Oracle Cryptographic Libraries for SSL 10*g* (9.0.4)

by Oracle Corporation

Version 9.0.4

FIPS 140-2 Non-Proprietary Security Policy

Version 1.5

Level 2 Validation

February 15 2003

Table of Contents

Introduction

Purpose

This is a non-proprietary Cryptographic Module Security Policy for the Oracle Cryptographic Libraries for SSL 10*g* by Oracle Corporation (Version 9.0.4). This security policy describes how the Oracle Cryptographic Libraries for SSL 10*g* by Oracle Corporation (Version 9.0.4) meets the security requirements of FIPS 140-2 and how to run the module in a FIPS 140-2 approved mode of operation. This policy was prepared as part of the Level 2 FIPS 140-2 validation of the module.

FIPS 140-2 (Federal Information Processing Standards Publication 140-2 — *Security Requirements for Cryptographic Modules*) details the U.S. Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the NIST website at http://csrc.nist.gov/cryptval/.

The Oracle Cryptographic Libraries for SSL 10*g* by Oracle Corporation (Version 9.0.4) is referred to in this document as the Cryptographic Library for SSL or the module.

References

This document deals only with operations and capabilities of the module in the technical terms of a FIPS 140-2 cryptographic module security policy. More information is available on the module from the following sources:

- The Oracle website (www.oracle.com) contains information on the full line of products from Oracle.
- The NIST Validated Modules website (http://csrc.ncsl.nist.gov/cryptval/) contains contact information for answers to technical or sales-related questions for the module.

Document Organization

The Security Policy document is one document in a FIPS 140-2 Submission Package. In addition to this document, the Submission Package contains:

- Vendor Evidence document
- Finite State Machine
- Other supporting documentation as additional references

This Security Policy and the other validation submission documentation was produced by Corsec Security, Inc. under contract to Oracle. With the exception of this Non-Proprietary Security Policy, the FIPS 140-2 Validation Documentation is proprietary to Oracle and is releasable only under appropriate non-disclosure agreements. For access to these documents, please contact Oracle.

ORACLE CRYPTOGRAPHIC LIBRARIES FOR SSL 10*G* BY ORACLE CORPORATION (VERSION 9.0.4)

Overview

The Cryptographic Library for SSL is a generic module used by Oracle in a variety of its application suites. The module is used to provide support for cryptography, authentication, SSL cryptographic primitives (the module does not implement the full SSL protocol), and PKCS and certificate management for applications like the Oracle database (Server & Client), Oracle Applications Server, Oracle Internet Directory, Web Cache and Apache. It provides a rich set of functionality and uses a PKCS based wallet structures for managing identities and trust points.

The module's cryptographic boundary consists of the following components:

- NZ library: This is the interface available to the oracle products for SSL protocol implementation and the cryptographic support required for it. It is a wrapper on top of Certicom and RSA BSAFE, BCERT toolkits.

- RSA BSafe library: The RSA BSafe library is used by the Cryptographic Library for SSL, and provides algorithm implementations and RNG implementation.

- RSA BCert v1.0.1: RSA BCert is used for certificate processing and management

- Certicom SSL plus v3.09 library: This is used for the PKCS#12 structure. For the cryptographic functionality, the BSafe library is used.

Module Interfaces

The Oracle Cryptographic Libraries for SSL 10*g* by Oracle Corporation (Version 9.0.4) is classified as a Multi-chip standalone module for FIPS 140-2 purposes. The module is evaluated on a Solaris operating system version 8 with Admin Suite 3.0.1 (CC EAL 4 certified against the Protection Profile – 'Controlled Access Protection Profile, Issue 1.d, 8 October 1999'). The physical boundary of the Oracle Cryptographic Library for SSL is defined by the metal enclosure over the board. The module supports the physical interfaces of a standard PC. The physical interfaces include the computer keyboard port, CD-ROM drive, floppy disk, mouse, serial ports, parallel ports, networks ports, monitor port and power plug. The functional module interface exists in the software. The module provides scripts and graphical user interfaces to interact with the components.

All of these physical interfaces are separated into logical interfaces defined by FIPS 140-2, as described in the following table:

Module Physical Interface	FIPS 140-2 Logical Interface	Module Mapping
PC Network Port, Keyboard Interface, Mouse port	Data Input Interface	The NZ library is the data input interface. It provides APIs to interact with the module. Any program wishing to utilize the functionality of the module should compile and link the module into its code.
PC Network Port	Data Output Interface	The API to the NZ library is the interface for data output. The data output can be in the form of function input/output variables or return values.
PC Network Port, Serial Port, Keyboard port, Mouse port, PC Power Button	Control Input Interface	The Operating system's file system is used to input control data to the module. It is used to establish keys and configure user passwords. Also the 'sqlnet.ora' is the control data file. This is used to put the module in a FIPS mode of operation.
LEDs, PC Monitor, PC Network Port	Status Output Interface	The appropriate function calls and return values for function calls and log files are the interfaces for status output.
PC Power Interface	Power Interface	Not Applicable.

Table 1 – FIPS 140-2 Logical Interfaces

Roles and Services

The module supports role-based authentication. There are two main roles in the module (as required by FIPS 140-2) that operators may assume: a Crypto-Officer role and User role. They are the operators on the Solaris Operating System on which the module is evaluated. The root user on the Solaris platform and the user that installs the module are the Crypto-Officers; other users on the system are classified as Users.

The Crypto-Officer is responsible for initializing the module and configuring it to run in a FIPS approved mode. After the switch to the FIPS approved mode of operation both the Crypto-Officer and User will be able to utilize the functionality of the module. The various services offered by the module are described below.

Service	Description & CSPs accessed	Role
Uninstalling/Removing the module	The Crypto-Officer has the ability to delete the module.	Crypto-Officer
Configuring in FIPS Mode	The Crypto-Officer has to edit the 'sqlnet.ora' file to enable FIPS mode.	Crypto-Officer/User
Deleting private/public key pairs	Every User can delete his/her own public/private key pair. The 'root' user can delete the key pairs of all the users.	Crypto-Officer/User
Cryptographic functionality offered by the library	These are used by the applications that the library is linked against. The applications will use the functionality of the library to process SSL protocol packets.	Crypto-Officer/User

Table 2 –Services, Descriptions, Inputs and Outputs

Authentication Mechanisms

The module supports a username password based authentication mechanism.

Authentication Mechanism	Strength of the implementation
Username Password based	The implementation is the one that is used by the CC EAL 4 certified Solaris Operating system of the module. The minimum length of the password is 8 and the module requires the password to be alpha numeric. Assuming only 36 characters (A-Z, a-z, and 0-9) with repetition, the chance of a random attempt falsely succeeding is 1 in 2821109907456.

Table 3 – Estimated Strength of Authentication Mechanisms

Physical Security

The Cryptographic Library for SSL is a multi-chip standalone module and has met all physical security requirements for FIPS 140-2 Level 2. The platform provides production grade equipment, industry-standard and a strong enclosure, and the system meets Federal Communication Commission (FCC) Electromagnetic Interference (EMI) compatibility requirements. Tamper-Evidence labels should be used across the module's cover so that the Crypto-Officer can detect any attempt to open the box. It also recommends logging all access to the box by logging each time the tamper evidence seals are broken with permission of the Crypto-Officer.

Cryptographic Key Management

The module supports the following Approved cryptographic algorithms:

Symmetric Key Algorithms

Algorithm	Modes Implemented	Key Sizes	Certificate Number
DES (FIPS 46-3 – for legacy use only)	CBC	56Bits	215
Triple DES (FIPS 46-3)	CBC	168 Bits	170

Hashing Algorithms

Algorithm	Certificate Number
SHA-1 (FIPS 180-2)	154
HMAC SHA-1	Vendor Affirmed; 154

Public Key Algorithms

Algorithm	Key Sizes	Certificate Number
RSA (PKCS#1 v1 and v1.5 key wrapping only)	512, 768,1024, 2048, 4096 bits	Vendor Affirmed

Non-FIPS Approved Algorithms

- Symmetric Key Algorithms

Algorithm	Modes Implemented	Key Sizes
RC4	CBC	40, 128Bits

The module uses the PKCS#5 standard based encryption method to encrypt the wallets.

- Hashing Algorithms

Algorithm
MD5
HMAC MD5

- Key Exchange Algorithm

Algorithm
Diffie-Hellman

The module supports the following critical security parameters:

Cryptographic Key	Description	Key Type	Storage and Zeroization
Private/public keys	The private keys are part of the wallets used by the module. The private keys are generated in the module. The Crypto-Officer/User has to generate the private/public keys using the Oracle Wallet manager.	They are RSA public/private keys used by the module. They can be 512, 768, 1024, 2048 or 4096 bit keys.	The private and public keys are stored in Wallets which are PKCS#12 format based structures encrypted with PKCS#5 password-based encryption (considered plaintext for FIPS purpose). The Private/Public keys are zeroized by overwriting the wallet with a new wallet.
Public/private key values used in Diffie-Hellman key exchange	These are public/private key values used to agree upon a key using Diffie-Hellman key exchange mechanism.	They are derived using the FIPS 186-2 RNG in the module.	They are not stored on the disk. They are stored in memory and zeroized when the module shuts down.
Symmetric keys	These are keys used in the symmetric key algorithms used in the bsafe library by an external application.	They can be either DES or 3DES keys as the module supports only DES and 3DES algorithms.	They are not stored in disk. They are stored in memory and are zeroized when the module shuts down.

Table 4 – Keys and CSPs

Key Generation

The module implements a FIPS-approved PRNG based on the FIPS 186-2 specification. It uses this to generate the private/public key pairs for the wallet and also to any intermediate keys required in exchanging SSL session keys requested by the Certicom library from outside the cryptographic boundary.

Key Establishment

Private/Public Keys are generated in the module. Private keys are generated in the module and stored in the wallet by the Crypto-Officer/User. Public key certificates are also established by the Crypto-Officer/User by getting them certified by a Certificate Authority (CA). Symmetric keys used are established using the Diffie-Hellman protocol or RSA. The ephemeral keys for Diffie-Hellman key negotiations are generated in the module.

Key Storage

All keys are stored in the filesystem in plaintext form as described in the table above. The module runs on a CC EAL 4 evaluated OS and so the OS also protects the CSPs in the filesystem.

Key Zeroization

All ephemeral keys are zeroized explicitly by writing zeroes into them before deleting. Private keys are zeroized by overwriting them with new keys. Also, as the Operating System is CC EAL 4 evaluated, it enforces zeroization of all CSPs.

Self-Tests

The module performs power-up and conditional self-tests to ensure the secure and proper operation, and the sections below provide details on the module's self-tests.

Power-Up Self-Tests

The power-up self tests implemented include known answer tests for DES, 3DES, SHA-1 RSA. Also executed at power-up is a PRNG KAT and a software/firmware integrity check with HMAC SHA-1. Power-up self tests are executed automatically when the module is started. Because it is a software library, the power-up self-tests are run when the module is linked by the application and used.

Conditional Self-Tests

The module performs two conditional self-tests: a pair-wise consistency test each time a module generates RSA public/private key and a continuous random number generator test each time the module produces random data.

The module logs the result of self-tests. If any of the self tests fail, the module logs them in a file and enters the error state.

Design Assurance

Oracle follows highly stabilized and popular design procedures for both Software implementations. The design for the module follows FIPS provided guidelines. The software design both go through many phases of review and inspections. The code and design documents are securely stored and the delivery is also secure. Detailed proprietary design, configuration management, and secure delivery documentation has been provided by Oracle as required for FIPS 140-2 validation.

Mitigation of Other Attacks

In a FIPS mode of operation the module does not claim to mitigate any attacks.

SECURE OPERATION

The Oracle Cryptographic Library for SSL meets Level 2 requirements for FIPS 140-2. The sections below describe how to place and keep the module in FIPS-approved mode of operation.

Initial Setup

The module is to be installed on the Sun Solaris operating system version 8 with Admin Suite 3.0.1. The operating system has to be installed and configured to meet the CC EAL 4 evaluated installation and configuration procedures.

Crypto-Officer Guidance

The Crypto-Officer is required to configure the module to run in a FIPS mode of operation. For this the Crypto-Officer has to edit the 'sqlnet.ora' file and set the variable 'SQLNET.SSLFIPS_140' to true. Also the Crypto-Officer has to make sure the trace level in the file is set to at least 1. The crypto-officer also has to verify that the files in the module have the permissions set to at least 755 (-rwxr-xr-x).

User guidance

If the module is being run in the client setting, the User has to run the module in a FIPS approved mode of operation. For this the variable 'SQL.SSLFIPS_140' in the 'sqlnet.ora' file has to be set to true. This will make the module run in a FIPS approved mode of operation. Also the User has to make sure the trace level in the file is set to at least 1. The user should select a strong password and should safeguard the password properly.

ACRONYMS

ANSI	American National Standards Institute
CMVP	Cryptographic Module Validation Program
CSE	Communications Security Establishment
CSP	Critical Security Parameter
DES	Data Encryption Standard
EDC	Error Detection Code
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
FCC	Federal Communication Commission
FIPS	Federal Information Processing Standard
KAT	Known Answer Test
LED	Light Emitting Diode
MAC	Message Authentication Code
NIST	National Institute of Standards and Technology
NVLAP	National Voluntary Laboratory Accreditation Program
PKCS	Public Key Cryptographic System
RAM	Random Access Memory
RSA	Rivest Shamir and Adleman
SHA	Secure Hash Algorithm