Migrating to and using ICSF HCR77A0

This topic describes our migration to and use of ICSF HCR77A0, which became generally available as a web deliverable in September, 2012. Our experiences with this level of ICSF include:

- Crypto Express4 Feature
- Simplified PKDS administration in a sysplex
- New callable services to translate cipher text
- Stress testing

Crypto Express4 Feature

Available on the IBM zEnterprise EC12, the new Crypto Express4 Feature can be configured as a Crypto Express4 Accelerator (CEX4A), a Crypto Express4 CCA coprocessor (CEX4C) or a Crypto Express4 PKCS #11 coprocessor (CEX4P). In our environment we configured all three.

New with ICSF HCR77A0 and the Crypto Express4 PKCS #11 coprocessor is the function of secure PKCS #11 keys. In order to take advantage of secure PKCS #11 keys, in addition to the coprocessor and ICSF HCR77A0, you need an initialized TKDS and the new P11 master key or P11-MK set. When you have set up all of these functions, secure PKCS #11 keys will be encrypted under the P11 master key and stored in the TKDS.

In order to set the P11 master key, a TKE workstation is required to load key parts. For the setting of the P11 master key, we had to switch between the TKE workstation and our ICSF host session.

- For information on how to use the TKE to set the P11 master key, see the Trusted Key Entry Workstation User's Guide, SA23-2211-07
- For information on what to do on the ICSF side to set the P11 master key, see z/OS Cryptographic Services Integrated Cryptographic Service Facility Administrator’s Guide , SA22-7521-16.

When we had set up our P11 master keys we ran various workloads to test out this new function.

Simplified PKDS administration in a sysplex

HCR77A0 provides new function to simplify PKDS administration in a sysplex. New panels and function were added to drive a PKDS change master key operation or a PKDS refresh operation from a single instance of ICSF across all sysplex members sharing the

same active PKDS. This single ICSF instance will drive the operation across the sysplex using sysplex messaging to other members sharing the same active PKDS.

We tested the coordinated PKDS that we refreshed in our environment. While we were running our stress workload, we did the following:

1. From the ICSF - Master Key Management panel, selected option 2, PKDS MK MANAGEMENT
2. From the ICSF- PKDS Master Key Management panel, selected option 4, COORDINATED PKDS REFRESH
3. From the ICSF- Coordinated KDS Refresh panel, the field **`KDS Type`** was pre-filled with `PKDS` and the **`Active KDS`** field was filled in. Also, the **`Rename Active to Archived and New to Active`** field was pre-filled with an `N` for no. Because we wanted to perform a coordinated PKDS refresh to the current Active KDS, we did not have to fill in the **`New KDS`** field; we left it blank.
4. Pressed `Enter` to perform the coordinated KDS refresh.
5. From the ICSF- Coordinated KDS Refresh Confirmation panel, confirmed that we wanted to perform the coordinated KDS refresh on the active KDS. We entered a `Y` to confirm.

When we pressed `Enter` this last time, we received the `REFRESH SUCCESSFUL` message in the upper-right corner of the ICSF - Coordinated KDS Refresh panel.

The coordinated PKDS refresh was initiated from a single ICSF instance and then carried out across all other sysplex members sharing the same active PKDS. This resulted in the in-storage copy of the PKDS being updated for all ICSF instances in the sysplex that share the same active PKDS as the initiator. From the ICSF syslogs, we can see that an update took place on both the initiating system and on the other systems that are sharing the same active CKDS. From the `syslog` of the ICSF instance from which we issued the coordinated refresh, we saw the following messages:

```
CSFM622I COORDINATED REFRESH PROGRESS: NEW IN-STORAGE KDS CONSTRUCTED.
CSFM622I COORDINATED REFRESH PROGRESS: MKVPS VERIFIED BETWEEN CURRENT
ACTIVE AND TARGET DATA SETS.
CSFM622I COORDINATED REFRESH PROGRESS: NEW IN-STORAGE KDS LOADED ON
REMOTE SYSTEMS.
CSFM622I COORDINATED REFRESH PROGRESS: OPERATION TERMINATION IS
TEMPORARILY INHIBITED.
CSFM622I COORDINATED REFRESH PROGRESS: ALL FINAL PKDS DSN REFERENCES
UPDATED.
CSFM622I COORDINATED REFRESH PROGRESS: SWITCHED THE ACTIVE PKDS HASH
TABLE TO NEW.
CSFM622I COORDINATED REFRESH PROGRESS: OPERATION TERMINATION IS NOW
REENABLED.
CSFM622I COORDINATED REFRESH PROGRESS: COMPLETING CORE WORK.
CSFM617I COORDINATED REFRESH ACTION COMPLETED SUCCESSFULLY.
CSFU006I REFRESH FEEDBACK: RC=00000000 RS=00000000 SUPRC=00000000
SUPRS=00000000 FLAGS=00000000.
```

From the ICSF syslog of the system sharing the same active PKDS as the initiator, we saw the following messages:

```
CSFM622I COORDINATED REFRESH PROGRESS: ALL FINAL PKDS DSN REFERENCES
UPDATED.
CSFM622I COORDINATED REFRESH PROGRESS: SWITCHED THE ACTIVE PKDS HASH
TABLE TO NEW.
```

From these messages, we are able to see that the coordinated refresh was successful.

New callable services to translate cipher text

These new services, CSNBCTT2 or CSNBCTT3 and CSNECTT2 or CSNECTT3, provide the ability for customer applications to reencipher text from one cipher key to another key without having the text appearing in the clear outside the cryptographic coprocessor. This function is available on the zEnterprise EC12.

We successfully tested this new function using workloads that we implemented in our environment.

Stress testing

We ran stress tests against HCR77A0, which included running all of our ICSF workloads at high levels for an extended period of time. These tests ran without error.