaded into the new master key register. The coordinated KDS change master key utility that is described in this section requires the master key to be in the new master key register and not set. Newer versions of the TKE workstation allow the master keys to be set from the TKE workstation. This utility fails if the master key is not in the new master key register.

The coordinated KDS change master key utility, for the CKDS, PKDS, and TKDS, reenciphers the active key data set disk-copy to a new key data set using the master key values into the new master key registers. Before performing the coordinated change master key function, you must load the new master key registers.

For CKDS, the coordinated change master key utility can be used to change the DES master key, the AES master key, or both.

After reenciphering the active key data set disk-copy, the initiating system will send sysplex messages to the other members sharing the same active key data set to inform them to re-load their in-store key data set from the new reenciphered key data set. Next, the initiating system sets the master keys for the new master key registers and make the new key data set the active key data set.

Finally, the initiating system sends sysplex messages to the other members of the key data set sysplex cluster to inform them to set their master keys for the new master key registers and to make the new key data set their active key data set.

Dynamic KDS update controls

When performing a coordinated change master key on the CKDS, PKDS, or TKDS, it is not required to disable dynamic KDS updates within the sysplex while performing a coordinated change master key. This is an enhancement over the local master key change functions, for which disallowing dynamic KDS update services is recommended.

During a coordinated change master key, dynamic key data set update requests are routed to, and processed by, the ICSF instance that initiated the coordinated change master key. The initiator processes dynamic key data set updates against the active key data set during the coordinated change master key. When the initiating system has reenciphered the key data set, and before it coordinates the key data set master key change across the sysplex, there is a brief suspension to dynamic KDS update processing. During this brief suspension, dynamic key data set updates that were processed by the initiator are applied to the new reenciphered key data set.

If you cannot tolerate a temporary suspension of dynamic KDS update services in your workload while processing a coordinated change master key and would prefer that update requests are failed instead, you should disallow dynamic KDS access prior to performing coordinated change master key.

For a coordinated CKDS and PKDS refresh, dynamic KDS update processing is internally suspended by the initiator until the coordinated refresh completes. However, IBM still recommends that you disallow dynamic access prior to performing a coordinated refresh.

Before performing any master key change, you should ensure that your KDS has no key error, using the Key check utility

Key check utility

Note: The key check utility is available on IBM z13 and later servers.

The key check utility validates each key in the key data set in the same manner as the KDS reencipher utility without reenciphering the key. The utility can be run prior to reenciphering on a key data set to ensure the KDS reencipher and change master key utilities will complete successfully. The utility checks the active key data sets. The results of the check are written to the ICSF joblog.

The key check utility can be invoked from the CKDS Management and PKDS Management panels or by writing an application to invoke the ICSF Multi-Purpose Service (CSFMPS) callable service. For more information, see z/OS Cryptographic Services ICSF Application Programmer's Guide.

Steps for checking the CKDS

1. Enter option 2, KDS MANAGEMENT, on the ICSF Primary Menu panel: ICSF Primary Menu panel to access the Key Data Set Management panel: CSFMKM10 — Key Data Set Management panel
2. Enter option 1, CKDS MK MANAGEMENT and the CKDS Management panel appears: CSFMKM20 — CKDS Management panel
3. Enter option 7 for CKDS KEY CHECK to run the utility on the active CKDS.

The panel message will indicate if the check was successful or if there are errors. CSFM661I messages are written to the ICSF joblog with warnings and errors.

When we executed the key check utility, it returned 'CHECK SUCCESSFUL', as shown below.

```
--------------------------- ICSF - CKDS Management --------- CHECK SUCCESSFUL
 OPTION ===> 7

 Enter the number of the desired option.

   1  CKDS OPERATIONS    -  Initialize a CKDS, activate a different CKDS,
                            (Refresh), or update the header of a CKDS and make
                            it active
   2  REENCIPHER CKDS    -  Reencipher the CKDS prior to changing a symmetric
                            master key
   3  CHANGE SYM MK      -  Change a symmetric master key and activate the
                            reenciphered CKDS
   4  COORDINATED CKDS REFRESH - Perform a coordinated CKDS refresh
   5  COORDINATED CKDS CHANGE MK - Perform a coordinated CKDS change master key
   6  COORDINATED CKDS CONVERSION - Convert the CKDS to use KDSR record format
   7  CKDS KEY CHECK     - Check key tokens in the active CKDS for format errors

 Press ENTER to go to the selected option.
 Press END   to exit to the previous menu.
```

The To-Do List…

__ 1. Step 1: Allocate New Key Data Sets

__ 2. Step 2: Load the New Master Key Registers

__ 3. Step 3: Initiate Coordinated CKDS Master Key Change

__ 4. Step 4: Verify the Master Keys are Active

Step 1 in our ToDo list: Allocate New Key Data Sets:

Allocate new key data sets. For example, for the CKDS …

The current / active key data set containing the existing keys is

```
PLEX75.SHARED.SCSFCKDS
```

The new key data setto contain the reenciphered keys will be

```
PLEX75.SHARED3..SCSFCKDS
```

The backup key data set to contain the reenciphered keys will be

```
PLEX75.SHARED3.COPY.SCSFCKDS
```

We allocated the new CKDS datasets (SHARED3 and SHARED3.COPY) using the following JCL (based from SYS1.SAMPLIB(CSFCKD3))

```
//DEFINE  EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DEFINE CLUSTER (NAME(PLEX75.SHARED3.COPY.SCSFCKDS)   -
                  VOLUMES(BH5CAT)              -
                  RECORDS(100 50)              -
                  RECORDSIZE(372,2048)         -
                  KEYS(72 0)                   -
                  FREESPACE(10,10)             -
                  SHAREOPTIONS(2,3))           -
           DATA (NAME(PLEX75.SHARED3.COPY.SCSFCKDS.DATA)      -
                  BUFFERSPACE(100000)          -
                  ERASE                        -
                  WRITECHECK)                  -
         INDEX (NAME(PLEX75.SHARED3.COPY.SCSFCKDS.INDEX))
/*
```

The To-Do List…

Step 1: Allocate New Key Data Sets

Step 2: **Load the New Master Key Registers**

Step 3: Initiate Coordinated CKDS Master Key Change

Step 4: Verify the Master Keys are Active

Step 2 in our todo list will consist in the following steps:

__ 1. Step 1: Generate a random number for the AES Master Key Part

__ 2. Step 2: Generate a checksum for the AES Master Key Part

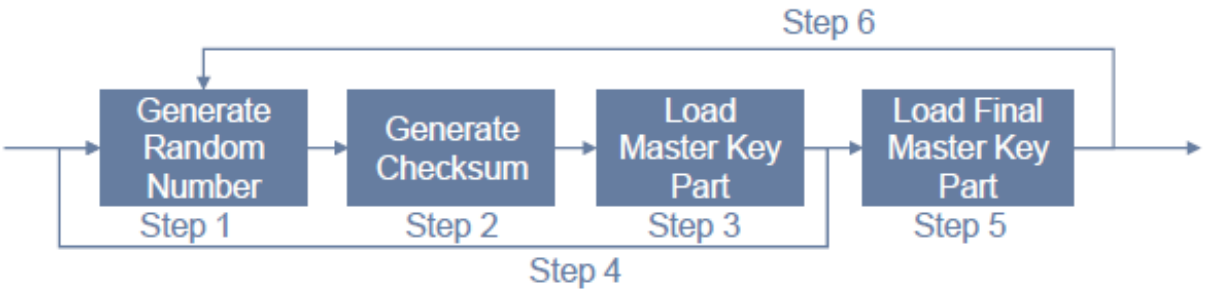
__ 3. Step 3: Load the first AES Master Key Part

__ 4. Step 4: Repeat Steps 1 -3 for the desired number of middle key parts

__ 5. Step 5: Load the final AES Master Key Part

__ 6. Step 6: Generate and load the remaining Master Keys

__ 7. Step 7: Verify the new Master Key Registers



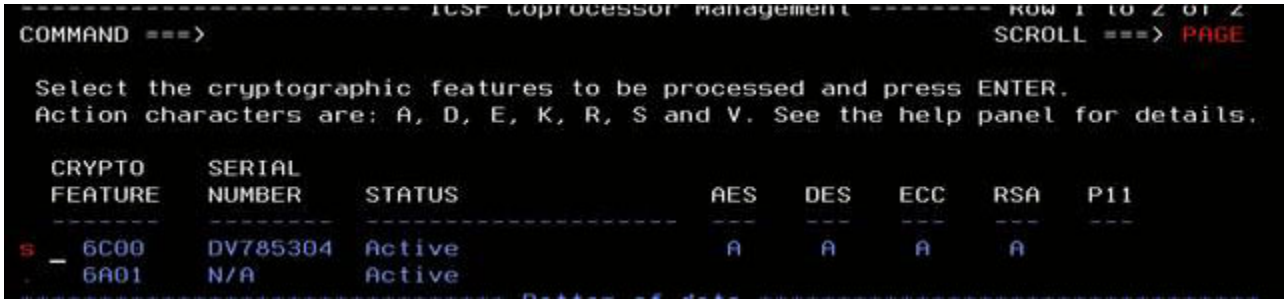
WARNING : **the screenshots show a change of MASTER KEY for AES, but the process is the same for all other types of Master keys (DES, RSA,ECC)**

Step 1: Generate a random number for the **AES Master Key Part (1 of 7)**

From the ICSF Panels, choose Option 5 -Utility

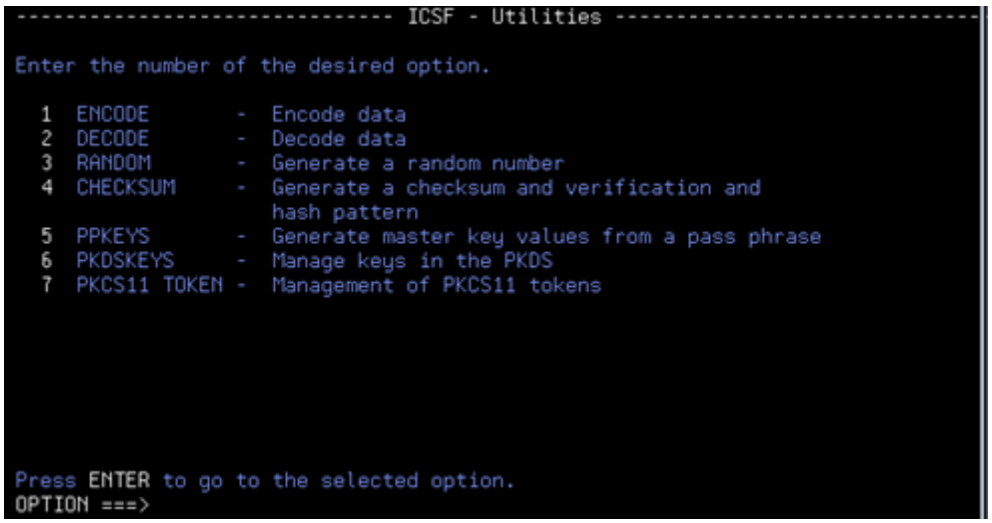
```
HCR77C1  --------------- Integrated Cryptographic Service Facility -------
OPTION ===>
System Name:  SC74                              Crypto Domain: 3
Enter the number of the desired option.

  1  COPROCESSOR MGMT -  Management of Cryptographic Coprocessors
  2  KDS MANAGEMENT   -  Master key set or change, KDS Processing
  3  OPSTAT           -  Installation options
  4  ADMINCNTL        -  Administrative Control Functions
  5  UTILITY          -  ICSF Utilities
  6  PPINIT           -  Pass Phrase Master Key/KDS Initialization
  7  TKE              -  TKE PKA Direct Key Load
  8  KGUP             -  Key Generator Utility processes
  9  UDX MGMT         -  Management of User Defined Extensions

     Licensed Materials - Property of IBM
     5650-ZOS Copyright IBM Corp. 1989, 2017.
     US Government Users Restricted Rights - Use, duplication or
     disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Press ENTER to go to the selected option.
Press END   to exit to the previous menu.
```

Check that the new Master key register is currently empty



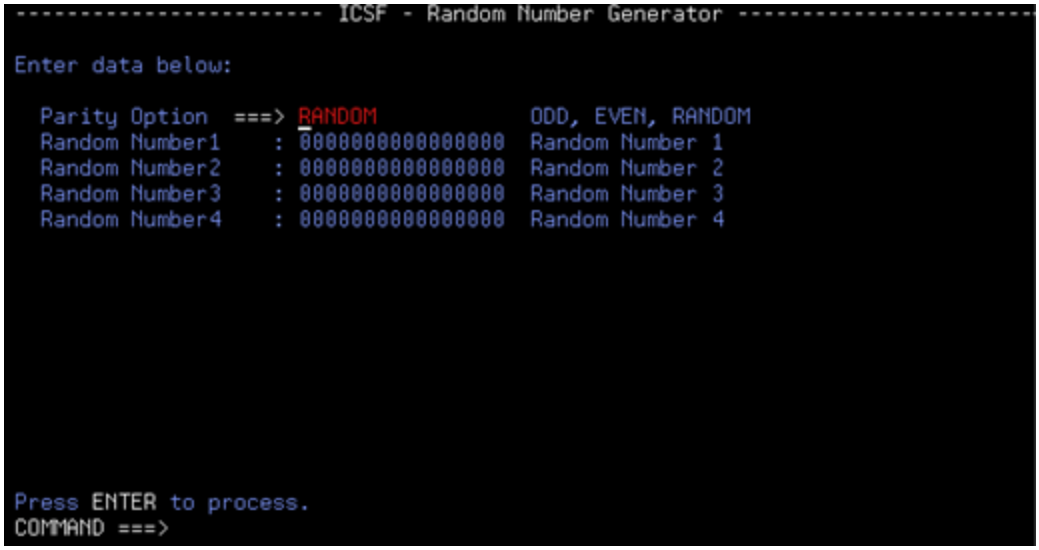
Step 1: Generate a random number for the **AES Master Key Part (2 of 7)**

Choose Option 3 -Random



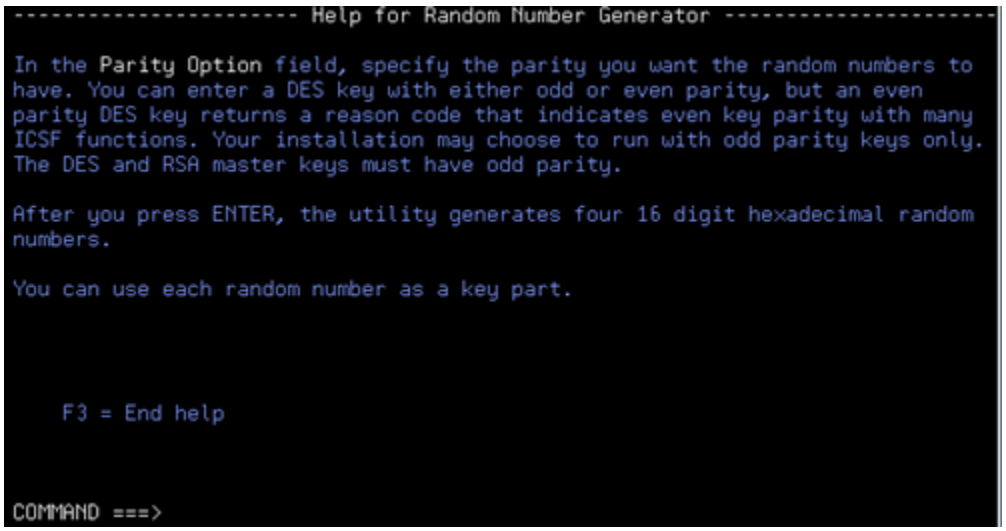
Step 1: Generate a random number for the **AES Master Key Part (3 of 7)**

View the Random Number Generator (RNG) Panel



Step 1: Generate a random number for the **AES Master Key Part (4 of 7)**

Press the <F1> key for Help



Step 1: Generate a random number for the **AES Master Key Part (5 of 7)**

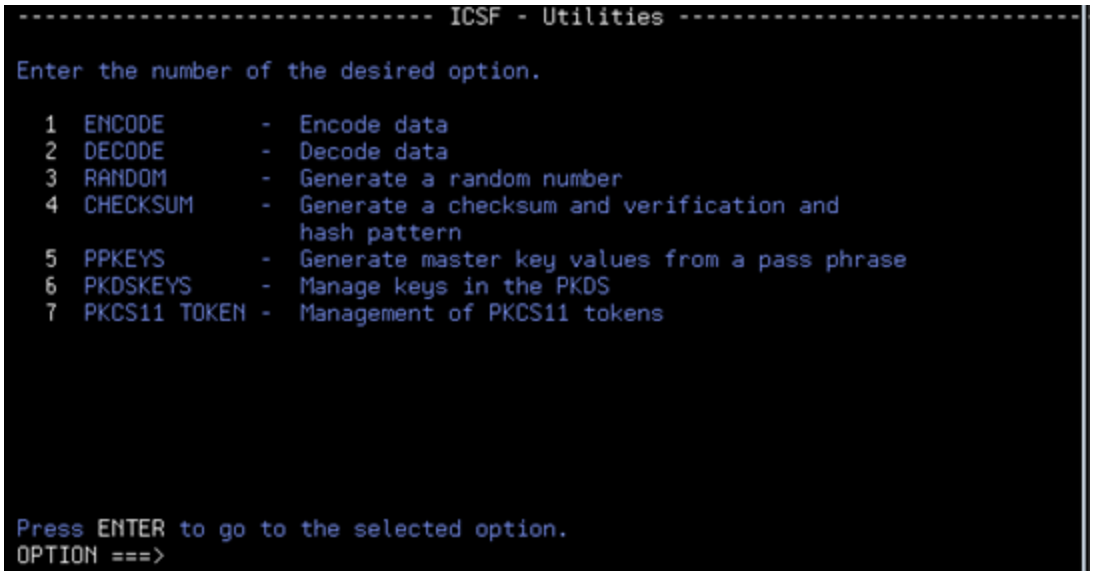
Press <F3> to return to the RNG Panel

Step 1: Generate a random number for the **AES Master Key Part (6 of 7)**

Type “RANDOM” and press <ENTER>

Step 1: Generate a random number for the **AES Master Key Part (7 of 7)**

Press <F3> to return to the Utilities Panel

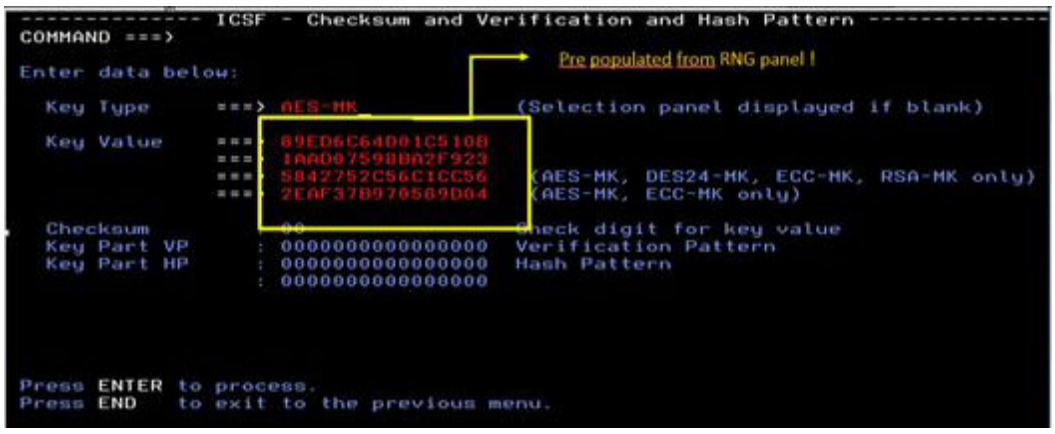


Step 2: Generate a checksum for the **AES Master Key Part (1 of 9)**

Choose Option 4 –Checksum

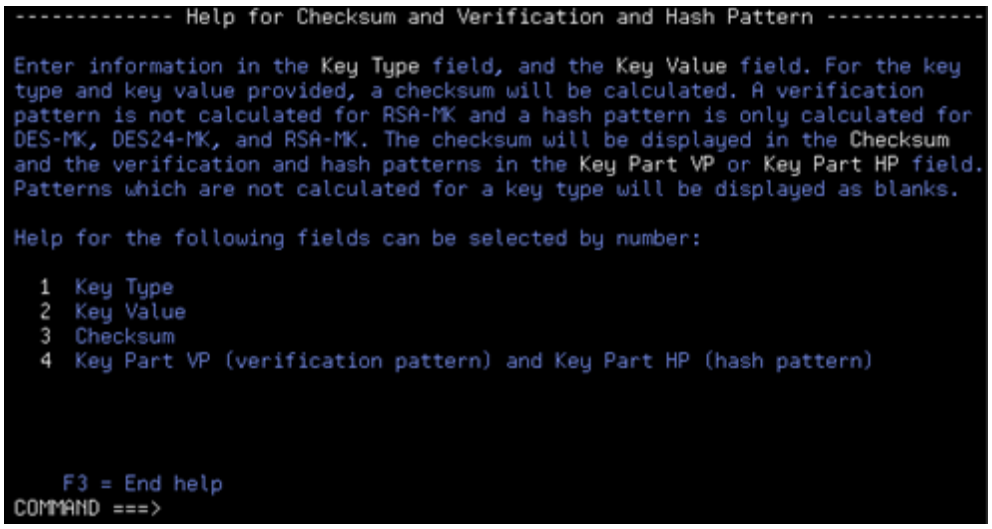
Step 2: Generate a checksum for the **AES Master Key Part (2 of 9)**

View the Checksum Panel



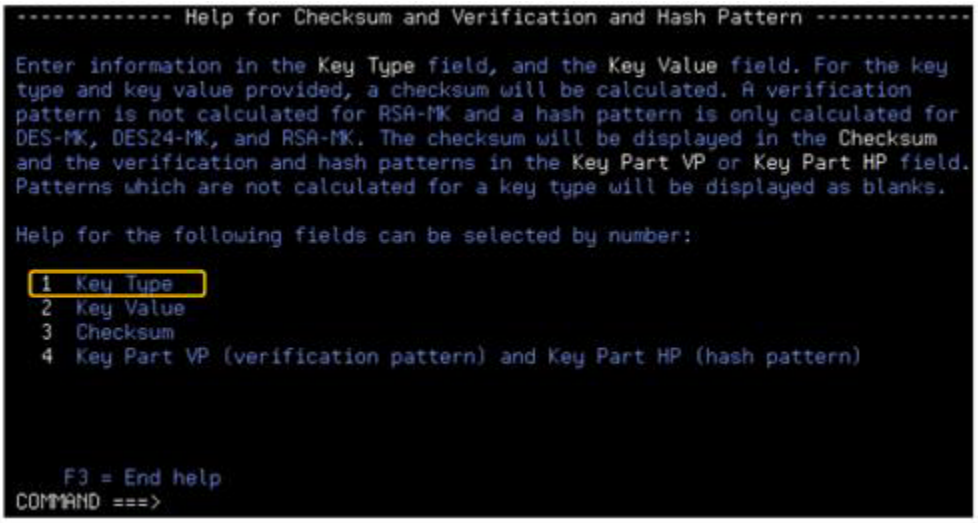
Step 2: Generate a checksum for the **AES Master Key Part (3 of 9)**

Press <F1> for Help



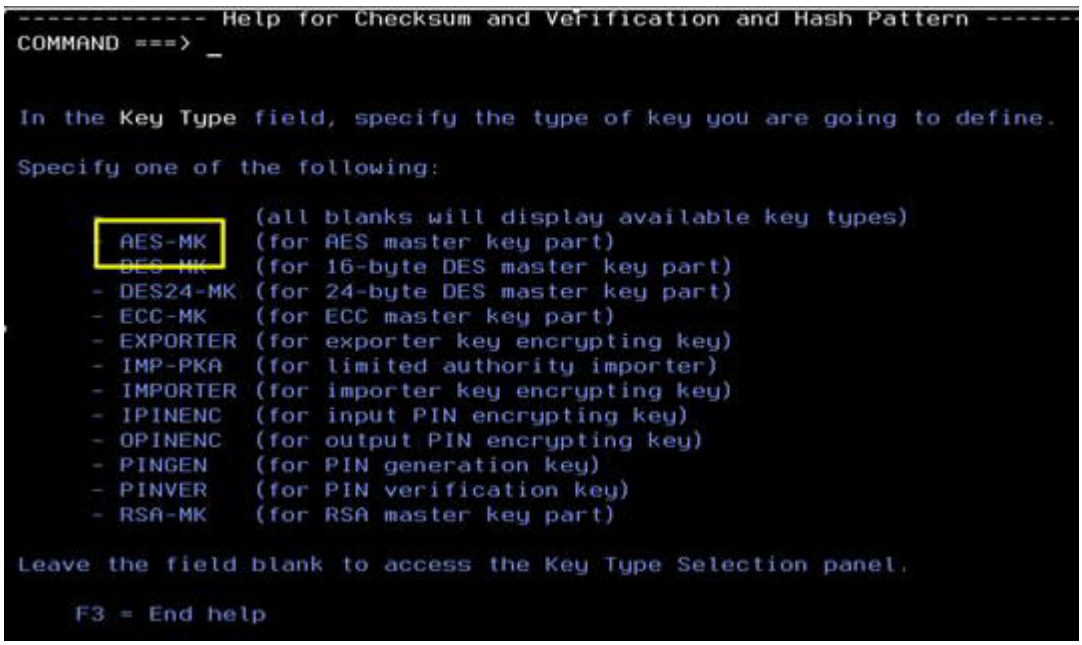
Step 2: Generate a checksum for the **AES Master Key Part (4 of 9)**

Choose Option 1 for Key Type



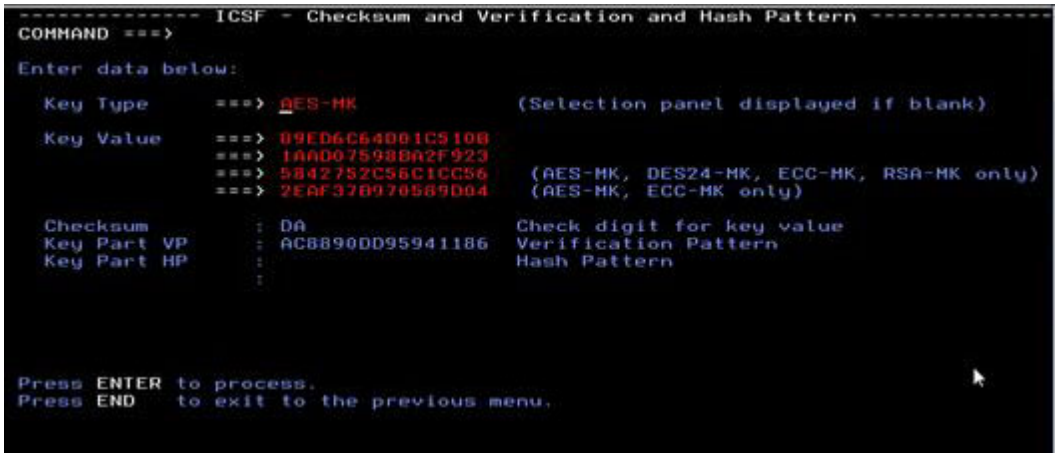
Step 2: Generate a checksum for the **AES Master Key Part (5 of 9)**

View the possible key type values and lengths



Step 2: Generate a checksum for the **AES Master Key Part (6 of 9)**

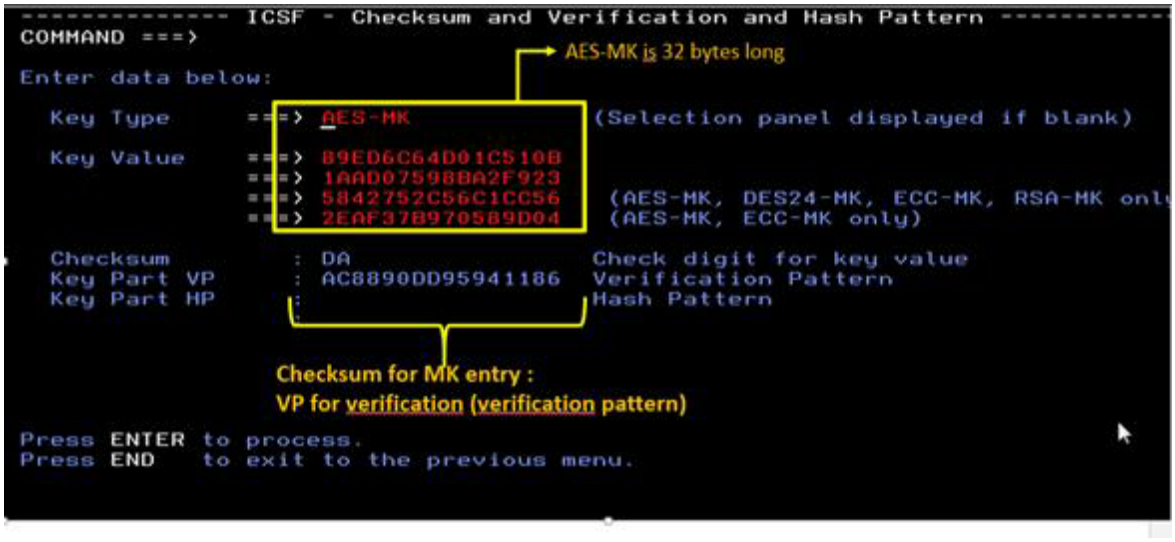
Press <F3> to return to the Checksum Panel



Step 2: Generate a checksum for the **AES Master Key Part (7 of 9)**

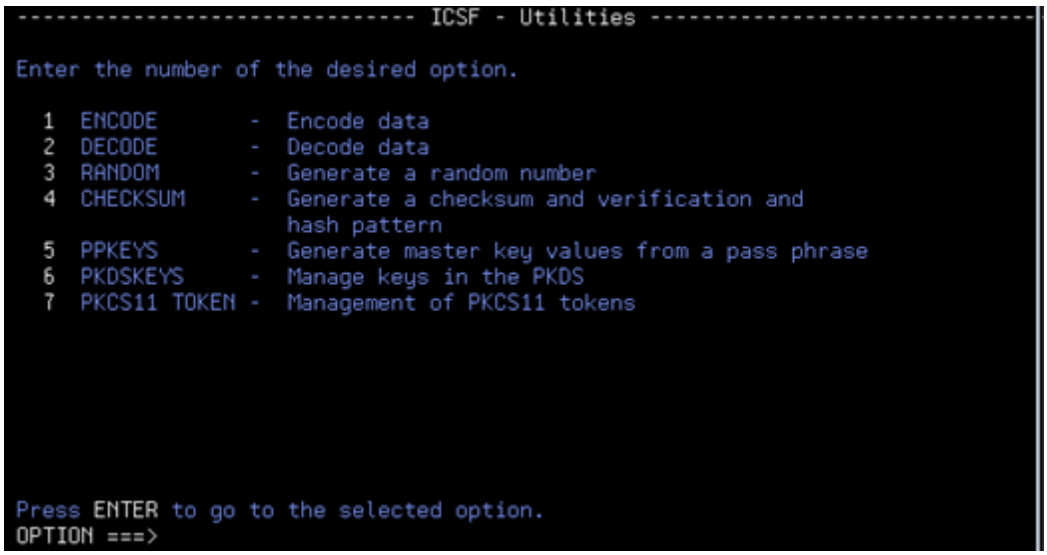
Type " AES-MK" for the key type and hit <ENTER>

(note: you would use AES-MK for dataset encryption)



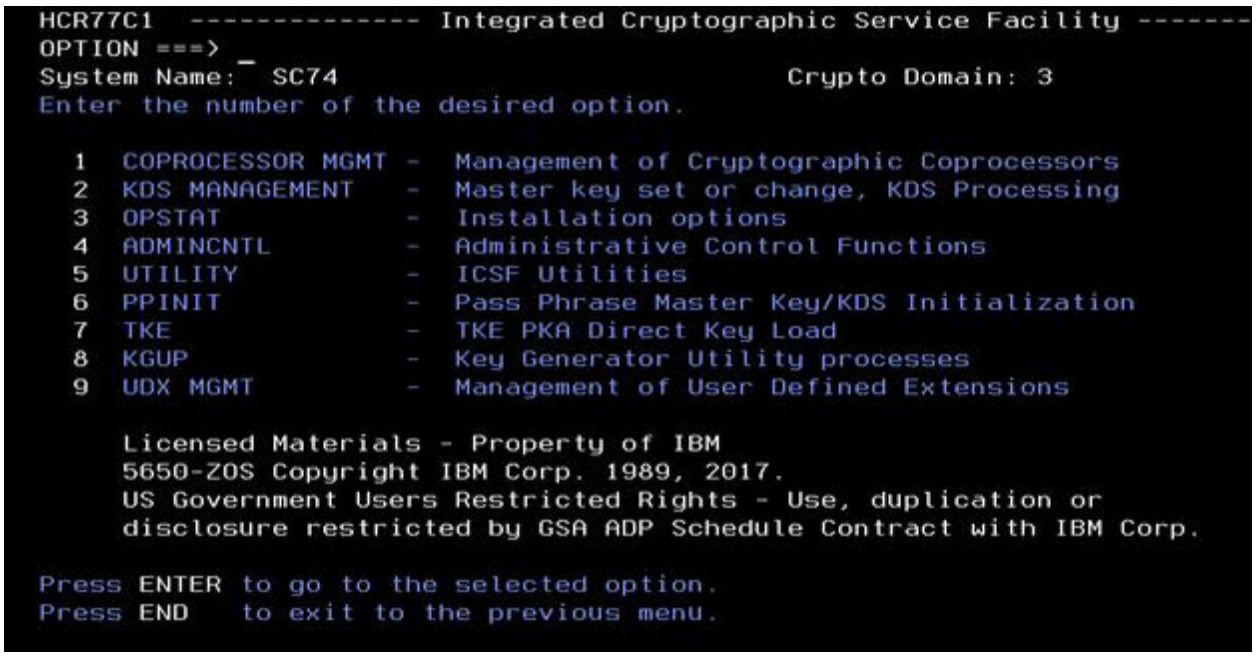
Step 2: Generate a checksum for the **AES Master Key Part (8 of 9)**

Press <F3> to return to the Utility Panel



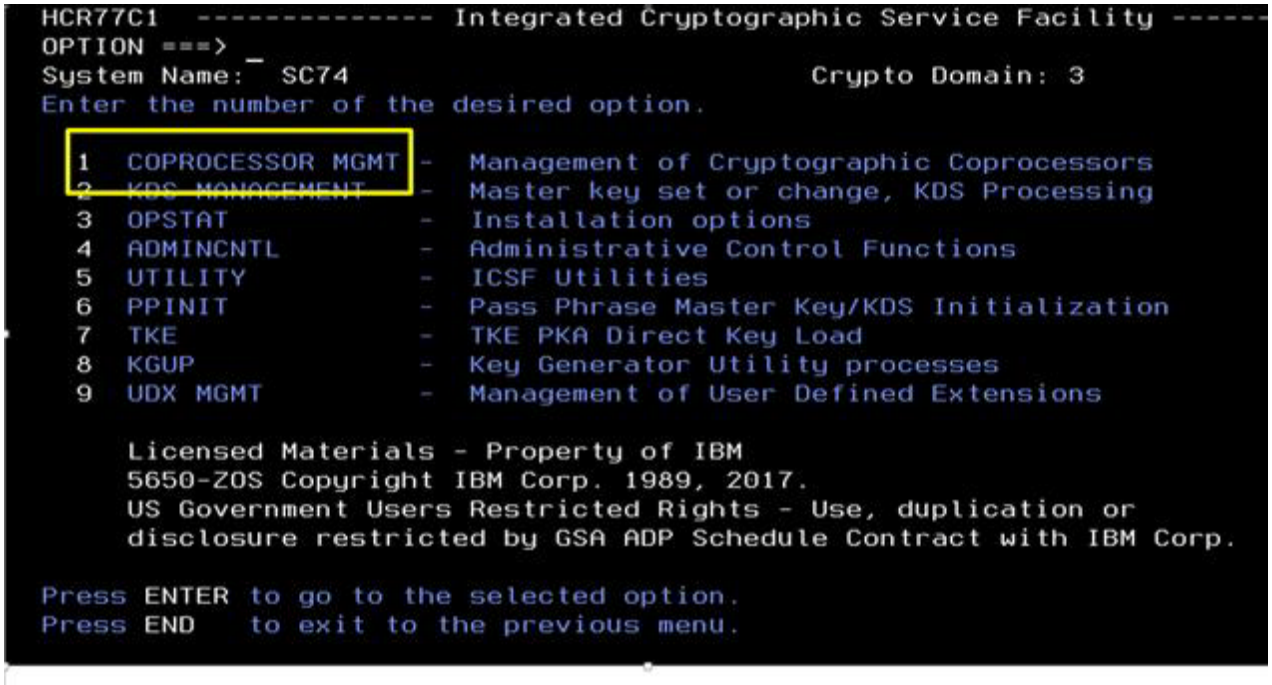
Step 2: Generate a checksum for the **AES Master Key Part (9 of 9)**

Press <F3> to return to the Main ICSF Panel



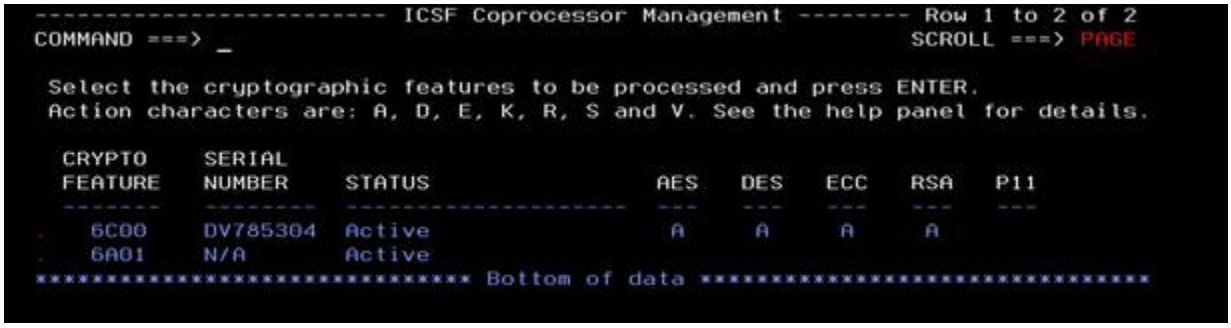
Step 3: Load the **AES Master Key Part (1 of 12)**

Choose Option 1 –Coprocessor Mgmt



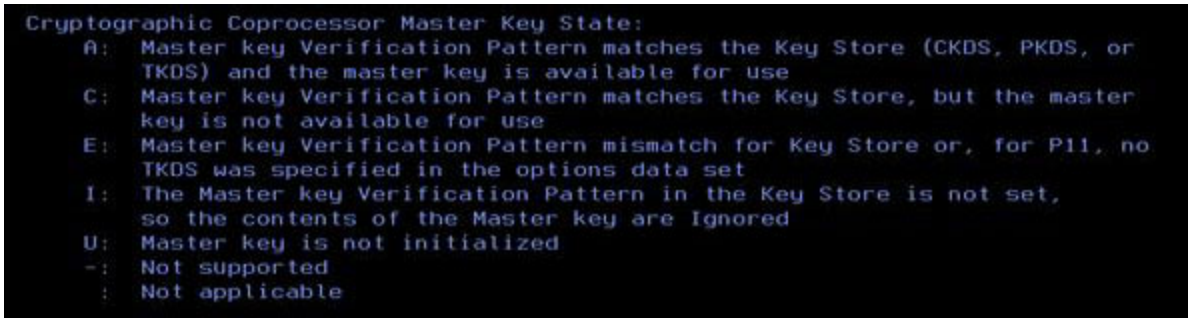
Step 3: Load the **AES Master Key Part (2 of 12)**

View the Coprocessor Management Panel



Step 3: Load the **AES Master Key Part (3 of 12)**

Press <F1> for Help, scroll down (i.e. press <ENTER>) to see the Master Key State definitions



Step 3: Load the **AES Master Key Part (4 of 12)**

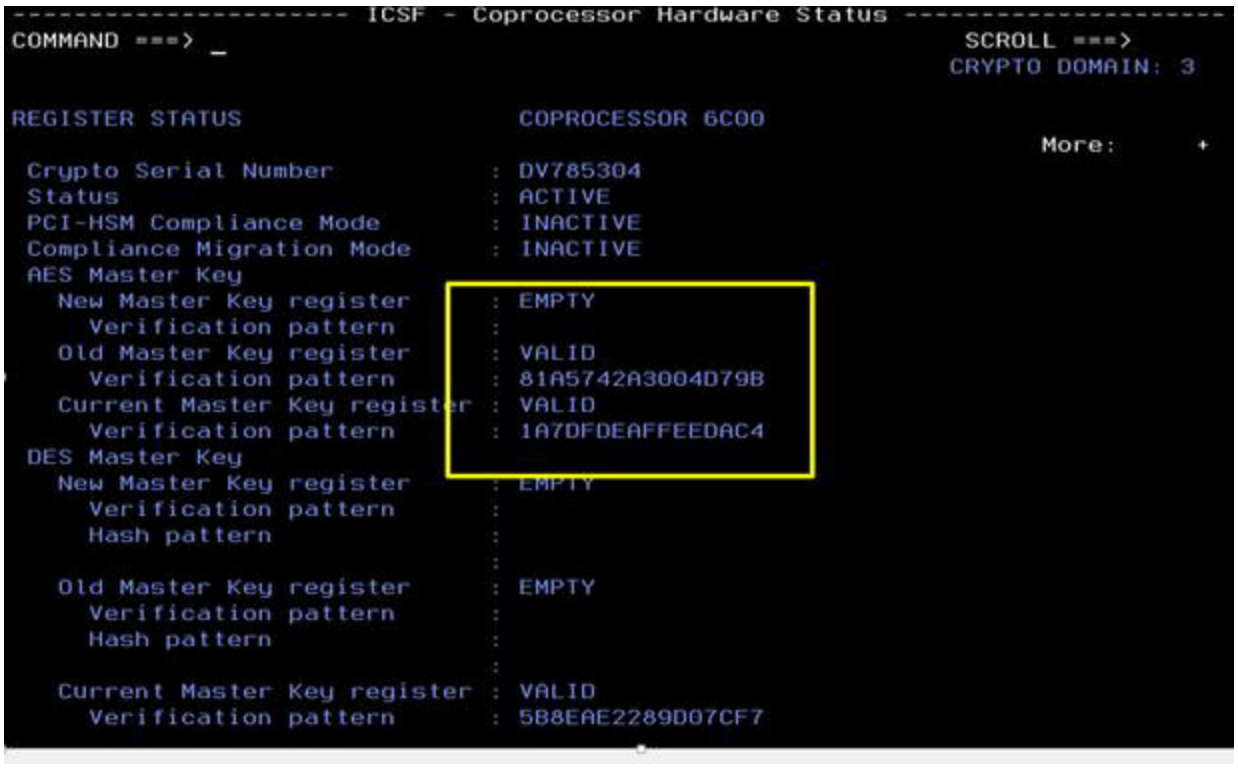
Press <F3> to return to the Coprocessor MgmtPanel

Step 3: Load the **AES Master Key Part (5 of 12)**

Type “s” next to the crypto feature to view status

Step 3: Load the **AES Master Key Part (6 of 12)**

View the coprocessor hardware status panel

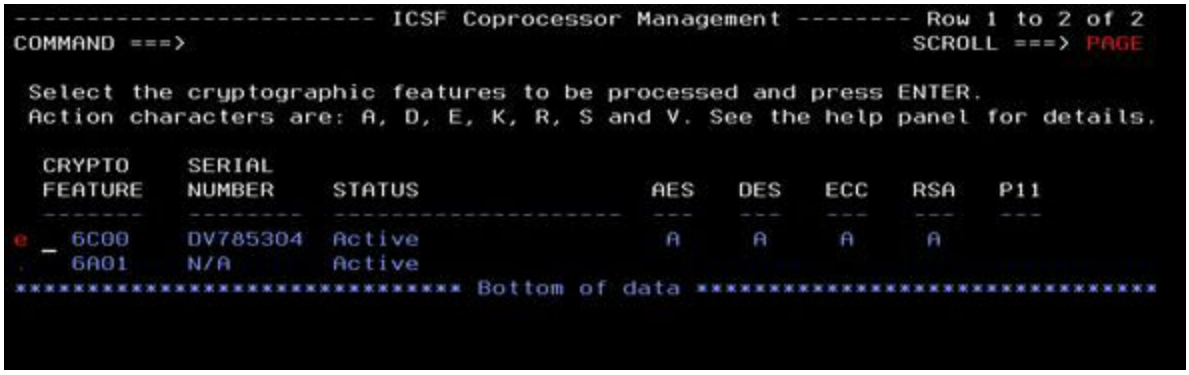


Step 3: Load the **AES Master Key Part (7 of 12)**

Press <F3> to return to the Coprocessor MgmtPanel

Step 3: Load the **AES Master Key Part (8 of 12)**

Type “e” next to the crypto feature to enter key parts



Step 3: Load the **AES Master Key Part (9 of 12)**

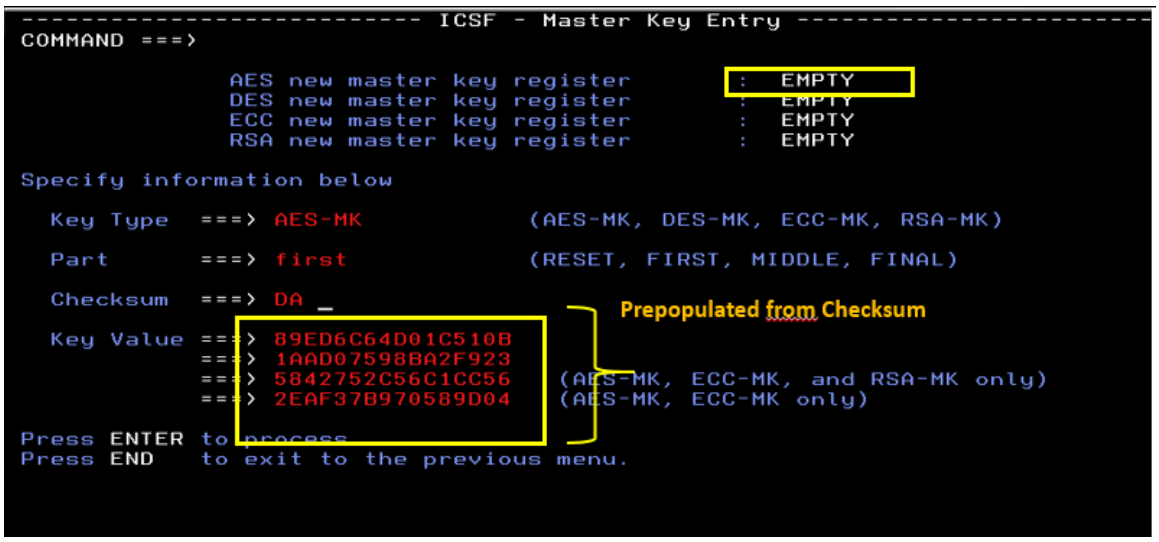
View the Master Key Entry panel

Step 3: Load the **AES Master Key Part (10 of 12)**

Enter “ AES-MK” for key type and “FIRST” for part.

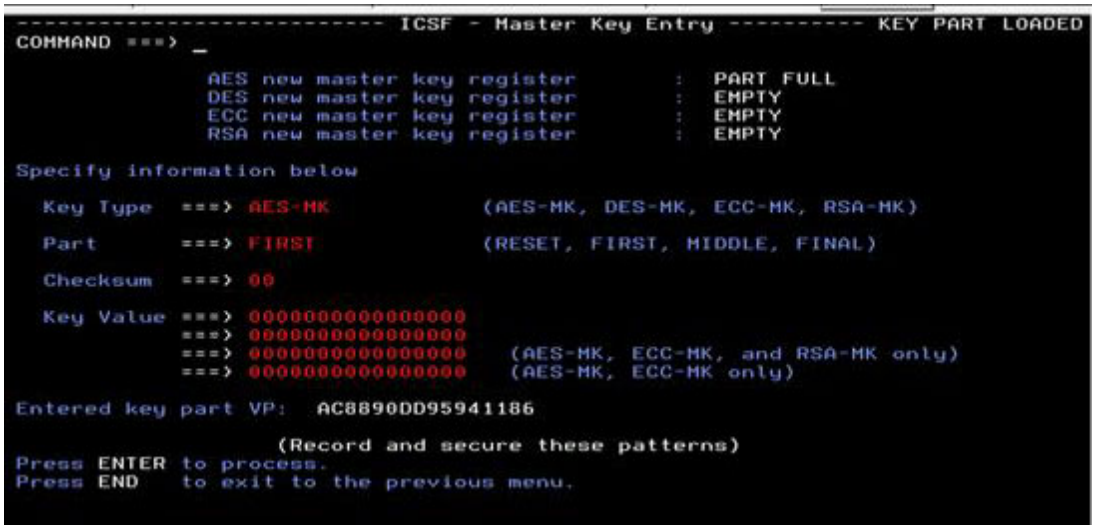
Step 3: Load the **AES Master Key Part (11 of 12)**

Enter “ AES-MK” for key type and “FIRST” for part.



Step 3: Load the **AES Master Key Part (12 of 12)**

Check the new master key register status is PART FULL



Step 4: Repeat Steps 1 -3 for the **desired number of key parts**

Repeat the steps for each intermediate AES Master Key Part. Each key custodian would generate and load (and securely save) their individual key part.

1. Generate a random key (and save the results)
   - ICSF Option 5.3
2. Generate a checksum, VP (and save the results)
   - ICSF Option 5.4

3. Load the key part into the new master key register
   - ICSF Option 1.e (with part MIDDLE)

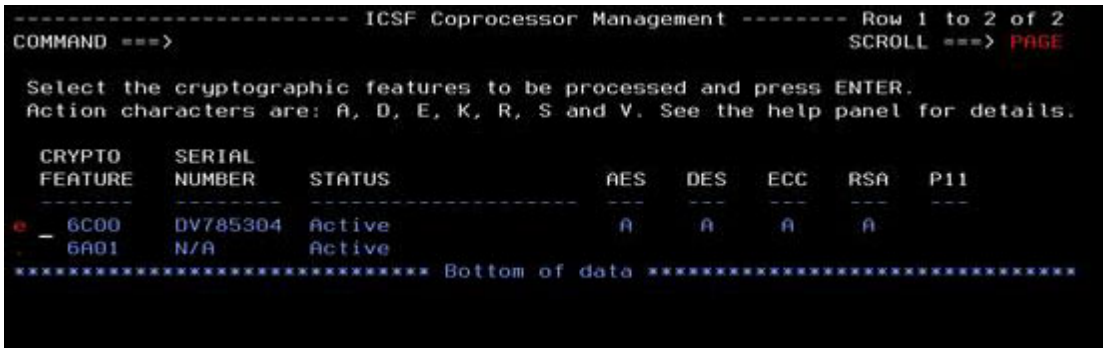
After all intermediate key parts have been generated and saved…

Step 5: Load the final **AES Master Key Part (1 of 9)**

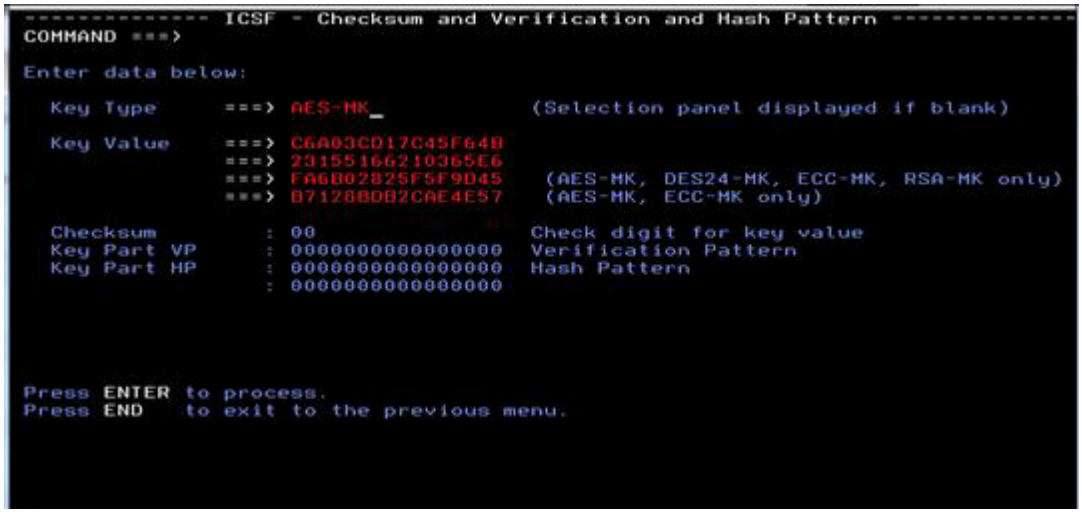
Choose Option 1 -Coprocessor MgmtPanel

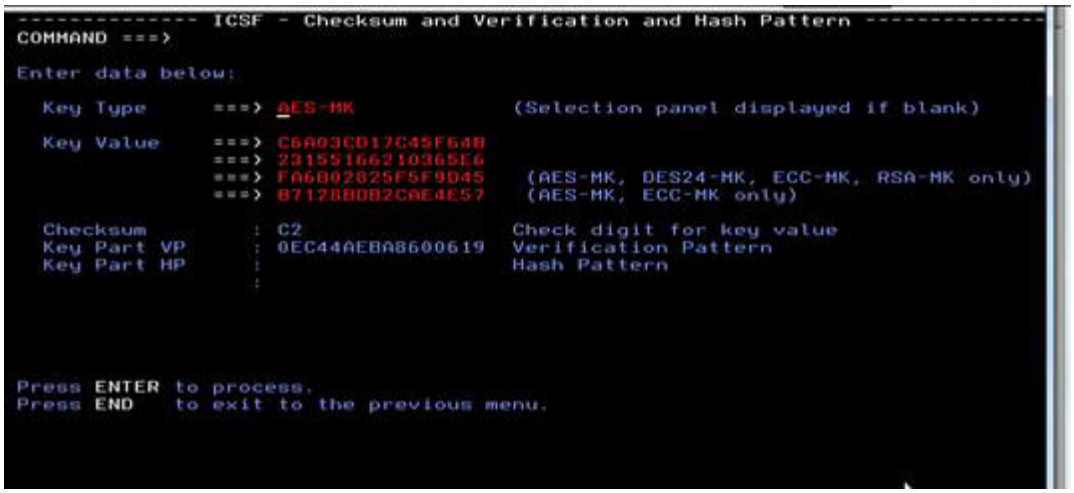
Step 5: Load the final **AES Master Key Part (2 of 9)**

Type “e” next to the crypto feature to enter the final key part



Step 5: Load the final **AES Master Key Part (3 of 9)**





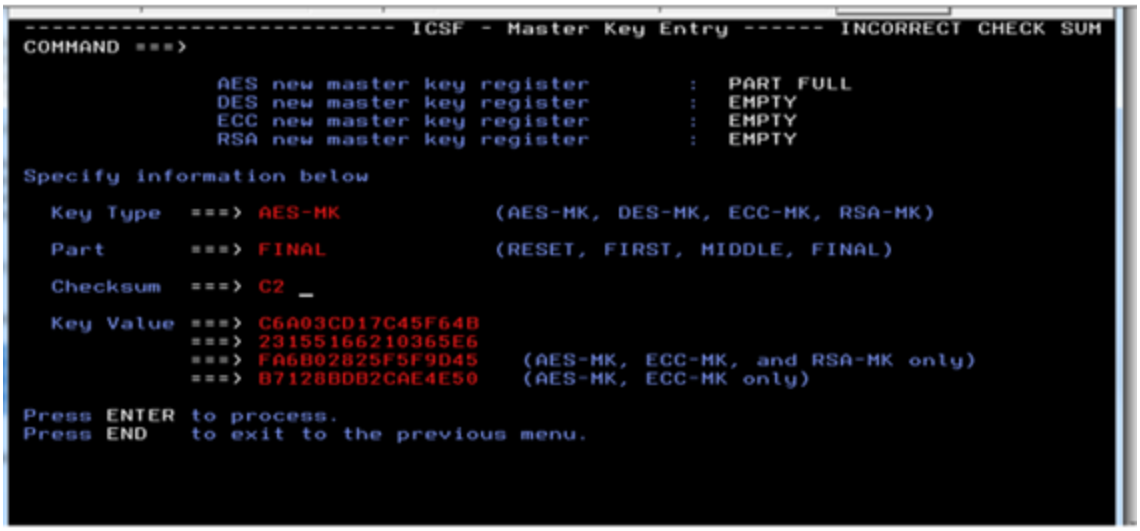
View the Master Key Entry Panel

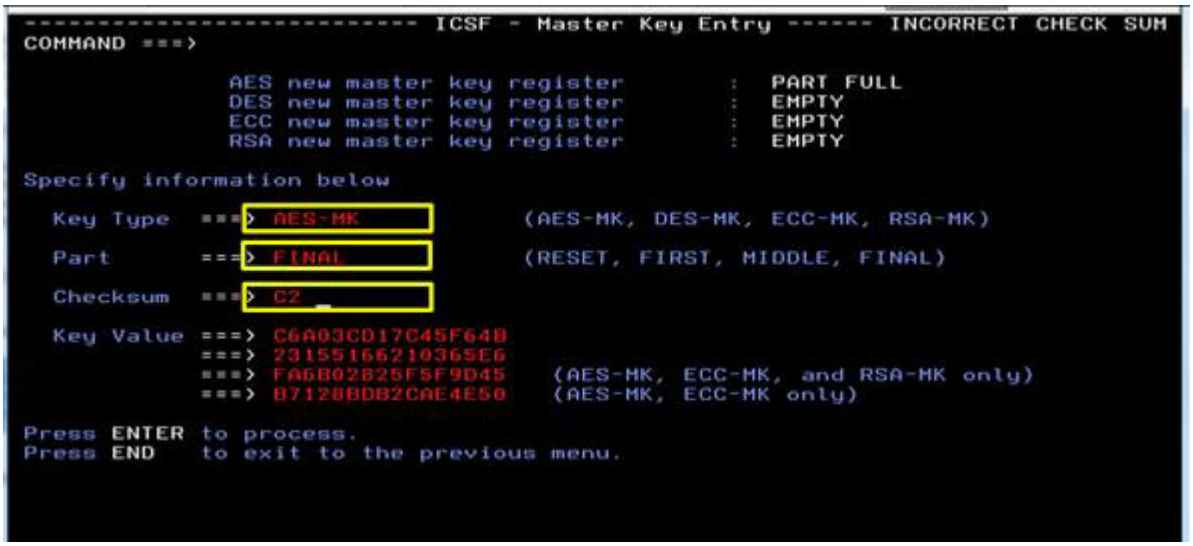
Step 5: Load the final **AES Master Key Part (4 of 9)**

Enter " AES-MK" for key type and "FINAL" for part.

Note: IF you enter an incorrect checksum value, then ICSF you warn you ('incorrect checksum value' message on the top right corner.

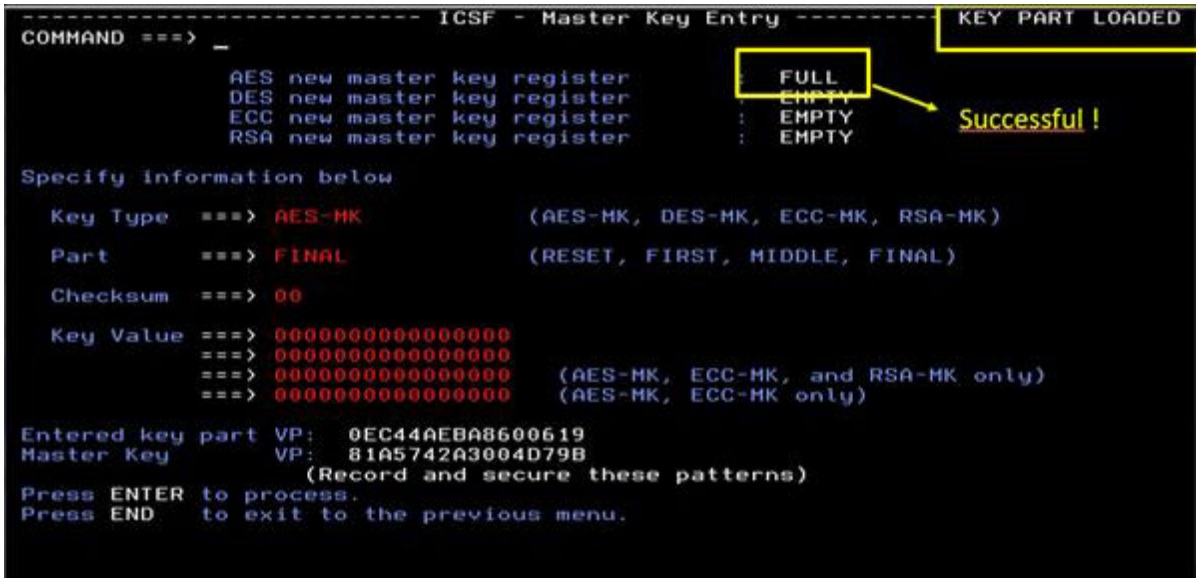
Meke sure you enter the correct value that was reported as the generated checksum value.





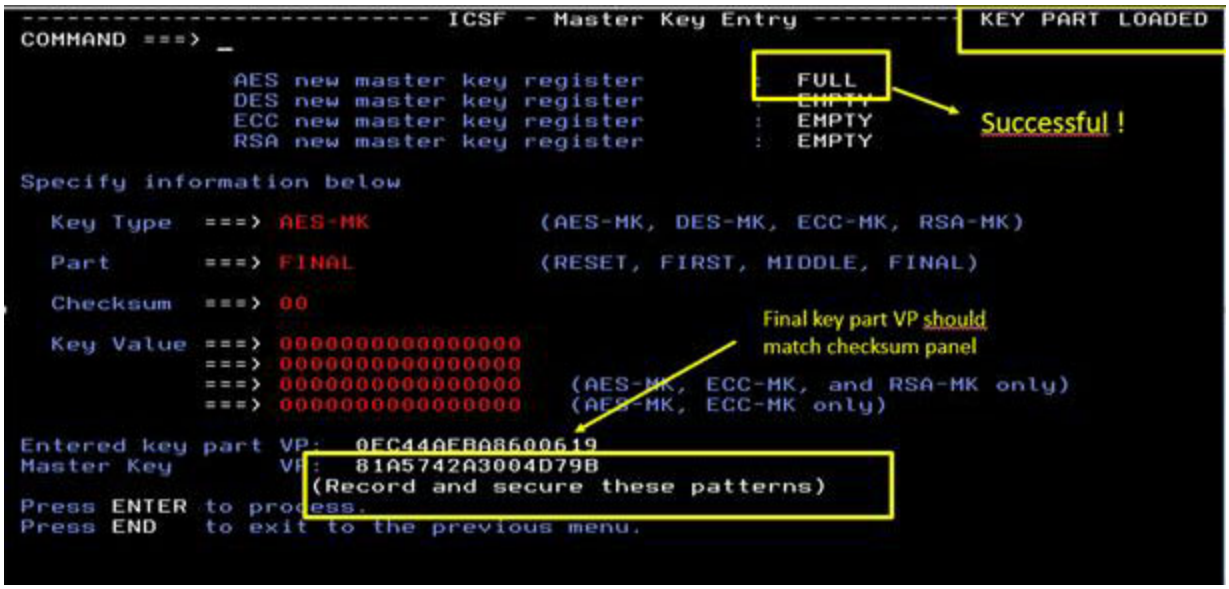
Step 5: Load the final **AES Master Key Part (5 of 9)**

Check the new master key register status is FULL



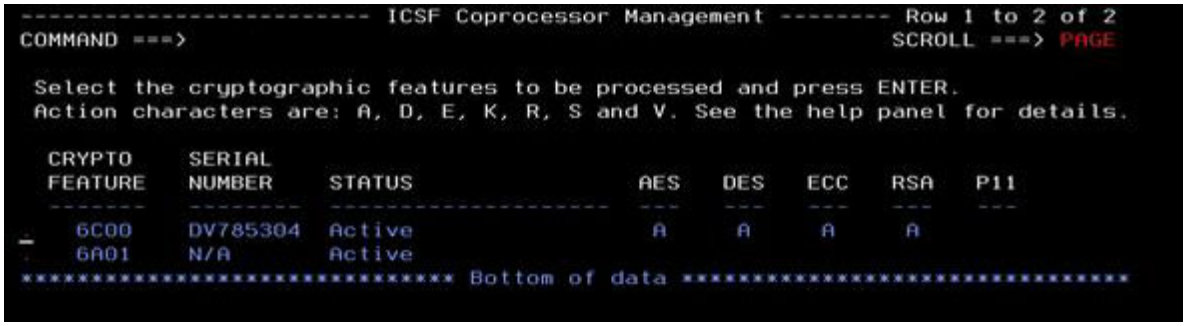
Step 5: Load the final **AES Master Key Part (6 of 9)**

Scroll down to capture the final Master VP .



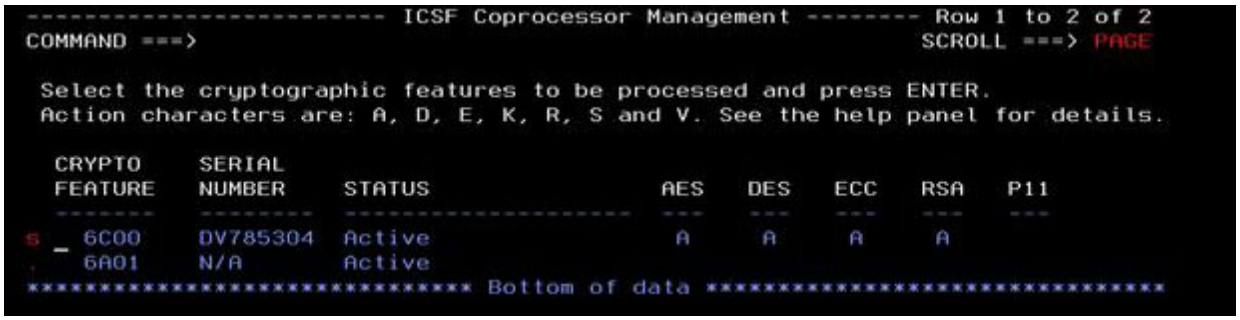
Step 5: Load the final **AES Master Key Part (7 of 9)**

Press <F3> to return to the Coprocessor MgmtPanel



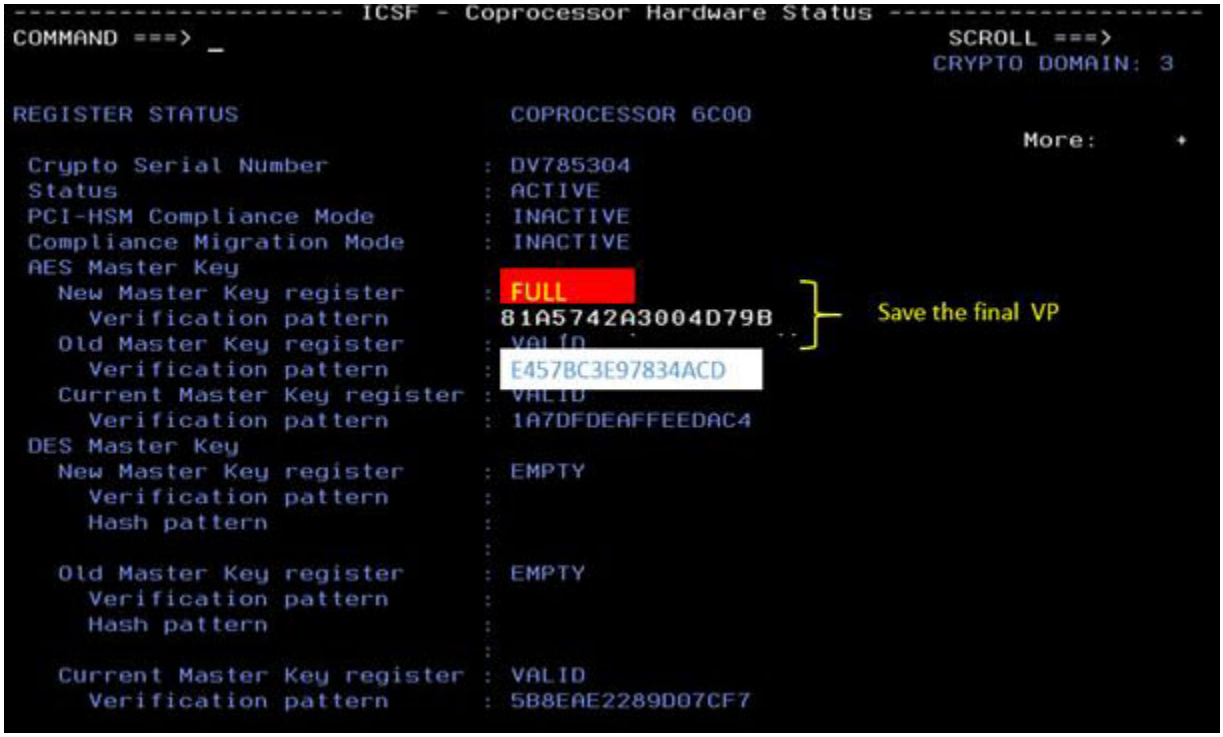
Step 5: Load the final **AES Master Key Part (8 of 9)**

Type “s” next to the crypto feature to view status



Step 5: Load the final **AES Master Key Part (9 of 9)**

View the coprocessor hardware status panel



If you have other members in your sysplex sharing the same CKDS and using a different domain number, you should repeat the same process for each member of the sysplex, except for steps 1 and 2 which must be avoided.

~~Step 1: Generate a random number for the AES Master Key Part~~

~~Step 2: Generate a checksum for the AES Master Key Part~~

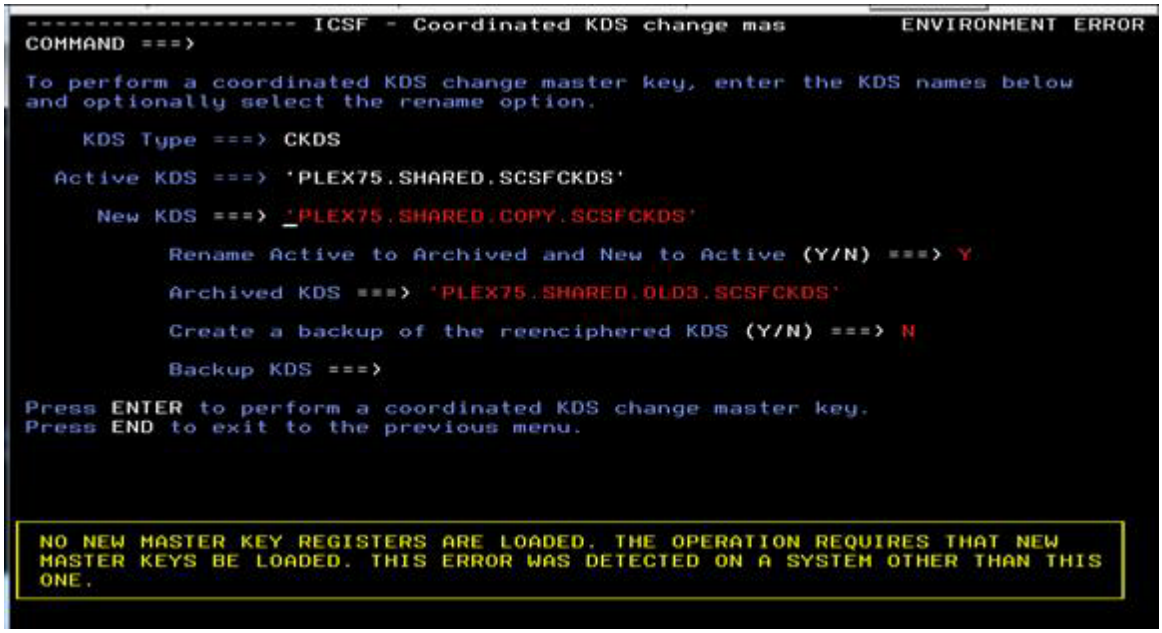
Step 3: Load the first AES Master Key Part

Step 4: Repeat Steps 1 -3 for the desired number of middle key parts

Step 5: Load the final AES Master Key Part

Step 6: Verify the new Master Key Registers

In our scenario, we did not repeat this for the second member (SC75) of our sysplex which is sharing the CKDS, and when we tried to perform the coordinated master key change we received the following error messages: 'ENVIRONMENT ERROR'




```
CSFM615I COORDINATED CHANGE-MK FAILED.  NEW MASTER KEYS INCORRECT
ON
SC75. RC = 12, RSN =
3098.
IEF196I CSFM615I COORDINATED CHANGE-MK FAILED.  NEW MASTER
KEYS
IEF196I INCORRECT ON SC75. RC = 12, RSN =
3098.
CSFM636I SYSTEM SC75 FAILURE FOR COORDINATED CKDS ACTIVITY.
MSGTYPE=00
0001 RC=12
RSN=3098.
CSFM616I COORDINATED CHANGE-MK FAILED, RC=0000000C RS=
00000C1A
SUPRC= 00000000 SUPRS= 00000000 FLAGS=
48000000.
CSFU006I CHANGE-MK FEEDBACK: RC=0000000C RS=00000C1A
SUPRC=00000000
SUPRS=00000000
FLAGS=48000000.
IEF196I CSFU006I CHANGE-MK FEEDBACK: RC=0000000C
RS=00000C1A
IEF196I SUPRC=00000000 SUPRS=00000000
FLAGS=48000000.
```

Also note that if your system is using multiple coprocessors, they must have the same master key or keys. When you load a new master key or keys in one coprocessor, you

should load the same new master key or keys in the other coprocessors. Therefore, to reencipher a key data set under a new master key, the new master key registers in all coprocessors must contain the same value.

Step 6: Generate & Load the Remaining Master Keys (optional **for other Master key types)**

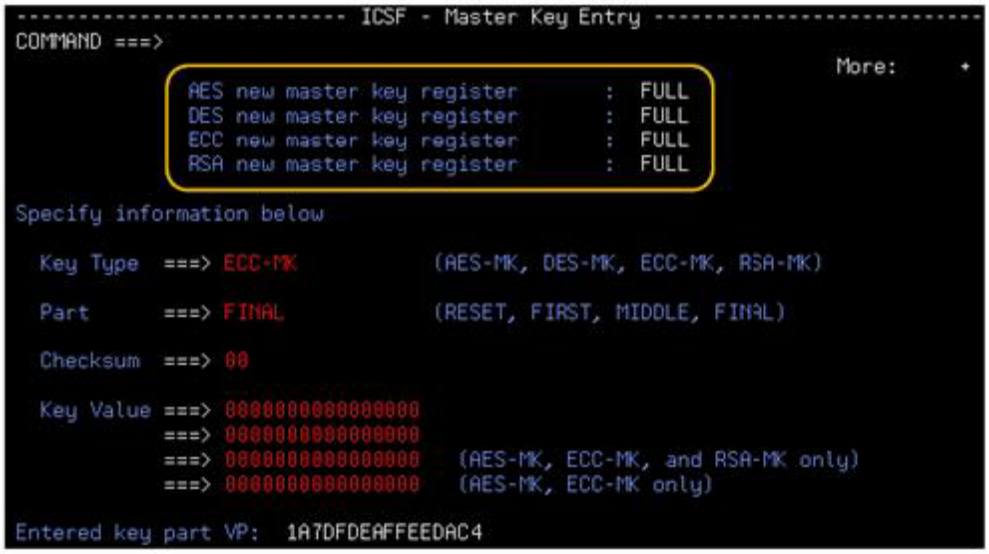
Follow the same sequence for each remaining Master Key (i.e. AES, RSA, ECC) and each key custodian / key part.

1. Generate a random key (and save the results)
   - ICSF Option 5.3
2. Generate a checksum, VP, HP (and save the results)
   - ICSF Option 5.4
3. Load the key part into the new master key register
   - ICSF Option 1.e
4. Repeat steps 1 –3 for each key part
5. Repeat steps 1 –4 for each Master Key type

When you initialize or reencipher a CKDS or PKDS, ICSF places the verification pattern for the master keys into the key data set header record.

Step 7: Verify the New Master Key Registers

View the Master Key Entry panel: it should at least FULL for AES new master key register if you have followed the previous steps.Depending if you have repated the steps for DES RSA and ECC , their status may also be FULL.



This is the end of STEP 2 In our todo list

The To-Do List…

Step 1: Allocate New Key Data Sets

Step 2: Load the New Master Key Registers

Step 3: Initiate Coordinated CKDS Master Key Change

Step 4: Verify the Master Keys are Active

Changing the master keys consists of:

a) Loading the new master key registers on all coprocessors on all member of the sysplex sharing the active CKDS.

b) Allocating a new CKDS.

c) Reenciphering the CKDS.

d) Setting the master keys and making the reenciphered CKDS the active CKDS.

Changing the master keys in a sysplex can be done by using the coordinated CKDS change master key utility. After loading the new master keys and allocating the new CKDS, the utility is initiated on one LPAR and all members of the sysplex sharing the CKDS will participate. The CKDS will be reenciphered on the initiating LPAR, all members will refresh to the reenciphered CKDS, and set the master keys. See Symmetric master keys and the CKDS for additional information.

Prior to reenciphering a CKDS, consider temporarily disallowing dynamic CKDS update services. For more information, see Steps for enabling and disabling PKA callable services and Dynamic CKDS/PKDS access controls.

Before beginning this procedure, you must:

1. Enter the key parts of the new master key that you want to replace the current master key into all coprocessors on your system. For information about how to do this procedure, see Entering master key parts. The new master key register must be full when you change the master key.
2. Create a new VSAM data set in which the reenciphered keys will be placed to create the new reenciphered CKDS. This data set must be allocated and empty and must contain the same data set attributes as the active CKDS. For more information about defining a CKDS, see z/OS Cryptographic Services ICSF System Programmer's Guide.

Now we must initiate a coordinated a CKDS master key change.

This is step 3 in our ToDo list:

The To-Do List…

Step 1: Allocate New Key Data Sets

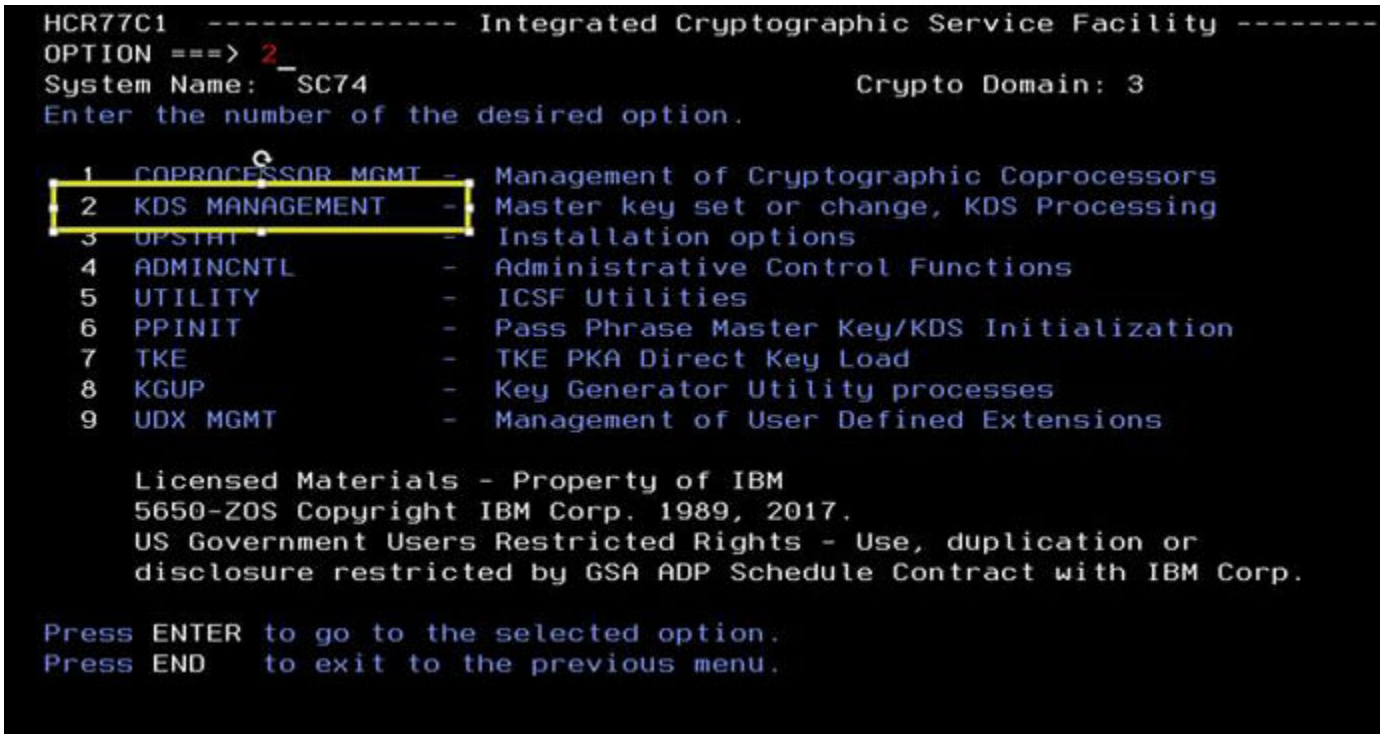
Step 2: Load the New Master Key Registers

Step 3: **Initiate Coordinated CKDS Master Key Change**

Step 4: Verify the Master Keys are Active

Step 3: Initiate CKDS Coordinated Change MK (1 of 10)

Choose Option 2 –KDS Management

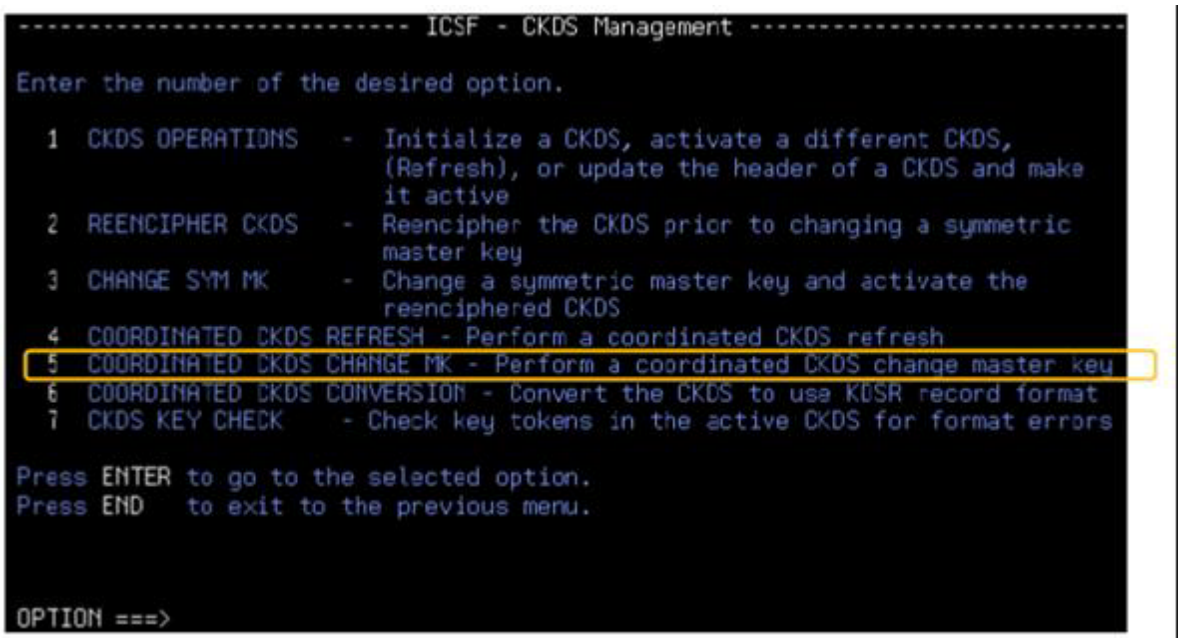


Step 3: Initiate CKDS Coordinated Change MK (2 of 10)

Choose Option 1–CKDS Management

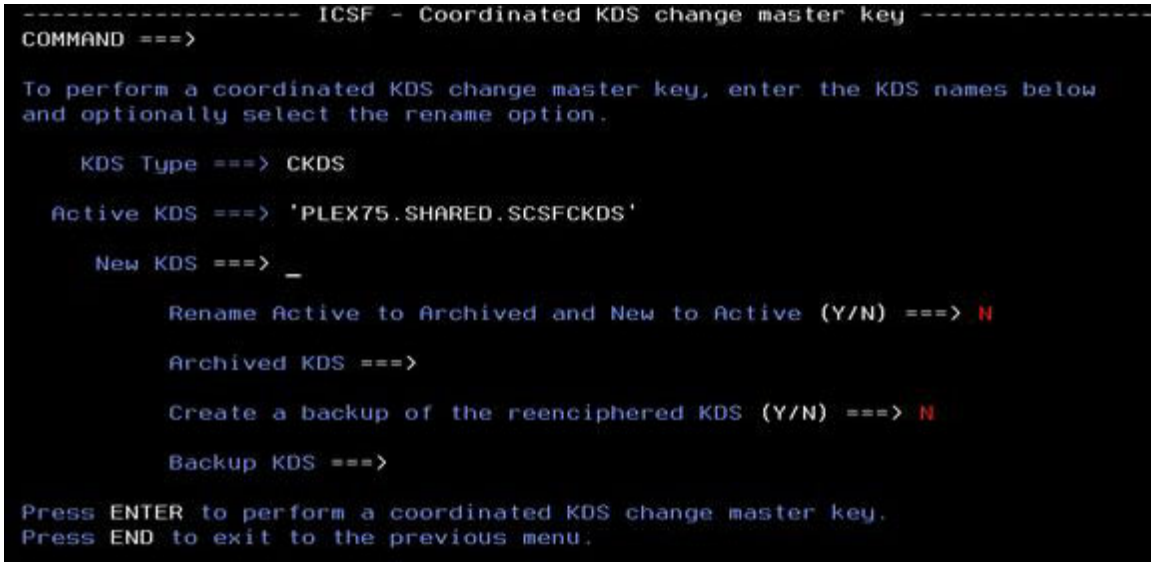
Step 3: Initiate CKDS Coordinated Change MK (3 of 10)

Choose Option 5–Perform a coordinated CKDS Change Master Key



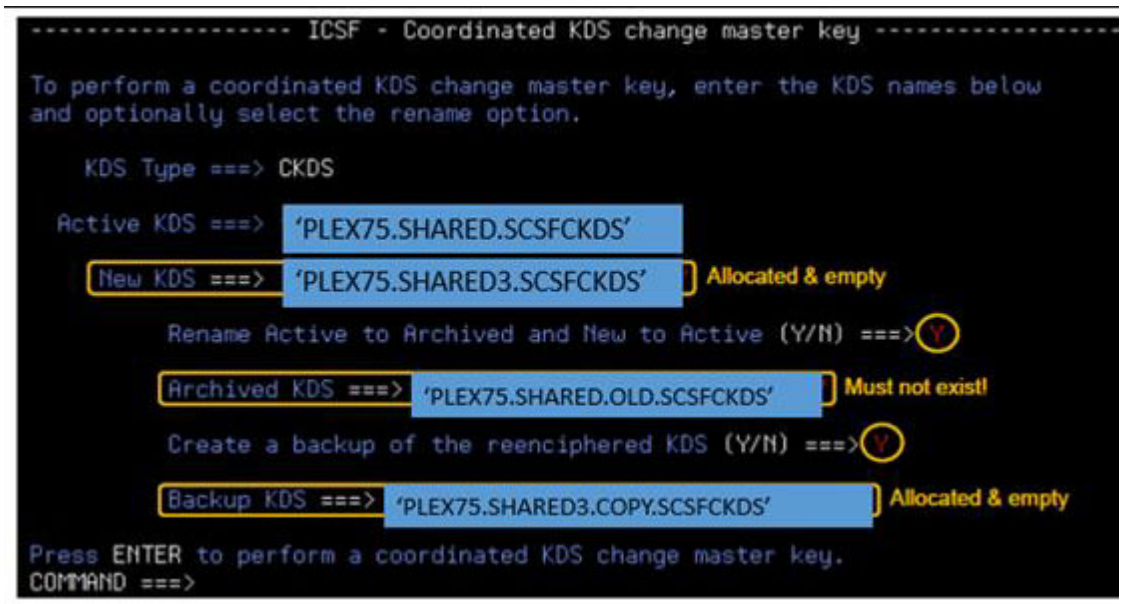
Step 3: Initiate CKDS Coordinated Change MK (4 of 10)

View the Coordinated KDS Change MK Panel



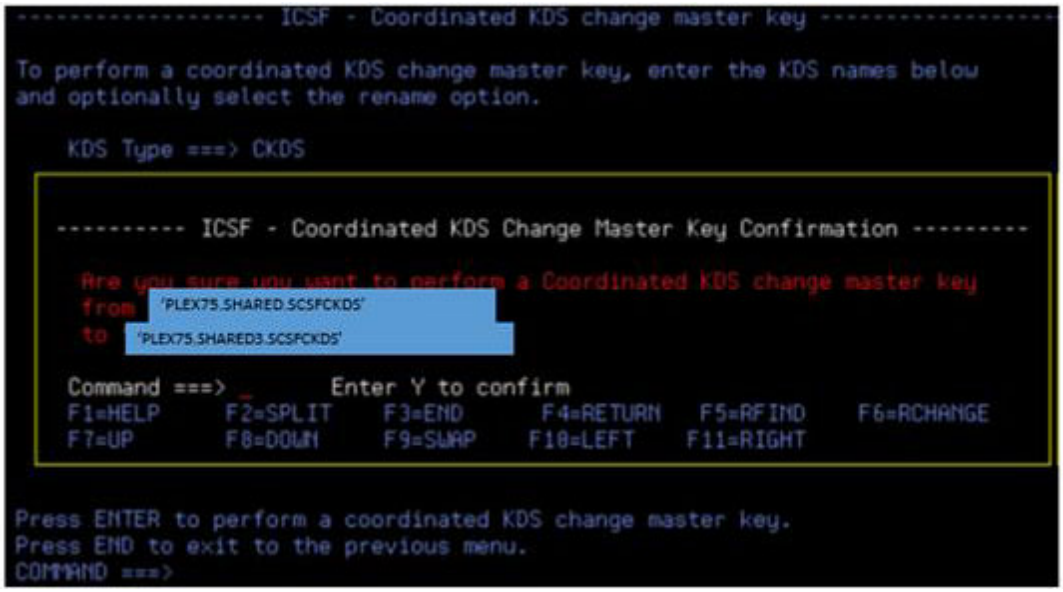
Step 3: Initiate CKDS Coordinated Change MK (5 of 10)

Type Y for both options and update the New, Archived and Backup KDS names.



Step 3: Initiate CKDS Coordinated Change MK (6 of 10)

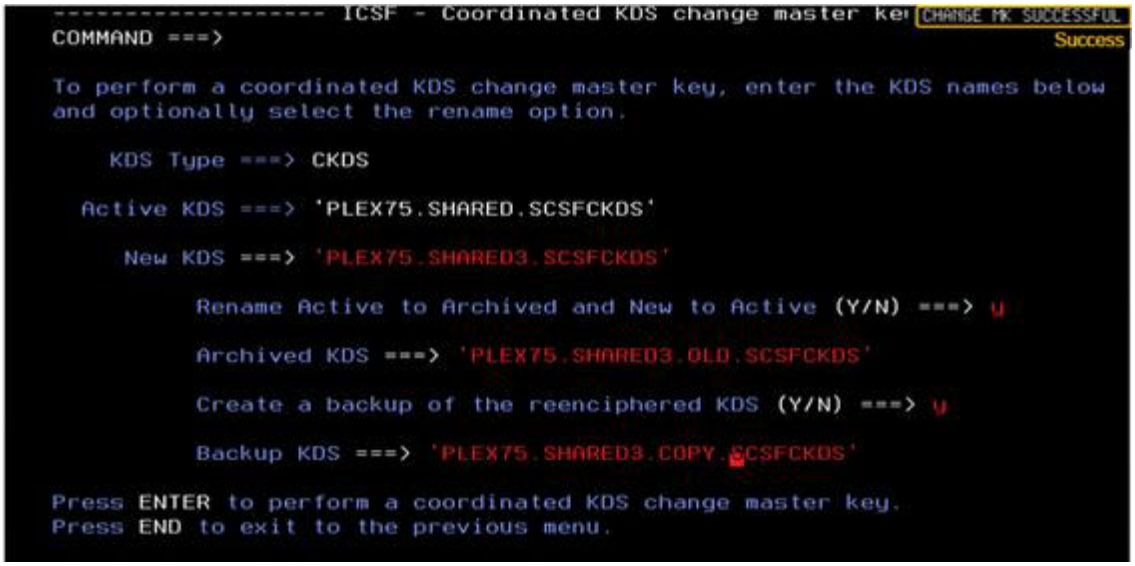
Type Y to confirm the operation.



If you don't have access to resource CSFDKCS (READ), you will get the following error message:

Step 3: Initiate CKDS Coordinated Change MK (7 of 10)

Check the status message



Step 3: Initiate CKDS Coordinated Change MK (8 of 10)

Check for the MVS Console messages…

In the SYSLOG, we saw the following set of messages:

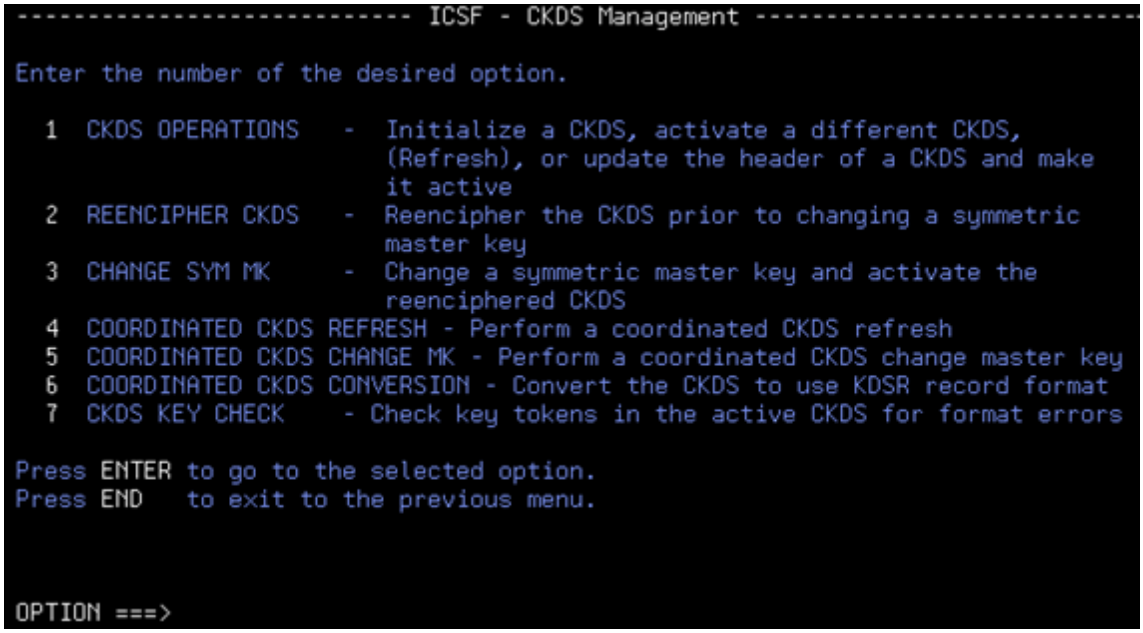
```
CSFM618I CKDS DATA SET PLEX75.SHARED3.SCSFCKDS.INDEX RENAMED TO
PLEX75.SHARED.SCSFCKDS.INDEX
IEF196I CSFM618I CKDS DATA SET PLEX75.SHARED3.SCSFCKDS.INDEX RENAMED
TO
IEF196I PLEX75.SHARED.SCSFCKDS.INDEX
CSFM622I COORDINATED CHANGE-MK PROGRESS: DATA SET RENAMING COMPLETE.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: DATA SET RENAMING
IEF196I COMPLETE.
IEF196I IEF237I 9788 ALLOCATED TO SYS00005
CSFM653I CKDS LOADED 9 RECORDS WITH AVERAGE SIZE 253
IEF196I CSFM653I CKDS LOADED 9 RECORDS WITH AVERAGE SIZE 253
IEF196I IGD104I PLEX75.SHARED.SCSFCKDS                      RETAINED,
IEF196I DDNAME=SYS00005
CSFM622I COORDINATED CHANGE-MK PROGRESS: NEW IN-STORAGE KDS LOADED ON
REMOTE SYSTEMS.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: NEW IN-STORAGE KDS
IEF196I LOADED ON REMOTE SYSTEMS.
CSFM622I COORDINATED CHANGE-MK PROGRESS: OPERATION TERMINATION IS
TEMPORARILY INHIBITED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: OPERATION TERMINATION
IEF196I IS TEMPORARILY INHIBITED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: A NEW CKDS HASH TABLE WAS
CONSTRUCTED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: A NEW CKDS HASH TABLE
IEF196I WAS CONSTRUCTED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CHANGE MASTER KEY PROCESSING
COMPLETED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: CHANGE MASTER KEY
IEF196I PROCESSING COMPLETED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN REFERENCES
 UPDATED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN
IEF196I REFERENCES UPDATED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE CKDS
HASH TABLE TO NEW.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE
IEF196I CKDS HASH TABLE TO NEW.
CSFM622I COORDINATED CHANGE-MK PROGRESS: OPERATION TERMINATION IS NOW
REENABLED.
```

```
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: OPERATION TERMINATION
IEF196I IS NOW REENABLED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: COMPLETING CORE WORK.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: COMPLETING CORE WORK.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: REENCIPHERING KEYS IN CFRM
CDS.
CSFM622I COORDINATED CHANGE-MK PROGRESS: A NEW CKDS HASH TABLE WAS
CONSTRUCTED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: A NEW CKDS HASH TABLE
IEF196I WAS CONSTRUCTED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: ADMINISTRATIVE DATA KEYS
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CHANGE MASTER KEY PROCESSING
STARTED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: ACTIVE POLICY DATA HAS BEEN
UPDATED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CF STRUCTURE UPDATES PENDING.
CSFM622I COORDINATED CHANGE-MK PROGRESS: CHANGE MASTER KEY PROCESSING
COMPLETED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: CHANGE MASTER KEY
IEF196I PROCESSING COMPLETED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN REFERENCES
 UPDATED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN
IEF196I REFERENCES UPDATED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE CKDS
HASH TABLE TO NEW.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE
IEF196I CKDS HASH TABLE TO NEW.
CSFM617I COORDINATED CHANGE-MK ACTION COMPLETED SUCCESSFULLY.
CSFU006I CHANGE-MK FEEDBACK: RC=00000000 RS=00000000 SUPRC=00000000
SUPRS=00000000 FLAGS=40000000.
IEF196I CSFU006I CHANGE-MK FEEDBACK: RC=00000000 RS=00000000
IEF196I SUPRC=00000000 SUPRS=00000000 FLAGS=40000000.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CF STRUCTURE UPDATES PENDING.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CF STRUCTURE UPDATES
COMPLETED.
```

Note that all the IXC messages are sent by XES because we had enabled CF structure encryption in our test environment, and XES Detected a change in the AES master key, so that it drove a CF Structure change in the CFRM couple dataset.

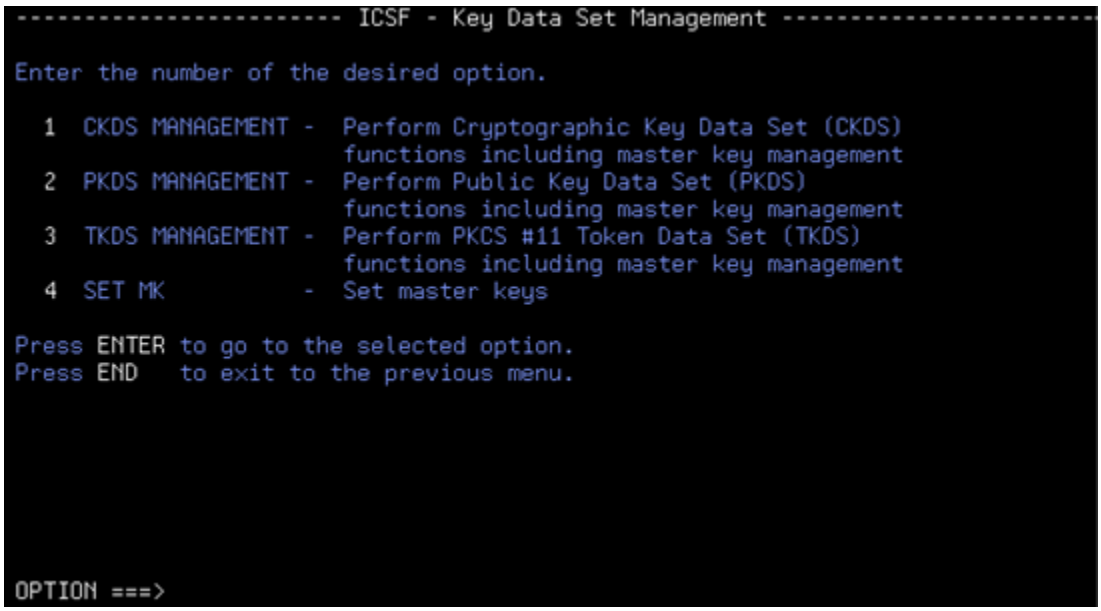
Step 3: Initiate CKDS Coordinated Change MK (9 of 10)

Press <F3> to return to the CKDS Management Panel



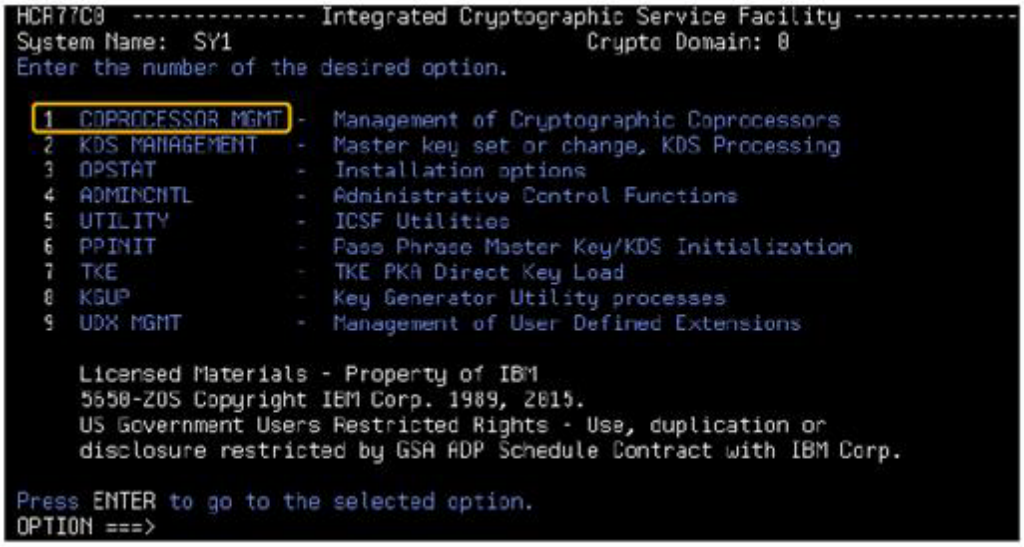
Step 3: Initiate CKDS Coordinated Change MK (10 of 10)

Press <F3> to return to the KDS Management Panel



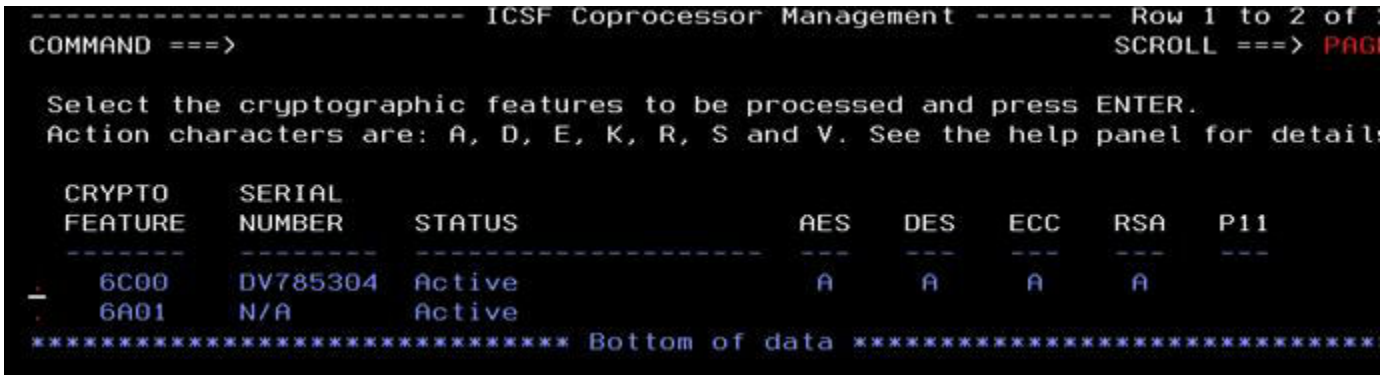
Step 4: Verify the Master Keys are Active (1 of 4)

Choose Option 1 –Coprocessor Mgmt



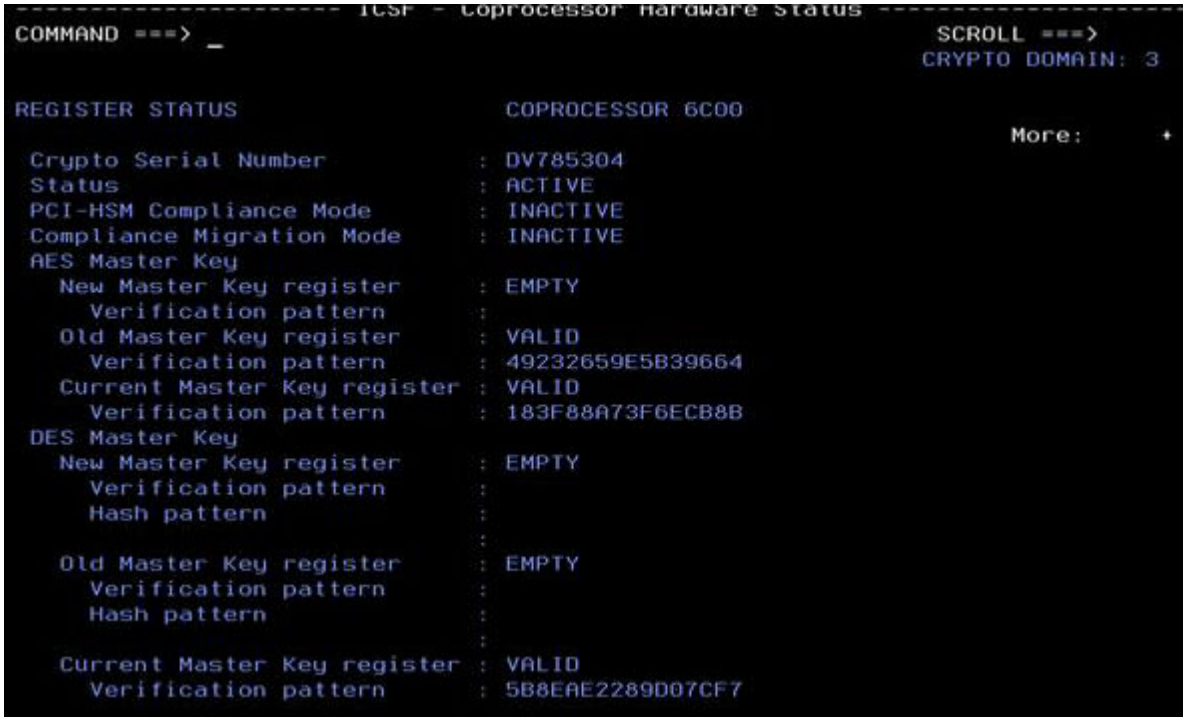
Step 4: Verify the Master Keys are Active (2 of 4)

View the Master Key Status



Step 4: Verify the Master Keys are Active (3 of 4)

Type “s” next to the crypto feature to view status



Step 4: Verify the Master Keys are Active (4 of 4)

For all Master Keys…

- The New Master Key register has become EMPTY.
- The Current Master Key register is VALID with a new Verification Pattern.
- The Old Master Key register is VALID with the Verification Pattern from the previous Master Key.

Warning:

In the knowledge center for z/OS Cryptographic Services ICSF Administrator's Guide at

https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.3.0/com.ibm.zos.v2r3.csfb300/ccxange.htm there is a note which says:

Important

The master keys are loaded into the new master key register. The utilities that are described in this section require the master keys to be in the new master key register and not set. Newer versions of the TKE workstation allow the master keys to be set from the TKE workstation. This option must not be used for a coprocessor or domain that is visible to ICSF. Setting the master keys should be done by using ICSF utilities so ICSF knows that the master keys have changed. Changing the master keys without using ICSF may cause an outage.

You should not be confused here , what it really boils down to is:

"***Newer versions of the TKE workstation allow the master keys to be set from the TKE workstation***. This option must not be used for a coprocessor or domain that is visible to ICSF. "

For the utilities in that section of the Admin Guide, TKE can be used to load the master keys but should not set the master keys.

- Loading a master key involves storing key material in the new master key register.
- Setting a master key involves moving key material from the new master key register to the current master key register

Whether the new master key is loaded with TKE or with the ICSF panels, when ICSF sets the master key, it performs checks to prevent inconsistencies.

Notes about Coordinated KDS Administration:

In HCR7790, the coordinated KDS administration callable service, CSFCRC, introduced the coordinated CKDS refresh and coordinated CKDS change-mk functions. In HCR77A0, this callable service has been extended to provide coordinated PKDS refresh, coordinated PKDS change-mk and coordinated TKDS change-mk.

When used for coordinated change-mk, applications may run KDS update workloads in parallel, and ICSF guarantees that any dynamic updates will be reflected in the target data set.

For coordinated refresh (CKDS and PKDS only) it is recommended to disable KDS update workloads when refreshing to a target data set that is different from the currently active KDS. Updates occurring to the current active KDS would not be guaranteed to be reflected in the target data set. ICSF does not enforce that dynamic KDS updates be manually disabled prior to coordinated refresh, and will itself internally suspend such updates until the coordinated refresh operation completes.

Dynamic KDS updates occurring during a coordinated refresh in place are guaranteed to be accounted in the resulting in-storage KDS when the operation completes.

In a sysplex environment, the coordinated operations are invoked from a single ICSF instance and processed across all members sharing the same active KDS (sysplex cluster).

The callable service name for AMODE(64) invocation is CSFCRC6.

The CRC service may be called from ICSF dialogs.

This service can be protected with RACF.

This support additionally includes a new sysplex communication protocol that provides better performance and servicability characteristics

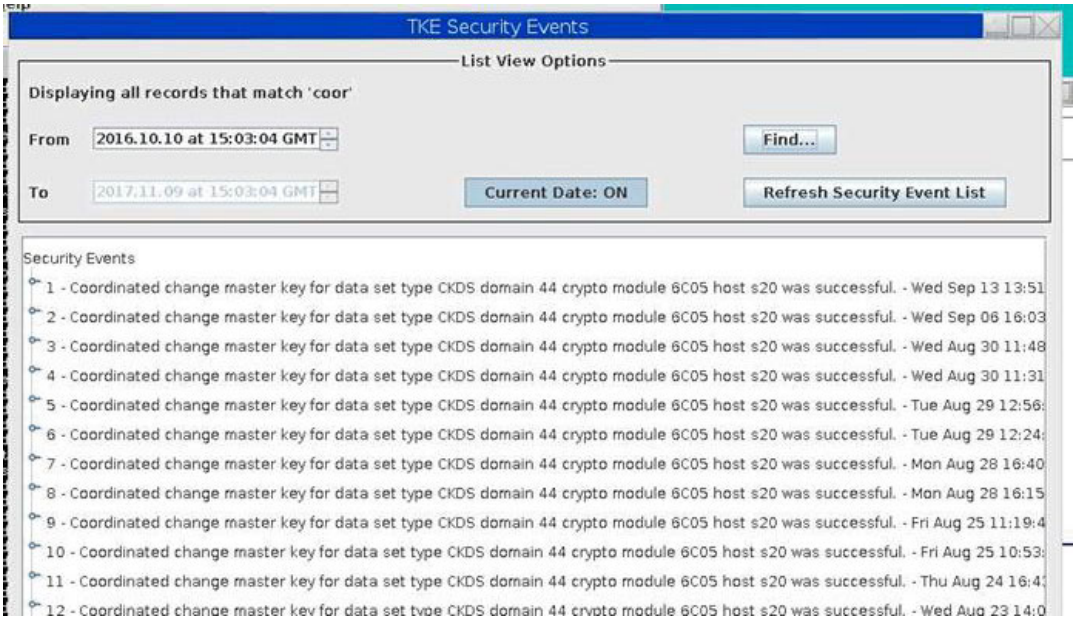
---

Auditing the master key change:

There are no SMF records showing a MK Change and that there are no indicators in the panels or via the commands.

Even with the SMF Type 82 Subtype 18 record, there is no information about master keys and/or cards being activated.

Initiating a Change MK from a TKE Workstation creates audit logs of the activity .



The auditors will also ask you: when was the Master key last changed, and can you prove it?

In the tests that we have done during our project residency, when we performed the master key change ceremony, we tried to collect and print out all SMF 82 records, but none of them gave any evidence or proof of a master key change event

The only SMF82 records we got were:

- 1 Subtype 1 (ICSF started)
- 6 Subtype 14 (3 clears on 2 systems (our MK is made of 3 parts) => but these records are just for the MASTER KEY NEW registers, not for the actual new MASTER key change
- 1 Subtype 20 (heartbeat)
- 6 Subtype 21 (when member joins the group)
- 1 Subtype 24 (1 duplicate entry)

Where can we get this information if (and it will be) requested by auditors ?

The best way for a client to prove audit compliance (and the most likely way that clients are doing it today), is to capture the console / job log messages.

The only thing we can provide is a copy of the syslog/operlog with all the ICSF messages.

The question is: is this going to be enough ? it can be tampered with... SMF records may not be if they are signed….

```
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: CHANGE MASTER KEY
IEF196I PROCESSING COMPLETED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN REFERENCES
 UPDATED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN
IEF196I REFERENCES UPDATED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE CKDS
HASH TABLE TO NEW.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE
IEF196I CKDS HASH TABLE TO NEW.
CSFM622I COORDINATED CHANGE-MK PROGRESS: OPERATION TERMINATION IS NOW
REENABLED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: OPERATION TERMINATION
IEF196I IS NOW REENABLED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: COMPLETING CORE WORK.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: COMPLETING CORE WORK.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: REENCIPHERING KEYS IN CFRM
CDS.
CSFM622I COORDINATED CHANGE-MK PROGRESS: A NEW CKDS HASH TABLE WAS
CONSTRUCTED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: A NEW CKDS HASH TABLE
IEF196I WAS CONSTRUCTED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: ADMINISTRATIVE DATA KEYS
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CHANGE MASTER KEY PROCESSING
STARTED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: ACTIVE POLICY DATA HAS BEEN
UPDATED.
IXC121I CFRM ENCRYPT CHANGE-MK PROGRESS: CF STRUCTURE UPDATES PENDING.
CSFM622I COORDINATED CHANGE-MK PROGRESS: CHANGE MASTER KEY PROCESSING
COMPLETED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: CHANGE MASTER KEY
IEF196I PROCESSING COMPLETED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN REFERENCES
 UPDATED.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: ALL FINAL CKDS DSN
IEF196I REFERENCES UPDATED.
CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE CKDS
```

```
HASH TABLE TO NEW.
IEF196I CSFM622I COORDINATED CHANGE-MK PROGRESS: SWITCHED THE ACTIVE
IEF196I CKDS HASH TABLE TO NEW.
CSFM617I COORDINATED CHANGE-MK ACTION COMPLETED SUCCESSFULLY.
CSFU006I CHANGE-MK FEEDBACK: RC=00000000 RS=00000000 SUPRC=00000000
SUPRS=00000000 FLAGS=40000000.
IEF196I CSFU006I CHANGE-MK FEEDBACK: RC=00000000 RS=00000000
IEF196I SUPRC=00000000 SUPRS=00000000 FLAGS=40000000.
```

However you must remember that all the ICSF administration services, utilities, and panels are access controlled with SAF resources. They can all be audited on the racroute call - which would generate a RACF SMF record. It does log the execution time and who did it, which should satisfy audit needs.

For critical services - set master key for example - you could set the resource to audit successes if you need an audit record for that event. If you don't need the record - then you can go with default and just audit the failure. The switch is in the SAF profile and can be changed with a ralter command.

The fact that a change MK can be audited should satisfy an auditor's requirements. The fact that the SMF record is in RACF vs ICSF would not cause an audit to fail.

So ICSF does have the capability of generating an audit record of the event, but not the result.

For the SAF calls, is it more typical that clients audit failures but not successes? So, they would be more likely to see that someone attempted to perform a CCMK or local Change MK without the correct authorization level than someone performing a CCMK that has authority.

The RACF profiles that are checked for when doing a master key change are the following:

- CSFCRC(READ)
- CSFSMK(READ)

In order to change the master key through the ICSF panels - the following profiles in CSFSERV class are used and required:

CSFSERV class entry	Controls access to ..	ICSF option	RACF command to allow access to user
CSFOWH	- generate random number	**5.3**	PE CSFOWH CLASS(CSFSERV) ID(CSFSERV) ID(*PE10*) ACC(READ)

CSFDKCS	- enter master key via coprocessor management	**1 ; then action character 'e'**	PE CSFDKCS CLASS(CSFSERV) ID(CSFSERV) ID(*PE10*) ACC(READ)
CSFCRC	- coordinated ckds change MK	**2.1.5**	PE CSFCRC CLASS(CSFSERV) ID(CSFSERV) ID(*PE10*) ACC(READ)

Therefore, we recommend to turn auditing on the above profiles, at least the CSFCRC entry with the following command: RALT CSFSERV CSFCRC AUDIT(ALL)

Note: SETR RACLIST(CSFSERV) REFRESH command to be entered after each change to take effect

Our recommendations for auditing Master key change:

We would recommend entering the following commands before the start of the Master key change ceremony:

RALT CSFSERV CSFCRC UACC(NONE) AUDIT(ALL) --- for changing master key with option 2.1.5 5 COORDINATED CKDS CHANGE MK

RALT CSFSERV CSFSMK UACC(NONE) AUDIT(ALL) -- for setting master key (option 2.4)

Then, permit the Security administrator that will carry out the Master key change on the ICSF panels:

PERMIT CSFCRC CLASS(CSFSERV) ID(PEADMIN) ACCESS(READ)

PERMIT CSFSMK CLASS(CSFSERV) ID(PEADMIN) ACCESS(READ)

This will generate an SMF record for both success and failure authorization checks for those resource names (see table below).

Then after the change of the master key has completed, you can reset the auditing level to failure only

RALT CSFSERV CSFCRC UACC(NONE) AUDIT(FAILURE) --- for changing master key

RALT CSFSERV CSFSMK UACC(NONE) AUDIT(FAILURE) -- for setting master key

Table 36. Resource names for ICSF TSO panels, utilities, and compatibility services for PCF macros

Resource Name	Utility and Callable Service Description
CSFBRCK	CKDS Browser.
CSFCMK	Change master key utility, including the panel for a local change master key, the Coordinated KDS Administration service, and CSFEUTIL.
CSFCONV	PCF CKDS to ICSF CKDS conversion utility.
CSFCRC	Coordinated KDS Administration.
CSFDKCS	Master key entry utility.
CSFEDC	Compatibility service for the PCF CIPHER macro.
CSFEMK	Compatibility service for the PCF EMK macro.
CSFGKC	Compatibility service for the PCF GENKEY macro.

Table 36. Resource names for ICSF TSO panels, utilities, and compatibility services for PCF macros (continued)

Resource Name	Utility and Callable Service Description
CSFGKF	Generate key fingerprint. Required by KGUP if key lifecycle auditing is enabled.
CSFKGUP	Key generation utility program.
CSFOPKL	Operational key load.
CSFPCAD	Cryptographic processors management (activate/deactivate).
CSFPKDR	PKDS reencipher and PKDS refresh utilities.
CSFPMCI	Pass phrase master key/KDS initialization utility.
CSFREFR	Refresh CKDS or PKDS utility, including the panels for a local refresh, the Coordinated KDS Administration service, and CSFEUTIL (CKDS) and CSFPUTIL (PKDS).
CSFRENC	Reencipher CKDS or PKDS utility, including the panels for a local refresh, the Coordinated KDS Administration service, and CSFEUTIL (CKDS) and CSFPUTIL (PKDS).
CSFRSWS	Administrative control functions utility (ENABLE).
CSFRWP	CKDS Conversion2 - rewrap option.
CSFRTC	Compatibility service for the CUSP or PCF RETKEY macro.
CSFSMK	Set master key utility.
CSFSSWS	Administrative control functions utility (DISABLE).
CSFUDM	User Defined Extensions (UDX) management functions.

Note: an RFE has been submitted during our residency to request that an SMF record be cut whenever the master key is changed, so that the event can be audited. The RFE can be found at: https://www.ibm.com/developerworks/rfe/execute?use_case=viewRfe&CR_ID=113488

Description: ICSF currently cuts SMF records for load master key (MK) parts (which does not necessary lead to an actual change of MK), but it doesn't cut SMF record for the actual change of MK. If/when auditor request evidence of a MK change, we cannot show that without the necessary SMF record(s).

Use case: Having ICSF cut SMF records for MK changes, we can then simply run (an updated) supplied CSFSMFJ/CSFSMFR in SAMPLIB to produce a report showing the MK change.

You may consider 'voting' for this RFE to raise its priority of being considered by the product development team for inclusion into the product. There is a clickable 'vote' link at the bottom of the page of the above url.

References

Using new DFSMS functions in z/OS V2R3
https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.idak100/encryption23.htm

IBM Crypto Education Community
https://www.ibm.com/developerworks/community/groups/community/crypto

Load AES MK presentation
https://www.ibm.com/developerworks/community/wikis/form/api/wiki/be0cb4c9-e5c5-4588-8d23-c896b7ec8ba3/page/a0439101-53d0-41a0-a933-7b32a84f9023/attachment/0393b6bb-7dfc-4ef5-8334-76abb570c410/media/Step4_LoadAESMK.pdf

Dataset Encryption FAQs
https://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/FQ131494