Oracle Key Manager

OKM-ICSF Integration Guide

Version 2.3

November 2010
Revision AA

Submit comments about this document by clicking the Feedback [+] link at: http://docs.sun.com

Oracle Key Manager (OKM) Integration Guide

Contents

Revision History

Date	Revision	Description
November, 2010	AA	*Oracle Key Manager 2.3 OKM-ICSF Integration Guide*

Preface

This guide provides information for the interface between the Oracle Key Manager (OKM) and the IBM® Integrated Cryptography Service Facility (ICSF). It is intended for mainframe system programmers and operators responsible for configuring and maintaining the OKM software at their site.

Note – The product name, Key Management System (KMS), has been renamed to Oracle Key Manager (OKM). References to the KMS, most of its components and concepts, have been changed accordingly.

Related Documentation

The following list contains the names and order numbers of publications that provide additional information about OKM.

The online documentation is available at:

http://www.oracle.com/technetwork/indexes/documentation/index.html

Function	Title
	Hardware Publications
Installation planning for the encryption solution	*Systems Assurance Guide*
	Software Publications
OKM software configuration and maintenance	*Administration Guide*
Interface between the OKM and IBM Integrated Cryptography Service Facility (ICSF)	*Integration Guide*

Documentation, Support, and Training

Function	URL
Documentation ■ Customer: ■ Employee: ■ Partner:	 ■ `http://docs.sun.com` ■ `http://docs.sfbay.sun.com/` ■ `https://spe.sun.com/spx/control/Login`
Downloads ■ Customer: ■ Employee:	 ■ `http://www.sun.com/download/index.jsp` ■ https://dlrequest-zn-dlapps1.sfbay.sun.com/usr/login
Support	■ `http://www.sun.com/support/`
Training	■ http://www.oracle.com/global/us/education/sun_select_country.html
Sun Online Account	■ `https://reg.sun.com/register`

Oracle Welcomes Your Comments

Oracle is interested in improving its documentation and welcomes your comments and suggestions. Submit your comments by clicking the Feedback[+] link at:

`http://docs.sun.com`

Please include the title of your document with your feedback:

OKM-ICSF Integration Guide

CHAPTER 1

OKM-ICSF Integration Overview

Key Stores and Master Key Mode

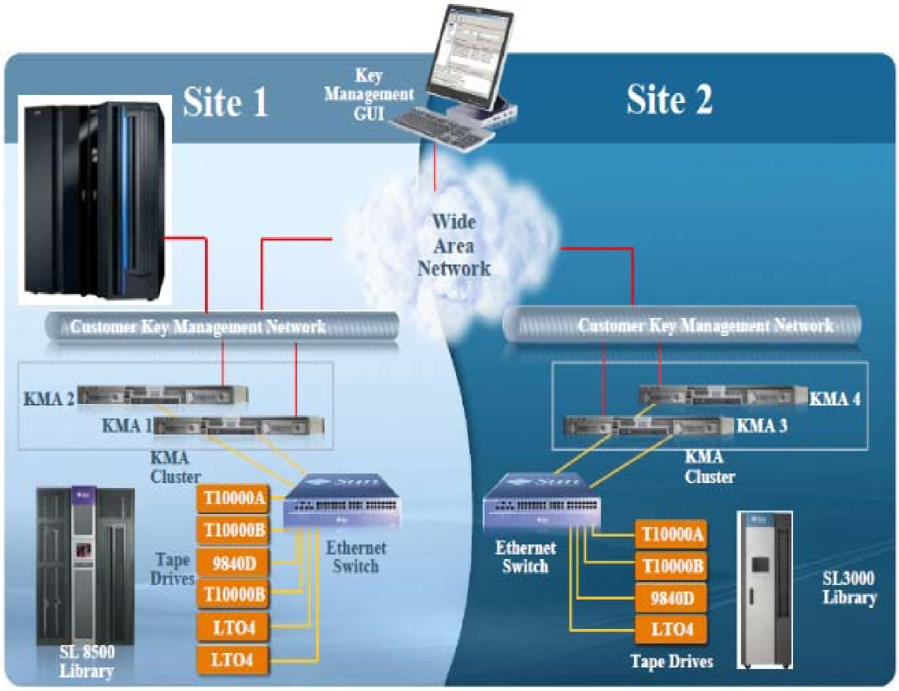
In KMS 2.0.x and KMS 2.1, the KMAs in an Oracle Key Manager (OKM) Cluster generate their own keys using their Sun Cryptographic Accelerator (SCA) 6000 cards. Some customers prefer to have the KMAs use master keys that are created and stored in an external key store.

KMS 2.2 introduces a Master Key Mode feature. When this feature is enabled, the OKM Cluster derives tape keys from a set of master keys. The master keys are created and stored in an external key store. Full disaster recovery is possible with just the tapes, the master keys, and factory default OKM equipment.

Understanding the Solution

In this solution, the external key store resides in an IBM mainframe and is accessed using a TLS/XML protocol. This protocol is supported in the IBM mainframe with the keys stored in a Token Data Set in the IBM Integrated Cryptography Service Facility (ICSF). FIGURE 1-1 shows a typical configuration.

FIGURE 1-1 Site Configurations

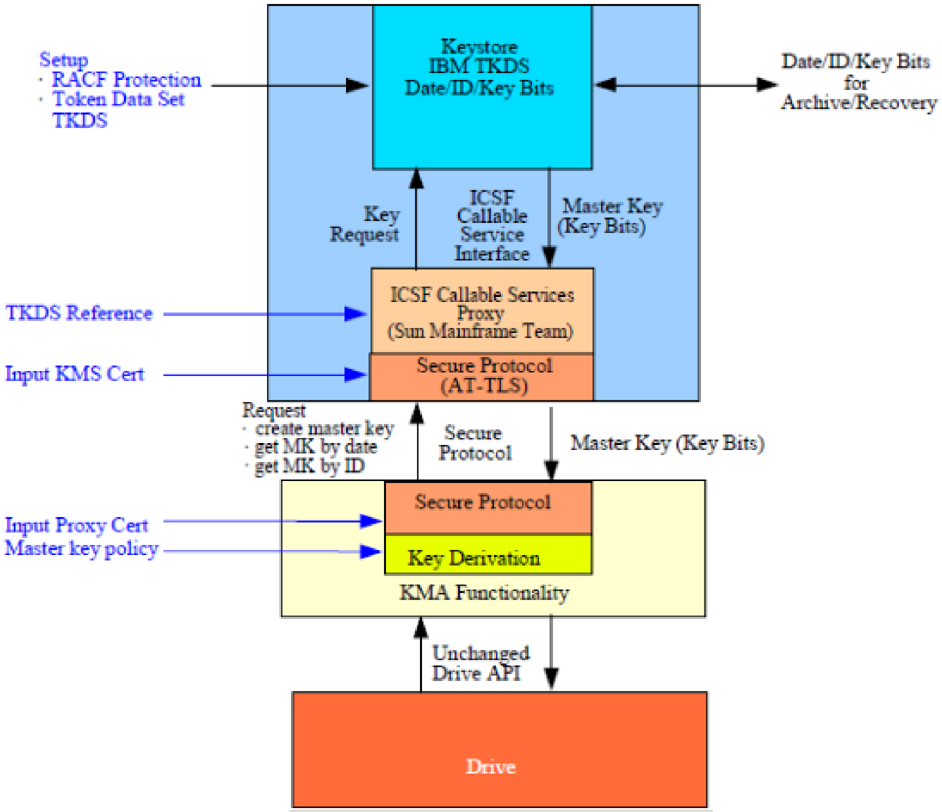


The OKM Cluster periodically issues requests to the IBM mainframe, asking to create new master keys (referred to as *application keys* in ICSF) and to return them to the OKM Cluster. The KMAs then use these new master keys to derive new tape keys.

Defining the System Components

The following components comprise the integration solution and are discussed in this section:

- "KeyStore" on page 4
- "Interface" on page 4
- "Transfer Security" on page 4
- "Key Derivation" on page 4
- "Key Policy" on page 5
- "Key Recovery" on page 5



KeyStore

Master (application) keys are stored in the Token Data Set (TKDS), as defined in the IBM ICSF documentation. The TKDS is identified in the ICSF installation options data set. The z/OS system programmer can create the TKDS by using the IDCAMS utility.

Keys stored in the TKDS are not encrypted, but access to the data set itself, as well as Callable Services and Tokens (key sets), is controlled by RACF or an equivalent. Access to the TKDS can be defined by the current policy for backup and restore of Master Keys.

Interface

You must add a module to the existing Sun Mainframe Software to implement an ICSF Callable Services Proxy. This Proxy allows the OKM Cluster to call PKCS#11 functions to access the KeyStore. Secure communication with the OKM Cluster is implemented using the z/OS Application Transparent - Transport Layer Security (AT-TLS) on the IBM mainframe.

AT-TLS is an encryption solution for TCP/IP applications that is completely transparent to the application client and server. Packet encryption and decryption occurs in the z/OS TCPIP address space at the TCP protocol level. The encrypted packet payload is unintelligible when sniffed or traced, but by the time it is delivered to the application the payload is once again readable.

Transfer Security

The OKM Cluster implements a Transport Layer Security (TLS) protocol to communicate with the Proxy on the IBM mainframe.

The z/OS system programmer generates and then exports two self-signed X.509v3 certificates and one RSA 2048-bit public key pair, and then transfers them (using FTP) off the IBM mainframe. The first certificate is a Root Certificate Authority (CA) certificate. The system programmer uses this Root CA certificate to generate the Client Certificate and Key Pair. These certificates and the key pair are manually installed in the IBM mainframe and configured using RACF and AT-TLS so that the Proxy can identify a valid OKM request. The certificates and the private key of the key pair are installed in the OKM Cluster so that it can authenticate the Proxy. As a result, only KMAs in a valid OKM Cluster can issue requests to the Proxy, and they accept a response only from a valid Proxy.

Key Derivation

The OKM Cluster accepts a Master Key Value and 18-byte Master Key ID from the Proxy. It creates a 30-byte Key ID by concatenating a 2-byte header and the 18-byte Master Key ID with an internally generated 10-byte value. It then creates a Derived Key Value by encrypting the Key ID (padded to 32 bytes) with the Master Key Value.

Key management between Drives and the OKM Cluster continue to use the current OKM strategy. Thus, no firmware upgrades are required.

Key Policy

The OKM Cluster controls the Master Key lifecycle. It requests a current Master Key value from the Proxy based on the current date. The Proxy retrieves the current Master Key from the TKDS using a sequence of PKCS#11 function calls. If there is no current Master Key Value, the OKM Cluster issues a Create Master Key request to the Proxy. The OKM can then re-submit the request for a current Master Key Value from the Proxy.

Key Recovery

The OKM Cluster retains all derived Keys and Key IDs it creates. If the Cluster does not have the Key for a specified set of written data, it can re-derive the Key by forming the Master Key ID from the Key ID and then issuing a retrieve request to the Proxy to get the Master Key Value stored in the TKDS. The OKM can then re-derive the Key Value to enable its Agent to read the data.

This key recovery mechanism allows “ground-level up” recovery of all tapes encrypted by this system, based only on availability of archived Master Keys in the TKDS.

System Requirements

The IBM mainframe and the OKM Cluster both have system requirements in this solution.

IBM Mainframe

The IBM z/OS mainframe must be running ICSF HCR-7740 or higher and Sun ELS 7.0 or NCS 6.2 along with associated PTFs. A CEX2C cryptographic card must also be installed on the IBM mainframe.

OKM Cluster

The OKM Cluster must be running KMS 2.2 or higher and must be using Replication Version 11 or higher. KMAs are shipped with SCA 6000 cards.

CHAPTER 2

Installing and Configuring ICSF

IBM Mainframe

Various steps are required to configure a z/OS system to be used as an external key store for a OKM Cluster.

Installing and Configuring the CEX2C Cryptographic Card

Refer to documentation that accompanies this card.

Installing Sun ELS or NCS PTF

The PTFs associated with Sun ELS 7.0 and NCS 6.2 are available at the Oracle Software Downloads site:

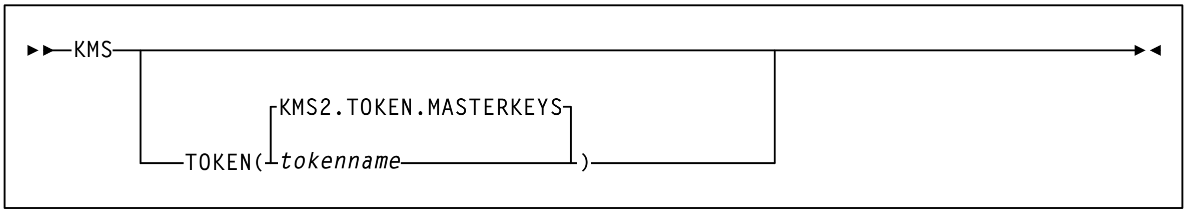
http://www.oracle.com/technetwork/indexes/downloads/index.html

Systems programmers can download the appropriate PTF and install it following standard procedures.

ELS 7.0 Setup

For ELS 7.0, the OKM-ICSF function is provided through an ELS PTF. The OKM-ICSF proxy is an HTTP server CGI routine. The SMC HTTP server must be active on a system with the ICSF PKCS11 function active.

The OKM command is valid from the SMCPARMS data set only.



KMS

The command name.

TOKEN

tokenname

Specifies the PKCS11 token name for the OKM-ICSF interface. The first character of the name must be alphabetic or a national character (#, $, or @). Each of the remaining characters can be alphanumeric, a national character, or a period (.). The maximum length is 32 characters.

KMS2.TOKEN.MASTERKEYS

Specifies the default PKCS11 token name.

NCS 6.2 Setup

For NCS 6.2, the OKM-ICSF function is provided through an SMC PTF. The OKM-ICSF proxy is an HTTP server CGI routine. The SMC loadlib must be included in the STEPLIB for the HTTP server (SSKY500). The PKCS11 token name is KMS2.TOKEN.MASTERKEYS and cannot be changed.

Add the following to the SSKY500 HTTP Server startup parameters:

```
LOADMODULE SMCGCSF
```

See the *Storage Management Component (SMC) 6.2 Configuration and Administration Guide* for additional information about the HTTP Server startup parameters.

Preparing ICSF

The following items activate the ICSF PKCS#11 function:

- Ensure that ICSF is at HCR7740 or higher.
- Define the Token Data Set (TKDS) in MVS. The TKDS is the repository for the keys used by PKCS#11. The TKDS is a key-sequenced VSAM data set.

  Keys within the Token Data Set are not encrypted. Therefore, it is important that the security administrator create a RACF profile to protect the Token Data Set from unauthorized access.
- The ICSF installation options data set contains two options related to the Token Data Set:
  - **TKDSN(*datasetname*)**

    Identifies the VSAM data set that contains the token data set. It must be specified for ICSF to provide PKCS#11 services.
  - **SYSPLEXTKDS(YES | NO,FAIL(YES | NO)**

    Specifies whether the token data set should have sysplex-wide data consistency.

See the *IBM z/OS Cryptographic Services ICSF System Programmer's Guide* (SA22-7520) for additional information on ICSF initialization.

ICSF uses profiles in the SAF CRYPTOZ class to control access to PKCS#11 tokens. The userid of the HTTP Server started task must have the following SAF access level for the defined PKCS#11 token:

- SO.token_name CONTROL
- USER.token_name UPDATE

Configuring AT-TLS

The document *Using AT-TLS with Sun Microsystems HSC Client/Server z/OS Solution, Implementation Example* (October 2008) shows examples for configuring AT-TLS on the IBM mainframe.

AT-TLS is an encryption solution for TCP/IP applications that is completely transparent to the application server and client. Packet encryption and decryption occurs in the z/OS TCPIP address space at the TCP protocol level.

To implement AT-TLS encryption for the OKM to NCS/ELS HTTP server connection, the minimum level needed for the Communication Server is z/OS 1.9. The following available IBM PTFs (for APAR PK69048) should be applied for best performance:

- Release 1A0 : UK39417 available 08/10/07 z/OS 1.10
- Release 190 : UK39419 available 08/10/07 z/OS 1.9

See the following IBM publications for detailed information about the IBM z/OS Communications Server Policy Agent configuration and RACF definitions for AT-TLS:

- *IP Configuration Guide*, SC31-8775
- *IP Configuration Reference*, SC31-8776
- *Security Server RACF Security Administrator's Guide*, SA22-7683
- *Security Server RACF Command Language Reference*, SA22-7687
- *IBM Redbook Communications Server for z/OS V1R7 TCP/IP Implementation*, Volume 4, *Policy-Based Network Security*, SG24-7172

TCPIP OBEY Parameter

Specify the following parameter in the TCPIP profile data set to activate the AT-TLS function:

```
TCPCONFIG TTLS
```

This statement may be placed in the TCP OBEY file.

Policy Agent (PAGENT) Configuration

The Policy Agent address space controls which TCP/IP traffic is encrypted. A sample PAGENT configuration follows.

PAGENT JCL

PAGENT started task JCL:

```
//PAGENT PROC
//*
//PAGENT EXEC PGM=PAGENT,REGION=0K,TIME=NOLIMIT,
// PARM='POSIX(ON) ALL31(ON) ENVAR("_CEE_ENVFILE=DD:STDENV")/-d1'
//*
//STDENV DD DSN=pagentdataset,DISP=SHR//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//*
//CEEDUMP DD SYSOUT=*,DCB=(RECFM=FB,LRECL=132,BLKSIZE=132)
```

The *pagentdataset* data set contains the PAGENT environment variables.

PAGENT Environment Variables

This is a sample PAGENT environment variable file:

```
LIBPATH=/lib:/usr/lib:/usr/lpp/ldapclient/lib:.
PAGENT_CONFIG_FILE=/etc/pagent.conf
PAGENT_LOG_FILE=/tmp/pagent.log
PAGENT_LOG_FILE_CONTROL=3000,2
_BPXK_SETIBMOPT_TRANSPORT=TCPIP
TZ=MST7MDT
```

/etc/pagent.conf contains the PAGENT configuration parameters.

PAGENT Configuration

This is a sample PAGENT configuration:

```
TTLSRule                            KMS-TO-ZOS
{
   LocalAddr                        localtcpipaddress
   RemoteAddr                       remotetcpipaddress
   LocalPortRange                   localportrange
   RemotePortRange                  remoteportrange
   Jobname                          HTTPserverJobname
   Direction                        Inbound
   Priority                         255
   TTLSGroupActionRef               gAct1~KMS_ICSF
   TTLSEnvironmentActionRef         eAct1~KMS_ICSF
   TTLSConnectionActionRef          cAct1~KMS_ICSF
}
TTLSGroupAction                     gAct1~KMS_ICSF
{
   TTLSEnabled                      On
   Trace                            2
}
TTLSEnvironmentAction               eAct1~KMS_ICSF
{
   HandshakeRole Server
   EnvironmentUserInstance          0
   TTLSKeyringParmsRef              keyR~ZOS
}
TTLSConnectionAction                cAct1~KMS_ICSF
{
   HandshakeRole                    ServerWithClientAuth
   TTLSCipherParmsRef               cipher1~AT-TLS__Gold
   TTLSConnectionAdvancedParmsRef   cAdv1~KMS_ICSF
   CtraceClearText                  Off
   Trace                            2
}
```

```
TTLSConnectionAdvancedParms         cAdv1~KMS_ICSF
{
   ApplicationControlled            Off
   HandshakeTimeout                 10
   ResetCipherTimer                 0
   CertificateLabel                 certificatelabel
   SecondaryMap                     Off
}
TTLSKeyringParms                    keyR~ZOS
{
   Keyring                          keyringname
}
TTLSCipherParms                     cipher1~AT-TLS__Gold
{
   V3CipherSuites                   TLS_RSA_WITH_3DES_EDE_CBC_SHA
   V3CipherSuites                   TLS_RSA_WITH_AES_128_CBC_SHA
}
```

where:

localtcpipaddress

local TCP/IP address (address of HTTP server)

remotetcpipaddress

remote TCP/IP address (address of OKM client) can be ALL for all TCP/IP addresses

localportrange

local port of HTTP server (specified in the HTTP or SMC startup)

remoteportrange

remote port range (1024-65535 for all ephemeral ports)

HTTPserverJobname

jobname of the HTTP Server

certificatelabel

label from certificate definition

keyringname

name from RACF keyring definition

RACF Definitions

Activate the following RACF classes. Either the RACF panels or the CLI may be used.

- DIGTCERT
- DIGTNMAP
- DIGTRING

SERVAUTH CLASS must be RACLISTed to prevent PORTMAP and RXSERV from abending TTLS is activated.

RACF Commands

The RACF commands to achieve the above:

- SETROPTS RACLIST(SERVAUTH)
- RDEFINE SERVAUTH ** UACC(ALTER) OWNER (RACFADM)
- RDEFINE STARTED PAGENT*.* OWNER(RACFADM) STDATA(USER(TCPIP) GROUP(STCGROUP)
- RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE) OWNER(RACFADM)
- RDEFINE FACLITY IRR.DIGTCERT.LIST UACC(NONE) OWNER(RACFADM)
- RDEFINE FACILITY IRR.DIGTCERT.GENCERT UACC(NONE) OWNER (RACFADM)

RACF Certificate Creation Commands

The *IBM Communications Server for z/OS V1R10 TCP/IP Implementation Volume 4: Security and Policy-Based Networking* document outlines the procedure required to create and export digital certificates on the z/OS system.

The RACDCERT utility creates and manages digital certificates within RACF. Verify that RACDCERT is in the AUTHCMD section of the IKJTSO*xx* member in SYS1.PARMLIB.

The RACF commands to create Keyrings and certificates for use by the AT-TLS function follow:

- RACDCERT ID(*stcuser*) ADDRING(*keyringname*)

where:

stcuser

RACF user id associated with the TCPIP address space

keyringname

Name of keyring, must match the Keyring specified in the PAGENT configuration

- RACDCERT ID(*stcuser*) GENCERT CERTAUTH SUBJECTSDN(CN('*serverdomainname*') O('*companyname*') OU('*unitname*') C('*country*')) WITHLABEL('*calabel*') TRUST SIZE(1024) KEYUSAGE(HANDSHAKE,DATAENCRYPT,CERTSIGN)

where:

stcuser

RACF user id associated with the TCPIP address space

serverdomainname

Domain name of the z/OS server (e.g., MVSA.COMPANY.COM)

companyname

Organization name

unitname

Organizational unit name

country

Country

calabel

Label for certificate authority (e.g., CAKMSSERVER)

Note – This is the CA certificate for the OKM system.

- RACDCERT ID(*stcuser*) GENCERT SUBJECTSDN(CN('*serverdomainname*') O('*companyname*') OU('*unitname*') C('*country*')) WITHLABEL('*serverlabel*') TRUST SIZE(1024) SIGNWITH(CERTAUTH LABEL('*calabel*'))

where:

stcuser

RACF user id associated with the TCPIP address space

serverdomainname

Domain name of the z/OS server (e.g., MVSA.COMPANY.COM)

companyname

Organization name

unitname

Organizational unit name

country

Country

serverlabel

Label for the server certificate (e.g., KMSSERVER)

calabel

Label for certificate authority, specified in the CA certificate definition

Note – This is the SERVER certificate.

- RACDCERT ID(*stcuser*) GENCERT SUBJECTSDN(CN('*clientdomainname*') O('*companyname*') OU('*unitname*') C('*country*')) WITHLABEL('*clientlabel*') TRUST SIZE(1024) SIGNWITH(CERTAUTH LABEL('*calabel*'))

where:

stcuser

RACF user id associated with the TCPIP address space

clientdomainname

Domain name of the OKM client (e.g., KMSA.COMPANY.COM)

companyname

Organization name

unitname

Organizational unit name

country

Country

clientlabel

Label for the server certificate – KMSCLIENT

calabel

Label for certificate authority, specified in the CA certificate definition.

Note – This is the CLIENT certificate.

The following commands connect the CA, SERVER and CLIENT certificates to the keyring specified in the PAGENT configuration:

- RACDCERT ID(*stcuser*) CONNECT(CERTAUTH LABEL('*calabel*') RING('*keyringname*') USAGE(CERTAUTH))

where:

stcuser

RACF user id associated with the TCPIP address space

calabel

Label for certificate authority, specified in the CA certificate definition

keyringname

Name of keyring, must match the Keyring specified in the PAGENT configuration

- RACDCERT ID(*stcuser*) CONNECT(ID(*stcuser*) LABEL('*serverlabel*') RING('*keyingname*') DEFAULT USEAGE(PERSONAL)

where:

stcuser

RACF user id associated with the TCPIP address space

serverlabel

Label for the server certificate

keyringname

Name of keyring, must match the Keyring specified in the PAGENT configuration

- RACDCERT ID(*stcuser*) CONNECT(ID(*stcuser*) LABEL('*clientlabel*') RING('*keyingname*') USEAGE(PERSONAL)

where:

stcuser

RACF user id associated with the TCPIP address space

clientlabel

Label for the client certificate

keyringname

Name of keyring, must match the Keyring specified in the PAGENT configuration

The following commands export the CA and client certificates for transmission to the OKM:

- RACDCERT EXPORT (LABEL('*calabel*')) CERTAUTH DSN('*datasetname*') FORMAT(CERTB64)

where:

calabel

Label for certificate authority, specified in the CA certificate definition

datasetname

Data set to receive the exported certificate

- RACDCERT EXPORT (LABEL('*clientlabel*')) ID(stcuser) DSN('*datasetname*') FORMAT(PKCS12DER) PASSWORD(' *password* ')

where:

clientlabel

Label for the client certificate

stcuser

RACF user id associated with the TCPIP address space

datasetname

Data set to receive the exported certificate

password

Password for data encryption. Needed when the certificate is received on the OKM. The password must 8 characters or more.

The export data sets are now transmitted to the OKM, and FTP can be used. The CA certificate is transmitted with an EBCDIC to ASCII conversion. The CLIENT certificate is transmitted as a BINARY file and contains both the client certificate and its private key.

RACF List Commands

The following RACF commands list the status of the various RACF objects:

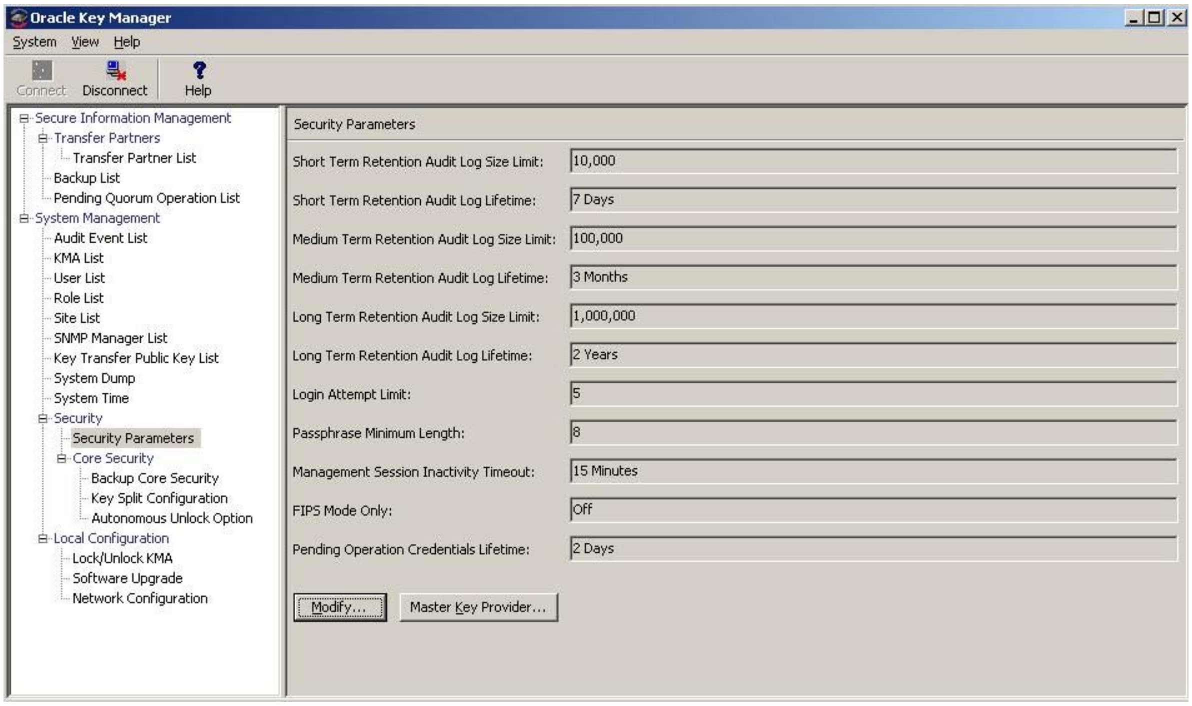
- RLIST STARTED PAGENT.* STDATA ALL
- RLIST DIGTRING * ALL
- RLIST FACILITY IRR.DIGTCERT.LISTRING ALL
- RLIST FACILITY IRR.DIGCERT.LST ALL
- RLIST FACILITY IRR.DIGCERT.GENCERT ALL
- RACDCERT ID(stcuser) LIST
- RACDCERT ID(stcuser) LISTRING(keyringname)
- RACDCERT CERTAUTH LIST

Updating OKM Cluster Information

After the IBM mainframe has been configured, the z/OS systems programmer must provide the following information to the administrator of the OKM Cluster:

- Host name or IP address of the mainframe
- Port number (such as 9889)
- Web application path (such as "/cgi/smcgcsf")
- File containing the client "user certificate" (exported and transferred off of the mainframe)
- File containing the client private key (exported and transferred off of the mainframe)
- Password that was used when the client private key was created
- File containing the Root CA certificate (exported and transferred off of the mainframe)

The administrator of the OKM Cluster enters this information as the Master Key Provider settings in the Security Parameters panel of the OKM GUI.



Security Parameters

Short Term Retention Audit Log Size Limit: 10,000

Short Term Retention Audit Log Lifetime: 7 Days

Medium Term Retention Audit Log Size Limit: 100,000

Medium Term Retention Audit Log Lifetime: 3 Months

Long Term Retention Audit Log Size Limit: 1,000,000

Long Term Retention Audit Log Lifetime: 2 Years

Login Attempt Limit: 5

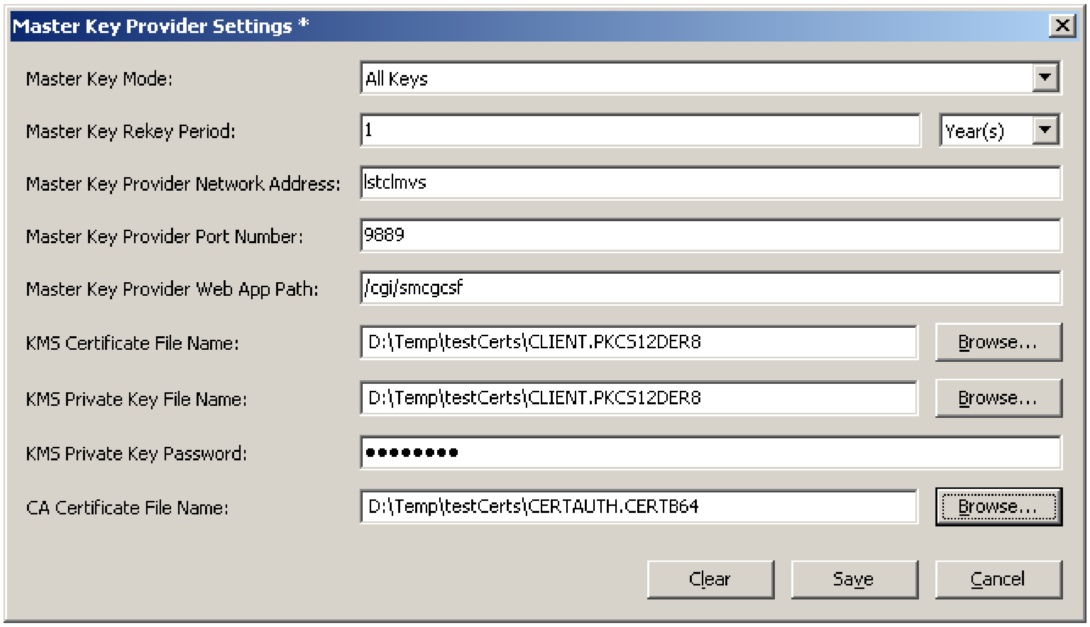
Passphrase Minimum Length: 8

Management Session Inactivity Timeout: 15 Minutes

FIPS Mode Only: Off

Pending Operation Credentials Lifetime: 2 Days

The client "user certificate" and the client private key might appear in the same file when they are exported from the IBM mainframe. If so, then the administrator should specify the same file in the OKM Certificate File Name and OKM Private Key File Name fields in the Master Key Provider settings.



Master Key Provider Settings *

Master Key Mode: All Keys

Master Key Rekey Period: 1 Year(s)

Master Key Provider Network Address: lstclmvs

Master Key Provider Port Number: 9889

Master Key Provider Web App Path: /cgi/smcgcsf

KMS Certificate File Name: D:\Temp\testCerts\CLIENT.PKCS12DER8

KMS Private Key File Name: D:\Temp\testCerts\CLIENT.PKCS12DER8

KMS Private Key Password: ●●●●●●●●●

CA Certificate File Name: D:\Temp\testCerts\CERTAUTH.CERTB64

The fields and their descriptions are given below:

Master Key Mode

Select “Off,” “All Keys,” or “Recover Keys Only.” A value of “Off” means that the KMAs in this OKM Cluster create their own keys and do not derive keys from a Master Key Provider. A value of “All Keys” means that the KMAs in this OKM Cluster contact the Master Key Provider defined in the settings on this screen in order to create and retrieve master keys, and then use these master keys to derive keys for Agents. A value of “Recover Keys Only” means that the KMAs in this OKM Cluster contact the Master Key Provider defined in the settings on this screen to retrieve (but not create) master keys and then use these master keys to derive keys for Agents. The “All Keys” and “Recover Keys Only” values can be set only if the Replication Version is at least 11.

Master Key Rekey Period

Type the amount of time that defines how often this KMA should contact the Master Key Provider to create and retrieve new master keys. The default is 1 day. The minimum value is 1 day; maximum value is 25,185 days (approximately 69 years).

Master Key Provider Network Address

Type the host name or IP address of the host where the Master Key Provider resides.

Master Key Provider Port Number

Type the port number on which the Master Key Provider listens for requests from the KMAs in this OKM Cluster.

Master Key Provider Web App Path

Type the web application path that forms part of the URL that is used to contact the Master Key Provider (for example, "/cgi/smcgcsf").

KMS Certificate File Name:

Specify the name of the file that contains the KMS certificate that was exported from the Master Key Provider host. The Master Key Provider uses this certificate to verify requests from KMAs in this OKM Cluster.

KMS Private Key File Name

Specify the name of the file that contains the OKM private key that was exported from the Master Key Provider host. The Master Key Provider uses this private key to verify requests from KMAs in this OKM Cluster.

KMS Private Key Password

Type the OKM private key password as it was generated on the Master Key Provider host. The Master Key Provider uses this private key password to verify requests from KMAs in this KMS Cluster.

CA Certificate File Name

Specify the name of the file that contains the CA (Certificate Authority) certificate that was exported from the Master Key Provider host. The KMA uses this CA certificate to verify responses back from the Master Key Provider.

After the administrator saves these settings, the KMS Cluster begins to issue requests to the Proxy on the IBM mainframe.