

ORACLE®

Linux

FIPS 140-2 Non-Proprietary Security Policy

Oracle Linux 7 Unbreakable Enterprise Kernel (UEK 6) Cryptographic Module

FIPS 140-2 Level 1 Validation

Software Version: R7-7.8.0

Date: August 3, 2023



Title: Oracle Linux 7 Unbreakable Enterprise Kernel (UEK 6) Cryptographic Module Security Policy

Date: August 3, 2023

Author: Oracle Security Evaluations – Global Product Security

Contributing Authors:

Oracle Linux Engineering

atsec information security

Oracle Corporation
World Headquarters
2300 Oracle Way
Austin, TX 78741
U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

www.oracle.com



Copyright ©2023, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. Oracle specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may reproduced or distributed whole and intact including this copyright notice.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Hardware and Software, Engineered to Work Together



TABLE OF CONTENTS

| Section | Title | Page |
|------------|--|-----------|
| 1. | Introduction | 1 |
| 1.1 | Overview..... | 1 |
| 1.2 | Document Organization | 1 |
| 2. | Oracle Linux 7 Unbreakable Enterprise Kernel Cryptographic Module | 2 |
| 2.1 | Functional Overview..... | 2 |
| 2.2 | FIPS 140-2 Validation Scope..... | 2 |
| 3. | Cryptographic Module Specification | 3 |
| 3.1 | Definition of the Cryptographic Module | 3 |
| 3.2 | Definition of the Physical Cryptographic Boundary | 4 |
| 3.3 | Modes of Operation | 4 |
| 3.4 | Approved or Allowed Security Functions | 5 |
| 3.5 | Non-Approved Security Functions | 12 |
| 4. | Module Ports and Interfaces | 13 |
| 5. | Physical Security | 14 |
| 6. | Operational Environment..... | 15 |
| 6.1 | Tested Environments..... | 15 |
| 6.2 | Vendor Affirmed Environments | 15 |
| 6.3 | Vendor Affirmed Environments | 15 |
| 7. | Roles, Services and Authentication..... | 16 |
| 7.1 | Roles | 16 |
| 7.2 | FIPS Approved Operator Services and Descriptions | 16 |
| 7.3 | Non-FIPS Approved Services and Descriptions | 17 |
| 7.4 | Operator Authentication | 17 |
| 8. | Key and CSP Management | 18 |
| 8.1 | Random Number Generation | 18 |
| 8.2 | Key Entry/Output | 19 |
| 8.3 | Key/CSP Storage | 19 |
| 8.4 | Key/CSP Zeroization | 19 |
| 8.5 | Key establishment / Key transport..... | 19 |
| 9. | Self-Tests..... | 20 |
| 9.1 | Power-Up Self-Tests | 20 |
| 9.1.1 | Integrity Tests..... | 21 |
| 9.2 | Conditional Self-Tests..... | 21 |
| 10. | Crypto-Officer and User Guidance | 23 |
| 10.1 | Crypto-Officer Guidance..... | 23 |
| 10.1.1 | Secure Installation and Startup | 23 |
| 10.1.2 | FIPS 140-2 and AES Hardware Acceleration Support | 25 |
| 10.2 | User Guidance | 26 |
| 10.2.1 | AES-XTS Usage | 26 |
| 10.2.2 | AES-GCM Usage..... | 26 |
| 10.2.3 | Triple-DES Usage | 27 |
| 10.3 | Handling Self-Test Errors | 27 |

| | |
|---|----|
| 11. Mitigation of Other Attacks..... | 28 |
| Acronyms, Terms and Abbreviations | 29 |
| References | 30 |

List of Tables

| | |
|--|----|
| Table 1: FIPS 140-2 Security Requirements..... | 2 |
| Table 2: FIPS Approved or Allowed Security Functions | 11 |
| Table 3: Non-Approved Security Functions..... | 12 |
| Table 4: Mapping of FIPS 140 Logical Interfaces to Logical Ports | 13 |
| Table 5: Tested Operating Environment..... | 15 |
| Table 6: Vendor Affirmed Operating Environment | 15 |
| Table 7: FIPS Approved Operator Services and Descriptions | 16 |
| Table 8: Non-FIPS Approved Operator Services and Descriptions..... | 17 |
| Table 9: CSP Table..... | 18 |
| Table 10: Power-On Self-Tests | 21 |
| Table 11: Conditional Self-Tests..... | 22 |
| Table 12: Acronyms..... | 29 |
| Table 13: References | 30 |

List of Figures

| | |
|--|---|
| Figure 1: Oracle Linux UEK Logical Cryptographic Boundary..... | 4 |
| Figure 2: Oracle Linux UEK Hardware Block Diagram | 4 |



1. Introduction

1.1 Overview

The Unbreakable Enterprise Kernel (UEK 6), included as part of Oracle Linux, based on the upstream Linux kernel version 5.4.17, provides the latest open source innovations, key optimizations and security for enterprise cloud workloads. This Linux kernel powers Oracle Cloud and Oracle Engineered Systems such as Oracle Exadata Database Machine. Oracle tests UEK intensively with demanding Oracle workloads, and recommends UEK for Oracle deployments and all other enterprise deployments.

Oracle contributes to upstream Linux kernel development with enhancements that benefit Oracle Database, middleware, applications and hardware, as well as our broad partner ecosystem. These enhancements are distributed to customers through UEK for Oracle Linux.

By selectively integrating the latest open source Linux capabilities into UEK while still providing application binary compatibility with the Red Hat Compatible Kernel, Oracle makes it easy to run the most demanding cloud and enterprise workloads without compromising stability and security. We test all our on-premises software, and run Oracle Cloud on UEK, ensuring you can achieve the highest scalability and performance with your current workloads and those of the future.

This document is the Security Policy for the Oracle Linux 7 Unbreakable Enterprise Kernel (UEK 6) Cryptographic Module by Oracle Corporation. Oracle Linux 7 UEK 6 Cryptographic Module is also referred to as “the Module or Module”. This Security Policy specifies the security rules under which the module shall operate to meet the requirements of FIPS 140-2 Level 1. It also describes how the Oracle Linux 7 UEK 6 Cryptographic Module functions in order to meet the FIPS requirements, and the actions that operators must take to maintain the security of the module.

This Security Policy describes the features and design of the Oracle Linux 7 UEK 6 Cryptographic Module using the terminology contained in the FIPS 140-2 specification. FIPS 140-2, Security Requirements for Cryptographic Module specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information. The NIST/CCCS Cryptographic Module Validation Program (CMVP) validates cryptographic module to FIPS 140-2. Validated products are accepted by the Federal agencies of both the USA and Canada for the protection of sensitive or designated information.

1.2 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:

- Oracle Linux 7 Unbreakable Enterprise Kernel (UEK 6) Cryptographic Module Non-Proprietary Security Policy
- Other supporting documentation as additional references

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.

2. Oracle Linux 7 Unbreakable Enterprise Kernel Cryptographic Module

2.1 Functional Overview

The Oracle Linux 7 Unbreakable Enterprise Kernel Cryptographic Module is a software only cryptographic module that provides general-purpose cryptographic services to the remainder of the Linux kernel. The Oracle Linux 7 UEK Cryptographic Module is software only, security level 1 cryptographic module, running on a multi-chip standalone platform.

2.2 FIPS 140-2 Validation Scope

The following table shows the security level for each of the eleven sections of the validation. See Table 1 below.

| Security Requirements Section | Level |
|---|-------|
| Cryptographic Module Specification | 1 |
| Cryptographic Module Ports and Interfaces | 1 |
| Roles and Services and Authentication | 1 |
| Finite State Machine Model | 1 |
| Physical Security | N/A |
| Operational Environment | 1 |
| Cryptographic Key Management | 1 |
| EMI/EMC | 1 |
| Self-Tests | 1 |
| Design Assurance | 3 |
| Mitigation of Other Attacks | N/A |

Table 1: FIPS 140-2 Security Requirements

3. Cryptographic Module Specification

3.1 Definition of the Cryptographic Module

The Oracle Linux 7 UEK 6 Cryptographic Module is a software-only multi-chip standalone module as defined by the requirements within FIPS PUB 140-2. The logical cryptographic boundary of the module consists of binary files and their integrity check HMAC files, which are delivered through the Oracle Public Yum Package Manager (RPM) as listed below:

The list of components required for the module version R7-7.8.0 running on Oracle Linux 7.8 to operate are defined below:

- Oracle Linux 7 Unbreakable Enterprise Kernel Cryptographic Module with the version of the RPM file [kernel-uek-5.4.17-2102.200.13.el7uek.x86_64.rpm](#), [kernel-uek-5.4.17-2102.200.13.el7uek.aarch64.rpm](#), and [kernel-uekemb-5.4.17-2102.200.13.1.el7uek.mips64.rpm](#).
- The configuration of the FIPS mode is provided by the dracut-fips version [dracut-fips-033-572.0.5.el7.x86_64.rpm](#), [dracut-fips-033-572.0.5.el7.aarch64.rpm](#), or [dracut-fips-033-572.0.201.el7.mips64.rpm](#), and the dracut-fips-aesni package with the version [dracut-fips-aesni-033-572.0.5.el7.x86_64.rpm](#), or [dracut-fips-aesni-033-572.0.5.el7.aarch64.rpm](#).
- [libkcapi-1.2.0-2.0.1.el7.x86_64.rpm](#), [libkcapi-1.2.0-2.0.1.el7.aarch64.rpm](#), and [libkcapi-1.2.0-2.0.200.el7.mips64](#).
- [libkcapi-hmaccalc-1.2.0-2.0.1.el7.x86_64.rpm](#), [libkcapi-hmaccalc-1.2.0-2.0.1.el7.aarch64.rpm](#), and [libkcapi-hmaccalc-1.2.0-2.0.200.el7.mips64](#).

The Oracle Linux UEK 6 RPM package of the Module includes the binary files, integrity check HMAC files and Man Pages. The files comprising the module are the following:

- kernel loadable components `/lib/modules/$(uname -r)/kernel/crypto/*.ko`
- kernel loadable components `/lib/modules/$(uname -r)/kernel/arch/x86/crypto/*.ko`
- kernel loadable components `/lib/modules/$(uname -r)/kernel/arch/aarch/crypto/*.ko`
- static kernel binary `/boot/vmlinuz-$(uname -r)`
- static kernel binary HMAC file `/boot/.vmlinuz-$(uname -r).hmac`
- sha512hmac binary file for performing the integrity checks: `usr/bin/sha512hmac`
- sha512hmac binary HMAC file: `/usr/lib64/hmaccalc/sha512hmac.hmac`
- libkcapi library: `/usr/lib64/libkcapi.so.1.2.0`
- libkcapi library HMAC file: `/usr/lib64/fipscheck/libkcapi.so.1.2.0.hmac`

Figure 1 shows the logical block diagram of the module executing in memory on the host system.

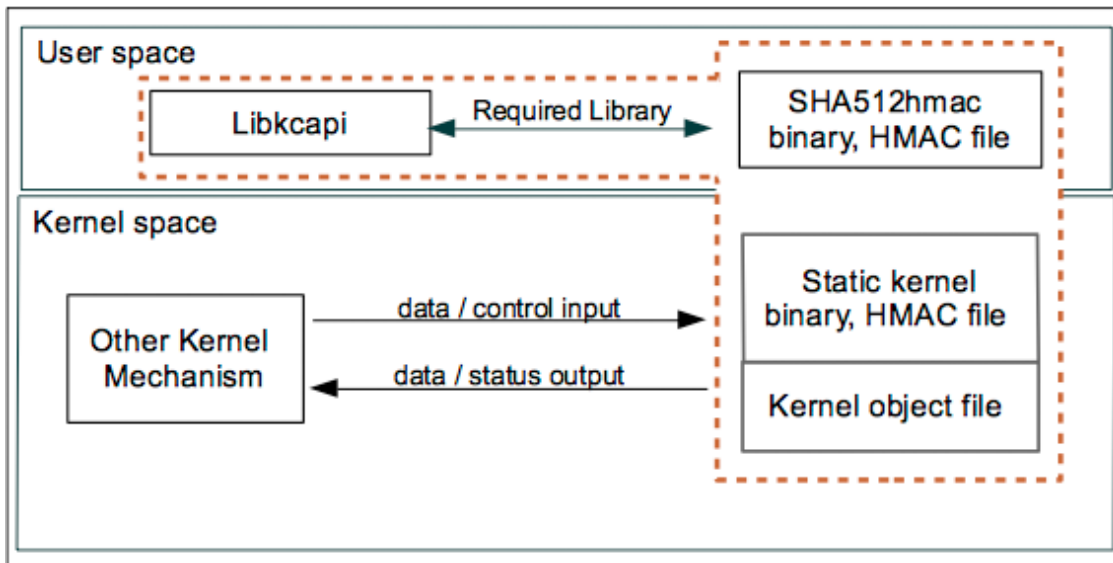


Figure 1: Oracle Linux UEK Logical Cryptographic Boundary

3.2 Definition of the Physical Cryptographic Boundary

The physical cryptographic boundary is defined as the hard enclosure of the host system on which it runs. See figure 2 below. No components are excluded from the requirements of FIPS PUB 140-2.

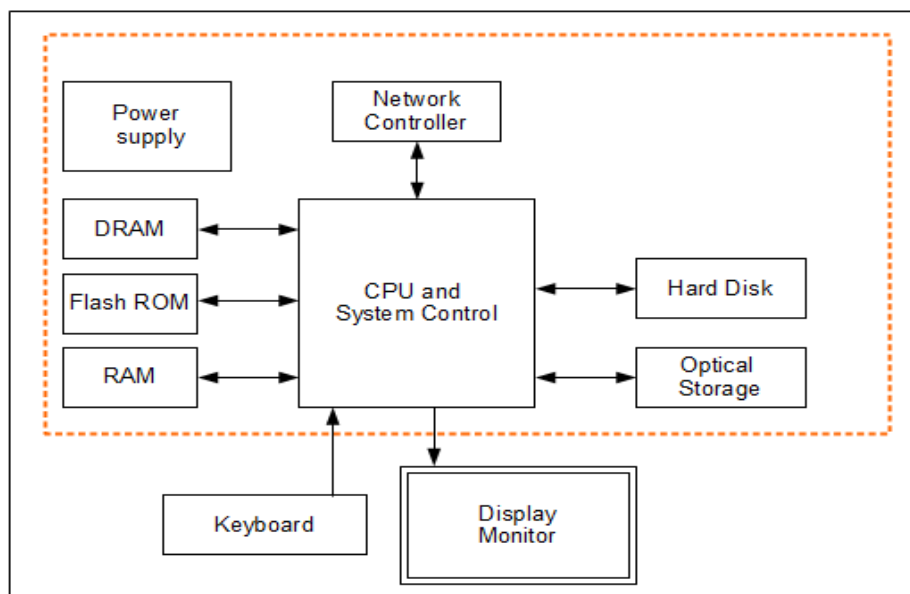


Figure 2: Oracle Linux UEK Hardware Block Diagram

3.3 Modes of Operation

The module supports two modes of operation: the FIPS approved and non-approved modes.

Section 10 describes the Crypto Officer and User Guidance to correctly install, configure, and use the module in the FIPS Approved mode of operation. The module turns to FIPS Approved mode after correct initialization and successful completion of power-on self-tests.

Invoking a non-Approved algorithm or a non-Approved key size with an Approved algorithm as listed in **Table 3** will result in the module implicitly entering the non-FIPS mode of operation. The critical security parameters (CSPs) used or stored in approved mode are not used in non-approved mode and vice versa. Once the module is operational, the mode of operation is implicitly assumed depending on the security function invoked and the security strength of the cryptographic keys.

The Approved services available in FIPS mode can be found in section 7.2, **Table 7**. The non-approved services not available in FIPS mode can be found in section 7.3, **Table 8**.

3.4 Approved or Allowed Security Functions

The Oracle Linux 7 Unbreakable Enterprise Kernel (UEK 6) Cryptographic Module contains the following FIPS Approved Algorithms:

Note: Not all algorithms tested with the CAVP certificates are implemented in the module.

| Approved or Allowed Security Functions | | Certificate |
|--|---|-------------|
| Symmetric Algorithms | | |
| AES | AESNI_C: AES in CBC, ECB, CTR, CCM, CMAC (MAC generation and verification), GCM (with external IV, only decryption is approved), GMAC, XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 1044 |
| | C_C AES in CBC, ECB, CTR, CCM, CMAC (MAC generation and verification), GCM (with external IV, only decryption is approved), GMAC, XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 1038 |
| | RFC4106IIV_AESNI_ASM: AES in ECB and GCM Modes; Internal IV (E; Key Sizes 128, 192, 256. With internal IV only encryption is approved) | A 1046 |
| | RFC4106IIV_AESNI_C: AES in ECB and GCM Modes; Internal IV (E; Key Sizes 128, 192, 256. With internal IV only encryption is approved) | A 1043 |
| | RFC4106IIV_C_C: AES in ECB and GCM Modes; Internal IV (E; Key Sizes 128, 192, 256. With internal IV only encryption is approved) | A 1034 |
| | RFC4106EIV_C_C: AES in ECB and GCM Modes; External IV (D; Key Sizes 128, 192, 256. With external IV, only decryption is approved) | A 1033 |
| | RFC4106EIV_AESNI_C: AES in ECB and GCM Modes; External IV (D; Key Sizes 128, 192, 256. With external IV, only decryption is approved) | A 1042 |
| | RFC4106EIV_AESNI_ASM: AES in ECB and GCM Modes; External IV (D; Key Sizes 128, 192, 256. With external IV, only decryption is approved) | A 1045 |
| | CTS_AESNI_C: AES in CBC-CS3 Mode (E/D; Key Sizes 128, 192, 256) | A 1055 |
| | CFB_AESNI_C: AES in CFB128 Mode (E/D; Key Sizes 128, 192, 256) | A 1041 |

| Approved or Allowed Security Functions | | Certificate |
|--|--|-------------|
| | CFB_C_C: AES in CFB128 Mode (E/D; Key Sizes 128, 192, 256) | A 1036 |
| | CFB_CTI_C: AES in CFB128 Mode (E/D; Key Sizes 128, 192, 256) | A 1049 |
| | AESNI_ASM: AES in CBC, ECB, CTR, GCM (with external IV, only decryption is approved), XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 1047 |
| | OFB_C_C: AES in OFB Mode (E/D; Key Sizes 128, 192, 256) | A 1035 |
| | OFB_CTI_C: AES in OFB Mode (E/D; Key Sizes 128, 192, 256) | A 1048 |
| | OFB_AESNI_C: AES in OFB Mode (E/D; Key Sizes 128, 192, 256) | A 1056 |
| | CTI_C: AES in CBC, ECB, CTR, CCM, CMAC (MAC generation and verification), GCM (with external IV, only decryption is approved), GMAC, XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 1032 |
| | RFC4106IIV_CTI_C: AES in ECB and GCM Modes; Internal IV (E; Key Sizes 128, 192, 256. With internal IV only encryption is approved) | A 1031 |
| | RFC4106EIV_CTI_C: AES in ECB and GCM Modes; External IV (D; Key Sizes 128, 192, 256. With external IV, only decryption is approved) | A 1050 |
| | ARM64_CE: AES in CBC, ECB, CTR, CCM, CMAC (MAC generation and verification), XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 2374 |
| | ARM64_CE_C: AES in CBC, CBC-CS3, ECB, CFB128, OFB, CTR, CCM, CMAC (MAC generation and verification), GCM (with external IV, only decryption is approved), GMAC, XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 2377 |
| | RFC4106IIV_ARM64_CE_C: AES in ECB and GCM Modes; Internal IV (E; Key Sizes 128, 192, 256. With internal IV only encryption is approved) | A 2378 |
| | RFC4106EIV_ARM64_CE_C: AES in ECB and GCM Modes; External IV (D; Key Sizes 128, 192, 256. With external IV only decryption is approved) | A 2379 |
| | ARM64_NEON: AES in CBC, ECB, CTR, XTS Modes (E/D; Key Sizes 128, 192, 256 for all modes except XTS Mode where key sizes are 128 and 256) | A 2380 |
| Triple DES | C_C: TDES in CBC, ECB, CTR, and CMAC (MAC generation and verification) Modes (E/D; Key option 1) | A 1038 |
| | X86ASM_ASM: TDES in CBC, ECB, and CTR, Modes (E/D; Key option 1) | A 1054 |

| Approved or Allowed Security Functions | | Certificate |
|--|--|-------------|
| | CFB C C: TDES in CFB64 Mode (E/D; Key option 1) | A 1036 |
| | OFB C C: TDES in OFB Mode (E/D; Key option 1) | A 1035 |
| Secure Hash Standard (SHS) | | |
| SHS | AVX: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 | A 1052 |
| | AVX2: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 | A 1051 |
| | SSSE3: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 | A 1053 |
| | C C: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 | A 1038 |
| | SHA-3 C C: SHA3-224, SHA3-256, SHA3-384, SHA3-512 (Supports empty message) | A 1037 |
| | ARM64 CE: SHA-1, SHA-224, SHA-256 | A 2374 |
| | ARM64 CE SHA-3: SHA3-224, SHA3-256, SHA3-384, SHA3-512 | A 2375 |
| | ARM64 CE SHA512: SHA-384, SHA-512 | A 2376 |
| | ARM64 NEON: SHA-224, SHA-256 | A 2380 |
| | ARM64 ASM: SHA-224, SHA-256, SHA-384, SHA-512 | A 2381 |
| | COP2 SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 | A 3482 |
| Data Authentication Code | | |
| HMAC | AVX: HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | A 1052 |
| | AVX2: HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | A 1051 |
| | SSSE3: HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | A 1053 |
| | C C: HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | A 1038 |
| | SHA-3 C C: HMAC-SHA3-224, HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 | A 1037 |
| | ARM64 CE: HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256 | A 2374 |
| | ARM64 CE SHA-3: HMAC-SHA3-224, HMAC-SHA3-256, HMAC-SHA3-384, HMAC-SHA3-512 | A 2375 |
| | ARM64 CE SHA512: SHA-384, SHA-512 | A 2376 |

| Approved or Allowed Security Functions | | Certificate |
|--|--|-------------|
| | <u>ARM64 NEON:</u> HMAC-SHA-224, HMAC-SHA-256 | A 2380 |
| | <u>ARM64 ASM:</u> HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | A 2381 |
| | <u>COP2</u> HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512 | A 3482 |
| Asymmetric Algorithms | | |
| RSA | <u>AVX:</u> PKCS 1.5 (Sig Ver); Modulus Sizes 2048, 3072, 4096 with Hash sizes SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. | A 1052 |
| | <u>AVX2:</u> PKCS 1.5 (Sig Ver); Modulus Sizes 2048, 3072, 4096 with Hash sizes SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. | A 1051 |
| | <u>SSSE3:</u> PKCS 1.5 (Sig Ver); Modulus Sizes 2048, 3072, 4096 with Hash sizes SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. | A 1053 |
| | <u>C_C:</u> PKCS 1.5 (Sig Ver); Modulus Sizes 2048, 3072, 4096 with Hash sizes SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. | A 1038 |
| Random Number Generation | | |
| DRBG | <u>RFC4106IIV_AESNI_ASM:</u> CTR_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] | A 1046 |
| | <u>RFC4106EIV_C_C:</u> CTR_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] | A 1033 |
| | <u>RFC4106EIV_AESNI_C:</u> CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128 , AES-192 , AES-256)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] | A 1042 |
| | <u>RFC4106EIV_AESNI_ASM:</u> CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] | A 1045 |

| Approved or Allowed Security Functions | | Certificate |
|---|--------|-------------|
| <p><u>AESNI C:</u> CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1044 | |
| <p><u>RFC4106IIV AESNI C:</u> CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128 , AES-192 , AES-256)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> | A 1043 | |
| <p><u>AESNI ASM:</u> CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128 , AES-192 , AES-256)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> | A 1047 | |
| <p><u>RFC4106IIV C C:</u> CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> | A 1034 | |
| <p><u>AVX2:</u> Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)] HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> | A 1051 | |
| <p><u>X86ASM ASM:</u> HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1054 | |
| <p><u>SSSE3:</u> HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1053 | |
| <p><u>AVX:</u> HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)] Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1052 | |
| <p><u>C C:</u></p> | A 1038 | |

| Approved or Allowed Security Functions | | Certificate |
|--|---|--|
| | <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> <p>HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> <p>Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | |
| | <p>CTI C:</p> <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> <p>HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> <p>Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1032 |
| | <p>RFC4106IIV CTI C:</p> <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> <p>HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> <p>Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1031 |
| | <p>RFC4106EIV CTI C:</p> <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> <p>HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> <p>Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 1050 |
| | <p>ARM64 CE C:</p> <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> | A 2377 |
| | <p>RFC4106IIV ARM64 CE C:</p> <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> <p>HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> <p>Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 2378 |
| | <p>RFC4106EIV ARM64 CE C:</p> <p>CTR_DRBG: [With Derivation: Prediction Resistance Tested: Enabled and Not Enabled; Supports Reseed: (AES-128, AES-192, AES-256)]</p> <p>HMAC_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, HMAC-SHA-512)]</p> <p>Hash_DRBG: [Prediction Resistance Tested: Enabled and Not Enabled; (SHA-1, SHA-256, SHA-384, SHA-512)]</p> | A 2379 |
| Key Transport Scheme | | |
| KTS | AES-GCM key wrapping with 128, 192 and 256 bit keys | A 1031 A 1034 A 1043 A 1046 |

| Approved or Allowed Security Functions | | Certificate |
|--|---|--|
| | | A 2378 A 2379 |
| | AES-CCM key wrapping with 128, 192 and 256 bit keys | A 1032 A 1038 A 1044 A 2374 A 2377 |
| | AES-CBC with HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, or HMAC-SHA-512 key wrapping with 128, 192 and 256 bit keys | A 1032 A 1038 A 1044 A 1047 A 1038 A 1051 A 1052 A 1053 A 2374 A 2376 A 2377 A 2380 A 2381 A 3482* (*excluding HMAC-SHA-224) |
| | Triple-DES CBC with HMAC-SHA-1, HMAC-SHA-224, HMAC-SHA-256, HMAC-SHA-384, or HMAC-SHA-512 key wrapping with 192 ¹ bit key. | A 1038 A 1054 A 1051 A 1052 A 1053 A 2374 A 2376 A 2380 A 2381 A 3482* (*excluding HMAC-SHA-224) |
| Entropy | | |
| ENT(NP) | NIST SP 800-90B | E 37 ² |

Table 2: FIPS Approved or Allowed Security Functions

¹ Though the key size is 192 bits, the strength of that key and therefore the KTS implementation for Triple-DES is 112 bits only according to IG 7.5.

² The entropy source validation public use document for cert E 37 can be found [here](#).

3.5 Non-Approved Security Functions

The following algorithms are considered non-Approved and may not be used in a FIPS-approved mode of operation. The services associated with these algorithms are specified in section 7.3.

| Algorithm | Usage |
|-------------------------|------------------------------|
| AES-XTS (192 bit) | Encrypt/Decrypt |
| AES GCM | Encryption with external IV |
| RSA | Encrypt/Decrypt |
| RSA | Signature Generation |
| Diffie-Hellman | Key Agreement |
| EC Diffie-Hellman | Key Agreement |
| SHA-1 (multiple-buffer) | Hashing |
| HMAC | HMAC Keys less than 112 bits |

Table 3: Non-Approved Security Functions

4. Module Ports and Interfaces

The module interfaces can be categorized as follows:

- Data Input Interface
- Data Output Interface
- Control Input interface
- Status Output Interface

The module can be accessed by utilizing the API it exposes. Table below, shows the mapping of ports and interfaces as per FIPS 140-2 Standard.

| FIPS 140 Interface | Module Interfaces |
|--------------------|---|
| Data Input | API input parameters |
| Data Output | API output parameters |
| Control Input | API function calls, kernel command line |
| Status Output | API return codes, kernel logs |

Table 4: Mapping of FIPS 140 Logical Interfaces to Logical Ports



5. Physical Security

The Module is comprised of software only and thus does not claim any physical security.

6. Operational Environment

6.1 Tested Environments

The module operates in a modifiable operational environment per FIPS 140-2 level 1 specifications. The Module was tested on the following environments:

| Module Version | Operating Environment | Processor | Hardware |
|----------------|-------------------------|---------------------------|--|
| R7-7.8.0 | Oracle Linux 7.8 64 bit | Intel® Xeon® 8167M | Oracle Server X7-2C with and without PAA (i.e. AES-NI) |
| R7-7.8.0 | Oracle Linux 7.8 64 bit | AMD EPYC™ 7551 | Oracle Server E1-2C with and without PAA (i.e. AES-NI) |
| R7-7.8.0 | Oracle Linux 7.8 64 bit | Ampere®Altra® Neoverse-N1 | Oracle Server A1-2C with and without PAA (i.e. AES-NI) |
| R7-7.8.0 | Oracle Linux 7.8 64 bit | Marvell OCTEON III | Marvell Liquid IO II without PAA |

Table 5: Tested Operating Environment

6.2 Vendor Affirmed Environments

The following platforms have not been tested as part of the FIPS 140-2 level 1 certification however Oracle “vendor affirms” that these platforms are equivalent to the tested and validated platforms. Additionally, Oracle affirms that the module will function the same way and provide the same security services on any of the systems listed below.

| Operating Environment | Hardware |
|-----------------------|--|
| Oracle Linux 7 64-bit | Oracle X Series Servers |
| Oracle Linux 7 64-bit | Oracle E Series Servers |
| Oracle Linux 7 64-bit | Oracle A Series Servers |
| Oracle Linux 7 64-bit | Marvell CN23XX OCTEON (MIPS) SmartNIC |
| Oracle Linux 7 64-bit | Marvell CN93XX LiquidIO III (ARM) SmartNIC |
| Oracle Linux 7 64-bit | Pensando DSC-200 (ARM) SmartNIC |

Table 6: Vendor Affirmed Operating Environment

CMVP makes no statement as to the correct operation of the module or the security strengths of the generated keys when so ported if the specific operational environment is not listed on the validation certificate.

6.3 Vendor Affirmed Environments

The operating system is restricted to a single operator (concurrent operators are explicitly excluded). The application that request cryptographic services is the single user of the module, even when the application is serving multiple clients.

In FIPS Approved mode, the ptrace(2) system call, the debugger (gdb(1)), and strace(1) shall be not used.

7. Roles, Services and Authentication

7.1 Roles

The roles are implicitly assumed by the entity accessing the module services. The module supports the following roles:

- **User Role:** performs symmetric encryption/decryption, keyed hash, message digest, random number generation, show status, zeroization.
- **Crypto Officer Role:** performs the module installation and configuration, module's initialization, self-tests.

7.2 FIPS Approved Operator Services and Descriptions

The below table provides a full description of FIPS Approved services provided by the module and the roles allowed to invoke each service.

| U | CO | Service Name | Service Description | Keys and CSP(s) | Access Type(s) |
|---|----|-----------------------------------|--|--|----------------|
| X | | Symmetric Encryption/Decryption | Encrypts or decrypts a block of data using 3-Key Triple-DES or AES in FIPS mode | AES or 3-Key Triple-DES Key | R, W, X |
| X | | Keyed Hash (HMAC) | Sign and or authenticate data using HMAC-SHA | HMAC Key | R, W, X |
| X | | Hash (SHS) | Hash a block of data. | None | N/A |
| X | | Random Number Generation | Generate random numbers based on the NIST SP 800-90A Standard | Entropy input string, seed, internal state | R, W, X |
| X | | Authenticated Encryption | Encrypt-then-MAC cipher (authenc) used for IPsec | AES key, Triple-DES Key, HMAC key | R, W, X |
| X | | Key Wrapping | NIST SP 800-38F key wrapping with AES and Triple-DES | AES key, Triple-DES Key, HMAC Key | R, W, X |
| X | | Show Status | Show status of the module state via verbose mode, exit codes and kernel logs (dmesg) | None | N/A |
| | X | Self-Test | Initiate power-on self-tests | None | N/A |
| X | | Zeroize | Zeroize all critical security parameters | All keys and CSP's | Z |
| | X | Module Initialization | Initialize the module into the FIPS Approved Mode | None | N/A |
| | X | Installation and Configuration | Install and configure the module. | None | N/A |
| X | | Error detection code ³ | Error detection code using crc32c, crct10dif | None | N/A |
| X | | Data compression ¹ | Performs data compression using deflate, lz4, lz4hc, lzo, zlib | None | N/A |

R – Read, W – Write, X – Execute, Z – Zeroize

Table 7: FIPS Approved Operator Services and Descriptions

³ The algorithms used in this service do not provide cryptographic attribute.

7.3 Non-FIPS Approved Services and Descriptions

The following table lists the non-Approved services available in non-FIPS mode.

| U | CO | Service Name | Service Description | Keys | Access Type(s) |
|---|----|---|--|---|----------------|
| X | | Symmetric Encryption/Decryption | Encrypts or decrypts using non-Approved algorithms from Table 3 | AES-XTS (192-bit key) and AES GCM encryption with external IV | R, W, X |
| X | | Asymmetric Encryption/Decryption | Encrypts or decrypts using non-Approved algorithms from Table 3 | RSA public and private keys | R, W, X |
| X | | Digital Signature Generation/Verification | Signs or verifies using non-Approved algorithms from Table 3 | RSA | R, W, X |
| X | | Key Agreement | Diffie-Hellman and EC Diffie-Hellman key agreement | Public and private keys | R, W, X |
| X | | Message Digest | Hashing using hash functions from SHA-1 mb | None | N/A |
| X | | Keyed Hash | HMAC Keys < 112 bits. | HMAC keys < 112 bits. | R, W, X |

R – Read, W – Write, X – Execute, Z – Zeroize

Table 8: Non-FIPS Approved Operator Services and Descriptions

7.4 Operator Authentication

The module is a Level 1 software-only cryptographic module and does not implement authentication. The role is implicitly assumed based on the service requested.

8. Key and CSP Management

The following keys, cryptographic key components and other critical security parameters are contained in the module.

| CSP Name | Generation | Entry/Output | Storage | Zeroization |
|---|--|---|---|---|
| AES Keys (128, 192, 256 bits) | N/A | The Key is passed into the module via API input parameter | kernel memory | Memory is automatically overwritten by zeroes when freeing the cipher handler |
| Triple-DES Keys (192 bits) | N/A | The Key is passed into the module via API input parameter | kernel memory | Memory is automatically overwritten by zeroes when freeing the cipher handler |
| RSA Public Key (only used for integrity test) | N/A | N/A | kernel memory | N/A |
| DRBG Entropy Input String | Obtained from entropy source | N/A | kernel memory | Memory is automatically overwritten by zeroes when freeing the cipher handler |
| DRBG seed, internal state (V, key and C values) | Derived from Entropy input as defined in NIST SP 800-90A | N/A | kernel memory | Memory is automatically overwritten by zeroes when freeing the cipher handler |
| HMAC Keys (≥ 112 bits) | N/A | The Key is passed into the module via API input parameter | kernel memory | Automatically zeroized when freeing the cipher handle |
| HMAC Integrity Key | N/A Installed with the module | N/A | Plaintext as part of the hmacsha512 application | Zeroized in memory by hmacsha512 |

Table 9: CSP Table

8.1 Random Number Generation

The module employs the Deterministic Random Bit Generator (DRBG) based on [SP 800-90A] for the random number generation. The DRBG supports the Hash_DRBG, HMAC_DRBG and CTR_DRBG mechanisms. The module performs the DRBG health tests as defined in section 11.3 of [SP 800-90A]. The module uses CPU jitter as an entropy source for seeding the DRBG. The source is compliant with [SP 800-90B] and marked as ENT(NP) on the certificate with cert #E37. The entropy source is tested with RCT and APT Health tests as required by section 4 of [SP 800-90B]. The DRBG is seeded with (DRBG_security_strength * 1.5) bits of random data from the CPU jitter RNG containing at least DRBG_security_strength bits of entropy. (e.g. 384 bits for the CTR_DRBG using AES-256). Therefore, the module ensures that during initialization (seed) and reseeding, the entropy source provides the required amount of entropy to meet the security strength of the respective DRBG methods.



The module does not provide any key generation service or perform key generation for any of its Approved algorithms. Keys are passed in from calling application via API parameters.

8.2 Key Entry/Output

An authorized application as user (the User role) has access to all key data generated during the operation of the module. Moreover, the module does not support the output of intermediate key generation values during the key generation process. The module does not support manual key entry.

8.3 Key/CSP Storage

Symmetric keys are provided to the module by the calling process and are destroyed when released by the appropriate API function calls. The module does not perform persistent storage of keys. The RSA public key used for signature verification of the kernel loadable components is stored outside of the module's boundary, in a keyring file in `/proc/keys/`.

8.4 Key/CSP Zeroization

The application that uses the module is responsible for appropriate destruction and zeroization of the key material. The module provides functions for key allocation and destruction. When a calling kernel components calls the appropriate API function that operation overwrites memory with 0's and then frees that memory.

8.5 Key establishment / Key transport

The module provides SP 800-38F compliant key wrapping using AES with GCM and CCM block chaining modes, as well as a combination of AES-CBC for encryption/decryption and HMAC for authentication. The module also provides SP 800-38F compliant key wrapping using a combination of Triple-DES-CBC for encryption/decryption and HMAC for authentication.

According to "Table 2: Comparable strengths" in [SP 800-57], the key sizes of AES and Triple-DES provides the following security strength in FIPS mode of operation:

- KTS (AES Certs. [#A1031](#), [#A1032](#), [#A1034](#), [#A1038](#), [#A1043](#), [#A1044](#), [#A1046](#), [#A2374](#), [#A2377](#), [#A2378](#) and [#A2379](#); key establishment methodology provides between 128 and 256 bits of encryption strength).
- KTS (AES Certs. [#A1032](#), [#A1038](#), [#A1044](#), [#A1047](#), [#A2374](#), [#A2377](#), and [#A2380](#) and HMAC Certs. [#A1038](#), [#A1051](#), [#A1052](#), [#A1053](#), [#A2374](#), [#A2376](#), [#A2380](#), [#A2381](#), and [#A3482](#); key establishment methodology provides between 128 and 256 bits of encryption strength).
- KTS (Triple-DES Certs. [#A1038](#) and [#A1054](#) and HMAC Certs. [#A1038](#), [#A1051](#), [#A1052](#), [#A1053](#), [#A2374](#), [#A2376](#), [#A2380](#), [#A2381](#), and [#A3482](#); key establishment methodology provides 112 bits of encryption strength).

9. Self-Tests

FIPS 140-2 requires that the Module perform self-tests to ensure the integrity of the Module and the correctness of the cryptographic functionality at start up. On successful completion of the power-up tests, the module is operational, and the crypto services are available. A failure of any of the self-tests panics the Module and no crypto operations are possible. The only recovery is to reboot the module. See section 10.3 for details. No operator intervention is required during the running of the self-tests.

9.1 Power-Up Self-Tests

The module performs power-up self-tests at module initialization without operator intervention. While the module is performing the power-up tests, services are not available, and input or output is not possible. The on-demand power up self-tests can be performed by power cycling the Module or by rebooting the operating system. Input, output, and cryptographic functions cannot be performed while the Module is in a self-test or error state. The Module is single-threaded during the self-tests and will stop the boot procedure, and therefore any subsequent operation before any other kernel component can request services from the Module. The Crypto Officer with physical or logical access to the Module can run the POST (Power-On Self-Tests) on demand by power cycling the Module or by rebooting the operating system. The table below summarizes the power-on self-tests performed by the module. If the known answer does not match the test fails. The different implementations of the same algorithms listed in **Table 2** are tested separately by performing the known-answer tests using the same test vectors.

| Algorithm | Test |
|-------------------------------|--|
| AES | KAT, encryption and decryption are tested separately. <ul style="list-style-type: none"> • ECB (128, 192 and 256-bit keys) • CBC (128, 192 and 256-bit keys) • CTR (128, 192 and 256-bit keys) • CCM (128, 192 and 256-bit keys) • CFB (128, 192 and 256-bit keys) • XTS (128, 192 and 256-bit keys) • CMAC (128, 192 and 256-bit keys) |
| AES-GCM | KAT, encryption and decryption are tested separately with 128, 192 and 256-bit keys. |
| Triple-DES | KAT, encryption and decryption are tested separately. <ul style="list-style-type: none"> • ECB • CBC • CTR • CMAC |
| SP 800-90A CTR_DRBG | KAT |
| SP 800-90A Hash_DRBG | KAT |
| SP 800-90A HMAC_DRBG | KAT |
| HMAC | (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512) KAT |
| HMAC-SHA-3 | (SHA3-224, SHA3-256, SHA3-384, SHA3-512) KAT |
| SHA-1, -224, -256, -384, -512 | KAT |
| SHA3-224, -256, -384, -512 | KAT |
| Module Integrity test | HMAC SHA-512, RSA signature verification |
| RSA Signature Verification | Part of the integrity test (considered as a KAT) |

Table 10: Power-On Self-Tests

9.1.1 Integrity Tests

An HMAC SHA-512 calculation is performed on the sha512hmac utility and static Linux kernel binary to verify their integrity. The Linux kernel crypto API kernel components, and any additional code components loaded into the Linux kernel are checked with the RSA signature verification implementation of the Linux kernel when loading them into the kernel to confirm their integrity.

NOTE: The fact that the kernel integrity check passed, which requires the loading of sha512hmac with the self-tests implies a successful execution of the integrity and self-tests of sha512hmac (the HMAC is stored in /usr/lib/hmaccalc/sha512hmac.hmac).

With respect to the integrity check of kernel loadable components providing the cryptographic functionality, the fact that the self test of these cryptographic components are displayed implies that the integrity checks of each kernel component passed successfully.

9.2 Conditional Self-Tests

The module performs conditional tests on the cryptographic algorithms shown in the following table:

| Algorithm | Test |
|-----------|---|
| ENT(NP) | The module performs SP 800-90B health tests of RCT and APT |
| DRBG | DRBG health tests as specified in section 11.3 of NIST SP 800-90Ar1 |

Table 11: Conditional Self-Tests

10. Crypto-Officer and User Guidance

This section provides guidance for the Cryptographic Officer and the User to maintain proper use of the module per FIPS 140-2 requirements.

10.1 Crypto-Officer Guidance

To operate the UEK module, the operating system must be restricted to a single operator mode of operation. (This should not be confused with single user mode which is runlevel 1 on Oracle Linux. This refers to processes having access to the same cryptographic instance which Oracle Linux ensures cannot happen by the memory management hardware.)

10.1.1 Secure Installation and Startup

Crypto Officers use the Installation instructions to install the Module in their environment. The version of the RPM containing the FIPS validated module is stated in section 3.1 above.

The RPM package of the Module shall be downloaded from the Oracle Linux 7 "Security Validation (Update 8)" yum repository and installed by standard tools recommended for the installation of Oracle packages on an Oracle Linux system (for example, yum, RPM, and the RHN remote management tool). The integrity of the RPM is automatically verified during the installation of the Module and the Crypto Officer shall not install the RPM file if the [Oracle Linux Yum Server](#) indicates an integrity error. The RPM files listed in section 3 are signed by Oracle and during installation; Yum performs signature verification which ensures as secure delivery of the cryptographic module. If the RPM packages are downloaded manually, then the CO should run 'rpm -K <rpm-file-name>' command after importing the builder's GPG key to verify the package signature. In addition, the CO can also verify the hash of the RPM package to confirm a proper download.

To configure the operating environment to support FIPS perform the following steps:

1. Install Kernel UEK RPM file e.g for x86_64 or aarch use yum command
 # yum install [kernel-uek-5.4.17-2102.200.13.el7uek.x86_64.rpm](#), [kernel-uek-5.4.17-2102.200.13.el7uek.aarch64.rpm](#), or [kernel-uekemb-5.4.17-2102.200.13.1.el7uek.mips64.rpm](#)
2. Install libkcapi RPM file
 # yum install [libkcapi-1.2.0-2.0.1.el7.x86_64.rpm](#), [libkcapi-1.2.0-2.0.1.el7.aarch64.rpm](#), or [libkcapi-1.2.0-2.0.200.el7.mips64](#)
3. Install libkcapi-hmaccalc RPM file
 # yum install [libkcapi-hmaccalc-1.2.0-2.0.1.el7.x86_64.rpm](#), [libkcapi-hmaccalc-1.2.0-2.0.1.el7.aarch64.rpm](#), or [libkcapi-hmaccalc-1.2.0-2.0.200.el7.mips64](#)
4. Install the dracut-fips package:
 # yum install [dracut-fips-033-572.0.5.el7.x86_64.rpm](#), [dracut-fips-033-572-0.5.el7.aarch64.rpm](#), or [dracut-fips-033-572.0.201.el7.mips64.rpm](#)
5. Install the dracut-fips-aesni package (if AES-NI or ARM optimizations are supported):
 To check if AES-NI is supported run:
 # grep aes /proc/cpuinfo
 If it is supported, run:
 # yum install [dracut-fips-aesni-033-572.0.5.el7.x86_64.rpm](#) or [dracut-fips-aesni-033-572.0.5.el7.aarch64.rpm](#)
6. Recreate the INITRAMFS image:
 # dracut -f

7. Perform the following steps to configure the boot loader so that the system boots into FIPS mode:

- a) Identify the boot partition and the UUID of the partition. If /boot or /boot/efi resides on a separate partition, the kernel parameter `boot=<partition of /boot or /boot/efi>` must be supplied. The partition can be identified with the command:

```
# df /boot or df /boot/efi
```

| <u>Filesystem</u> | <u>1K-blocks</u> | <u>Used</u> | <u>Available</u> | <u>Use%</u> | <u>Mounted on</u> |
|-------------------|------------------|-------------|------------------|-------------|-------------------|
| /dev/sda1 | 233191 | 30454 | 190296 | 14% | /boot |

```
# blkid /dev/sda1
```

```
/dev/sda1: UUID="6046308a-75fc-418e-b284-72d8bfad34ba" TYPE="xfs"
```

- b) As the root user, edit the /etc/default/grub file as follows:

- i. Add the `fips=1` option to the boot loader configuration.


```
GRUB_CMDLINE_LINUX="vconsole.font=latarcyrheb-sun16
rd.lvm.lv=ol/swap rd.lvm.lv=ol/root crashkernel=auto
vconsole.keymap=uk rhgb quiet fips=1"
```
- ii. If the contents of /boot reside on a different partition to the root partition, you must use the `boot=UUID=boot_UUID` line to the boot loader configuration to specify the device that should be mounted onto /boot when the kernel loads.


```
GRUB_CMDLINE_LINUX="vconsole.font=latarcyrheb-sun16
rd.lvm.lv=ol/swap rd.lvm.lv=ol/root crashkernel=auto
vconsole.keymap=uk rhgb quiet
boot=UUID=6046308a-75fc-418e-b284-72d8bfad34ba fips=1"
```
- iii. Save the changes.

This is required for FIPS to perform kernel validation checks, where it verifies the kernel against the provided HMAC file in the /boot directory.

Note:

On systems that are configured to boot with UEFI, /boot/efi is located on a dedicated partition as this is formatted specifically to meet UEFI requirements. This does not automatically mean that /boot is located on a dedicated partition.

Only use the `boot=` parameter if /boot is located on a dedicated partition. If the parameter is specified incorrectly or points to a non-existent device, the system may not boot.

If the system is no longer able to boot, you can try to modify the kernel boot options in grub to specify an alternate device for the `boot=UUID=boot_UUID` parameter, or remove the parameter entirely.

8. Rebuild the GRUB configuration as follows:

On BIOS-based systems, run the following command:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

On UEFI-based systems, run the following command:

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

To ensure proper operation of the in-module integrity verification, prelinking must be disabled on all system files. By default, the prelink package is not installed on the system. However, if it is installed, disable prelinking on all libraries and binaries as follows:

Set `PRELINKING=no` in the `/etc/sysconfig/prelink` configuration file.

If the libraries were already prelinked, undo the prelink on all of the system files as follows:

```
# prelink -u -a
```

9. Reboot the system

10. Verify that FIPS Mode is enabled by running the command:

```
# cat /proc/sys/crypto/fips_enabled
```

The response should be “1”

The version of the RPM containing the validated Modules is the version listed in Section 3. The integrity of the RPM is automatically verified during the installation of the Modules and the Crypto Officer shall not install the RPM file if the RPM tool indicates an integrity error.

10.1.2 FIPS 140-2 and AES Hardware Acceleration Support

According to the UEK FIPS 140-2 Security Policy, the UEK module supports the AES-NI Intel processor instruction and ARM AES optimizations set as an approved cipher. Both architecture optimizations are used by the Module.

In case you configured a full disk encryption using AES, you *may* use the aforementioned optimizations for a higher performance compared to the software-only implementation. To utilize the AES-NI support, the mentioned Module must be loaded during boot time by installing a plugin.

Before you install the plugin, you **MUST** verify that your processor offers AES hardware acceleration by calling the following command:

```
cat /proc/cpuinfo | grep aes
```

If the command returns a list of properties, including the “aes” string, your CPU provides the AES hardware acceleration. If the command returns nothing, AES hardware acceleration is not supported.

You **MUST NOT** install the following plugin if your CPU does not support AES-NI because the kernel will panic during boot.

The support for the AES-NI instruction set during boot time is enabled by installing the following plugin (make sure that the version of the plugin RPM matches the version of the installed RPMs!):

```
# install the dracut-fips-aesni package
yum install dracut-fips-aesni-*
# recreate the initramfs image
dracut -f
```

The changes come into effect during the next reboot.

10.2 User Guidance

CTR and RFC 3686 mode must only be used for IPsec. It must not be used otherwise.

When using the Module, the user shall utilize the Oracle Linux UEK provided memory allocation mechanisms. In addition, the user shall not use the function `copy_to_user()` on any portion of the data structures used to communicate with the Oracle Linux UEK 6. Only the cryptographic mechanisms provided with the Oracle Linux UEK are considered for use.

10.2.1 AES-XTS Usage

As specified in SP800-38E, the AES algorithm in XTS mode is designed for the cryptographic protection of data on storage devices. Thus, it can only be used for the disk encryption functionality offered by dm-crypt (i.e., the hard disk encryption scheme). For dm-crypt, the length of a single data unit encrypted with AES XTS mode is at most 65,536 bytes (64KiB of data), which does not exceed 220 AES blocks (16MiB of data).

To meet the requirement stated in IG A.9, the module implements a check to ensure that the two AES keys used in AES XTS mode are not identical.

Note: AES-XTS shall be used with 128 and 256-bit keys only. AES-XTS with 192-bit keys is not an Approved service.

10.2.2 AES-GCM Usage

The GCM with internal IV generation in FIPS mode is in compliance with RFC4106 and shall only be used in conjunction with the IPsec stack of the kernel to be compliant with IG A.5. Any other usage of GCM will be considered non-Approved.

In case the module's power is lost and then restored, the key used for the AES-GCM encryption or decryption shall be redistributed.

When a GCM IV is used for decryption, the responsibility for the IV generation lies with the party that performs the AES-GCM encryption therefore there is no restriction on the IV generation.

The `nonce_explicit` part of the IV does not exhaust the maximum number of possible values for a given session key. The design of the IPsec protocol ensures that the `nonce_explicit`, or counter portion, of the IV will not exhaust all of its possible values.

10.2.3 Triple-DES Usage

According to IG A.13, the same Triple-DES key shall not be used to encrypt more than 2^{16} 64-bit blocks of data. It is the user's responsibility to make sure that the module complies with this requirement and that the module does not exceed this limit.

10.3 Handling Self-Test Errors

Self test failure within the UEK 6 Module or the dm-crypt kernel component will panic the kernel and the operating system will not load.

Recover from this error by trying to reboot the system. If the failure continues, you must reinstall the software package being sure to follow all instructions. If you downloaded the software verify the package hash to confirm a proper download. Contact Oracle if these steps do not resolve the problem.

The UEK 6 Module performs a power-on self test that includes an integrity check and known answer tests for the available cryptographic algorithms.

The kernel dumps self-test success and failure messages into the kernel message ring buffer. Post boot, the messages are moved to `/var/log/messages`. Use **dmesg** to read the contents of the kernel ring buffer. The format of the ringbuffer (**dmesg**) output is:

```
alg: self-tests for %s (%s) passed
```

Typical messages are similar to "alg: self-tests for hmac(sha1-generic) (hmac(sha1)) passed" for each algorithm/sub-algorithm type.



11. Mitigation of Other Attacks

The module does not claim to mitigate against any attacks.

Acronyms, Terms and Abbreviations

| Term | Definition |
|-------|--|
| AES | Advanced Encryption Standard |
| CAVP | Cryptographic Algorithm Validation Program |
| CMVP | Cryptographic Module Validation Program |
| CCCS | Canadian Centre for Cyber Security |
| CSP | Critical Security Parameter |
| COP2 | Coprocessor 2 |
| DH | Diffie-Hellman |
| DHE | Diffie-Hellman Ephemeral |
| DRBG | Deterministic Random Bit Generator |
| ECDH | Elliptic Curve Diffie-Hellman |
| ECDSA | Elliptic Curve Digital Signature Algorithm |
| EDC | Error Detection Code |
| HMAC | (Keyed) Hash Message Authentication Code |
| IKE | Internet Key Exchange |
| KAT | Known Answer Test |
| KDF | Key Derivation Function |
| LRNG | Linux Random Number Generator |
| NIST | National Institute of Standards and Technology |
| PAA | Processor Algorithm Acceleration |
| PBKDF | Password Based Key Derivation Function |
| POST | Power On Self-Test |
| PR | Prediction Resistance |
| PSS | Probabilistic Signature Scheme |
| PUB | Publication |
| SHA | Secure Hash Algorithm |

Table 12: Acronyms

References

The FIPS 140-2 standard, and information on the CMVP, can be found at <http://csrc.nist.gov/groups/STM/cmvp/index.html>. More information describing the module can be found on the Oracle web site at <https://www.oracle.com/linux/>

This Security Policy contains non-proprietary information. All other documentation submitted for FIPS 140-2 conformance testing and validation is “Oracle - Proprietary” and is releasable only under appropriate non-disclosure agreements.

| Document | Author | Title |
|------------------------|------------------|---|
| FIPS PUB 140-2 | NIST | FIPS PUB 140-2: Security Requirements for Cryptographic Modules |
| FIPS IG | NIST | Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program |
| FIPS PUB 140-2 Annex A | NIST | FIPS 140-2 Annex A: Approved Security Functions |
| FIPS PUB 140-2 Annex B | NIST | FIPS 140-2 Annex B: Approved Protection Profiles |
| FIPS PUB 140-2 Annex C | NIST | FIPS 140-2 Annex C: Approved Random Number Generators |
| FIPS PUB 140-2 Annex D | NIST | FIPS 140-2 Annex D: Approved Key Establishment Techniques |
| DTR for FIPS PUB 140-2 | NIST | Derived Test Requirements (DTR) for FIPS PUB 140-2, Security Requirements for Cryptographic Modules |
| NIST SP 800-67 | NIST | Recommendation for the Triple Data Encryption Algorithm TDEA Block Cypher |
| FIPS PUB 197 | NIST | Advanced Encryption Standard |
| FIPS PUB 198-1 | NIST | The Keyed Hash Message Authentication Code (HMAC) |
| FIPS PUB 186-4 | NIST | Digital Signature Standard (DSS) |
| FIPS PUB 180-4 | NIST | Secure Hash Standard (SHS) |
| NIST SP 800-131A | NIST | Recommendation for the Transitioning of Cryptographic Algorithms and Key Sizes |
| PKCS#1 | RSA Laboratories | PKCS#1 v2.1: RSA Cryptographic Standard |

Table 13: References