

*Odyssey Security Component User Mode
Version 2.0 and Odyssey Security
Component Kernel Mode Version 2.0
Security Policy
Document Version 1.0*

Juniper Networks, Inc.

September 23, 2009

TABLE OF CONTENTS

1. MODULE OVERVIEW.....3

2. SECURITY LEVEL.....4

3. MODES OF OPERATION.....5

4. PORTS AND INTERFACES.....5

5. IDENTIFICATION AND AUTHENTICATION POLICY.....6

6. ACCESS CONTROL POLICY.....7

 ROLES AND SERVICES.....7

 DEFINITION OF CRITICAL SECURITY PARAMETERS (CSPs).....8

 DEFINITION OF CSPs MODES OF ACCESS.....8

7. CRYPTOGRAPHIC KEY MANAGEMENT.....10

 KEY GENERATION.....10

 KEY STORAGE.....10

 ZEROIZATION.....10

8. OPERATIONAL ENVIRONMENT.....10

9. SECURITY RULES.....11

10. PHYSICAL SECURITY.....12

11. MITIGATION OF OTHER ATTACKS POLICY.....12

12. DEFINITIONS AND ACRONYMS.....12

1. Module Overview

The Odyssey Security Component User Mode (OSCU) (SW Version 2.0) and Odyssey Security Component Kernel Mode (OSCKM) (SW Version 2.0) are two software modules that both implement a set of cryptographic algorithms for use by a software application. This Security Policy document details the OSCU Version 2.0 and OSCKM Version 2.0.

The Odyssey Security Component User Mode is comprised of a dynamic link library that resides in user space, odFIPS2.dll. The Odyssey Security Component Kernel Mode is comprised of a kernel level module, odFIPS2.sys. They are two separate configurations that provide the same functionality and services, but run independently of one another. The modules have a multi-chip standalone embodiment as defined by FIPS 140-2.

The modules only implement an Approved mode of operation.

The cryptographic modules run on the Microsoft Windows XP and Microsoft Windows 2000 operating systems. The modules were operational tested on the following operating systems:

- Microsoft Windows XP SP2 (in single-user mode)
- Microsoft Windows 2000 SP3 (in single-user mode)

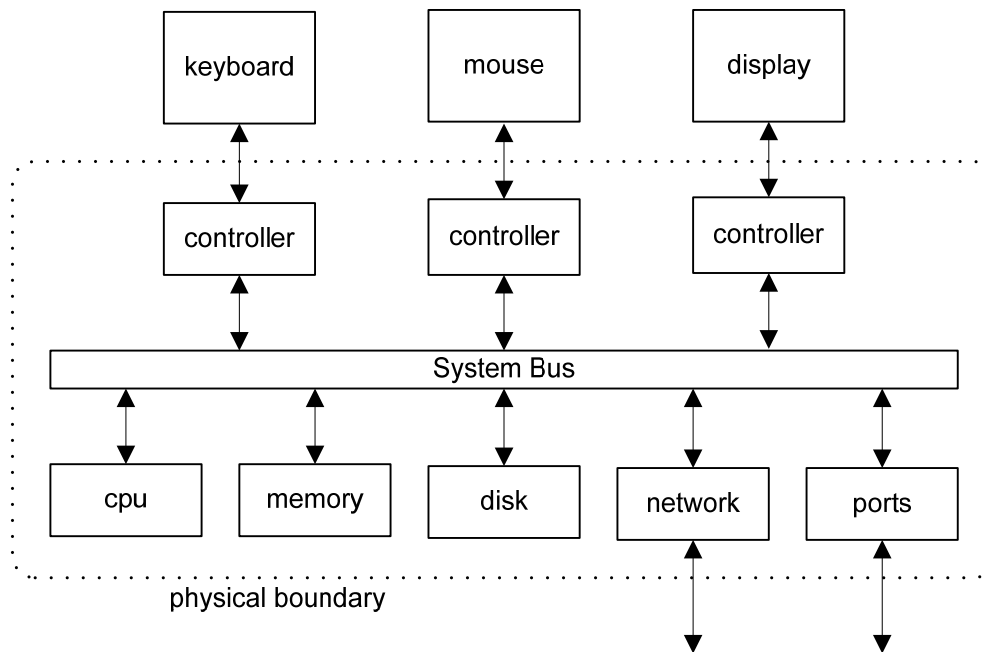


Figure 1: Hardware Diagram Showing PC Containing Cryptographic Module

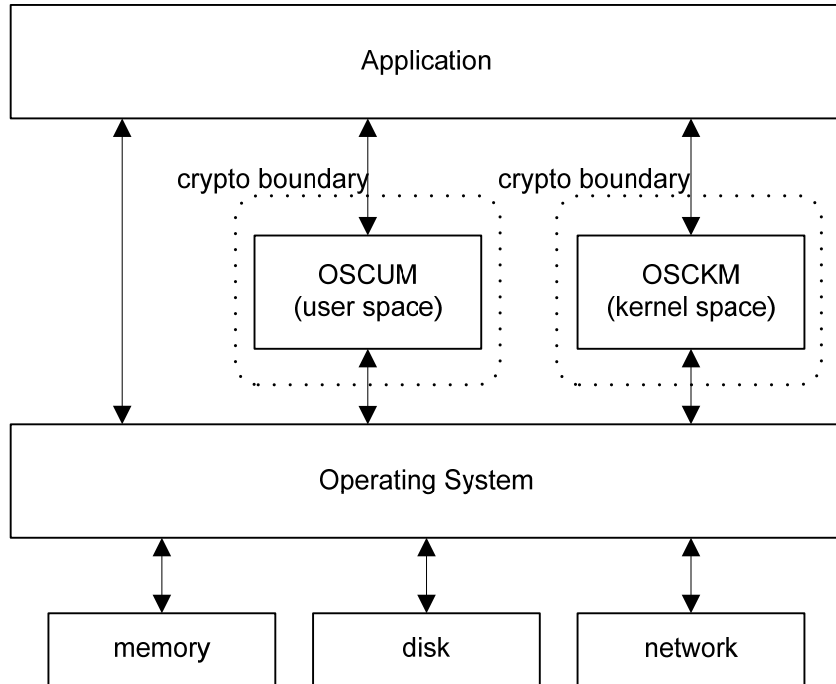


Figure 2: Software Diagram Showing Cryptographic Boundary

2. Security Level

The OSCUM and OSCKM meet the overall requirements applicable to Level 1 security of FIPS 140-2.

Table 1 - Module Security Level Specification

| Security Requirements Section | Level |
|------------------------------------|-------|
| Cryptographic Module Specification | 1 |
| Module Ports and Interfaces | 1 |
| Roles, Services and Authentication | 1 |
| Finite State Model | 1 |
| Physical Security | N/A |
| Operational Environment | 1 |
| Cryptographic Key Management | 1 |
| EMI/EMC | 1 |
| Self-Tests | 1 |
| Design Assurance | 2 |
| Mitigation of Other Attacks | N/A |

3. Modes of Operation

Approved mode of operation

In FIPS mode, the OSCUM and OSCKM support the following FIPS Approved algorithms:

- AES 128, 192, 256 – ECB, CBC, and Counter modes (See certificate #783)
- AES-CCM – Key sizes 128, 192, and 256 (See certificate #784)
- Triple-DES – TEBC and TCBC modes (See certificate #679)
- SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 (See certificate #787)
- HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 (See certificate #430)
- DSA Sign/Verify, Key Gen, and PQG Gen/Verify (See certificate #293)
- RSA Sign/Verify (See certificate #373)
- FIPS 186-2 RNG (See certificate #451)

Each module only supports an Approved mode of operation. Once loaded into memory and executed, the module is running in FIPS mode. An operator of the module can verify that the module is running in the FIPS Approved mode of operation by executing the “Get State” service, which shall return the following: `OD_FIPS_STATE_ENABLED`.

The cryptographic modules provide the following allowed cryptographic algorithms:

- RSA Encrypt/Decrypt (for Key Transport only) (key wrapping; key establishment methodology provides between 80 and 128 bits of encryption strength)

The following non-Approved algorithm is also available in the Approved mode of operation:

- RSA Encrypt/Decrypt (for bulk data) - No security is claimed for data that has been encrypted using this RSA.

4. Ports and Interfaces

All FIPS ports and interfaces are defined as the API of each cryptographic module. The API contains all data input, data output, control input, and status output interfaces to and from the module.

5. Identification and Authentication Policy

Assumption of roles

The OSCUM and OSCKM cryptographic modules support a Cryptographic Officer role and a User role. The system administrator implicitly assumes the Crypto Officer role, and is responsible for installing the module.

Table 2 - Roles and Required Identification and Authentication

| Role | Type of Authentication | Authentication Data |
|-----------------------|-------------------------------|----------------------------|
| User | N/A | N/A |
| Cryptographic Officer | N/A | N/A |

6. Access Control Policy

Roles and Services

Table 3 lists each role and the services authorized for each role.

Table 3 – Services Authorized for Roles

| Role | Authorized Services |
|---------------------------------|--|
| User and Cryptographic Officer: | <ul style="list-style-type: none"> • <u>AES Encrypt/Decrypt</u> • <u>TDES Encrypt/Decrypt</u> • <u>RSA Sign/Verify</u> • <u>DSA Sign/Verify</u> • <u>Generate Random Number</u> • <u>AES CCM</u> • <u>HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512</u> • <u>RSA Encrypt/Decrypt</u> (for key transport only) - Note: This service is also used for encrypting/decrypting bulk data. However, no security is claimed for data that has been protected by RSA. • <u>RSA Key Generation</u> • <u>DSA Key Generation</u> • <u>AES Key Encryption</u> • <u>Generate Prime Number</u> – Generates a prime number using the FIPS 186-2 RNG • <u>Modular Exponentiation</u> • <u>EnableFIPSMModule</u> – Enables all authorized services • <u>DisableFIPSMModule</u> – Disables all authorized services and returns the module to a pre-operational state • <u>GetState</u> – Returns the current state of the cryptographic module • <u>GetError</u> – Returns a specific error code when the module is in an error state • <u>Run Self-tests</u> – This service executes the suite of power up self-tests required by FIPS 140-2 by calling the API command. |

Note: In addition to the authenticated "Run Self-Tests" service, self-tests can also be initiated by any operator without authentication by reloading the module into memory.

Definition of Critical Security Parameters (CSPs)

The Critical Security Parameters (CSPs) defined for the OSCUM and OSCKM consist of cryptographic keys and random numbers used as seeding material. The modules do not persistently store CSPs within the logical boundary.

The following secret keys, private keys, and CSPs are supported by the modules:

- AES Keys: 128, 192 and 256 bit keys used to AES encrypt/decrypt data.
- TDES Keys: 3 separate 128 bit DES keys used to TDES encrypt/decrypt data.
- AES CCM Key: 128, 192, or 256 bit AES Key used for AES CCM operations.
- HMAC Keys: For use during HMAC operations.
- DSA Signing Private Key: Used to digitally sign data.
- RSA Private Key: Used to digitally sign data.
- AES Key Encryption Key: 128 bit AES key for use in AES key wrapping operations.
- FIPS 186-2 PRNG Seed and Seed Key: Used for the generation of CSPs and Keys. These values are entered into the module (not internally generated) and the strength of the keys generated depends on the strength of these parameters.
- HMAC Integrity Key: HMAC-SHA-512 key used during the Software Integrity Test. (Note: This key is only used for power up self-tests and is not considered a CSP per CMVP IG 7.4.)

Definition of Public Keys

The following are the public keys contained in the two modules:

- RSA Verifying Public Key: This is the public part of the cryptographic module's RSA Public/Private key pair used to verify RSA signatures.
- DSA Public Key: This is the public part of the cryptographic module's DSA Public/Private key pair used to verify DSA signatures.
- RSA Wrapping Key: Used to perform RSA key transport of keys.

Definition of CSPs Modes of Access

Table 5 defines the relationship between access to CSPs and the different module services. The modes of access shown in the table are defined as follows:

- Read
- Write
- Execute

Each service's API indicates the type of access to CSPs defined by that API. When a CSP is used by the API call to perform particular services, read and execute access is indicated. When a CSP is generated, modified or deleted by the API call, write access is indicated.

Table 4 – Key and CSP Access Rights within Services

| Approved Services | Keys/CSPs | Authorized Roles | Type of Access |
|--|-----------------------------------|-------------------------|-----------------------|
| Symmetric Encryption/Decryption Services | | | |
| AES Encrypt/Decrypt | AES Key | User/CO | read, execute |
| TDES Encrypt/Decrypt | TDES Key | User/CO | read, execute |
| Asymmetric Encryption/Decryption for Key Wrapping Services | | | |
| RSA Encrypt | RSA Wrapping Public Key | User/CO | read, execute |
| RSA Decrypt | RSA Private Key | User/CO | read, execute |
| Message Authentication Service | | | |
| AES-CCM | AES-CCM Key | User/CO | read, execute |
| HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | HMAC Key | User/CO | read, execute |
| Digital Signature Generation/Verification Services | | | |
| RSA Verify | RSA Verifying Public Key | User/CO | read, execute |
| RSA Sign | RSA Private Key | User/CO | read, execute |
| DSA Verify | DSA Public Key | User/CO | read, execute |
| DSA Sign | DSA Signing Private Key | User/CO | read, execute |
| Symmetric Key Wrapping | | | |
| AES Key Encryption | AES Key Encryption Key | User/CO | read, execute |
| Symmetric Key Generation Service | | | |
| Generate Random Number | FIPS 186-2 PRNG Seed and Seed Key | User/CO | read, execute |
| Asymmetric Key Generation Services | | | |
| RSA Key Generation | RSA Public/Private Key Pair | User/CO | write, execute |
| DSA Key Generation | DSA Public/Private Key Pair | User/CO | write, execute |
| Other Services | | | |
| Generate Prime Number | FIPS 186-2 PRNG Seed and Seed Key | User/CO | read, execute |
| Modular Exponentiation | N/A | User/CO | N/A |

| Approved Services | Keys/CSPs | Authorized Roles | Type of Access |
|--------------------------|------------------|-------------------------|-----------------------|
| EnableFIPSMModule | N/A | User/CO | N/A |
| DisableFIPSMModule | N/A | User/CO | N/A |
| Run Self-Tests | N/A | User/CO | N/A |
| GetState | N/A | User/CO | N/A |
| GetError | N/A | User/CO | N/A |

7. Cryptographic Key Management

Key Generation

The cryptographic modules support generation of DSA and RSA public and private keys, using the Approved FIPS 186-2 deterministic random number generator.

Key Storage

The modules do not persistently store keys. Key material is provided for use through a defined API, stored in RAM, and then destroyed once processing is terminated. If the operator wishes to store keys they are responsible for doing so outside of the cryptographic module's logical boundary.

Zeroization

All key data exists in data structures allocated within the cryptographic module, and can only be returned to an authorized user using the defined API. The operating system protects system memory and process space from access by unauthorized users. The operator of the cryptographic module should follow the steps outlined in the module's API specification to ensure sensitive data is protected by zeroizing the data from memory when it is no longer needed.

8. Operational Environment

The FIPS 140-2 Area 6 Operational Environment requirements are applicable because each module operates in a modifiable operational environment. The modules were operational tested on the following platforms:

- Windows 2000 (SP3) (running in single-user mode)
- Windows XP (SP2) (running in single-user mode)

9. Security Rules

Each cryptographic module's design corresponds to the module's security rules. This section documents the security rules enforced by the cryptographic modules to implement the security requirements of FIPS 140-2 Level 1.

1. The cryptographic modules shall provide two roles. These are the User role, and the Cryptographic Officer role.
2. The cryptographic modules shall perform the following tests:
 - A. Power up Self-Tests:
 1. Cryptographic algorithm tests:
 - a. AES KAT
 - b. AES CCM KAT
 - c. TDES KAT
 - d. RSA Sign/Verify KAT
 - e. RSA Encrypt/Decrypt KAT (for key transport only)
 - f. DSA Sign/Verify KAT
 - g. HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 KATs
 - h. SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 KATs
 - i. FIPS 186-2 DRNG KAT
 2. Software Integrity Test (HMAC SHA-512)
 - B. Conditional Self-Tests:
 1. Continuous Random Number Generator (RNG) test – performed on DRNG
 2. DSA pairwise consistency test
 3. RSA pairwise consistency test
3. The operator shall be capable of commanding the modules to perform the power-up self-test by reloading the modules into memory or by calling the `odFIPS_RunSelfTestAsynch` function.
4. Prior to each use, the internal RNG shall be tested using the conditional test specified in FIPS 140-2 §4.9.2.
5. Data output shall be inhibited during key generation, self-tests, zeroization, and error states.
6. Status information shall not contain CSPs or sensitive data that if misused could lead to a compromise of the module.
7. The modules do not support concurrent operators.

10. Physical Security

The FIPS 140-2 Area 5 Physical Security requirements are not applicable because the devices are software only modules.

11. Mitigation of Other Attacks Policy

The modules have not been designed to mitigate any specific attacks outside the scope of FIPS 140-2 requirements.

12. Definitions and Acronyms

| | |
|-------|---|
| AES | Advanced Encryption Standard |
| API | Application Program Interface |
| CO | Cryptographic Officer |
| CSP | Critical Security Parameter |
| DLL | Dynamic Link Library |
| DRNG | Deterministic Random Number Generator |
| DSA | Digital Signature Algorithm |
| EMC | Electromagnetic Compatibility |
| EMI | Electromagnetic Interference |
| FIPS | Federal Information Processing Standard |
| GPC | General Purpose Computer |
| HMAC | Keyed-Hash Message Authentication Code |
| OSCKM | Odyssey Security Component Kernel Mode |
| OSCU | Odyssey Security Component User Mode |
| RAM | Random Access Memory |
| RNG | Random Number Generator |
| RSA | Rivest, Shamir and Adleman Algorithm |
| TDES | Triple-DES |
| SHA | Secure Hash Algorithm |