z/OS
Cryptographic Services
Integrated Cryptographic Service Facility

Migration Health Checks - APAR OA24221

z/OS
Cryptographic Services
Integrated Cryptographic Service Facility

Migration Health Checks - APAR OA24221

Contents

Chapter 1. Health checker introduction 1

Chapter 2. Installation, Initialization, and Customization 3
Steps for installation and initialization 3
Steps to create the PKDS 3
Security considerations at first-time startup 5
Customizing ICSF after the first start 6
Changing parameters in the installation options data set 6

Chapter 3. Migration 9
Migrating from earlier software releases 9
Function Restrictions 9

Chapter 4. CSFHnnnn Messages (IBM Health Checker Processing) 11

Appendix. Diagnosis Reference Information 13
Data Areas 13
The Cryptographic Communication Vector Table (CCVT) 13

Chapter 1. Health checker introduction

The IBM Health Checker for z/OS ships as a component part of the operating system for z/OS versions 1.7 and later. The general objective of Health Checker is to periodically check settings, definitions and conditions for z/OS, the sysplex and major z/OS component middleware and facilities to detect potential problems before they can impact availability or cause outages. It is not intended to be a diagnostic or general monitoring tool, but rather a preventative resource to identify potential problems. Specific check capability is provided by the underlying component parts of the environment, and tied back to the Health Checker infrastructure for common administration, controls and reporting purposes.

There are two ICSF migration checks for the Health Checker:

1. ICSFMIG7731_ICSF_RETAINED_RSAKEY - Detection of the existence of retained RSA private keys on a PCICC or PCIXCC/CEX2C cryptographic card.

   A PCICC or PCIXCC/CEX2C card may possess the only copy of a retained RSA private key. Customers that run applications and middleware that utilize the retained key functionality of these cards are exposed to the loss of keys upon hardware failure, which may result from a problem as simple as an exhausted or malfunctioning card battery. Lost retained keys have the further implication of lost data, for retained key management keys, and an inability to verify signatures, for retained signature keys. Beginning with HCR7750, ICSF begins stepping down its support of retained keys, along with IBM cryptographic card firmware (at certain firmware release levels). An ICSF migration check informs customers that ICSF retained key functionality is deprecating. The customer or application/middleware provider must migrate to an alternative retained key strategy.
2. ICSFMIG7731_ICSF_PKDS_TO_4096BIT - Verification that the PKDS size in an ICSF pre-HCR7750 environment is sufficiently allocated to support 4096-bit RSA keys.

   HCR7750 introduces ICSF support for 4096-bit RSA keys, which requires a larger PKDS than prior ICSF releases needed. If a pre-HCR7750 ICSF customer migrates to HCR7750 without first reallocating the PKDS for 4096-bit key support, ICSF will fail to start. An ICSF migration check detects the case where the currently active PKDS is not sufficiently allocated for the HCR7750 environment and informs the customer that a PKDS reallocation action is necessary.

Health Checker resolves check routines from the current system linklist. The z/OS linklist concatenation needs to properly reflect the operating level release of ICSF (for example, HCR7731 must be higher in the list than HCR7750), or Health Checker must be started with a steplib addressing the operational level release of ICSF. Otherwise, Health Checker will fail to resolve the correct level of ICSF migration check modules.

For information on activating these checks, consult *IBM Health Checker for z/OS: User's Guide*.

Chapter 2. Installation, Initialization, and Customization

Steps for installation and initialization

Steps to create the PKDS

The PKDS must be allocated and the PKDS data set name must be specified on the PKDSN parameter of the options data set when you first start ICSF. ICSF support for the PCI Cryptographic Coprocessor or PCI X Cryptographic Coprocessor/Crypto Express2 Coprocessor requires a PKDS. Even if not available at first time start up, a PCICC, PCIXCC or CEX2C can be dynamically configured online. Since ICSF can not tell if a PCICC, PCIXCC or CEX2C will be added, it requires the PKDS to be available at start up.

The PKDS must be a key-sequenced data set with variable length records. Allocate the PKDS on a permanently resident volume.

1. Determine the amount of primary space you need to allocate for the PKDS.

   This should reflect the total number of entries you expect the data set to contain originally. The PKDS will contain both public and private PKA keys. Each record has a maximum size of 3.8 KB. The average record length for a private key is 1 KB, and for a public key is 0.5 KB. Allocate space for a minimum of two private keys, one for digital signatures, and another for encipherment. In addition, allocate enough space for the number of public keys you expect to store in the PKDS. The number of public keys varies from system to system. Generally, only those keys that are received from other users or systems are stored in the PKDS. The public keys are used to send messages to the owners of the public keys.
2. Determine the amount of secondary space to allocate for the PKDS.

   This should reflect the total number of entries you expect to add to the data set.

   To access keys, VSAM uses the key label as the VSAM key. This means that VSAM adds keys to the data set in collating sequence. That is, if two keys named A and B are in the data set, A appears earlier in the data set than B. As a result, adding keys to the data set can cause multiple VSAM control interval splits and control area splits. For example, a split might occur if the data set contains keys A, B, and E and you add C. In this case, C must be placed between B and E.

   The amount of secondary space you allocate must take into account the number of control interval and control area splits that might occur. If the PKDS uses a significant amount of secondary space, you can copy it into another disk copy that you created with more primary space. You can do this by using the Access Method Services (AMS) REPRO command or the AMS EXPORT/IMPORT commands.

   The BUFFERSPACE parameter on the AMS DEFINE CLUSTER command lets VSAM optimize space for control area and control interval splits.
3. Create an empty VSAM data set to use as the PKDS. Use the AMS DEFINE CLUSTER command to define the data set and to allocate its space. ICSF provides a sample job to define the PKDS in member CSFPKDS of SYS1.SAMPLIB.

   Note: To improve security and reliability of the data that is stored on the PKDS:

   - Use the ERASE and WRITECHECK parameters on the AMS DEFINE CLUSTER command. ERASE overwrites data records with binary

zeros when the PKDS cluster is deleted. WRITECHECK provides hardware verification of all data that is written to the data set.

- Create a Security Server (RACF) data set profile for the PKDS.
- The CISZ(8192) coded in this sample in the DATA section is a hardcoded requirement.

4. Allocate a disk copy of the PKDS by defining a VSAM cluster as in this SYS1.SAMPLIB CSFPKDS member sample:

```
//CSFPKDS JOB = JOB CARD PARAMETERS
//*******************************************************************
//* Licensed Materials - Property of IBM                            *
//* 5694-A01                                                        *
//* (C) Copyright IBM Corp. 2002, 2007                              *
//*                                                                 *
//* THIS JCL DEFINES A VSAM PKDS TO USE FOR ICSF                    *
//*                                                                 *
//* CAUTION: This is neither a JCL procedure nor a complete JOB.    *
//* Before using this JOB step, you will have to make the following *
//* modifications:                                                  *
//*                                                                 *
//* 1) Add the job parameters to meet your system requirements.     *
//* 2) Be sure to change CSF to the approrpriate HLQ if you choose  *
//*    not to use the default.                                      *
//* 3) Change xxxxxx to the volid where you want your PKDS to       *
//*    reside. The PKDS needs to be on a permanently resident       *
//*    volume.                                                      *
//*                                                                 *
//*******************************************************************
//DEFINE EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
  DEFINE CLUSTER (NAME(CSF.CSFPKDS) -
                  VOLUMES(XXXXXX) -
                  RECORDS(100,50) -
                  RECORDSIZE(350,3800) -
                  KEYS(72 0) -
                  FREESPACE(0,0) -
                  SHAREOPTIONS(2,3)) -
           DATA (NAME(CSF.CSFPKDS.DATA) -
                  BUFFERSPACE(100000) -
                  ERASE -
                  CISZ(8192) -
                  WRITECHECK) -
         INDEX (NAME(CSF.CSFPKDS.INDEX))
/*
```

You can change and use the Job Control Language according to the needs of your installation. Please note that the JCL to define the PKDS differs from the JCL that defines the CKDS (RECORDSIZE and CISZ parameters).

Migrating to a larger PKDS

For FMID HCR7750, the LRECL for the PKDS has changed and customers must migrate to a new PKDS. The PKDS must be migrated before the customers can start HCR7750.

To create a larger PKDS, use the JCL in SYS1.SAMPLIB(CSFPKDS).

If the PKDS is shared with down-level systems, install the toleration APAR on those systems to allow continued sharing of the PKDS. ICSF will be able to interact with both sizes of the PKDS. The toleration APAR number is OA21807.

The steps to migrate are:

1. Make a backup copy of the PKDS you are currently using.

2. Create a larger PKDS - use the JCL in SYS1.SAMPLIB(CSFPKDS) from the HCR7750 system. If the PKDS will be shared, place the VSAM dataset where it can be shared.
3. Copy the old PKDS to the larger PKDS using the JCL in SYS1.SAMPLIB(CSFPKDCP) from the HCR7750 system. Protect the VSAM data sets from viewing by non-authorized personnel.

```
//CSFPKDCP <JOB CARD PARAMETERS>
//*********************************************************************
//* Licensed Materials - Property of IBM                              *
//* 5694-A01                                                          *
//* Copyright IBM Corp. 2007                                          *
//*                                                                   *
//* THIS JCL COPIES ONE VSAM PKDS TO THE LARGER PKDS                  *
//*                                                                   *
//* CAUTION: This is neither a JCL procedure nor a complete JOB.      *
//* Before using this JOB step, you will have to make the following   *
//* modifications:                                                    *
//*                                                                   *
//* 1) Add the job parameters to meet your system requirements.       *
//* 2) Be sure to change CSF to the appropriate HLQ if you choose     *
//* not to use the default.                                           *
//*                                                                   *
//*********************************************************************
//STEP1 EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD SYSOUT=*
//INDD DD DSN=CSF.CSFPKDS.OLD,DISP=SHR
//OUTDD DD DSN=CSF.CSFPKDS,DISP=SHR
//SYSIN DD *
REPRO INFILE(INDD) OUTFILE(OUTDD)
/*
```

4. Update the ICSF started procedures on all systems to reference the new PKDS. If the new PKDS was empty when ICSF was started, you must initialize it before PKA callable services can be enabled.
5. Stop and restart ICSF on each system. If the system has not been IPLed since HCR7750 was installed, then an IPL is required.

Security considerations at first-time startup

ICSF provides two migration checks to provide customer installations the advantages offered by Health Checker. An ICSF migration check informs customers that ICSF retained key functionality is deprecating and that the customer or application/middleware provider must migrate to an alternative retained key strategy. Another migration check verifies that the PKDS size in an ICSF pre-HCR7750 environment is sufficiently allocated to support 4096-bit RSA keys.

If the ICSFMIG7731_ICSF_RETAINED_RSAKEY migration check detects retained keys in a coprocessor, report messages CSFH008R-CSFH0011R will disclose key labels that could be considered security-sensitive information. The checks routine will only utilize HZSFMSG to convey this information. Consequently, the key label data will be written only to the Health Checker's message buffer, where it can be contained and controlled with Health Checker security mechanisms. For information on creating security definitions, consult *IBM Health Checker for z/OS: User's Guide*. Placing an S next to the Healthcheck to browse the message buffer will display the CSFH* messages.

Note: Most CSFH message IDs are not displayed.

If you configure CHECKAUTH(YES) in the ICSF options dataset, the Health Checker address space user identity must be authorized to the CSFRKL profile in

class CSFSERV for the ICSFMIG7731_ICSF_RETAINED_RSAKEY migration check to successfully execute. If you configure CHECKAUTH(NO), there is no requirement to authorize the Health Checker user identity for any ICSF profiles or classes, since the check routine executes in supervisor state. This is not an implementation consideration, but rather a check deployment or activation time customer administration consideration.

Customizing ICSF after the first start

The startup procedure includes a CSFPARM DD statement, which gives the name of the installation options data set. The installation options data set includes a CKDSN option, which gives the names of the CKDS, and a PKDSN option, which gives the name of the PKDS.

After the first start, whenever you restart ICSF, the CKDS named in the installation options data set becomes the active in-storage CKDS.

To change the active in-storage CKDS (or any other installation option), change the option value in the installation options data set and **stop and restart ICSF**.

Changing parameters in the installation options data set

The installation options data set is an intended programming interface.

When specifying parameter values within parentheses, leading and trailing blanks are ignored. Embedded blanks may cause unpredictable results.

Support is provided for the use of system symbols in the installation options data set. System symbols can be used as values for any of the parameters. System symbols are specified in the IEASYMxx member of SYS1.PARMLIB; the IEASYM statement of the LOADxx member of SYS1.PARMLIB specifies the IEASYMxx member(s) to be used for the resolution of system symbols. This example shows the use of a system symbol for specifying the domain to be used for the start of ICSF:

```
DOMAIN(&PARDOM.)
```

When the Installation Options Data Set is processed during the start of ICSF, the value of the system symbol PARDOM will be substituted as the value of the DOMAIN parameter.

For the first start, you specified an empty VSAM data set name for the CKDS in the CKDSN option, an empty VSAM data set name for the PKDS in the PKDSN option, and SSM(YES). You may want to change these and other options for subsequent starts. Here is a complete list of installation options:

CHECKAUTH(YES or NO)
: Indicates whether ICSF performs security access control checking of Supervisor State and System Key callers. If you specify CHECKAUTH(YES), ICSF issues RACROUTE calls to perform the security access control checking and the results are logged in RACF SMF records that are cut by RACF. If you specify CHECKAUTH(NO), the authorization checks against resources in the CSFSERV class are not performed resulting in a significant performance enhancement for supervisor state and system key callers. However, the authorization checks are not logged in the RACF SMF records.

If you do not specify the CHECKAUTH option, the default is CHECKAUTH(NO).

If you configure CHECKAUTH(YES) in the ICSF options dataset, the Health Checker address space user identity must be authorized to the CSFRKL profile in class CSFSERV for the ICSFMIG7731_ICSF_RETAINED_RSAKEY migration check to successfully execute. However, you have no action to take if you choose not to run the migration check. If you configure CHECKAUTH(NO), there is no requirement to authorize the Health Checker user identity for any ICSF profiles or classes, since the check routine executes in supervisor state. This is not an implementation consideration, but rather a check deployment or activation time customer administration consideration.

Chapter 3. Migration

This topic describes migration considerations.

Your plan for migrating to the new level of ICSF should include information from a variety of sources. These sources of information describe topics such as coexistence, service, hardware and software requirements, installation and migration procedures, and interface changes.

For complete information on the migration health checks, see *IBM Health Checker for z/OS: User's Guide*. For complete information on migration, see *IBM z/OS Migration*.

Attention: Although you are migrating to a new release, you should review the information in Chapter 2, "Installation, Initialization, and Customization," on page 3; especially review customization steps that may have changed since your last migration.

An IPL is required when installing a new release of ICSF (it is possible for ICSF control blocks like the DACC and CCVT to persist in storage across an ICSF restart).

Migrating from earlier software releases

These topics describe common activities and considerations that should be considered when you migrate from:

- an earlier release of ICSF to HCR7750

Function Restrictions

Retained keys are RSA private keys that are stored in a cryptographic coprocessor instead of in the public key storage data set. This change does not affect retained keys that you are currently using, that is, keys that are stored on the cryptographic coprocessor. However, starting with ICSF HCR7750, the ICSF services do not allow you to store RSA keys intended for key management use in a cryptographic coprocessor . Your applications can continue to store RSA private keys intended for signature usage in the cryptographic coprocessor. Although HCR7750 introduces support for 4096-bit modulus RSA keys, signature usage retained keys are limited to 2048-bit modulus.

The 2048-bit RSA keys may have an public exponent, e, in the range of $1<e<2^{2048}$. and e must be odd. The RSA public key exponents for 2049-bit to 4096-bit RSA keys are restricted to the values 3 and 65537

ICSF delivers the migration check support for PKDS compatibility with 4096-bit RSA key support for HCR7731 and z/OS V1.9 (ICSF release HCR7740) only, but not for any releases earlier or later. This check support also requires that its delivery PTF pre-req the PTF for the PKDS 4096-bit key toleration - APAR OA21807. Doing so ensures that you have enabled the opportunity for a sufficiently allocated PKDS, and avoids the problem where you attempt to properly allocate a PKDS for 4096-bit RSA keys, but then find the current ICSF service level fails to support it.

Chapter 4. CSFHnnnn Messages (IBM Health Checker Processing)

Chapter 4, "CSFHnnnn Messages (IBM Health Checker Processing)" describes messages that ICSF issues while processing health checks. See *IBM Health Checker for z/OS: User's Guide* for more information.

Information on the router and descriptor codes is found in *z/OS ICSF Messages*.

These messages are placed in the Health Check message buffer. CSFH0003E and CSFH0005E are also displayed in the syslog.

CSFH0001I The (IBMICSF, ICSFMIG7731_ICSF_RETAINED_RSAKEY) check found no apparent retained RSA key use on this system because there are no online coprocessors that support retained keys.

Explanation: A migration check (IBMICSF, ICSFMIG7731_ICSF_RETAINED_RSAKEY) indicates that there is no apparent retained RSA key use on this system because there are no online processors.

System action: Processing continues.

CSFH0002I Cryptographic coprocessors were examined and the (IBMICSF, ICSFMIG7731_ICSF_RETAINED_RSAKEY) check found no apparent retained RSA keys use on this system.

Explanation: A migration check (IBMICSF, ICSFMIG7731_ICSF_RETAINED_RSAKEY) indicates that there is no apparent retained RSA key use on this system.

System action: Processing continues.

CSFH0003E Cryptographic coprocessors were examined and found to possess retained RSA keys.

Explanation: Online coprocessors possess one or more retained RSA keys, implying retained RSA keys are potentially being used on this system. ICSF is deprecating its retained RSA key support. The migration check used is (IBMICSF, ICSFMIG7731_ICSF_RETAINED_RSAKEY).

System action: There is no effect on the system.

Operator response: Report this exception to the system programmer.

System programmer response: Alert the installation security administrator and application and middleware administrators for this system.

Problem determination: Investigate the cryptographic services utilized by the workload executed on this system. Determine application and middleware products using retained RSA key services for key management use and depend upon the key labels listed in the report. Develop an immediate strategy to remove any dependencies on creating new ICSF-supported retained RSA keys prior to migration to ICSF release level HCR7750, and an eventual strategy to remove any dependencies on ICSF-supported retained key interfaces.

CSFH0004I The (IBMICSF, ICSFMIG7731_ICSF_PKDS_TO_4096BIT) check found the currently active PKDS to be sufficiently allocated for ICSF HCR7750 release, and later, 4096-bit RSA key support.

Explanation: A migration check (IBMICSF, ICSFMIG7731_ICSF_PKDS_TO_4096BIT) indicates that the current PKDS is sufficiently allocated for 4096-bit key support.

System action: There is no effect on the system.

CSFH0005E The currently active PKDS is not sufficiently allocated for ICSF 4096-bit RSA key support.

Explanation: The active PKDS does not support a record size that is sufficiently large for 4096-bit RSA keys. ICSF, at the HCR7750 and later release levels, will be unable to successfully start with this PKDS. The migration check used is (IBMICSF, ICSFMIG7731_ICSF_PKDS_TO_4096BIT).

System action: There is no effect on the system.

Operator response: Report this exception to the system programmer.

System programmer response: Alert the installation security administrator for this system.

Problem determination: : Determine the currently active PKDS from the Administrative Control Functions option of the primary ICSF administrator panel. Refer to the ICSF (HCR7750 or later release) System Programmer's Guide for instruction on how to reallocate

| the (or allocate a new) PKDS data set for compatibility
| with 4096-bit RSA keys.
|

Appendix. Diagnosis Reference Information

See *IBM Health Checker for z/OS: User's Guide* for information on creating security definitions, viewing check status and working with Health Checker output. If a program error occurs, look for additional information in the system console log, the logrec data set, and the message buffer.

This appendix contains a description of the Cryptographic Communication Vector Table (CCVT).

Data Areas

This topic presents the format of the Cryptographic Communication Vector Table (CCVT).

The Cryptographic Communication Vector Table (CCVT)

The CCVT is the ICSF base control block and contains addresses of common areas for use by ICSF components. Indicators in the CCVT also provide ICSF status information. The CCVT is getmained in subpool 245 under the line.

Programming Interface information

CCVT

ONLY these fields are part of the programming interface:

- CCVTDACC
- CCVTCCVE
- CCVTHFLG
- CCVTPRPC
- CCVTINST
- CCVTINS2
- CCVTLNTH
- CCVTFMID
- CCVT_USERPARM

End of Programming Interface information

Table 1 describes the contents of the Cryptographic Communication Vector Table.

Table 1. Cryptographic Communication Vector Table

Offset (Dec)	Number of Bytes	Field Name	Description
0	4	CCVTID	EBCDIC Cryptographic Communication Vector Table ID. This field must contain the character string `CCVT`.
4	2	CCVTVER	Version. The version of the CCVT. This field must contain the character string `03`
6	2	CCVTLEN	Length. The length of the CCVT. The value of this field is 296 in decimal.

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
8	1	CCVTAUX	Auxilliary flags. **Bit** **Meaning When Set On** **0** ICSF is terminating.
9	5		Reserved.
14	2	CCVTRLVL	ICSF level.
16	4	CCVTCCVE	Cryptographic Communication Vector Table Extension (CCVE) address. The address of a private area extension of the CCVT. You should place any fields not needed by other address spaces in the CCVE.
20	4		Reserved
24	4	CCVTPC2	PC number for entry into module CSFASSPC.
28	4	CCVTPRPC	Entry point for the pre-PC processing module, CSFARPC.
32	4	CCVTINST	For installation use.
36	1	CCVTSFG1	Status byte. **Bit** **Meaning When Set On** **0** ICSF services are active. **1** At least one Integrated Cryptographic Feature has a valid master key. **2** ICSF initialization complete. **3** ICSF is active and PCF is not active. **4** Compatibility is permitted. COMPAT(YES) or COMPAT(COEXIST) is specified. **5** At least one Integrated Cryptographic Feature is valid. **6** SEC 250 or greater. **7** S/390 Enterprise Servers and S/390 Multiprise Cryptographic Coprocessor Feature is in use.

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
37	1	CCVTFLAG	Flag byte. **Bit** — **Meaning When Set On** **0** — VTAM puts terminal buffer greater than 16MB line. **1** — Hardware environment tested. **2** — GTMAC opcode in hardware. **3** — PCI Cryptographic Coprocessor hardware instructions available. **4** — At least one PCI Cryptographic Coprocessor is active. **5** — Additional hardware environment tested. **6** — At least one PCI Cryptographic Coprocessor is online. **7** — At least one PCI Cryptographic Coprocessor is present.
38	1	CCVTOFLG	Operational flag byte. **Bit** — **Meaning When Set On** **0** — Configuration is under PR/SM. **1** — Close the CKDS **2** — Key record create, key record delete, and key record write disallowed. **3** — I/O subtask is available. **4** — CCVT_DEF_ALG bit. If on, CDMF is the system default algorithm; if off, DES is the default. **5** — CCVT_CDMF_ENA bit. If on, hardware is capable of performing CDMF. **6** — PKA master keys are valid. **7** — Use ICSF reason codes.
39	1	CCVTSVCM	SVC number for key management. This is the PCF compatibility SVC.
40	1	CCVTCDX	Old CDX, constant this IPL.
41	1	CCVTSVCS	SVC number for DES interface SVC. This is the PCF compatibility SVC.
42	2	CCVTASID	ASID of ICSF address space.
44	4	CCVTIDNR	Subtask caller ID.
48	4	CCVTPC3	Entry point to CSFASSPA used by compatibility SVCs.
52	4	CCVTSRUT	Address of the access method module.
56	8	CCVTINS2	An 8-byte area for installation use.

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
64	4	CCVTMDS	Data space server PC. PC number for entry to data space server that adds and deletes the in-storage CKDS.
68	4	CCVTLNTH	Maximum installation data length.
72	4	CCVTASCB	ICSF ASCB address.
76	4	CCVTWLST	Address of CICS Wait List.
80	1	CCVTHFLG	Flag bytes. **Bit** — **Meaning When Set On** **0** — Crypto assist instructions available. **1** — Additional secure Crypto device available. **2** — Support for 64-bit callers. **3** — ICSF Cross-System Services environment is active for CKDS **4** — ICSF Cross-System Services environment is active for TKDS **5** — RSA 4096-bit function enabled and the RNGL service is available **6-7** — Reserved.
81	1	CCVTSFLG	Flag bytes. **Bit** — **Meaning When Set On** **0** — ICSF during initialization. **1-7** — Reserved.
82	2		Reserved.
84	4	CCVTENF	ECB for ENF listen.
88	4	CCVTTCB	ICSF maintask TCB address.
92	4	CCVTTRC	ECB for component trace.
96	4	CCVTECBA	Address of CAMQ ECB array.
100	4	CCVTENF1	Token for ENF listen exit.
104	4	CCVTENF2	Token for ENF listen exit.
108	4	CCVTLX	LX of ICSF address space.
112	8	CCVTDS	Bytes 0–3: Address of the beginning of the current data space. Bytes 4–7: ALET (Access list entry token) of the current data space.
120	4	CCVTLFDE	ECB to post to start the task to search for disabled Integrated Cryptographic Features.
124	4	CCVTIOSE	ECB to post to use I/O subtask.
128	4	CCVTIOSC	ECB posted by I/O subtask.

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
132	4	CCVTIOAS	ASCB address for non-CSF address space.
136	8	CCVTFMID	ICSF FMID.
144	8	CCVT_USERPARM	ICSF user parameter.
152	1	CCVTPKAF	PKA register clear key entry processing flags. **Bit** — **Meaning When Set On** **0** — KMMK is valid for CP0. **1** — SMK is valid for CP0. **2** — KMMK has been reset for CP0. **3** — SMK has been reset for CP0. **4** — KMMK is valid for CP1. **5** — SMK is valid for CP1. **6** — KMMK has been reset for CP1. **7** — SMK has been reset for CP1.
153	1	CCVTPKAR	**Bit** — **Meaning When Set On** **0 and 1** — SMK status for KSU0. **2 and 3** — KMMK status for KSU0. **4 and 5** — SMK status for KSU1. **6 and 7** — KMMK status for KSU1.
154	1	CCVTPKAX	PKA register status (reserved).
155	1	CCVTKAZ	PKA register status (reserved).
156	16	CCVTCCC	Cryptographic configuration control (CCC).
172	4	CCVTSPKB	Address of public key build.
176	4	CCVTSPKX	Address of public key extract.
180	4	CCVTPIOE	ECB for PKDS I/O subtask.
184	4	CCVTPIOC	ECB for PKDS I/O work complete.
188	4	CCVTPIOA	Address of ASCB task posting the PKDS I/O subtask.
192	4	CCVTIDNR_PKDS	I/O subtask caller identification.
196	1	CCVTPKDF	PKDS processing flags. **Bit** — **Meaning When Set On** **0** — PKDS available. **1** — Signal PKDS I/O to close PKDS. **2** — At least one PCICA is active. **3-7** — Reserved.
197	1	CCVTCICS	CICS processing flags. **Bit** — **Meaning When Set On** **0** — CSFAPRPD installed. **1** — CSFACKWL installed.

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
198	1	CCVTYAFF	**Bit** **Meaning When Set On** **0** ZKA compliance environment.
199	1	CSFTTKDF	TKDS processing flags **Bit** **Meaning When Set On** **0** TKDS available **1** Signal TKDS I/O to close TKDS **2-7** Reserved
200	4	CCVTPRPD	Address of CSFAPRPD.
204	4	CCVTCKWL	Address of CSFACKWL.
208	4	CCVTPC4	PC4 (CSFMCAMP) number.
212	4	CCVTPERR	Error routine for POST macro.
216	4	CCVTENF3	ENF token for PCI Cryptographic Coprocessor online event.
220	4	CCVTENFP	ECB for PCI Cryptographic Coprocessor online event.
224	4	CCVTENA1	Address of ENF1 listen exit.
228	4	CCVTENA2	Address of ENF2 listen exit.
232	4	CCVTENA3	Address of ENF3 listen exit.
236	4	CCVTPC5	PC5 (CSFMCCPP entry).
240	4	CCVTPC6	PC6 (CSFMWCFS entry).
244	16	CCVT_KXMD	Hardware feature status. **Bit** **Meaning When Set On** **0** Reserved. **1** SHA-1 enabled. **2** SHA-256 enabled. **3** SHA-512 enabled. **4-7** Reserved. Bytes 2–16 are reserved.
260	4	CCVTGSVT	Address of generic service vector.
264	4	CCVTGSFL	Flags.
268	4	CCVTCSVG	Address of CSFSCVG.
272	4	CCVTACVG	Address of CSFACVG.
276	4	CCVTDACC	ICSF DAC instructions control block for RMF.

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
280	16	CCVT_KMC_EXPORT	Hardware feature status. **Bit** **Meaning When Set On** **0** Reserved. **1** KMC DES enabled. **2** Reserved. **3** KMC TDES enabled. **4-17** Reserved. **18** KMC AES 128 key enabled. **19** KMC AES 192 key enabled. **20** KMC AES 256 key enabled. **21-23** Reserved. Change bytes 4-16 are reserved.
296	4	CCVTPC7	PC7 (CSFMGARM entry)
300	4	CCVTPC8	PC8 (CSFMGTRM entry)
304	8	CCVTGART	Token of CSFMGARC resource manager
312	8	CCVTGTRT	Token of CSFMGTRC resource manager
320	4	CCVTGARC	Address of CSFMGARC resource manager
324	4	CCVTGTRC	Address of CSFMGTRC resource manager
328	4	CCVT_IDENTITY	Identifier
332	6	CCVT_GARCDATE	Compile date of CSFMGARC (EBCDIC)
338	6	CCVT_GTRCDATE	Compile date of CSFMGTRC (EBCDIC)
344	4	CCVTEPRP	Address of CSFEPRPC
348	4	CCVTPKB6	Address of CSFSPKB6
352	4	CCVTVRET	Address of CSFVRET
356	4	CCVTWRET	Address of CSFWRET
360	4	CCVTENA4	Address of ENF4 listen exit
364	4	CCVTENF4	ENF Token for sysplex member leave/join event
368	4	CCVTSPXE	ECB to post for sysplex member leave/join event
372	4	CCVT_CCSTCB	Address of TCB of of Cross-System Services task
376	4	CCVTCSS	Address of Cross-Services Block
380	4	CCVTTIOE	ECB for TKDS I/O subtask
384	4	CCVTTIOC	ECB for TKDS I/O work complete
388	4	CCVTTIOA	Address of ASCB task posting the TKDS I/O subtask
392	4	CCVTIDNR TKDS	Subtask called ID for TKDS
396	4	*	Reserved
400	4	CCVTTPLEX_ECB	ECB to post for TKDS sysplex member leave/join event
404	4	CCVTPPLEX_ECB	ECB to post for PKDS sysplex member leave/join event
408	4	CCVT_CSST_TCB	Address of Cross-System Services TCB for TKDS

Table 1. Cryptographic Communication Vector Table (continued)

Offset (Dec)	Number of Bytes	Field Name	Description
412	4	CCVTCSST	Address of Cross-System Services block for TKDS sharing
416	4	CCVT_CSSP_TCB	Address of Cross-System Services TCB for PKDS
420	4	CCVTCSSP	Address of Cross-System Services block for PKDS sharing
424	4	CCVTTDSA	Address of beginning of TKDS data space
428	4	CCVTTDST	ALET of TKDS data space
432	8	CCVTTDSK	STOKEN of TKDS data space
440	4	CCVTDSR	ECB to post for data space management requests
444	4	CCVTDSC	ECB to post for data space management work complete
448	4	CCVTDSAS	ASCB address for non-CSF address
452	2	CCVT_PKDS_MAXLREC	Maximum logical record length for PKDS records
454	26	*	Reserved
480		*	Ensure CCVT ends on doubleword boundary.

Program Number: 5964-A01

Printed in USA