Security Policy

for

Oracle Advanced Security Option Cryptographic Module

Version 1.0

September 1999

Prepared by

Oracle Corporation

A. *Scope of Document*

This document describes the security policy for the Oracle Advanced Security Option (ASO) Cryptographic Module.

The Oracle Advanced Security Option is a service communication component for encrypting data crossing Oracle network connections and for detecting surreptitious alteration of the data.

The ASO employs the DES algorithm for encryption. Other encryption algorithms may be selected, such as RC4, for operation in non-FIPS approved modes.

B. *Security Level*

The cryptographic module is designed to meet the overall requirements applicable to Level 2 security of FIPS 140-1. Table 1 lists the security levels corresponding to each of the security requirement sections of FIPS 140-1

Table 1. Module Security Level Specification

Security Requirements Section	Level
Cryptographic Module	2
Module Interfaces	2
Roles and Services	*2
Finite State Machine	2
Physical Security	2
EFP/EFT	N/A
Software Security	2
Operating System Security	**2
Key Management	2
Cryptographic Algorithms	2
EMI/EMC	2
Self Test	2

* The Cryptographic Module does not perform operator authentication. Controlled access is provided through the C2 Operating System.

** The Cryptographic Module runs under an operating system that provides controlled access protection (TCSEC C2).

C. Roles and Services

Controlled access protection is provided by the TCSEC C2 or equivalent operating system as specified by NIST, of the Client or Server computer system in which the ASO Cryptographic Module software is installed.

The ASO *User Role* provides the services necessary for the secure transport of data over an insecure network. These services include the following:

Encryption Services

- Select Encryption Algorithm. This service selects the encryption algorithm to be employed for the duration of the network connection, which has been negotiated between the Client and Server.
- Diffie-Hellman Key Generation. This service generates the keys to be used for encryption. In DES algorithm, 16 keys are generated, and placed in queue, for use in re-synchronizing (re-selecting) the encryption key during a *Break* service.
- Encrypt Data. This service encrypts the input data stream from the Net 8 Session Layer and outputs the encrypted data stream to the Net 8 Session Layer.
- Decrypt Data. This service decrypts the encrypted data stream from the Net 8 Session Layer and outputs the decrypted data stream to the Net 8 Session Layer.
- Break - Encryption. This service resynchronizes the keys, upon receipt of a software interrupt.
- Terminate Encryption. This service destroys the connection-specific encryption data structures.

Checksum Services

- Select Data Integrity Algorithm. This service selects the negotiated data integrity algorithm to be employed for the duration of the network connection.
- Generate Checksum. This service generates the checksum on the data block.
- Verify Checksum. This service verifies the checksum for the received data block.
- Break - Checksum. This service handles software interrupts for the Data Integrity Algorithm.
- Terminate Checksum. This service destroys the connection-specific checksum data structures.

The Security Administrator (Crypto Officer) has access to all the services available to the User and in addition has access to the following service:

- Edit List. This service establishes the encryption and checksum algorithms available at Client and Server sites.

- DAC-Key Zeroization. DAC key is a fixed, hard coded value compiled into the cryptographic program image which the Crypto Officer could zeroize by deleting the program image file itself.

D. Security Rules

This section documents the security rules enforced by the cryptographic module.

a) When the server is configured with DES encryption *required* and have FIPS-140 set to *true*, the client configuration must be set to *accept* the server's configurations. Otherwise, the server configuration will prohibit the establishment of a connection.

b) The cryptographic module encrypts message traffic using the DES algorithm operated in Cipher Block Chaining Mode (CBC) as described in FIPS PUB 81.

c) The DES algorithm is tested by the use of a known answer for both encryption and decryption cryptographic functions.

d) The module generates a checksum on the data block using the selected checksum algorithm. The checksum is encrypted before transmission for purposes of ensuring data integrity.

e) The Diffie-Hellman generated key is based upon the time of day, crypto-seed parameter and operating system dependent data. The number of bits in the modulus, the modulus, and the number of bits in the exponent are sent from the Server to the Client. The public key for the Client is sent to the Server, and the public key for the Server is sent to the Client. A number of keys are generated and placed in queue, where the number of keys generated is a function of the encryption algorithm selected (16 keys are generated when the DES algorithm is selected).

f) The encryption key used for transmission is changed during a *Break* service.

g) Termination of a connection causes the encryption keys and checksum data structures to be destroyed.

h) The module performs a continuous random number generator test as specified in section 4.11.2 of FIPS 140-1.

i) For the Diffie-Hellman key exchange process, calculation of exponential is considered as a critical function and its modulo exponentiation function is tested as required in the Section 4.11.1 of FIPS 140-1 standard.

j) When the module is in the FIPS mode of operation, the Pseudorandom Number Generator for DSA private key as specified in Appendix 3.1 of FIPS 186-1, Digital Signature Standard (DSS) is used.

k) The module calculates the Data Authentication Code of the local encryption library, libncrypt8.a, and compares it with the DAC of the pre-built library file in order to detect if the library routines have been modified, in accordance with Sections 4.7 and 4.11.1 of FIPS 140-1 standard.

l) DAC-Key may be zeroized by the Crypto Officer through deleting the program image file itself.

m) The software security certification (UK ITSEC Certification Report No. P101) requires the use of a specified hardware and software configuration to meet the Operating System security requirements of FIPS 140-1. Under the Certification Report No. P101, the system certified is Sun Solaris Version 2.6 running on Sun Ultra SPARC-1 Workstation.

n) In non-FIPS mode of operation, cryptographic algorithms used are RC4-40bit, RC4-56bit, or RC4-128bit and MD-5.

o) Self-tests, as defined in c, d, h, i, and k, are performed each time a connection request is made from the client to the server. Upon failure of any self-tests, the connection is terminated and must be reinitialized for further connection attempts.

p) All error states and its associated indicators for Algorithm Negotiation, Diffie-Hellman Key Negotiation, Adaptor Initialization, Operational, Encryption and Decryption functions are specified in the vendor document Oracle Advanced Security, FIPS 140-1 Finite State Model.

E. Definitions of Security Relevant Data Items

There are 4 types of security relevant data items (SRDI's). These are:

a) Encryption Key (EK): This is a DES key used to encrypt data.

b) Message Initialization Vector (IV): This is a 64 bit fixed value used to initialize the DES encryption algorithm.

c) Checksum (CKSM): This is a checksum, calculated on the data block to be transmitted, which is used to verify that received data has not been modified.

d) DAC Key : This is a Data Authentication Code used to verify the integrity of the module's software and firmware from any modification. The fixed, hard coded DAC value is compiled into the cryptographic program image and its access is limited by the security provisions of the operating system.

F. Definitions of SRDI Modes of Access

Table 2 defines the relationship between access to SRDI's and the different module services. The modes of access shown in the table are defined as follows:

a) Generate EK: This operation generates an Encryption Key for the data to be encrypted.

b) Generate IV: Current version of ASO contains a constant value IV which is used to initialize the encryption algorithm.

c) Generate CKSM: This operation generates a checksum on the data block to be transmitted.

d) Encrypt: This operation encrypts data input to the ASO module.

e) Decrypt: This operation decrypts the input ciphertext data to the ASO module

f) Verify CKSM: This operation verifies the integrity of the received data.

g) Terminate: This operation releases the memory associated with encryption and checksum processing.

h) DAC-Key Zeroization: Deletes the hard coded DAC key which is used for comparison against calculated DAC value for verification against data modification.

i) Edit List: This service establishes the encryption and checksum algorithms available at Client and Server sites.

G. Service to SRDI Access Operation Relationship

Table 2. Service to SRDI Access Operation Relationships

Service	SRDI	Modes of Access	User Role	CO Role
Select Data Integrity Algorithm		Select	X	X
Diffie-Hellman Key Generation	EK's	Generate, Store	X	X
Data Integrity Algorithm Self Test		Run	X	X
Select Encryption Algorithm		Select	X	X
Encryption Algorithm Self Test		Run	X	X
Encrypt	Data –plain text Data – cipher text	Input Output	X	X
Add Checksum	CKSM	Add to output data transmitted	X	X
Decrypt	Data – cipher text Data – clear text	Input Output	X	X
Verify Checksum	CKSM	Verify	X	X
Break	EK	Select	X	X
Terminate Checksum	Checksum data	Destroy	X	X
Terminate Encryption	EK data	Destroy	X	X
DAC Key Zeroization	Key	Delete		X
Edit List	Encryption & Checksum algorithms	Modify		X