


# The OpenSSL Corporation

# OpenSSL FIPS Provider

FIPS 140-3 Non-Proprietary Security Policy

**Document Version 1.2**
**August 21, 2026**

Prepared by:




www.lightshipsec.com

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# Table of Contents

**1 General** ..... 5
1.1 Overview ..... 5
1.2 Security Levels ..... 5
1.3 Additional Information ..... 6
**2 Cryptographic Module Specification** ..... 7
2.1 Description ..... 7
2.2 Tested and Vendor Affirmed Module Version and Identification ..... 8
2.3 Excluded Components ..... 10
2.4 Modes of Operation ..... 10
2.5 Algorithms ..... 11
2.6 Security Function Implementations ..... 20
2.7 Algorithm Specific Information ..... 39
2.8 RBG and Entropy ..... 42
2.9 Key Generation ..... 43
2.10 Key Establishment ..... 43
2.11 Industry Protocols ..... 43
**3 Cryptographic Module Interfaces** ..... 44
3.1 Ports and Interfaces ..... 44
**4 Roles, Services, and Authentication** ..... 45
4.1 Authentication Methods ..... 45
4.2 Roles ..... 45
4.3 Approved Services ..... 45
4.4 Non-Approved Services ..... 55
4.5 External Software/Firmware Loaded ..... 56
4.6 Bypass Actions and Status ..... 56
4.7 Cryptographic Output Actions and Status ..... 56
**5 Software/Firmware Security** ..... 57
5.1 Integrity Techniques ..... 57
5.2 Initiate on Demand ..... 57
5.3 Open-Source Parameters ..... 57
**6 Operational Environment** ..... 58
6.1 Operational Environment Type and Requirements ..... 58
6.2 Configuration Settings and Restrictions ..... 58
**7 Physical Security** ..... 59
**8 Non-Invasive Security** ..... 60

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

**9 Sensitive Security Parameters Management** ..... 61
9.1 Storage Areas ..... 61
9.2 SSP Input-Output Methods ..... 61
9.3 SSP Zeroization Methods ..... 61
9.4 SSPs ..... 62
**10 Self-Tests** ..... 81
10.1 Pre-Operational Self-Tests ..... 81
10.2 Conditional Self-Tests ..... 81
10.3 Periodic Self-Test Information ..... 85
10.4 Error States ..... 89
10.5 Operator Initiation of Self-Tests ..... 89
**11 Life-Cycle Assurance** ..... 90
11.1 Installation, Initialization, and Startup Procedures ..... 90
11.2 Administrator Guidance ..... 91
11.3 Non-Administrator Guidance ..... 91
11.4 Design and Rules ..... 91
11.5 Maintenance Requirements ..... 91
11.6 End of Life ..... 92
**12 Mitigation of Other Attacks** ..... 93
12.1 Attack List ..... 93

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

## List of Tables

Table 1: Security Levels ..... 6
Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets) ..... 8
Table 3: Tested Operational Environments - Software, Firmware, Hybrid ..... 10
Table 4: Modes List and Description ..... 10
Table 5: Approved Algorithms ..... 18
Table 6: Vendor-Affirmed Algorithms ..... 19
Table 7: Non-Approved, Allowed Algorithms ..... 19
Table 8: Non-Approved, Allowed Algorithms with No Security Claimed ..... 19
Table 9: Non-Approved, Not Allowed Algorithms ..... 20
Table 10: Security Function Implementations ..... 38
Table 11: Ports and Interfaces ..... 44
Table 12: Roles ..... 45
Table 13: Approved Services ..... 53
Table 14: Non-Approved Services ..... 56
Table 15: Storage Areas ..... 61
Table 16: SSP Input-Output Methods ..... 61
Table 17: SSP Zeroization Methods ..... 61
Table 18: SSP Table 1 ..... 74
Table 19: SSP Table 2 ..... 80
Table 20: Pre-Operational Self-Tests ..... 81
Table 21: Conditional Self-Tests ..... 85
Table 22: Pre-Operational Periodic Information ..... 86
Table 23: Conditional Periodic Information ..... 89
Table 24: Error States ..... 89

## List of Figures

Figure 1: OpenSSL FIPS Provider Block Diagram ..... 8

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 1 General

## 1.1 Overview

### Introduction

Federal Information Processing Standards Publication 140-3 — Security Requirements for Cryptographic Modules specifies requirements for cryptographic modules to be deployed in a Sensitive but Unclassified environment. The National Institute of Standards and Technology (NIST) and Canadian Centre for Cyber Security (CCCS) Cryptographic Module Validation Program (CMVP) run the FIPS 140-3 program. The NVLAP accredits independent testing labs to perform FIPS 140-3 testing; the CMVP validates modules meeting FIPS 140-3 validation. Validated is the term given to a module that is documented and tested against the FIPS 140-3 criteria.

More information is available on the CMVP website at:

https://csrc.nist.gov/projects/cryptographic-module-validation-program

### About this Document

This document describes the non-proprietary Security Policy for the OpenSSL FIPS Provider cryptographic module (hereafter referred to as "the Module") from The OpenSSL Corporation. It contains specifications of the security rules under which the Module operates, including the security rules derived from the requirements of the FIPS 140-3 standard.

The OpenSSL Corporation may also be referred to as "OpenSSL" in this document.

The following trademarks are referenced within this Security Policy:

- Linux®: Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.
- FreeBSD®: FreeBSD is a registered trademark of FreeBSD Foundation.
- Microsoft Windows®: Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Copyright Notice

Copyright © 2025 The OpenSSL Corporation Authors.

## 1.2 Security Levels

The Module meets FIPS 140-3 overall Level 1 requirements, with security levels as follows:

The Module satisfies the overall requirements for FIPS 140-3 Security Level 1. The applicable security levels for each FIPS 140-3 requirement area are summarized in the following table.

| Section | Title | Security Level |
|---|---|---|
| 1 | General | 1 |
| 2 | Cryptographic module specification | 1 |
| 3 | Cryptographic module interfaces | 1 |
| 4 | Roles, services, and authentication | 1 |
| 5 | Software/Firmware security | 1 |
| 6 | Operational environment | 1 |
| 7 | Physical security | N/A |
| 8 | Non-invasive security | N/A |
| 9 | Sensitive security parameter management | 1 |
| 10 | Self-tests | 1 |

| Section | Title | Security Level |
|---|---|---|
| 11 | Life-cycle assurance | 3 |
| 12 | Mitigation of other attacks | 1 |
| | Overall Level | 1 |

Table 1: Security Levels

## 1.3 Additional Information

In accordance with AS02.05, [ISO19790] §7.7 *Physical Security* is optional and does not apply to the Module.

In accordance with current CMVP policy, [ISO19790] §7.8 *Non-Invasive Security* is not applicable.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 2 Cryptographic Module Specification

## 2.1 Description

### Purpose and Use:

The Module is a cryptographic software library providing a C-language application program interface (API) for use by applications that require cryptographic functionality and is designated as a software module with a multi-chip standalone embodiment based on the descriptions of [ISO19790] AS02.03. The Module is intended for use by US and Canadian Federal agencies and other markets that require FIPS 140-3 validated cryptographic functionality.

The Module's formal name and version are "OpenSSL FIPS Provider" and "3.5.9", respectively.

The Module design corresponds to the Module security rules. Security rules enforced by the Module are described in the appropriate context of this document.

**Module Type**: Software

**Module Embodiment**: Multi-Chip Standalone

### Cryptographic Boundary:

Figure 1 depicts the Module operational environment, with the cryptographic boundary highlighted in red inclusive of all Module entry points (API calls). The Module is defined as a Software module per [ISO19790] AS02.03. The cryptographic boundary of the Module is the FIPS Provider, a dynamically loadable library as well as the fipsmodule.cnf file which contains the integrity reference value. The Module performs no communication other than with the calling application via APIs that invoke the Module.

The pre-operational approved integrity test is performed over all components within the cryptographic boundary.

### Tested Operational Environment's Physical Perimeter (TOEPP):

The Tested Operational Environment's Physical Perimeter (TOEPP) is the General-Purpose Computer.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

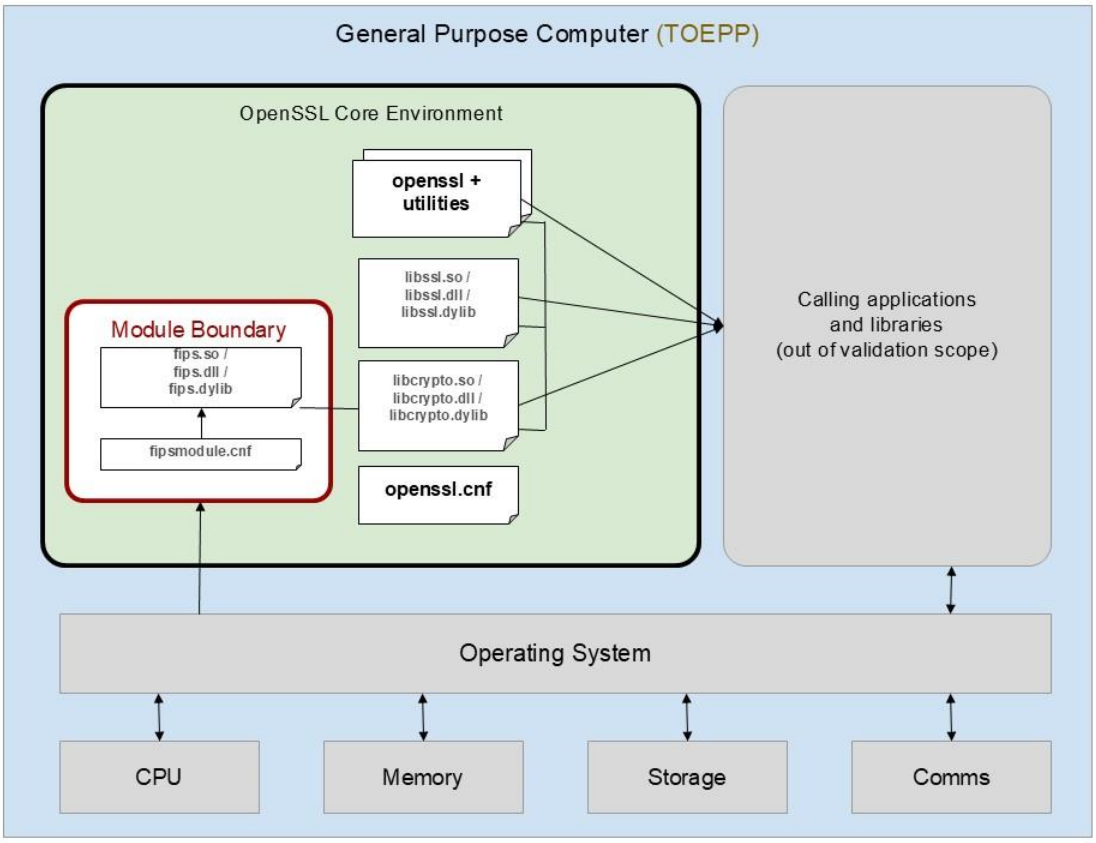



Figure 1: OpenSSL FIPS Provider Block Diagram

## 2.2 Tested and Vendor Affirmed Module Version and Identification

**Tested Module Identification – Hardware:**

N/A for this module.

**Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets):**

| Package or File Name | Software/ Firmware Version | Features | Integrity Test |
|---|---|---|---|
| fips.so | 3.5.9 | fips.so for Linux/FreeBSD platforms | HMAC-SHA2-256 |
| fips.dll | 3.5.9 | fips.dll for Windows platforms | HMAC-SHA2-256 |
| fips.dylib | 3.5.9 | fips.dylib for Mac platforms | HMAC-SHA2-256 |

Table 2: Tested Module Identification – Software, Firmware, Hybrid (Executable Code Sets)

**Tested Module Identification – Hybrid Disjoint Hardware:**

N/A for this module.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

**Tested Operational Environments - Software, Firmware, Hybrid:**

| Operating System | Hardware Platform | Processors | PAA/PAI | Hypervisor or Host OS | Version(s) |
|---|---|---|---|---|---|
| Ubuntu Linux 22.04.1 Server | Dell Inspiron 7573 | Intel i7-8550U | No | N/A | 3.5.9 |
| Ubuntu Linux 22.04.1 Server | Dell Inspiron 7573 | Intel i7-8550U | Yes | N/A | 3.5.9 |
| Ubuntu Linux 24.04.2 LTS (Noble Numbat) | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | No | N/A | 3.5.9 |
| Ubuntu Linux 24.04.2 LTS (Noble Numbat) | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | Yes | N/A | 3.5.9 |
| Debian 11.5 | Dell Inspiron 7573 | Intel i7-8550U | No | N/A | 3.5.9 |
| Debian 11.5 | Dell Inspiron 7573 | Intel i7-8550U | Yes | N/A | 3.5.9 |
| Debian 12 (Bookworm) | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | No | N/A | 3.5.9 |
| Debian 12 (Bookworm) | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | Yes | N/A | 3.5.9 |
| FreeBSD 13.1 | Dell Inspiron 7591 2 in 1 | Intel i7-10510U | No | N/A | 3.5.9 |
| FreeBSD 13.1 | Dell Inspiron 7591 2 in 1 | Intel i7-10510U | Yes | N/A | 3.5.9 |
| FreeBSD 14.3 | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | No | N/A | 3.5.9 |
| FreeBSD 14.3 | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | Yes | N/A | 3.5.9 |
| Windows 10 Pro | Dell Inspiron 7591 2 in 1 | Intel i7-10510U | No | N/A | 3.5.9 |
| Windows 10 Pro | Dell Inspiron 7591 2 in 1 | Intel i7-10510U | Yes | N/A | 3.5.9 |
| Microsoft Windows Server 2025 Standard 64-bit | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | No | N/A | 3.5.9 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Operating System | Hardware Platform | Processors | PAA/PAI | Hypervisor or Host OS | Version(s) |
|---|---|---|---|---|---|
| Microsoft Windows Server 2025 Standard 64-bit | Supermicro AS-1015A-MT | AMD EPYC Raphael 4124P 4-Core Processor | Yes | N/A | 3.5.9 |
| macOS 15.5 | Apple M4 Mac Mini | Apple M4 10-core CPU | No | N/A | 3.5.9 |
| macOS 15.5 | Apple M4 Mac Mini | Apple M4 10-core CPU | Yes | N/A | 3.5.9 |

Table 3: Tested Operational Environments - Software, Firmware, Hybrid

**Vendor-Affirmed Operational Environments - Software, Firmware, Hybrid:**

N/A for this module.

## 2.3 Excluded Components

No components are excluded from [FIPS140-3] requirements.

## 2.4 Modes of Operation

**Modes List and Description:**

| Mode Name | Description | Types of Services | Status Indicator |
|---|---|---|---|
| Approved mode | The module must be installed and configured per instructions provided in Section 11 of this document and the module is in the Approved mode by default as a result. The installation of the Module as described in Section 11 results in the settings described below this table, which are required for operation in the Approved mode | Approved | fips-indicator = 1 |
| Non-Approved mode | The module is in the Approved mode of operation by default. Use of the non-Approved Algorithms Not Allowed in the Approved Mode will place the module in the non-approved mode of operation. | Non-Approved | fips-indicator = 0 |

Table 4: Modes List and Description

The Module supports an Approved mode and a non-Approved mode of operation. The module automatically switches between the approved and non-approved modes depending on the services requested by the operator.

The inherent properties of the Module are:

1. Manual key entry is not supported.
2. Data output is inhibited during self-tests, zeroisation, SSP generation and error states.
3. The Module does not perform any cryptographic function if any self-test has failed.

The conditions for using the Module in the [FIPS140-3] Approved mode of operation are:

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

1. Installation of the Module as described in Section 11 results in the settings described below, which are required for operation in the Approved mode:
    a. security-checks = 1

    Enforce minimum key strengths and approved curve names.

    b. conditional-errors = 1

    Enforce the Module entering the error state on conditional test errors such as PCT failure.

    c. drbg-no-trunc-md=1

    Disallow use of truncated digests with HASH and HMAC DRBGs (IG D.R)

    d. tls1-prf-ems-check=1

    Enforce Extended Master Secret (EMS) use with TLS 1.2 (IG D.Q)

Changing these settings will result in a non-compliant module

2. The Module is a cryptographic library used by a calling application. The calling application is responsible for:
    a. Use of the primitives in the correct sequence.
    b. Use of keys in accordance with [SP800-140Dr2] (as the keys used by the Module for cryptographic purposes are provided over the call stack by the calling application).
    c. Use of a [SP800-90B] compliant entropy source. Entropy is supplied to the Module via callback functions. The callback functions return an error if the minimum entropy strength cannot be met.
    d. Indicator callbacks. The call to registered application-level functions to inform the application of various errors that change the operational mode of the module should be made when various validation/functional errors are encountered.
    e. The self-test callback informs the application of self-test results.

**Mode Change Instructions and Status:**

Use of the Approved algorithms and Non-Approved Algorithms Allowed in the Approved Mode will ensure operation of the module in the Approved mode of operation. Use of the non-Approved Algorithms Not Allowed in the Approved Mode will place the module in the non-approved mode of operation.

**Degraded Mode Description:**

The module does not support a degraded mode of operation.

## 2.5 Algorithms

**Approved Algorithms:**

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| AES-CBC | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-CBC-CS1 | A9077 | Direction - decrypt, encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-CBC-CS2 | A9077 | Direction - decrypt, encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-CBC-CS3 | A9077 | Direction - decrypt, encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| AES-CCM | A9077 | Key Length - 128, 192, 256 | SP 800-38C |
| AES-CFB1 | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-CFB128 | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-CFB8 | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-CMAC | A9077 | Direction - Generation, Verification<br>Key Length - 128, 192, 256 | SP 800-38B |
| AES-CTR | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-ECB | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-GCM | A9077 | Direction - Decrypt, Encrypt<br>IV Generation - External, Internal<br>Key Length - 128, 192, 256<br>IV Generation Mode - 8.2.1 | SP 800-38D |
| AES-GMAC | A9077 | Direction - Decrypt, Encrypt<br>IV Generation - External, Internal<br>IV Generation Mode - 8.2.1, 8.2.2<br>Key Length - 128, 192, 256 | SP 800-38D |
| AES-KW | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38F |
| AES-KWP | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38F |
| AES-OFB | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 192, 256 | SP 800-38A |
| AES-XTS Testing Revision 2.0 | A9077 | Direction - Decrypt, Encrypt<br>Key Length - 128, 256 | SP 800-38E |
| Counter DRBG | A9077 | Prediction Resistance - Yes<br>Mode - AES-128, AES-192, AES-256<br>Derivation Function Enabled - Yes | SP 800-90A Rev. 1 |
| DSA PQGVer (FIPS186-4) | A9077 | L - 1024, 2048, 3072<br>N - 160, 224, 256<br>Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256 | FIPS 186-4 |
| DSA SigVer (FIPS186-4) | A9077 | L - 1024, 2048, 3072<br>N - 160, 224, 256<br>Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256 | FIPS 186-4 |
| ECDSA KeyGen (FIPS186-5) | A9077 | Curve - B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521<br>Secret Generation Mode - testing candidates | FIPS 186-5 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| ECDSA KeyVer (FIPS186-5) | A9077 | Curve - B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521 | FIPS 186-5 |
| ECDSA SigGen (FIPS186-5) | A9077 | Curve - B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521<br>Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512<br>Component - Yes | FIPS 186-5 |
| ECDSA SigVer (FIPS186-5) | A9077 | Curve - B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521<br>Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512 | FIPS 186-5 |
| EDDSA KeyGen | A9077 | Curve - ED-25519, ED-448 | FIPS 186-5 |
| EDDSA KeyVer | A9077 | Curve - ED-25519, ED-448 | FIPS 186-5 |
| EDDSA SigGen | A9077 | Curve - ED-25519, ED-448<br>PreHash - Yes<br>Pure - Yes | FIPS 186-5 |
| EDDSA SigVer | A9077 | Curve - ED-25519, ED-448<br>PreHash - Yes<br>Pure - Yes | FIPS 186-5 |
| Hash DRBG | A9077 | Prediction Resistance - Yes<br>Mode - SHA-1, SHA2-256, SHA2-512, SHA3-256, SHA3-512 | SP 800-90A Rev. 1 |
| HMAC DRBG | A9077 | Prediction Resistance - Yes<br>Mode - SHA-1, SHA2-256, SHA2-512, SHA3-256, SHA3-512 | SP 800-90A Rev. 1 |
| HMAC-SHA-1 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA2-224 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA2-256 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA2-384 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA2-512 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA2-512/224 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA2-512/256 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA3-224 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA3-256 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| HMAC-SHA3-384 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| HMAC-SHA3-512 | A9077 | Key Length - Key Length: 112-524288 Increment 8 | FIPS 198-1 |
| KAS-ECC CDH-Component SP800-56Ar3 (CVL) | A9077 | Curve - B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521 | SP 800-56A Rev. 3 |
| KAS-ECC-SSC Sp800-56Ar3 | A9077 | Domain Parameter Generation Methods - B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521<br>Scheme -<br>ephemeralUnified -<br>KAS Role - initiator, responder | SP 800-56A Rev. 3 |
| KAS-FFC-SSC Sp800-56Ar3 | A9077 | Domain Parameter Generation Methods - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192<br>Scheme -<br>dhEphem -<br>KAS Role - initiator, responder | SP 800-56A Rev. 3 |
| KAS-IFC-SSC | A9077 | Modulo - 2048, 3072, 4096, 6144, 8192<br>Key Generation Methods - rsakpg1-basic, rsakpg1-crt, rsakpg1-prime-factor, rsakpg2-basic, rsakpg2-crt, rsakpg2-prime-factor<br>Scheme -<br>KAS1 -<br>KAS Role - initiator, responder<br>KAS2 -<br>KAS Role - initiator, responder | SP 800-56A Rev. 3 |
| KDA HKDF SP800-56Cr2 | A9077 | Derived Key Length - 2048<br>Shared Secret Length - Shared Secret Length: 224-65536 Increment 8<br>HMAC Algorithm - SHA1, SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512 | SP 800-56C Rev. 2 |
| KDA OneStep SP800-56Cr2 | A9077 | Auxiliary Function Methods -<br>MAC Salting Methods - default, random<br>Derived Key Length - 2048<br>Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 | SP 800-56C Rev. 2 |
| KDA TwoStep SP800-56Cr2 | A9077 | MAC Salting Methods - default<br>KDF Mode - feedback<br>Derived Key Length - 2048<br>Shared Secret Length - Shared Secret Length: 224-8192 Increment 8 | SP 800-56C Rev. 2 |
| KDF ANS 9.42 (CVL) | A9077 | KDF Type - DER<br>Hash Algorithm - SHA-1, SHA2-224, SHA2-256, | SP 800-135 Rev. 1 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| | | SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512<br>Key Data Length - Key Data Length: 8-4096 Increment 8 | |
| KDF ANS 9.63 (CVL) | A9077 | Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512<br>Key Data Length - Key Data Length: 128, 4096 | SP 800-135 Rev. 1 |
| KDF KMAC Sp800-108r1 | A9077 | Derived Key Length - Derived Key Length: 112-4096 Increment 8 | SP 800-108 Rev. 1 |
| KDF SP800-108 | A9077 | KDF Mode - Counter, Feedback<br>Supported Lengths - Supported Lengths: 8, 72, 128, 776, 3456, 4096 | SP 800-108 Rev. 1 |
| KDF SSH (CVL) | A9077 | Cipher - AES-128, AES-192, AES-256, TDES<br>Hash Algorithm - SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512 | SP 800-135 Rev. 1 |
| KMAC-128 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Key Data Length - Key Data Length: 128-1024 Increment 8 | SP 800-185 |
| KMAC-256 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Key Data Length - Key Data Length: 128-1024 Increment 8 | SP 800-185 |
| KTS-IFC | A9077 | Modulo - 2048, 3072, 4096, 6144<br>Key Generation Methods - rsakpg1-basic, rsakpg1-crt, rsakpg1-prime-factor, rsakpg2-basic, rsakpg2-crt, rsakpg2-prime-factor<br>Scheme -<br>KTS-OAEP-basic -<br>KAS Role - initiator, responder<br>Key Transport Method -<br>Key Length - 1024 | SP 800-56B Rev. 2 |
| ML-DSA KeyGen | A9077 | Parameter Sets - ML-DSA-44, ML-DSA-65, ML-DSA-87 | FIPS 204 |
| ML-DSA SigGen | A9077 | Signature Interfaces - external, internal<br>Pre Hash - pure<br>Deterministic - No, Yes<br>External Mu - No, Yes<br>Parameter Sets - ML-DSA-44, ML-DSA-65, ML-DSA-87<br>Message Length - Message Length: 8-65536 Increment 8<br>Context Length - Context Length: 0-2040 Increment 8 | FIPS 204 |
| ML-DSA SigVer | A9077 | Signature Interfaces - external, internal<br>Pre Hash - pure | FIPS 204 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| | | External Mu - No, Yes<br>Parameter Sets - ML-DSA-44, ML-DSA-65, ML-DSA-87<br>Message Length - Message Length: 8-65536 Increment 8<br>Context Length - Context Length: 0-2040 Increment 8 | |
| ML-KEM EncapDecap | A9077 | Parameter Sets - ML-KEM-1024, ML-KEM-512, ML-KEM-768<br>Functions - Decapsulation, Decapsulation Key Check, Encapsulation, Encapsulation Key Check | FIPS 203 |
| ML-KEM KeyGen | A9077 | Parameter Sets - ML-KEM-1024, ML-KEM-512, ML-KEM-768 | FIPS 203 |
| PBKDF | A9077 | Iteration Count - Iteration Count: 1-10000 Increment 1<br>Password Length - Password Length: 8-128 Increment 8 | SP 800-132 |
| RSA KeyGen (FIPS186-5) | A9077 | Key Generation Mode - probableWithProbableAux<br>Modulo - 2048, 3072, 4096, 8192<br>Primality Tests - 2powSecStr<br>Private Key Format - standard | FIPS 186-5 |
| RSA SigGen (FIPS186-5) | A9077 | Hash Pair -<br>Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512<br>Modulo - 2048, 3072, 4096<br>Signature Type - pkcs1v1.5, pss | FIPS 186-5 |
| RSA Signature Primitive (CVL) | A9077 | Private Key Format - crt | FIPS 186-5 |
| RSA SigVer (FIPS186-4) | A9077 | Signature Type - ANSI X9.31<br>Modulo - 1024, 2048, 3072, 4096<br>Hash Pair -<br>Hash Algorithm - SHA-1, SHA2-256, SHA2-384, SHA2-512 | FIPS 186-4 |
| RSA SigVer (FIPS186-5) | A9077 | Hash Pair -<br>Hash Algorithm - SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512<br>Modulo - 2048, 3072, 4096<br>Signature Type - pkcs1v1.5, pss | FIPS 186-5 |
| Safe Primes Key Generation | A9077 | Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192 | SP 800-56A Rev. 3 |
| Safe Primes Key Verification | A9077 | Safe Prime Groups - ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192 | SP 800-56A Rev. 3 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| SHA-1 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA2-224 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA2-256 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA2-384 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA2-512 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA2-512/224 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA2-512/256 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 180-4 |
| SHA3-224 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 202 |
| SHA3-256 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 202 |
| SHA3-384 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 202 |
| SHA3-512 | A9077 | Message Length - Message Length: 0-65536 Increment 8<br>Large Message Sizes - 1, 2, 4, 8 | FIPS 202 |
| SHAKE-128 | A9077 | Output Length - Output Length: 16-65536 Increment 8 | FIPS 202 |
| SHAKE-256 | A9077 | Output Length - Output Length: 16-65536 Increment 8 | FIPS 202 |
| SLH-DSA KeyGen | A9077 | Parameter Sets - SLH-DSA-SHA2-128f, SLH-DSA-SHA2-128s, SLH-DSA-SHA2-192f, SLH-DSA-SHA2-192s, SLH-DSA-SHA2-256f, SLH-DSA-SHA2-256s, SLH-DSA-SHAKE-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-256f, SLH-DSA-SHAKE-256s | FIPS 205 |
| SLH-DSA SigGen | A9077 | Deterministic - No, Yes<br>Signature Interfaces - external, internal<br>Pre Hash - pure<br>Parameter Sets - SLH-DSA-SHA2-128f, SLH-DSA- | FIPS 205 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm | CAVP Cert | Properties | Reference |
|---|---|---|---|
| | | SHA2-128s, SLH-DSA-SHA2-192f, SLH-DSA-SHA2-192s, SLH-DSA-SHA2-256f, SLH-DSA-SHA2-256s, SLH-DSA-SHAKE-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-256f, SLH-DSA-SHAKE-256s<br>Message Length - Message Length: 8-65536 Increment 8<br>Context Length - Context Length: 0-2040 Increment 8 | |
| SLH-DSA SigVer | A9077 | Signature Interfaces - external, internal<br>Pre Hash - pure<br>Parameter Sets - SLH-DSA-SHA2-128f, SLH-DSA-SHA2-128s, SLH-DSA-SHA2-192f, SLH-DSA-SHA2-192s, SLH-DSA-SHA2-256f, SLH-DSA-SHA2-256s, SLH-DSA-SHAKE-128f, SLH-DSA-SHAKE-128s, SLH-DSA-SHAKE-192f, SLH-DSA-SHAKE-192s, SLH-DSA-SHAKE-256f, SLH-DSA-SHAKE-256s<br>Message Length - Message Length: 8-65536 Increment 8<br>Context Length - Context Length: 0-2040 Increment 8 | FIPS 205 |
| TDES-CBC | A9077 | Direction - Decrypt | SP 800-67 Rev. 2 |
| TDES-ECB | A9077 | Direction - Decrypt | SP 800-67 Rev. 2 |
| TLS v1.2 KDF RFC7627 (CVL) | A9077 | Hash Algorithm - SHA2-256, SHA2-384, SHA2-512 | SP 800-135 Rev. 1 |
| TLS v1.3 KDF (CVL) | A9077 | HMAC Algorithm - SHA2-256, SHA2-384<br>KDF Running Modes - DHE, PSK, PSK-DHE | SP 800-135 Rev. 1 |

Table 5: Approved Algorithms

The Module implements the Approved cryptographic functions listed in Table 5.

**Vendor-Affirmed Algorithms:**

| Name | Properties | Implementation | Reference |
|---|---|---|---|
| DSA PQGVer [FIPS186-4] | Key Size, Key Strength:L = 1024/N = 160 (s < 112), L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128)<br>Mode/Method:PQGVer using SHA3 | OpenSSL FIPS Provider 3.5 | Vendor affirmed per IG C.C and IG C.B Resolution (bullet point #3) |
| DSA SigVer | Key Size, Key Strength:L = 1024/N = 160 (s < 112), L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = | OpenSSL FIPS Provider 3.5 | Vendor affirmed per IG C.C and IG C.B |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Properties | Implementation | Reference |
|---|---|---|---|
| [FIPS186-4] | 112), L = 3072/N = 256 (s = 128) Mode/Method:SigVer using SHA3 | | Resolution (bullet point #3) |
| CKG - Section 4 | Key Type:Symmetric and Asymmetric | N/A | NIST SP800-133r2 Section 4: Using the Output of a Random Generator, Example 1 |
| CKG - Section 6.3 | Key Type:Symmetric | N/A | NIST SP800-133r2, Section 6.3: Symmetric Keys Produced by Combining Multiple Keys and Other Data, Method 1 |

Table 6: Vendor-Affirmed Algorithms

**Non-Approved, Allowed Algorithms:**

| Name | Properties | Implementation | Reference |
|---|---|---|---|
| AES | AES (Any non-authenticated mode) (Cert. #A9077):Symmetric key unwrapping | OpenSSL FIPS Provider 3.5 | Per IG C.M Resolution 2 |

Table 7: Non-Approved, Allowed Algorithms

**Non-Approved, Allowed Algorithms with No Security Claimed:**

| Name | Caveat | Use and Function |
|---|---|---|
| X25519 | X25519: Must only be used in the context of an approved hybrid KEM (enforced by the module per IG D.S Scenario 2 requirements) | Implement the X25519MLKEM768 hybrid scheme |

Table 8: Non-Approved, Allowed Algorithms with No Security Claimed

**Non-Approved, Not Allowed Algorithms:**

| Name | Use and Function |
|---|---|
| Triple-DES | Provides 3-Key ECB and CBC mode, Encryption |
| X448 | Provides 224 bits of security, Key Agreement set as fips=no |
| X448MLKEM1024 | Provides 256 bits of security for the X448MLKEM1024 hybrid Key Agreement scheme set as fips=no |
| X25519 | Provides 128 bits of security, Key Agreement, set as fips=no |
| ECDSA SigVer Component | Provides between 80 and 256 bits for security, Curves: B-163, B-233, B-283, B-409, B-571, K-163, K-233, K-283, K-409, K-571, P-192, P-224, P-256, P-384, P-521, Digital Signature Verification |
| X963KDF | Usage of X963KDF with PRF SHA-1 |
| HKDF | Provides < 112 bits of security, Usage of HKDF with key length less than 112 bits |
| HMAC | Provides < 112 bits of security, Usage of HMAC with key length less than 112 bits for MAC generation |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Use and Function |
|---|---|
| Hash and HMAC DRBG | Usage of Hash and HMAC DRBGs with PRFs SHA2-224, SHA2-384, SHA2-512/224 and SHA2-512/256 |
| FIPS 186-4 DSA KeyGen/PQGen | DSA PQGGen/KeyGen [FIPS 186-4] (VA) : L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128) |
| FIPS 186-4 DSA Digital Signature | SigGen: using SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3 : L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128) |
| FIPS 186-4 RSA Digital Signature | SigGen ANSI X9.31: using SHA2-256, SHA2-384, SHA2-512 : k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152) |
| RSA SigVer Component | RSA-2048 SigVer Component PKCS#1 - Provides between 80 and 112 bits of security |
| AES GCM encryption with external IV | Encryption |
| ML-KEM encapsulation with user-provided randomness | Key Encapsulation |
| FIPS 186-2 RSA SigGen | Provides < 112 bits of security, RSA signature generation per FIPS 186-2 |

Table 9: Non-Approved, Not Allowed Algorithms

## 2.6 Security Function Implementations

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| Symmetric Encryption or Decryption | BC-Auth BC-UnAuth | Symmetric Encryption or Decryption | Key Length:128, 192 and 256 bits Key Length (XTS):128 and 256 bits Keying option (TDES):Option 1 | AES-CBC: (A9077) AES-CBC-CS1: (A9077) AES-CBC-CS2: (A9077) AES-CBC-CS3: (A9077) AES-CCM: (A9077) AES-CFB1: (A9077) AES-CFB128: (A9077) AES-CFB8: (A9077) AES-CTR: (A9077) AES-ECB: (A9077) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | | AES-GCM: (A9077) AES-OFB: (A9077) AES-XTS Testing Revision 2.0: (A9077) TDES-ECB: (A9077) TDES-CBC: (A9077) AES-KW: (A9077) AES-KWP: (A9077) |
| Message Digest | SHA XOF | Message Digest | SHA-1:(s = 160). Large Message Sizes 1, 2, 4, 8 gigabytes SHA2:SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA2-512/224 (s = 224), SHA2-512/256 (s = 256). Large Message Sizes 1, 2, 4, 8 gigabytes SHA3:SHA3-224 (s = 224), SHA3-256 (s = 256), SHA3-384 (s = 384), SHA3-512 (s = 512). See Note 1. Large Message Sizes 1, 2, 4, 8 gigabytes SHAKE:SHAKE-128 (s = 128), SHAKE-256 (s = 256). See Note 1. | SHA-1: (A9077) SHA2-224: (A9077) SHA2-256: (A9077) SHA2-384: (A9077) SHA2-512: (A9077) SHA2-512/224: (A9077) SHA2-512/256: (A9077) SHA3-224: (A9077) SHA3-256: (A9077) SHA3-384: (A9077) SHA3-512: (A9077) SHAKE-128: (A9077) SHAKE-256: (A9077) |
| Keyed Hash | MAC | Keyed Hash | HMAC-SHA-1 [FIPS198-1]:SHA-1 (s = 160) HMAC-SHA2 [FIPS198-1]:SHA2-224 (s = 224), SHA2- | HMAC-SHA-1: (A9077) HMAC-SHA2-224: (A9077) HMAC- |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | 256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA2-512/224 (s = 224), SHA2-512/256 (s = 256)<br>HMAC-SHA3 [FIPS198-1]:SHA3-224 (s = 224), SHA3-256 (s = 256), SHA3-384 (s = 384), SHA3-512 (s = 512)<br>KMAC:KMAC-128 (112 <= s <= 128), KMAC-256 (112 <= s <= 256). See Note 8<br>AES-CMAC:Key Length: 128, 192 and 256 bits<br>AES-GMAC:Key Length: 128, 192 and 256 bits | SHA2-256: (A9077)<br>HMAC-SHA2-384: (A9077)<br>HMAC-SHA2-512: (A9077)<br>HMAC-SHA2-512/224: (A9077)<br>HMAC-SHA2-512/256: (A9077)<br>HMAC-SHA3-224: (A9077)<br>HMAC-SHA3-256: (A9077)<br>HMAC-SHA3-384: (A9077)<br>HMAC-SHA3-512: (A9077)<br>AES-CMAC: (A9077)<br>AES-GMAC: (A9077)<br>KMAC-128: (A9077)<br>KMAC-256: (A9077) |
| RSA Digital Signature Generation and Verification | DigSig-SigGen<br>DigSig-SigVer | RSA Digital Signature Generation and Verification | SigGen type (PKCS 1.5) (FIPS 186-5):PKCS 1.5 tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512; k=2048 (s ~= 112), k=3072 (s ~= | RSA SigGen (FIPS186-5): (A9077)<br>RSA SigVer (FIPS186-5): (A9077)<br>RSA SigVer (FIPS186-4): (A9077) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | 128), k=4096 (s ~= 152)<br>SigGen type (PKCSPSS) (FIPS 186-5):PKCSPSS tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152)<br>SigVer type (ANSI X9.31):ANSI X9.31 tested with the listed moduli and the following hash algorithms: SHA-1*, SHA2-256, SHA2-384, SHA2-512; k=1024 (s <= 112), k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152)<br>SigVer type (PKCS 1.5) (FIPS 186-5):PKCS 1.5 tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152)<br>SigVer type (PKCSPSS) (FIPS 186-5):PKCSPSS | |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256, SHA3-224, SHA3-256, SHA3-384, SHA3-512; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152)<br>SigGen type (PKCS 1.5) (FIPS 186-4):PKCS 1.5 tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152)<br>SigGen type (PKCSPSS) (FIPS 186-4):PKCSPSS tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152)<br>SigVer type (PKCS 1.5) (FIPS 186-4):PKCS 1.5 tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, | |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | SHA2-512/256; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152) SigVer type (PKCSPSS) (FIPS 186-4):PKCSPSS tested with the listed moduli and the following hash algorithms: SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256; k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152) | |
| ECDSA Signature Generation and Signature Verification | DigSig-SigGen DigSig-SigVer | ECDSA Signature Generation and Signature Verification | SigGen (includes SigGen Component) (tested with SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512):B-233, K-233, P-224 (s ~= 112); B-283, K-283, P-256 (s ~= 128); B-409, K-409, P-384 (s ~= 192); B-571, K-571, P-521 (s ~= 256) SigVer (tested with SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA3-224, SHA3-256, SHA3-384, SHA3-512):B-233, K-233, P-224 (s ~= 112); B-283, K-283, P-256 (s ~= 128); B-409, K-409, P-384 (s ~= 192); B-571, K-571, P-521 (s ~= 256) | ECDSA SigGen (FIPS186-5): (A9077) ECDSA SigVer (FIPS186-5): (A9077) |
| EDDSA Signature Generation and Signature Verification | DigSig-SigGen DigSig-SigVer | EDDSA Signature Generation and Signature Verification | Ed25519 SigGen (tested with SHA2-512):Edwards25519 (s ~= 128) Ed25519 SigVer | EDDSA SigGen: (A9077) EDDSA |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | (tested with SHA2-512):Edwards25519 (s ~= 128)<br>Ed448 SigGen (tested with SHAKE256):Edwards448 (s ~= 224)<br>Ed448 SigVer (tested with SHAKE256):Edwards448 (s ~= 224) | SigVer: (A9077) |
| DSA Digital Signature Verification | DigSig-SigVer | DSA Digital Signature Verification | SigVer (tested with SHA-1, SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256);<br>SigVer using SHA3; no ACVP testing is available:L = 1024/N = 160 (s < 112), L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128) | DSA SigVer (FIPS186-4): (A9077)<br>DSA SigVer [FIPS186-4]: ()<br>Key Size, Key Strength: L = 1024/N = 160 (s < 112), L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128)<br>Mode/Method: SigVer using SHA3 |
| RSA Signature Primitive | DigSig-SigGen | Signature primitive | Private Key format:CRT<br>Public Exponent Mode:Fixed : k = 2048 | RSA Signature Primitive: (A9077) |
| Asymmetric Key Pair Generation | AsymKeyPair-KeyGen<br>AsymKeyPair-KeyVer<br>CKG | Generation of asymmetric key pairs | RSA KeyGen:k=2048 (s ~= 112), k=3072 (s ~= 128), k=4096 (s ~= 152), k=8192 (s ~= 208)<br>ECDSA KeyGen:Secret Generation Mode: Testing Candidates: B-233, K-233, P-224 | RSA KeyGen (FIPS186-5): (A9077)<br>ECDSA KeyGen (FIPS186-5): (A9077)<br>Safe Primes Key |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | (s ~= 112); B-283, K-283, P-256 (s ~= 128); B-409, K-409, P-384 (s ~= 192); B-571, K-571, P-521 (s ~= 256) Safe Primes Key Generation, Safe Primes Key Verification:ffdhe2048 (s = 112), ffdhe3072 (112 <= s <= 128), ffdhe4096 (112 <= s <= 152), ffdhe6144 (112 <= s <= 176), ffdhe8192 (112 <= s <= 200), MODP-2048 (s = 112), MODP-3072 (112 <= s <= 128), MODP-4096 (112 <= s <= 152), MODP-6144 (112 <= s <= 176), MODP-8192 (112 <= s <= 200) ECDSA KeyVer:B-233, K-233, P-224 (s ~= 112); B-283, K-283, P-256 (s ~= 128); B-409, K-409, P-384 (s ~= 192); B-571, K-571, P-521 (s ~= 256) DSA PQGVer [FIPS186-4], DSA PQGVer [FIPS186-4] (VA):L = 1024/N = 160 (s < 112), L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128) EDDSA KeyGen:Edwards25519 (s ~= 128), Edwards448 (s ~= 224) EDDSA KeyVer:Edwards25519 (s ~= 128), Edwards448 (s ~= 224) | Generation: (A9077) Safe Primes Key Verification: (A9077) ECDSA KeyVer (FIPS186-5): (A9077) DSA PQGVer (FIPS186-4): (A9077) DSA PQGVer [FIPS186-4]: () Key Size, Key Strength: L = 1024/N = 160 (s < 112), L = 2048/N = 224 (s = 112), L = 2048/N = 256 (s = 112), L = 3072/N = 256 (s = 128) Mode/Method: PQGVer using SHA3 EDDSA KeyGen: (A9077) EDDSA KeyVer: (A9077) CKG - Section 4: () Key Type: Symmetric and Asymmetric |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| Random Number Generation | DRBG | Random Number Generation - Hash_DRBG, CTR_DRBG and HMAC_DRBG | Counter DRBG [SP800-90Ar1]:AES-128 (s = 128), AES-192 (s = 192), AES-256 (s = 256)<br>Hash DRBG [SP800-90Ar1]:SHA-1 (s = 160), SHA2-256 (s = 256), SHA2-512 (s = 512), SHA3-256 (s = 256), SHA3-512 (s = 512)<br>HMAC DRBG [SP800-90Ar1]:SHA-1 (s = 160), SHA2-256 (s = 256), SHA2-512 (s = 512), SHA3-256 (s = 256), SHA3-512 (s = 512) | Counter DRBG: (A9077)<br>Hash DRBG: (A9077)<br>HMAC DRBG: (A9077) |
| Key Derivation | CKG<br>KAS-135KDF<br>KAS-56CKDF<br>KBKDF<br>PBKDF | Derive Keying Material | KDA HKDF:SHA-1 (s = 160), SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA2-512/224 (s = 224), SHA2-512/256 (s = 256), SHA3-224 (s = 224), SHA3-256 (s = 256), SHA3-384 (s = 384), SHA3-512 (s = 512)<br>KDA OneStep:SHA-1 (s = 160), SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA2-512/224 (s = 224), SHA2-512/256 (s = 256), SHA3-224 (s = 224), SHA3-256 (s = 256), SHA3-384 (s = 384), SHA3-512 (s = 512); HMAC-SHA-1 (s = 160), HMAC-SHA2-224 (s = 224), HMAC-SHA2-256 (s = 256), HMAC-SHA2-384 (s = 384), HMAC-SHA2-512 (s = 512), HMAC- | KDA HKDF SP800-56Cr2: (A9077)<br>KDA OneStep SP800-56Cr2: (A9077)<br>KDA TwoStep SP800-56Cr2: (A9077)<br>KDF ANS 9.42: (A9077)<br>KDF ANS 9.63: (A9077)<br>KDF KMAC Sp800-108r1: (A9077)<br>KDF SP800-108: (A9077)<br>KDF SSH: (A9077)<br>PBKDF: (A9077) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | SHA2-512/224 (s = 224), HMAC-SHA2-512/256 (s = 256), HMAC-SHA3-224 (s = 224), HMAC-SHA3-256 (s = 256), HMAC-SHA3-384 (s = 384), HMAC-SHA3-512 (s = 512); KMAC-128 (112 <= s <= 128), KMAC-256 (112 <= s <= 256) KDA TwoStep [SP800-56Cr2]:HMAC-SHA-1 (s = 160), HMAC-SHA2-224 (s = 224), HMAC-SHA2-256 (s = 256), HMAC-SHA2-384 (s = 384), HMAC-SHA2-512 (s = 512), HMAC-SHA2-512/224 (s = 224), HMAC-SHA2-512/256 (s = 256), HMAC-SHA3-224 (s = 224), HMAC-SHA3-256 (s = 256), HMAC-SHA3-384 (s = 384), HMAC-SHA3-512 (s = 512) KDF ANS 9.42 [SP800-135r1]:SHA-1 (s = 160), SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA2-512/224 (s = 224), SHA2-512/256 (s = 256), SHA3-224 (s = 224), SHA3-256 (s = 256), SHA3-384 (s = 384), SHA3-512 (s = 512) KDF ANS 9.63 [SP800-135r1]:SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA3-224 (s = 224), SHA3- | TLS v1.2 KDF RFC7627: (A9077) TLS v1.3 KDF: (A9077) CKG - Section 4: () Key Type: Symmetric and Asymmetric |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | 256 (s = 256), SHA3-384 (s = 384), SHA3-512 (s = 512)<br>KDF KMAC [SP800-108r1]:KMAC-128 (112 <= s <= 128), KMAC-256 (112 <= s <= 256)<br>KDF [SP800-108r1]:CMAC-AES128 (s = 128), CMAC-AES192 (s = 192), CMAC-AES256 (s = 256), HMAC-SHA-1 (s = 160), HMAC-SHA2-224 (s = 224), HMAC-SHA2-256 (s = 256), HMAC-SHA2-384 (s = 384), HMAC-SHA2-512 (s = 512), HMAC-SHA2-512/224 (s = 224), HMAC-SHA2-512/256 (s = 256), HMAC-SHA3-224 (s = 224), HMAC-SHA3-256 (s = 256), HMAC-SHA3-384 (s = 384), HMAC-SHA3-512 (s = 512)<br>KDF SSH [SP800-135r1]:TDES (s=168), AES-128 (s = 128), AES-192 (s = 192), AES-256 (s = 256); SHA-1 (s = 160), SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512)<br>PBKDF [SP800-132]:SHA-1 (s = 160), SHA2-224 (s = 224), SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512), SHA2-512/224 (s = 224), SHA2-512/256 (s = 256), SHA3-224 (s = 224), SHA3-256 (s = 256), SHA3-384 | |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | (s = 384), SHA3-512 (s = 512)<br>TLS v1.2 KDF RFC7627:TLS [RFC7627] key derivation with Extended Master Secret (EMS) support, using the listed hash algorithms: SHA2-256 (s = 256), SHA2-384 (s = 384), SHA2-512 (s = 512)<br>TLS v1.3 KDF [RFC8446]:HMAC-SHA2-256 (s = 256), HMAC-SHA2-384 (s = 384) | |
| KAS-1 | KAS-SSC | Scheme: EphemeralUnified, KAS Role: Initiator, Responder | SP800-56Ar3 KAS-ECC-SSC per IG D.F Scenario 2 path (1):B-233, K-233, P-224, B-283, K-283, P-256, B-409, K-409, P-384, B-571, K-571, and P-521 curves providing 112, 128, 192, or 256 bits of security strength | KAS-ECC-SSC Sp800-56Ar3: (A9077)<br>KAS-ECC CDH-Component SP800-56Ar3: (A9077) |
| KAS-2 | KAS-SSC | Scheme: dhEphem. KAS Role: Initiator, Responder | SP800-56Ar3 KAS-FFC-SSC IG D.F Scenario 2 path (1):2048, 3072, 4096, 6144, and 8192-bit key providing 112, 128, 152, 176, or 200 bits of security strength | KAS-FFC-SSC Sp800-56Ar3: (A9077) |
| KAS-3 | KAS-SSC | Scheme: KAS1, KAS2. KAS Role: Initiator, Responder | SP800-56Br2 KAS-IFC-SSC IG D.F Scenario 1 path (1):2048, 3072, 4096, 6144, and 8192-bit key providing 112, 128, 152, 176, or 200 bits of security strength | KAS-IFC-SSC: (A9077) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| Perform self-tests (All) | AsymKeyPair-KeyGen<br>BC-Auth<br>BC-UnAuth<br>DigSig-SigGen<br>DigSig-SigVer<br>DRBG<br>KAS-135KDF<br>KAS-56CKDF<br>KAS-SSC<br>KBKDF<br>KEM-Decap<br>KEM-Encap<br>MAC<br>PBKDF<br>SHA<br>XOF | All self-tests executed by the module at boot | | AES-ECB: (A9077)<br>AES-GCM: (A9077)<br>TDES-ECB: (A9077)<br>Counter DRBG: (A9077)<br>DSA SigVer (FIPS186-4): (A9077)<br>Hash DRBG: (A9077)<br>HMAC DRBG: (A9077)<br>HMAC-SHA2-256: (A9077)<br>KAS-ECC-SSC Sp800-56Ar3: (A9077)<br>KAS-FFC-SSC Sp800-56Ar3: (A9077)<br>KAS-IFC-SSC: (A9077)<br>KDF SP800-108: (A9077)<br>KDA OneStep SP800-56Cr2: (A9077)<br>KDA TwoStep SP800-56Cr2: (A9077)<br>KTS-IFC: (A9077)<br>PBKDF: (A9077)<br>RSA SigGen |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | | (FIPS186-5): (A9077) RSA SigVer (FIPS186-5): (A9077) SHA-1: (A9077) SHA2-512: (A9077) SHA3-256: (A9077) KDF ANS 9.42: (A9077) KDF ANS 9.63: (A9077) KDF SSH: (A9077) TLS v1.2 KDF RFC7627: (A9077) TLS v1.3 KDF: (A9077) KDF KMAC Sp800-108r1: (A9077) ECDSA SigGen (FIPS186-5): (A9077) ECDSA SigVer (FIPS186-5): (A9077) EDDSA SigGen: (A9077) EDDSA SigVer: (A9077) ML-KEM KeyGen: (A9077) ML-KEM EncapDecap: (A9077) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | | ML-DSA KeyGen: (A9077) ML-DSA SigGen: (A9077) ML-DSA SigVer: (A9077) SLH-DSA KeyGen: (A9077) SLH-DSA SigGen: (A9077) SLH-DSA SigVer: (A9077) |
| Cryptographic Key Generation (CKG) | CKG | Direct generation of symmetric keys per NIST SP800-133r2 Section 4 | | CKG - Section 4: () |
| Software Integrity Test | MAC | HMAC-SHA2-256 used to perform the software integrity test | Key size:256 bits | HMAC-SHA2-256: (A9077) |
| Cryptographic Key Generation (CKG) - AES XTS | CKG | AES XTS Key generated to comply with the approved key generation guidelines of NIST SP800-133r2 Section 6.3 | Key size:128, 256 bits | CKG - Section 6.3: () |
| ML-KEM EncapDecap | KEM-Decap KEM-Encap | Perform key encapsulation and decapsulation | Parameters:512, 768, 1024 Bit Strength:128, 192, 256 bits IG:IG D.S Scenario 1 | ML-KEM EncapDecap: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256). |
| ML-DSA Signature Generation | DigSig-SigGen DigSig-SigVer | ML-DSA Signature Generation and Signature Verification | Parameters:44, 65, 87 Bit Strength:128, 192, 256 bits | ML-DSA SigGen: (A9077) SHAKE: |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| and Verification | | | | SHAKE-128 (s=128), SHAKE-256 (s=256) ML-DSA SigVer: (A9077) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) |
| SLH-DSA Signature Generation and Verification | DigSig-SigGen DigSig-SigVer | SLH-DSA Signature Generation and Signature Verification | Parameters:SHA2-128s, SHA2-128f, SHA2-192s, SHA2-192f, SHA2-256s, SHA2-256f, SHAKE-128s, SHAKE-128f, SHAKE-192s, SHAKE-192f, SHAKE-256s, SHAKE-256f Bit Strength:128, 192, 256 bits | SLH-DSA SigGen: (A9077) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) SHA2: SHA2-256 (s=256), SHA2-512 (s=512) HMAC-SHA2 [FIPS198-1]: SHA2-256 (s=256), SHA2-512 (s=512) SLH-DSA SigVer: (A9077) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) SHA2: SHA2-256 (s=256), SHA2-512 (s=512) HMAC-SHA2 [FIPS198-1]: SHA2-256 (s=256), |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | | SHA2-512 (s=512) |
| PQC Key Pair Generation | AsymKeyPair-KeyGen CKG | Generation of asymmetric key pairs for PQC algorithms | ML-KEM [FIPS203] Parameters:512, 768, 1024<br>ML-DSA [FIPS204] Parameters:44, 65, 87<br>SLH-DSA [FIPS-205] Parameters:SHA2-128s, SHA2-128f, SHA2-192s, SHA2-192f, SHA2-256s, SHA2-256f, SHAKE-128s, SHAKE-128f, SHAKE-192s, SHAKE-192f, SHAKE-256s, SHAKE-256f<br>Bit Strength:128, 192, 256 bits | CKG - Section 4: ()<br>ML-KEM KeyGen: (A9077)<br>SHA3:: SHA3-256 s=256), SHA3-512 (s=512)<br>SHAKE:: SHAKE-128 (s=128), SHAKE-256 (s=256).<br>ML-DSA KeyGen: (A9077)<br>SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256)<br>SLH-DSA KeyGen: (A9077)<br>SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256)<br>SHA2: SHA2-256 (s=256), SHA2-512 (s=512)<br>HMAC-SHA2 [FIPS198-1]: SHA2-256 (s=256), SHA2-512 (s=512) |
| RSA Encapsulation and Unencapsulation | UNK | OAEP-basic (no key confirmation): RSA-OAEP, Key Encapsulation, Key Unencapsulation Key | SP800-56Br2 IFC (key encapsulation and un-encapsulation) per IG D.G:2048, 3072, 4096, and | KTS-IFC: (A9077) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | Generation Methods: rsakpg1-basic, rsakpg1-crt, rsakpg1-prime-factor, rsakpg2- basic, rsakpg2-crt, rsakpg2-prime-factor | 6144-bit key providing 112, 128, 152, or 176 bits of encryption strength | |
| KEM-Hybrid-TLS1 | AsymKeyPair-KeyGen KAS-SSC KEM-Decap KEM-Encap KEM-Hybrid | Hybrid KEM based on SecP384r1MLKEM1024 for the implementation of TLS 1.3 | IG and Key Combiner:IG D.S Scenario 2 Option NA: the approved key combiner is offered as a separate SFI/Service Non-ML-KEM:Approved per IG D.F Scenario [2] | ML-KEM KeyGen: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) ML-KEM EncapDecap: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) KAS-ECC-SSC Sp800-56Ar3: (A9077) |
| KEM-Hybrid-TLS2 | AsymKeyPair-KeyGen KAS-SSC KEM-Decap KEM-Encap KEM-Hybrid | Hybrid KEM based on SecP256r1MLKEM768 for the implementation of TLS 1.3 | IG and Key Combiner:IG D.S Scenario 2 Option NA: the approved key combiner is offered as a separate SFI/Service Non-ML-KEM:Approved per IG D.F Scenario [2] | ML-KEM KeyGen: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Type | Description | Properties | Algorithms |
|---|---|---|---|---|
| | | | | ML-KEM EncapDecap: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) KAS-FFC-SSC Sp800-56Ar3: (A9077) |
| KEM-Hybrid-TLS3 | AsymKeyPair-KeyGen KEM-Decap KEM-Encap KEM-Hybrid | Hybrid KEM based on X25519MLKEM768 for the implementation of TLS 1.3 | IG and Key Combiner:IG D.S Scenario 2 Option NA: the approved key combiner is offered as a separate SFI/Service Non-ML-KEM:Non-approved with no security claimed per IG 2.4.A | ML-KEM KeyGen: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) ML-KEM EncapDecap: (A9077) SHA3: SHA3-256 s=256), SHA3-512 (s=512) SHAKE: SHAKE-128 (s=128), SHAKE-256 (s=256) X25519: () |

Table 10: Security Function Implementations

Equivalent strength in bits is given for each key or algorithm type (as some algorithms do not use or produce keys). The term s is used throughout to indicate security strength, following the notation used in the majority of the sources.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

**Note 1:** Preimage resistance strength applies to hash algorithms used in DRBG, KDFs. Described also in [SP800-57P1r5] Table 3.

**Note 2:** Elliptic curve strengths are annotated as approximate (i.e., s ~=) since [SP800-186] Table 1 provides approximate security strengths.

**Note 3:** [SP800-186] (cited in [SP800-140Cr2]) and [FIPS140-3_IG] C.K indicate that the Binary (B-) and Koblitz (K-) curves are deprecated.

**Note 4:** Approved elliptic curves for ECC key agreement are given in [SP800-56Ar3] Table 24.

**Note 5:** In Digital Signature applications, security strength is primarily associated with the asymmetric key pair specification. The hash function used must have equivalent strength equal to or greater than the security strength of the associated key pair.

**Note 6**: Approved key types for FFC key agreement are given in [SP800-56Ar3] Tables 25, 26. The group notation of Table 26 is used for consistency with CAVP algorithm listings and ACVP capability registration.

**Note 7:** Approved key types for IFC key agreement are given in [SP800-56Br2] Table 4. IFC key types approved for Digital Signature Generation and Verification are given also in [SP800-57P1r5] Table 2. Equivalent strengths are annotated as approximate (i.e., s ~=) since [SP800-56Br2] Table 4 provides approximate security strengths.

**Note 8:** Security strengths for KDA One Step are given in [SP800-56Cr2] Table 1 (hash), Table 2 (HMAC) and Table 3 (KMAC).

**Note 9:** Security strength for L=2048/N=256 is determined in accordance with [FIPS140-3_IG] D.B Strength of SSP Establishment Methods as y = min(x, N/2), where x is 112 and therefore y = min(112, 128) = 112.

Other reference sources for the strengths are as follows:

- AES (AES-128, AES-192, AES-256): [SP800-57P1r5] Table 2.
- ECC (B-233, B-283, B-409, B-571, K-233, K-283, K-409, K-571, P-224, P-256, P-384, P-521): [SP800-186] Table 1.
- FFC (L=2048/N=224, L=2048/N=256, L=3072/N=256): [SP800-57P1r5] Table 2.
- FFC (ffdhe2048, ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, MODP-2048, MODP-3072, MODP-4096, MODP-6144, MODP-8192): [SP800-56Ar3] Tables 25 and 26.
- IFC (k=2048, k=3072, k=4096, k=6144, k=8192): [SP800-56Br2] Table 4.
- KMAC (KMAC128, KMAC256): [SP800-56Cr2] Table 3.
- SHA-1, SHA2 (SHA2-224, SHA2-256, SHA2-384, SHA2-512, SHA2-512/224, SHA2-512/256): [SP800-107] Table 1.
- SHA3 (SHA3-224, SHA3-256, SHA3-384, SHA3-512): [SP800-57P1r5] Table 3.
- SHAKE (SHAKE128, SHAKE256): [SP800-185] Section 8.1.

## 2.7 Algorithm Specific Information

a. AES-GCM Usage

AES-GCM IV generation must be compliant to [FIPS140-3_IG] C.H Key/IV Pair Uniqueness Requirements from SP 800-38D Scenario 1(a), tested per option (ii) under C.H TLS/DTLS 1.2 protocol IV generation per RFC7627, Scenario 1(d) SSHv2 per RFC4252, RFC4253 and RFC5647 and Scenario 5 TLS 1.3 per RFC8446. IV constructed in compliance with a protocol shall only be used in the context of the AES-GCM mode encryptions within the protocol.

The Module does not implement the TLS and SSH protocols itself, however, it provides the cryptographic functions required for implementing the protocols. AES-GCM encryption is used in the context of the SSH and TLS protocol versions 1.2 and 1.3. The module provides the primitives to support the AES-GCM ciphersuites from [SP800-52r1] Section 3.3.1. The module's implementation of AES-GCM is used together

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

with an application that runs outside the module's cryptographic boundary. The application negotiates the protocol session's keys and the 32-bit nonce value of the IV.

When the IV exhausts the maximum number of possible values for a given session key (2^64 - 1), this results in a failure in encryption and a handshake to establish a new encryption key will be required. It is the responsibility of the user of the module, i.e., the first party, client or server, to encounter this condition, to trigger this handshake in accordance with the TLS/SSH protocol.

The Module also supports internal IV generation using the module's approved DRBG. The IV is at least 96 bits in length per [SP800-38D] Section 8.2.2. Per [FIPS140-3_IG] C.H Scenario 2 and [SP800-38D], the approved DRBG generates outputs such that the (key, IV) pair collision probability is less than 2^-32.

In each case, in the event that the Module power is lost and restored the user must ensure that the AES-GCM encryption/decryption keys are re-distributed in accordance with IG C.H Scenario 3. The module does not support persistent storage of SSPs.

The Module also supports importing of GCM IVs when an IV is not generated within the Module. In the approved mode, an IV must not be imported for encryption from outside the cryptographic boundary of the Module as this will result in a non-conformance. This is in accordance with IG 2.4.A: If the module operator (e.g., calling application) can do things outside of the module's control/visibility that can take an otherwise approved algorithm and use it in a non-approved way (e.g., use PBKDF and/or AES XTS outside of storage applications), the corresponding module service may still be considered approved (and if so, shall have an approved indicator per AS02.24) and the Security Policy shall clarify how to use the service in an approved manner (per ISO 19790 B.2.2 on Overall security design and the rules of operation).

b. PBKDF Usage

The lower limit on the supported length of a password/passphrase used in key derivation is 1-character. The ASCII system comprises of 94 printable characters (letters, digits, punctuation, and symbols). For a 1-character password/passphrase chosen from 94 printable ASCII characters, the total combinations are: 94^1. Thus, the probability of guessing the correct password/passphrase on a random attempt is: 1/94^1 ~0.010.

The module being software, does not restrict the usage of a password/string used as the password and input to the PBKDF. The onus is on the calling application to provide a password of an appropriate length based on the intended security strength (and size) of the key to be derived.

In accordance with NIST SP 800-132, passwords shorter than 10 characters are usually considered to be weak. There are many other properties that may render a password weak. For example, it is not advisable to use sequences of numbers or sequences of letters as passwords. Easily accessed personal information, such as the user's name, phone number, and date of birth, should not be used directly as a password.

Passphrases frequently consist solely of letters, but they make up for their lack of entropy by being much longer than passwords, typically 20 to 30 characters. Passphrases shorter than 20 characters are usually considered weak.

The module complies with NIST SP 800-132 Section 5.4 Option 1 a) and IG D.N. The iteration count values used range from 1 to 10000 per NIST SP 800-132 Section 5.2 whereby the iteration count shall be selected as large as possible, as long as the time required to generate the key using the entered password is acceptable for the users. Keys derived from passwords, as shown in SP 800-132, may only be used in storage applications. The security strength of the derived key is at least 112 bits. The module implements CKG per NIST SP 800-133r2 Section 6.2.2.

c. AES-XTS Usage

AES-XTS is only approved for storage purposes. Per IG C.I, AES-XTS Key_1 and/or Key_2, when entered into the module by the operator, shall be generated and/or established independently of each other according to the rules for component symmetric keys from NIST SP 800-133rev2, Sec. 6.3. for an approved use of AES-XTS.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

d. Legacy Usage

The module supports the following implementations for legacy use/support per NIST SP 800-131Ar2:

- RSA x9.31 (modulus 1024, 2048, 3072 and 4096 bits) and DSA (modulus 1024, 2048 and 3072 bits) digital signature verification as per IG C.K Resolution 1.
- RSA (modulus 1024 bits), DSA (modulus 1024 bits) digital signature verification providing less than 112 bits of security strength.
- DSA PQG Verification (modulus 1024, 2048 and 3072 bits) as per IG C.M.
- RSA and DSA digital signature verification with SHA-1 used as the underlying hash algorithm.

e. Component Validation List (CVL)

In accordance with IG 2.4.B, all tested components that may be called during the operation of the module and shown in the module's CVL certificates have been listed individually in Table 5. All vendor affirmed components that may be called during the operation of the module have also been listed individually in Table 6 per IG 2.4.B.

f. FIPS 202 Usage

In accordance with IG C.C Resolution 2. a., each SHA-3 and SHAKE function has been tested and validated on all of the module's operating environments. Per Resolution 2. c., SHA-3 hash functions used as part of the higher-level DRBG algorithms for which the CAVP testing is not yet available have been vendor affirmed as documented in Table 6.

g. RSA Usage

- Per IG C.E, the module generates RSA signature keys using an approved key generation procedure per RSA KeyGen validated for conformance to FIPS 186-5 Cert. #A9077.
- Per IG C.F, the RSA SigGen and SigVer implementations have been tested for all implemented RSA modulus lengths where CAVP testing is available. The module supports generation of RSA keys with the following untested approved moduli/sizes: 8192 <nlen<= 16384. The module also supports the following untested, approved moduli for the RSA SigGen and SigVer: 4096 <nlen<= 16384.

h. TLS 1.2 KDF

In accordance with IG D.Q, the module has been implemented to enforce usage of the extended master secret in the TLS 1.2 KDF as specified per Section 2.4.1. d. (tls1-prf-ems-check set option set to 1 due to initialization of the module per Section 11). The module complies with RFC 7627.

i. DRBG Usage

Per IG D.R, the Hash_DRBG and HMAC_DRBG implementations use SHA-1, SHA2-256, SHA2-512, SHA3-256 and SHA3-512.

j. NIST SP 800-108 KDF Usage

Per IG D.M, the SP 800-108 KDF is not used to generate asymmetric keys directly in that the module restricts generation of keys to approved methods only. For keys passed into the module, the onus lies on the calling application to ensure correct generation of such keys using approved mechanisms. The module implements CKG per NIST SP 800-133r2 Section 6.2.3.

k. SHA-1 Usage:

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

The module implements SHA-1 for usage in the following (this can be vetted from the SFI Table 9 in the Security Policy):

i. As a PRF in the KDF ANS 9.42, KDA HKDF, KDA OneStep KDF, KDF SSH [SP800-135r1], PBKDF [SP800-132].

ii. As a standalone SHA-1 hash function.

iii. As a PRF in HMAC-SHA-1.

iv. As the underlying hash function for RSA SigVer [FIPS 186-4] and DSA SigVer for legacy use/support per NIST SP 800-131Ar2 as specified in the Security Policy Section 2.7 d.

v. As the underlying hash function in Hash DRBG and HMAC DRBG.

l. Key Agreement (KAS):

The module does not establish SSPs using an approved key agreement scheme (KAS). However, it does offer some or all of the underlying KAS cryptographic functionality to be used by an external operator/application as part of an approved KAS.

Note that the caller provides the KAS_Private and KAS_Public keys for shared secret computation; the caller's exchange and assurance of PSPs with the remote participant is outside the scope of the Module.

m. Key Transport (KTS):

The module does not establish SSPs using an approved key transport scheme (KTS). However, it does offer approved authenticated algorithms that can be used by an external operator/application as part of an approved KTS.

n. Key Encapsulation Mechanism (KEM):

The module does not establish SSPs using an approved key encapsulation mechanism (KEM). However, it does offer some or all the underlying KEM cryptographic functionality to be used by an external operator/application as part of an approved KEM.

It can be utilized as PQC Key-encapsulation mechanism or as part of a hybrid key establishment scheme based on combining a classic key agreement scheme with ML-KEM according to IG D.S and [SP800-227].

o. Hybrid Key Encapsulation Mechanism (KEM-Hybrid):

The module does not establish module SSPs using an approved hybrid KEM (thus, no KEM-HybridFull SFI type is claimed per supplemental guidance). However, it does offer hybrid KEM combination sets to be used by an external operator/application as part of an approved hybrid KEM.

The module implements the following pre-defined set of hybrid schemes: SecP256r1MLKEM768, SecP384r1MLKEM1024 and X25519MLKEM768.

The SecP256r1MLKEM768, SecP384r1MLKEM1024 and X25519MLKEM768 hybrid schemes are implemented to be utilized as part of a hybrid key exchange in TLS 1.3.

p. ML-DSA Usage:

The module supports ML-DSA signature generation and verification using both the internal and external μ interfaces. Per FIPS 204 for ML-DSA, the external Mu shall be provided by another cryptographic module.

## 2.8 RBG and Entropy

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

The Module relies on the use of a [SP800-90B] compliant entropy source outside the Module boundary. The calling application is responsible for use of an [SP800-90B] compliant entropy source with sufficient entropy based on the required security strength. Entropy is supplied to the Module via callback functions (see Section 2.4 2. c). Minimum Number of Bits of Entropy, depending on the target security strength of generated SSPs are 128, 192 or 256 bits. When using the Counter DRBG implementation without the derivation function enabled, full entropy from the entropy source is required. The following caveat applies to the module: No assurance of the minimum strength of generated SSPs (e.g., keys).

## 2.9 Key Generation

The module implements NIST SP 800-90Ar1 DRBGs and supports the following sections per NIST SP 800-133r2 (CKG):

- Section 4: Using the Output of a Random Generator, Example 1. For the purposes defined in the following sections of NIST SP 800-133r2.
    - Section 5.1: Key Pairs for Digital Signature Schemes.
    - Section 5.2: Key Pairs for Key Establishment.
    - Section 6.1: Direct Generation of Symmetric Keys.
    - Section 6.2: Derivation of Symmetric keys.
- Section: 6.3: Symmetric Keys Produced by Combining Multiple Keys and Other Data, Method 1.

## 2.10 Key Establishment

Per IG D.F:

The module offers as a service to the calling application CAVP-tested KAS SSC that map to [FIPS140-3_IG] D.F Scenario 1 path (1) or [FIPS140-3_IG] D.F Scenario 2 path (1) without establishing a module key/SSP used by the module for cryptographic protection. The KAS-1, KAS-2, KAS-3 in the SFI Table 9 have been documented accordingly. The Approved Algorithm list includes the tested components (KAS-ECC-SSC, KAS-FFC-SSC and KAS-IFC-SSC) as individual entries.

The Module obtains the [FIPS140-3_IG] D.F Additional Comment 5 required key agreement assurances:

- [SP800-56Ar3] in accordance with Section 5.6.2.
- [SP800-56Br2] in accordance with Section 6.4.

## 2.11 Industry Protocols

The Module conforms to Resolution 3 per [FIPS140-3_IG] D.C References to the Support of Industry Protocols: while it provides [SP800-56Ar3] conformant schemes and API entry points oriented to SSH and TLS usage, the Module does not contain the full implementation of SSH or TLS. The following caveat is required:

*No parts of the TLS protocol, other than the approved cryptographic algorithms and the KDFs, have been tested by the CAVP and CMVP.*

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 3 Cryptographic Module Interfaces

## 3.1 Ports and Interfaces

| Physical Port | Logical Interface(s) | Data That Passes |
|---|---|---|
| N/A | Control Input | API entry point: stack frame including non-sensitive parameters |
| N/A | Data Input | API call parameters passed by reference or value for cryptographic service input |
| N/A | Status Output | API return value: enumerated status resulting from call execution |
| N/A | Data Output | API call parameters passed by reference for cryptographic service output |

Table 11: Ports and Interfaces

Table 10 defines the Module's [FIPS140-3] logical interfaces; the Module does not interact with physical ports. The Control Output logical interface is not applicable to the Module and is intentionally omitted from Table 10.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 4 Roles, Services, and Authentication

## 4.1 Authentication Methods

N/A for this module.

The Module does not provide an authentication or identification method of its own. The CO role is assumed by meeting the conditions of Section 11 of this document.

## 4.2 Roles

| Name | Type | Operator Type | Authentication Methods |
|---|---|---|---|
| Crypto Officer | Role | Crypto Officer | None |

Table 12: Roles

The Module supports the mandatory Cryptographic Officer (CO) operational role only (implicitly identified) and does not support a maintenance role or a bypass capability.

## 4.3 Approved Services

The Approved Service table describes Module service access to SSPs; '--' indicates the cell is intentionally empty, not applicable or not relevant. The following annotations indicate the type of access by the Module service:

- G = Generate: The Module generates or derives the SSP.
- R = Read: The SSP is read from the Module (e.g. the SSP is output).
- W = Write: The SSP is updated, imported, or written to the Module.
- E = Execute: The Module uses the SSP in performing a cryptographic operation.
- Z = Zeroise: The Module zeroises the SSP.

Each service provides context sensitive status responses as described in the OpenSSL 3 API manual pages; generally, functions of return type int return the value 1 for success with other error codes as appropriate for the call (described in API documentation).

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| Initialize | Module initialization | | Core handle, dispatch in and out, provider context | Initialization status (1 = pass, 0 = fail) | Random Number Generation | Crypto Officer<br>- DRBG_EI: W,E,Z<br>- DRBG_State (IG D.L): G |
| Core (all except Teardown) (Show Status, Show Version) | Show status; Core operations dispatched by FIPS | | Provider context, parameters types | Parameter types (array) with: Name, | None | Crypto Officer |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | provider: Metadata (Gettable parameters; Get parameters; Get capabilities); Query; Self-test | | (array), capability, callback pointer and arguments, operation ID | Version, BuildInfo, Status, SecurityChecks; Status return, TLS group capabilities, Null or array of available operations | | |
| Core: Perform self-tests | Run the self-test sequence | | Provider context | Status (1 = pass, 0 = fail) | Perform self-tests (All) Software Integrity Test | Crypto Officer |
| Core: Teardown (Perform zeroisation) | Uninstantiate the module; includes Zeroise | | Provider context | None | None | Crypto Officer<br>- DS_SGK: Z<br>- DS_SVK: Z<br>- GKP_Private: Z<br>- GKP_Public: Z<br>- KAS_Private: Z<br>- KAS_Public: Z<br>- ML-KEM_Private: Z<br>- ML-KEM_Public: Z<br>- KD_KDK: Z<br>- PBKDF Password: Z<br>- KD_DKM: Z<br>- KH_Key: Z |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | | | | | | - RSA_KDK: Z<br>- RSA_KEK: Z<br>- DRBG_EI: Z<br>- DRBG_Seed (IG D.L): Z<br>- DRBG_State (IG D.L): Z<br>- SC_EDK: Z<br>- TDES_DK: Z<br>- GCM_IV: Z<br>- KAS_SS: Z<br>- KEM_SS: Z<br>- KASKEM_SS: Z<br>- ML-DSA_Seed: Z<br>- SLH_DSA_SKSEED: Z<br>- SLH_DSA_SKPRF: Z<br>- SLH_DSA_PKSEED: Z<br>- ML_KEM_SEED: Z |
| Asymmetric cipher (Key Transport) (Perform approved security functions) | Encapsulate or decapsulate key material on behalf of the calling process (does not establish keys into the module) | fips-indicator = 1 | Encapsulate: Key struct (RSA_KDK); Decapsulate (RSA_KEK) | Status return; Plaintext or ciphertext data, or wrapped key | RSA Encapsulation and Unencapsulation | Crypto Officer<br>- RSA_KDK: W,E<br>- RSA_KEK: W,E |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| Cipher (Encryption/Decryption and Key Wrapping) (Perform approved security functions) | Encrypt or decrypt data, including AEAD modes (CCM, GCM) and key wrap (KW, KWP) (CSPs are passed in by the calling process or generated within the module) | fips-indicator = 1 | SC_EDK, TDES_DK and KH_Key (for key wrapping); flags | Status return. Plaintext or ciphertext data, or wrapped key | Symmetric Encryption or Decryption Keyed Hash Cryptographic Key Generation (CKG) Cryptographic Key Generation (CKG) - AES XTS | Crypto Officer<br>- SC_EDK: G,R,W,E<br>- TDES_DK: W,E<br>- KH_Key: G,W,E<br>- GCM_IV: G,W,E |
| Key derivation (Perform approved security functions) | Derive keying material | fips-indicator = 1 | KD_KDK; flags | Status return; KD_DKM | Key Derivation | Crypto Officer<br>- KD_KDK: W,E<br>- PBKDF Password: W,E,Z<br>- KD_DKM: G,R |
| Key exchange (Perform approved security functions) | Perform key agreement primitives on behalf of the calling process (does not establish keys into the module) | fips-indicator = 1 | Key structs (KAS_Private and KAS_Public); flags | Status return; Shared Secret | KAS-1 KAS-2 KAS-3 | Crypto Officer<br>- KAS_Private: W,E<br>- KAS_Public: W,E |
| Key management (Perform approved security functions) | Generate asymmetric key pairs | fips-indicator = 1 | ECDSA: curve identifier; EDDSA: curve identifier; RSA: modulus size; | Status return; Key struct (GKP_Private, GKP_Public) | Asymmetric Key Pair Generation PQC Key Pair Generation | Crypto Officer<br>- GKP_Private: G,R<br>- GKP_Public: G,R<br>- ML-DSA_Seed: |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | | | ML-KEM: parameters; ML-DSA: parameters; SLH-DSA: parameters | | | G<br>- SLH_DSA_SKSEED: G<br>- SLH_DSA_SKPRF: G<br>- SLH_DSA_PKSEED: G<br>- ML_KEM_SEED: G |
| Message authentication (Perform approved security functions) | Generate or verify data integrity. (CSPs are passed in by the calling process or generated within the module) | fips-indicator = 1 | KH_Key | Status return; Tag value | Keyed Hash Cryptographic Key Generation (CKG) | Crypto Officer<br>- KH_Key: W,E |
| Message digest (Perform approved security functions) | Generate a message digest | fips-indicator = 1 | Message; flags | Status return; Hash value | Message Digest | Crypto Officer |
| Random (Perform approved security functions) | Generate random bits using the DRBG | fips-indicator = 1 | DRBG struct (RBG State); DRBG_EI | Status return; Random value | Random Number Generation | Crypto Officer<br>- DRBG_EI: E<br>- DRBG_Seed (IG D.L): G,E<br>- DRBG_State (IG D.L): E |
| Signature (Perform approved security functions) | Generate or verify digital signatures (SSPs are passed in by the calling process) | fips-indicator = 1 | [RSA, ECDSA, EDDSA Sign: Key struct (DS_SGK); messag | Status return; Signature value | RSA Digital Signature Generation and Verification ECDSA Signature | Crypto Officer<br>- DS_SGK: W,E<br>- DS_SVK: W,E<br>- SLH_DSA_SKSEED: W,E |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | | | e; RSA, ECDSA, EDDSA Verify: signature value; Key struct (DS_SVK); flags; sizes] [ML-DSA Sign: Key struct (DS_SGK); message (M) or Encoded Message(M') or Mu; ML-DSA Sign Verify: signature value; Key struct (DS_SVK); message (M) or Encoded Message(M') or Mu; flags; sizes] [SLH-DSA Sign: Key | | Generation and Signature Verification EDDSA Signature Generation and Signature Verification DSA Digital Signature Verification RSA Signature Primitive ML-DSA Signature Generation and Verification SLH-DSA Signature Generation and Verification | - SLH_DSA_SKPRF: W,E - SLH_DSA_PKSEED: W,E |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | | | struct (DS_SGK); message (M) or Encoded Message(M'); SLH-DSA Sign Verify: signature value; Key struct (DS_SVK); message (M) or Encoded Message(M'); flags; sizes] | | | |
| Key Encapsulation/Decapsulation (Perform approved security functions) | Perform key encapsulation and decapsulation primitives on behalf of the calling process (does not establish keys into the module) | fips-indicator = 1 | ML-KEM: Parameter, ML-KEM Public Key, ML-KEM Private Key | Status return; Shared secret | ML-KEM EncapDecap | Crypto Officer<br>- ML-KEM_Private: W,E<br>- ML-KEM_Public: W,E |
| Zeroise (Perform zeroisation) | The core Teardown operation zeroizes all Module scope SSPs; Call stack | | Memory pointer | Void | None | Crypto Officer<br>- DS_SGK: Z<br>- DS_SVK: Z<br>- GKP_Private: Z<br>- |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | cleanup is the duty of the application; Restarting the general-purpose computer clears all SSPs in RAM; OPENSSL_cleanse provides zeroisation of SSPs managed by the caller; See the notes below this table for additional explanation | | | | | GKP_Public: Z<br>- KAS_Private: Z<br>- KAS_Public: Z<br>- ML-KEM_Private: Z<br>- ML-KEM_Public: Z<br>- KD_KDK: Z<br>- PBKDF Password: Z<br>- KD_DKM: Z<br>- KH_Key: Z<br>- RSA_KDK: Z<br>- RSA_KEK: Z<br>- DRBG_EI: Z<br>- DRBG_Seed (IG D.L): Z<br>- DRBG_State (IG D.L): Z<br>- SC_EDK: Z<br>- TDES_DK: Z<br>- GCM_IV: Z<br>- KAS_SS: Z<br>- KEM_SS: Z<br>- KASKEM_SS: Z<br>- ML-DSA_Seed: Z<br>- SLH_DSA_SKSEED: Z<br>- SLH_DSA_S |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Indicator | Inputs | Outputs | Security Functions | SSP Access |
|---|---|---|---|---|---|---|
| | | | | | | KPRF: Z<br>-<br>SLH_DSA_PKSEED: Z<br>-<br>ML_KEM_SEED: Z |
| Hybrid KEM (Perform approved security functions) | Perform hybrid key encapsulation mechanism primitives on behalf of the calling process (does not establish keys into the module). | fips-indicator = 1 | Hybrid KEM Parameters: ML-KEM Public Key, ML-KEM Private Key | Status return; Concatenated shared secret | KEM-Hybrid-TLS1<br>KEM-Hybrid-TLS2<br>KEM-Hybrid-TLS3 | Crypto Officer<br>-<br>KAS_Private: W,E<br>-<br>KAS_Public: W,E<br>- ML-KEM_Private: W,E<br>- ML-KEM_Public: W,E<br>- KAS_SS: G,R<br>- KEM_SS: G,R<br>-<br>KASKEM_SS: G,R |

Table 13: Approved Services

Note: For every service in the Approved Services table that uses an algorithm context, the approved status of the service is defined solely by the value of the Approved Service Indicator, the integer parameter "fips-indicator" (OSSL_ALG_PARAM_FIPS_APPROVED_INDICATOR). The Indicator column gives the value the Module reports when the service is approved (1). The algorithm name, mode, curve, digest or key size is not itself the indicator and must not be used by the calling application to decide whether a service was approved; only the value of the fips-indicator parameter is used.

**How the indicator works:**

- The "fips-indicator" (OSSL_ALG_PARAM_FIPS_APPROVED_INDICATOR) is a gettable integer parameter of the algorithm context. A value of 1 means the operation was approved, meaning that every check required for the algorithm (key size, digest, mode, parameter set, IV source, etc.) passed. A value of 0 means the operation was not approved.
- The calling application must request the indicator from the same context that performed the service, after the operation has completed and before the context is freed or reused, using the getter listed below for the type of service. The value applies only to the operation just performed and must be read again after each operation.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

- The application can register an indicator callback (OSSL_INDICATOR_set_callback) to be notified when a check fails; the callback is informational and is not the indicator. The operator would still need to check OSSL_ALG_PARAM_FIPS_APPROVED_INDICATOR..

**Getter used to retrieve the fips-indicator, by type of service:**

- Message digest (SHA, SHAKE):
  - EVP_MD_CTX_get_params() after EVP_DigestFinal_ex() or EVP_DigestFinalXOF()
- Symmetric encryption and key wrapping (AES modes, Triple-DES):
  - EVP_CIPHER_CTX_get_params() after the final call of the operation (e.g. EVP_CipherFinal_ex())
- Message authentication (HMAC, CMAC, KMAC, GMAC):
  - EVP_MAC_CTX_get_params() after EVP_MAC_final()
- Key derivation (KDFs):
  - EVP_KDF_CTX_get_params() after EVP_KDF_derive()
- Key exchange (KAS-ECC-SSC, KAS-FFC-SSC):
  - EVP_PKEY_CTX_get_params() after EVP_PKEY_derive()
- Asymmetric cipher (RSA key transport), ML-KEM and Hybrid KEM:
  - EVP_PKEY_CTX_get_params() after EVP_PKEY_encrypt()/EVP_PKEY_decrypt() or EVP_PKEY_encapsulate()/EVP_PKEY_decapsulate()
- Signature generation and verification (RSA, ECDSA, DSA, EDDSA, ML-DSA, SLH-DSA, RSA Signature Primitive):
  - EVP_PKEY_CTX_get_params() after EVP_PKEY_sign()/EVP_PKEY_verify() or the message-based sign/verify calls; for EVP_DigestSign*/EVP_DigestVerify* use the EVP_PKEY_CTX returned by EVP_MD_CTX_get_pkey_ctx(). EVP_MD_CTX_get_params() on a DigestSign/DigestVerify context returns the indicator of the underlying digest, not of the signature, and must not be used for this purpose.
- Key pair generation (SafePrimes, RSA, ECDSA, EDDSA, ML-DSA, SLH-DSA, ML-KEM):
  - EVP_PKEY_CTX_get_params() on the key generation context after EVP_PKEY_generate()
- Random bit generators (Hash, HMAC and Counter DRBG):
  - EVP_RAND_CTX_get_params() after EVP_RAND_generate()

**Note on the Cipher service:**

Triple-DES (DES-EDE3-ECB and DES-EDE3-CBC) is approved for decryption only. Triple-DES encryption is rejected with an error in the approved configuration.

**Services that have no algorithm context:**

Initialize, the Core services including Show Status, Show Version and Perform self-tests, Teardown, and Zeroise do not provide a fips-indicator parameter. These services are always approved when they succeed.

**How the services with no algorithm context are called:**

- Initialize: OSSL_PROVIDER_load() for the FIPS provider (or loading it through the OpenSSL configuration file). Loading runs the Module initialization, including the integrity test and the self-tests, and returns the provider handle on success.
- Core (Show Status, Show Version and the other core operations): OSSL_PROVIDER_get_params() with the provider handle returns the name, version, build information and status (1 while the Module is running);

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

OSSL_PROVIDER_gettable_params(), OSSL_PROVIDER_query_operation() and OSSL_PROVIDER_get_capabilities() return the metadata, available operations and capabilities.

- Core: Perform self-tests: OSSL_PROVIDER_self_test() with the provider handle runs the self-test sequence on demand and returns 1 for pass or 0 for fail.
- Core: Teardown (Perform zeroisation): OSSL_PROVIDER_unload() with the provider handle; when the last reference to the provider is released the Module is torn down, which frees the Module library context and provider context and zeroises the SSPs they hold.
- Zeroise: OSSL_PROVIDER_unload() (the Teardown operation) zeroises all Module-scope SSPs; SSPs owned by the calling application are zeroised with OPENSSL_cleanse(); freeing an EVP context or key object (e.g. EVP_CIPHER_CTX_free(), EVP_PKEY_free()) zeroises the SSPs held by that object.

**Status of the Module:**

The OpenSSL toolkit *OSSL_PROVIDER_get_params* function when called with the Module's global handle and a pointer to a parameter structure (initialized using *provider_gettable_params* or the equivalent), can be used to retrieve the current status of the Module as well as the name and version; this information correlates to the validation listing. A 1 value returned in status indicates the Module is running without error (FIPS_OK); a 0 return indicates an error (with additional error details indicated as described in the release specific API documentation). Services are only operational in the running state. Any attempts to access services in any other state will result in an error being returned. If the integrity test or any CAST fails then any attempt to access any service will result in an error being returned.

## 4.4 Non-Approved Services

| Name | Description | Algorithms | Role |
|---|---|---|---|
| Signature | Generate signatures (SSPs are passed in by the calling process) | FIPS 186-4 DSA Digital Signature<br>FIPS 186-4 RSA Digital Signature<br>FIPS 186-2 RSA SigGen | Crypto Officer |
| Key Exchange | Perform key agreement primitives on behalf of the calling process (does not establish keys into the module) | X448<br>X25519 | Crypto Officer |
| Cipher (Encryption) | Encrypt data (CSPs are passed in by the calling process) | Triple-DES | Crypto Officer |
| ECDSA SigVer Component | Verify ECDSA digital signatures (SSPs are passed in by the calling process) | ECDSA SigVer Component | Crypto Officer |
| Key Derivation | Derive keys (key derivation key passed in by the calling process) | X963KDF<br>HKDF | Crypto Officer |
| Key Generation | Generate DSA public/private key pair per FIPS 186-4 | FIPS 186-4 DSA KeyGen/PQGen | Crypto Officer |
| Keyed Hash | Generate HMAC using key length less than 112 bits | HMAC | Crypto Officer |
| Random | Generate random bits using the non-approved Hash and HMAC DRBGs with PRFs SHA2-224, | Hash and HMAC DRBG | Crypto Officer |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Algorithms | Role |
|---|---|---|---|
| | SHA2-384, SHA2-512/224 and SHA2-512/256 | | |
| RSA SigVer Component | Verify RSA digital signatures (SSPs are passed in by the calling process) | RSA SigVer Component | Crypto Officer |
| Encryption | AES GCM (external IV) | AES GCM encryption with external IV | Crypto Officer |
| ML-KEM encapsulation with user-provided randomness | Key encapsulation | ML-KEM encapsulation with user-provided randomness | Crypto Officer |
| Key Agreement | X448MLKEM1024 hybrid Key Agreement scheme | X448MLKEM1024 | Crypto Officer |

Table 14: Non-Approved Services

## 4.5 External Software/Firmware Loaded

The module does not support loading of external software.

## 4.6 Bypass Actions and Status

The module does not support bypass.

## 4.7 Cryptographic Output Actions and Status

The module does not support self-initiated cryptographic output.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 5 Software/Firmware Security

## 5.1 Integrity Techniques

The Module uses HMAC-SHA2-256 as the approved integrity technique; the file fipsmodule.cnf contains the integrity reference value. The HMAC key used for the integrity test is considered a non-SSP. The HMAC-SHA2-256 CAST is performed prior to the software integrity test. The Module is provided in an executable form (as fips.so shared object for use in Linux environments, fips.dylib for use in Mac environments and fips.dll for use in Windows environments).

The module does not support loading of any additional software.

## 5.2 Initiate on Demand

The operator can initiate the integrity test on demand by calling *fips_self_test* (invoked using *OSSL_PROVIDER_self_test* called with the Module's global handle) or reloading the Module.

## 5.3 Open-Source Parameters

In accordance with [ISO19790] Annex B, as the Module is open source, the tools used to build the Module as tested are:

- gcc version 9.3.0
- perl v5.30.0
- gnu make v4.2.1

Compilers Used for Each Operational Environment

The specific compilers used to generate the Module for the respective operational environments are listed below:

- Ubuntu Linux 22.04.1 Server: gcc 11.2.0
- Ubuntu Linux 24.04.2 Server: gcc 13.2.0
- Debian GNU/Linux 11.5: gcc 10.2.1
- Debian GNU/Linux 12: gcc 12.2.0
- FreeBSD 13.1: clang 13.0.0
- FreeBSD 14.3: clang 13.0.6
- Windows 10 Pro: Visual Studio 2019
- Windows Server 2025 Standard 64-bit: Visual Studio 2019
- macOS 15.5: clang 16.0.0

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 6 Operational Environment

## 6.1 Operational Environment Type and Requirements

**Type of Operational Environment**: Modifiable

**How Requirements are Satisfied:**

The operational environment for the Module is modifiable as it runs in General Purpose Computers (GPC). The Module conforms to [FIPS140-3_IG] 2.3.C Processor Algorithm Accelerators (PAA) and Processor Algorithm Implementation (PAI). The AES-NI functions are identified by [FIPS140-3_IG] 2.3.C as a known PAA.

## 6.2 Configuration Settings and Restrictions

Table 3 lists the operational environments on which the Module was tested; no operational environment restrictions are required for operation in the approved mode.

All conditions for operation of the Module in the approved mode are given in Section 2.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 7 Physical Security

Physical Security requirements are not applicable for this software Module.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 8 Non-Invasive Security

In accordance with current CMVP policy, Non-Invasive Security is not applicable.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 9 Sensitive Security Parameters Management

## 9.1 Storage Areas

| Storage Area Name | Description | Persistence Type |
|---|---|---|
| RAM | Temporary, plaintext storage | Dynamic |

Table 15: Storage Areas

## 9.2 SSP Input-Output Methods

| Name | From | To | Format Type | Distribution Type | Entry Type | SFI or Algorithm |
|---|---|---|---|---|---|---|
| CALL STACK (API) INPUT PARAMETERS | Calling application | RAM | Plaintext | Manual | Electronic | |
| CALL STACK (API) OUTPUT PARAMETERS | RAM | Calling application | Plaintext | Manual | Electronic | |

Table 16: SSP Input-Output Methods

The Software Integrity key stored in the configuration file is not considered as a SSP.

The module is complaint with FIPS 140-3 IG 9.5.A MD/EE (CM Software to/from App via TOEPP Path).

No assurance of minimum security of SSPs (e.g., keys, bit strings) that are externally loaded, or of SSPs established with externally loaded SSPs apply

## 9.3 SSP Zeroization Methods

| Zeroization Method | Description | Rationale | Operator Initiation |
|---|---|---|---|
| OPENSSL_cleanse | Zeroisation of SSPs managed by the caller | The OPENSSL_cleanse provides zeroisation of SSPs managed by the caller | Module initiated |
| Cleared after use | Temporary copies of CSPs are zeroised within the relevant function for the scope within which they are used | CSPs with a lifetime associated with an OpenSSL object will be zeroised when reinitialized | Module initiated |
| Teardown | This operation triggers Module uninstantiation | CSPs with a lifetime associated with the Module are zeroised on Module uninstantiation | Operator initiated |
| Restarting the general-purpose computer | RAM (memory) is used for temporary storage of SSPs | Restarting the general-purpose computer clears all SSPs in RAM | Operator initiated |

Table 17: SSP Zeroization Methods

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

All CSPs are zeroised (overwritten with 0s) when they are no longer needed:

- Temporary copies of CSPs are zeroised within the relevant function for the scope within which they are used.
- CSPs with a lifetime associated with an OpenSSL object (e.g., EVP_PKEY) will be zeroised when reinitialized.
- CSPs with a lifetime associated with the Module are zeroised on Module uninstantiation (the Teardown operation).

The OPENSSL_cleanse function is used to zeroise CSPs owned by the caller.

## 9.4 SSPs

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| DS_SGK | Private key for signature generation | RSA: 2048, 3072 and 4096 bits; ECDSA: B-233, K-233, P-224; B-283, K-283, P-256; B-409, K-409, P-384; B-571, K-571, P-521; EDDSA: Edwards25519, Edwards448; ML-DSA: 20480, 32256, 39168 bits; SLH-DSA: 512, 768, 1024 bits - RSA: 112, 128, 152; ECDSA: 112, 128, 192, 521; EDDSA: 128, 224; ML-DSA: 128, 192, 256; SLH- | Private key - CSP | | | RSA Digital Signature Generation and Verification ECDSA Signature Generation and Signature Verification EDDSA Signature Generation and Signature Verification RSA Signature Primitive ML-DSA Signature Generation and Verification SLH-DSA Signature Generation and Verification |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | | DSA: 128, 192, 256 | | | | |
| DS_SVK | Public key for signature verification | RSA: 1024, 2048, 3072 and 4096 bits; DSA: 1024, 2048 and 3072 bits; ECDSA: B-233, K-233, P-224; B-283, K-283, P-256; B-409, K-409, P-384; B-571, K-571, P-521; EDDSA: Edwards25519, Edwards448; ML-DSA: 10496, 15616, 20736 bits; SLH-DSA: 256, 384, 512 bits - RSA: 80, 112, 128, 152; DSA: 80, 112 or 128; ECDSA: 112, 128, 192, 521; EDDSA: 128, 224; ML-DSA: 128, 192, 256; SLH-DSA: 128, 192, 256 | Public key - PSP | | | RSA Digital Signature Generation and Verification ECDSA Signature Generation and Signature Verification EDDSA Signature Generation and Signature Verification DSA Digital Signature Verification ML-DSA Signature Generation and Verification SLH-DSA Signature Generation and Verification |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| GKP_Private | Key pair (Private: DS_SGK, Public: DS_SVK) generated per caller request; the keypair purpose is unspecified | RSA: 2048, 3072, 4096 bits; ECDSA: B-233, K-233, P-224; B-283, K-283, P-256; B-409, K-409, P-384; B-571, K-571, P-521; EDDSA: Edwards25519, Edwards448; ML-DSA: 20480, 32256, 39168 bits; SLH-DSA: 512, 768, 1024 bits; ML-KEM: 13056, 19200, 25344 bits - RSA: 112, 128, 152; ECDSA: 112, 128, 192, 256; EDDSA: 128, 224; ML-DSA: 128, 192, 256; SLH-DSA: 128, 192, 256; ML-KEM: 128, 192, 256 | Private key - CSP | Asymmetric Key Pair Generation Random Number Generation PQC Key Pair Generation | | |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| GKP_Public | Key pair (Private: GPK_Private, Public: GPK_Public) generated per caller request; the keypair purpose is unspecified | RSA: 2048, 3072, 4096 bits; ECDSA: B-233, K-233, P-224; B-283, K-283, P-256; B-409, K-409, P-384; B-571, K-571, P-521; EDDSA: 128, 224; ML-DSA: 10496, 15616, 20736 bits; SLH-DSA: 256, 384, 512 bits; ML-KEM: 6400, 9472, 12544 bits - RSA: 112, 128, 152; ECDSA: 112, 128, 192, 256; EDDSA: 128, 224; ML-DSA: 128, 192, 256; SLH-DSA: 128, 192, 256; ML-KEM: 128, 192, 256 | Public key - PSP | Asymmetric Key Pair Generation<br>Random Number Generation<br>PQC Key Pair Generation | | |
| KAS_Private | Key pair component provided | FFC: MODP2048, ffdhe2048, | Private key - CSP | Asymmetric Key Pair Generatio | | KAS-1<br>KAS-2<br>KAS-3<br>KEM- |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | by the local participant, used for Diffie-Hellman shared secret generation | MODP3072, ffdhe3072, MODP4096, ffdhe4096, MODP6144, ffdhe6144, MODP8192, ffdhe8192; ECC: B-233, K-233, P-224, B-283, K-283, P-256, B-409, K-409, P-384, B-571, K-571, P-521; IFC: k=2048, 3072, 4096, 6144, 8192 bits - FFC: between 112 and 200; ECC: 112, 128, 192, 256; IFC [SP800-56Br2]: 112, 128 | | n Random Number Generation | | Hybrid-TLS1 KEM-Hybrid-TLS2 |
| KAS_Public | Key pair component provided by the local participant, used for Diffie-Hellman | FFC: MODP2048, ffdhe2048, MODP3072, ffdhe3072, MODP4096, ffdhe4096, | Public key - PSP | Asymmetric Key Pair Generation Random Number Generation | | KAS-1 KAS-2 KAS-3 KEM-Hybrid-TLS1 KEM-Hybrid-TLS2 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | shared secret generation | MODP6144, ffdhe6144, MODP8192, ffdhe8192; ECC: B-233, K-233, P-224, B-283, K-283, P-256, B-409, K-409, P-384, B-571, K-571, P-521; IFC: k=2048, 3072, 4096, 6144, 8192 bits - FFC: between 112 and 200; ECC: 112, 128, 192, 256; IFC [SP800-56Br2]: 112, 128 | | | | |
| ML-KEM_Private | ML-KEM Public Key for key encapsulation | 13056, 19200, 25344 bits - 128, 192, 256 | Private Key - CSP | | | ML-KEM EncapDecap KEM-Hybrid-TLS1 KEM-Hybrid-TLS2 KEM-Hybrid-TLS3 |
| ML-KEM_Public | ML-KEM Private Key for key | 6400, 9472, 12544 bits | Public Key - PSP | | | ML-KEM EncapDecap KEM- |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | decapsulation | - 128, 192, 256 | | | | Hybrid-TLS1 KEM-Hybrid-TLS2 KEM-Hybrid-TLS3 |
| KD_KDK | Key Derivation key derivation key | 128 - 256 bits - 128 - 256 | Derivation Key - CSP | | | Key Derivation |
| PBKDF Password | Input provided to the PBKDF | Recommended size is greater than 10 characters for passwords and greater than 20 characters for passphrases - 112 or greater | Password - CSP | | | Key Derivation |
| KD_DKM | Key Derivation derived keying material | HMAC PRF: 160, 224, 256, 384, 512 bits - HMAC PRF: 160, 224, 256, 384, 512 | Derived Keying Material - CSP | | Key Derivation | |
| KH_Key | Keyed Hash key | CMAC: 128, 192, 256 bits; GMAC: 128, 192, 256 bits; HMAC: 160, 256, 512 bits; KMAC: 128, 256 bits - | Symmetric key - CSP | Random Number Generation Cryptographic Key Generation (CKG) | | Keyed Hash |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | | CMAC: 128, 192, 256; GMAC: 128, 192, 256; HMAC: 160, 256, 512; KMAC: 128, 256 | | | | |
| RSA_KDK | Private (KDK) cmponent of an RSA key pair used for [SP800-56Br2] RSA OAEP | 2048, 3072, 4096 and 6144 bits - 112, 128, 152, 176 | Private key - CSP | | | RSA Encapsulation and Unencapsulation |
| RSA_KEK | Public (KEK) component of an RSA key pair used for [SP800-56Br2] RSA OAEP | 2048, 3072, 4096 and 6144 bits - 112, 128, 152, 176 | Public key - PSP | | | RSA Encapsulation and Unencapsulation |
| DRBG_EI | Entropy input from an external source used for DRBG seeding | 128 - 256 bits - 128 - 256 | Entropy input - CSP | | | Random Number Generation |
| DRBG_Seed (IG D.L) | Seed generated from the entropy input for the DRBG | 128 - 256 bits - 128 - 256 | DRBG seed - CSP | | | Random Number Generation |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| DRBG_State (IG D.L) | Hash DRBG: V and C. HMAC DRBG: V and Key CTR DRBG: V and Key | Hash DRBG: 160, 224, 256, 384, 512 bits; HMAC DRBG: 160, 224, 256, 384, 512 bits; CTR DRBG: 128, 192, 256 bits - Hash DRBG: 160, 224, 256, 384, 512; HMAC DRBG: 160, 224, 256, 384, 512; CTR DRBG: 128, 192, 256 | DRBG state - CSP | Random Number Generation | | Random Number Generation |
| SC_EDK | AES key used for symmetric encryption and decryption (including use in key wrapping) | AES: 128, 192, 256 bits; AES-CCM: 128, 192, 256 bits; AES-GCM: 128, 192, 256 bits; AES XTS: 128, 256 bits - AES: 128, 192, 256; AES-CCM: 128, 192, 256; AES-GCM: 128, 192, 256; AES XTS: 128, 256 | Symmetric key - CSP | Random Number Generation Cryptographic Key Generation (CKG) Cryptographic Key Generation (CKG) - AES XTS | | Symmetric Encryption or Decryption |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| TDES_DK | TDES key used for symmetric decryption | TDES ECB: 168 bits; TDES CBC: 168 bits - TDES ECB: 112; TDES CBC: 112 | Symmetric key - CSP | | | Symmetric Encryption or Decryption |
| GCM_IV | GCM IV as per IG C.H Additional Comment 10 | 96 - 1024 bits - 96 - 1024 | Initialization Vector - PSP | Random Number Generation | | Symmetric Encryption or Decryption |
| KAS_SS | Shared Secret (Z) produced by the Shared Secret Computation (SSC); intended for use as input to an approved KDF or utilized to be concatenated witha MK-KEM shared secret in a hybrid KEM | FFC: FB, FC, MODP2048, ffdhe2048, MODP3072, ffdhe3072, MODP4096, ffdhe4096, MODP6144, ffdhe6144, MODP8192, ffdhe 8192 ECC: B-233, K-233, P-224, B-283, K-283, P-256, B-409, K-409, P-384, B-571, K-571, P-521, IFC: k=2048, 3072, 4096, 6144, 8192 bits - FFC: between | Shared secret - CSP | | KAS-1 KAS-2 KAS-3 | KEM-Hybrid-TLS1 KEM-Hybrid-TLS2 |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | | 112 and 200 ECC: 112, 128, 192, 256 IFC: 112, 128 | | | | |
| KEM_SS | The shared secret key output by the ML-KEM component of the Hybrid KEM | 256 bits - 128, 192, 256 | Shared secret key - CSP | | ML-KEM EncapDecap | KEM-Hybrid-TLS1 KEM-Hybrid-TLS2 KEM-Hybrid-TLS3 |
| KASKEM_SS | Concatenated shared secret obtained by concatenating KAS_SS and KEM_SS as part of a Hybrid KEM | 512 bits - 192, 256 | Concatenated Shared secret - CSP | | KEM-Hybrid-TLS1 KEM-Hybrid-TLS2 KEM-Hybrid-TLS3 | |
| ML-DSA_Seed | ML-DSA random seed that is given as input to ML_DSA keyGen internal to produce the ML-DSA Private (DS_SGK) and ML-DSA public keys | 256 bits - 256 | ML-DSA Seed - CSP | Random Number Generation | | PQC Key Pair Generation |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | (DG_SVK) | | | | | |
| SLH_DSA_SK SEED | Secret seed used to derive the SLH-DSA private key material | 128, 192, 256 bits - 128, 192, 256 | SLH_DSA seed - CSP | Random Number Generation | | SLH-DSA Signature Generation and Verification |
| SLH_DSA_SK PRF | Secret PRF key used to generate per-signature randomness | 128, 192, 256 bits - 128, 192, 256 | Symmetric key - CSP | Random Number Generation | | SLH-DSA Signature Generation and Verification |
| SLH_DSA_PK SEED | Public seed used to parameterize the hash function computations during SLH-DSA key generation, signature generation, and signature verification | 128, 192, 256 bits - 128, 192, 256 | SLH_DSA seed - PSP | Random Number Generation | | SLH-DSA Signature Generation and Verification |
| ML_KEM_SEED | 64-byte secret seed containing the values (d, z) used to generate an ML- | 512 bits - 256 | ML_KEM seed - CSP | Random Number Generation | | PQC Key Pair Generation |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Description | Size - Strength | Type - Category | Generated By | Established By | Used By |
|---|---|---|---|---|---|---|
| | KEM key pair | | | | | |

Table 18: SSP Table 1

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| DS_SGK | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DS_SVK:Paired With |
| DS_SVK | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DS_SGK:Paired With |
| GKP_Private | CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | GKP_Public:Paired With |
| GKP_Public | CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | GKP_Private:Paired With |
| KAS_Private | CALL STACK (API) INPUT | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use | KAS_Public:Paired With |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| | PARAMETERS | | | Teardown Restarting the general-purpose computer | |
| KAS_Public | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | KAS_Private:Paired With |
| ML-KEM_Private | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | ML-KEM_Public:Paired With |
| ML-KEM_Public | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | ML-KEM_Private:Paired With |
| KD_KDK | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | KD_DKM:Derives |
| PBKDF Password | CALL STACK (API) INPUT | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown | KD_DKM:Derives |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| | PARAMETERS | | | Restarting the general-purpose computer | |
| KD_DKM | CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | KD_KDK:Derived From PBKDF Password:Derived From |
| KH_Key | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | |
| RSA_KDK | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | RSA_KEK:Paired With |
| RSA_KEK | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | RSA_KDK:Paired With |
| DRBG_EI | CALL STACK (API) INPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the | DRBG_Seed (IG D.L):Used to derive |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| | | | | general-purpose computer | |
| DRBG_Seed (IG D.L) | | RAM:Plain text | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DRBG_EI:Derived From |
| DRBG_State (IG D.L) | | RAM:Plain text | Until power-cycling of the underlying host platform | Teardown Restarting the general-purpose computer | DRBG_Seed (IG D.L):Derived From |
| SC_EDK | CALL STACK (API) INPUT PARAMETERS CALL STACK (API) OUTPUT PARAMETERS | RAM:Plain text | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | |
| TDES_DK | CALL STACK (API) INPUT PARAMETERS | RAM:Plain text | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | |
| GCM_IV | CALL STACK (API) INPUT | RAM:Plain text | Cleared after use | OPENSSL_cleanse Cleared after use Teardown | |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| | PARAMETERS | | | Restarting the general-purpose computer | |
| KAS_SS | | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | KAS_Private:Established using KAS_Public:Established using |
| KEM_SS | | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | ML-KEM_Private:Generated by ML-KEM Decapsulation using the ML-KEM Private Key ML-KEM_Public:Generated by ML-KEM Encapsulation using the ML-KEM Public Key |
| KASKEM_SS | CALL STACK (API) INPUT PARAMETERS CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | KEM_SS:Generated by concatenatin KAS_SS and KEM_SS KAS_SS:Generated by concatenatin KAS_SS and KEM_SS |
| ML-DSA_Seed | CALL STACK (API) INPUT PARAMETERS CALL STACK (API) | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DS_SGK:Derived From DS_SVK:Derived From |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| | OUTPUT PARAMETERS | | | | |
| SLH_DSA_SKSEED | CALL STACK (API) INPUT PARAMETERS CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DS_SGK:Component of SLH-DSA Private key (DS_SGK) |
| SLH_DSA_SKPRF | CALL STACK (API) INPUT PARAMETERS CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DS_SGK:Component of SLH-DSA Private key (DS_SGK) |
| SLH_DSA_PKSEED | CALL STACK (API) INPUT PARAMETERS CALL STACK (API) OUTPUT PARAMETERS | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general-purpose computer | DS_SGK:Component of SLH-DSA Private key (DS_SGK) DS_SVK:Component of SLH-DSA Private key (DS_SVK) |
| ML_KEM_SEED | CALL STACK (API) INPUT PARAMETERS CALL | RAM:Plaintext | Cleared after use | OPENSSL_cleanse Cleared after use Teardown Restarting the general- | ML-KEM_Private:Derived From ML-KEM_Public:Derived From |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Name | Input - Output | Storage | Storage Duration | Zeroization | Related SSPs |
|---|---|---|---|---|---|
| | STACK (API) OUTPUT PARAMETERS | | | purpose computer | |

Table 19: SSP Table 2

All SSPs used by the Module are described in this section, arranged for consistency with Table 12; '--' indicates the cell is intentionally empty, not applicable, or not relevant.

Keys used for CASTs and the temporary value used in the integrity test are not SSPs; however, the latter is deleted after use as required by AS05.10. Equivalent strength is given for each key or algorithm type (as some algorithms do not use or produce keys).

The Module maintains only the DRBG CSPs used for key generation as persistent CSPs; these are used exclusively for approved services.

DRBG outputs are used internally to the Module for asymmetric key pair generation and used by calling applications to generate a random value (potentially for use as a symmetric key).

The Module:

- Produces random values in accordance with [SP800-133r2] Section 4, in that the DRBG output is provided directly as the random output.
- SSPs used with symmetric key algorithms are provided by the calling application.
- Produces asymmetric keys in accordance with [SP800-133r2] Section 5, in that all asymmetric keys generated by the module (the Key management service) provide the output of the approved key generation algorithm with no post-processing or manipulation of the generated key pairs. As noted in the previous item, random values used in the asymmetric key generation algorithms are direct outputs of the DRBG. Keys produced by the module use an internal Counter DRBG for which the minimum key size and equivalent security strength is 128 bits.
- Supports direct generation of symmetric keys in accordance with [SP800-133r2] Section 6.1 and symmetric key derivation in accordance with [SP800-133r2] Section 6.2, using the approved and CAVP listed KDF algorithms. AES-KTS keys are generated in accordance with [SP800-133r2] Section 6.3.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 10 Self-Tests

## 10.1 Pre-Operational Self-Tests

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details |
|---|---|---|---|---|---|
| HMAC-SHA2-256 (A9077) | Key Length: 256 bits | Message Authentication over the complete module file image | SW/FW Integrity | Success: All self-tests passed (as expected) | MAC (HMAC-SHA2-256, A9077) |

Table 20: Pre-Operational Self-Tests

The module is complaint with FIPS 140-3 IG 10.2.A in that it performs a self-test, a Known Answer Test (KAT) for the HMAC-SHA2-256 algorithm.

## 10.2 Conditional Self-Tests

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| AES-ECB (A9077) | Key Length: 128 bits | KAT | CAST | FIPS_OK | Decrypt | On reloading the module |
| AES-GCM Encrypt | Key Length: 256 bits | KAT | CAST | FIPS_OK | Encrypt | On reloading the module |
| AES-GCM Decrypt | Key Length: 256 bits | KAT | CAST | FIPS_OK | Decrypt | On reloading the module |
| TDES-ECB (A9077) | Key Length: 168 bits | KAT | CAST | FIPS_OK | Decrypt | On reloading the module |
| Counter DRBG (A9077) | AES-CTR (128 bits) with derivation function | KAT | CAST | FIPS_OK | Generate, Reseed, Instantiate functions | On reloading the module |
| DSA SigVer (FIPS186-4) (A9077) | Modulus: 2048 bits; Hash: SHA2-256 | KAT | CAST | FIPS_OK | Verify | On reloading the module |
| Hash DRBG (A9077) | PRF: SHA2-256 | KAT | CAST | FIPS_OK | Generate, Reseed, Instantiate functions | On reloading the module |
| HMAC DRBG (A9077) | PRF: HMAC-SHA-1 | KAT | CAST | FIPS_OK | Generate, Reseed, Instantiate functions | On reloading the module |
| HMAC-SHA2-256 (A9077) | PRF: SHA2-256 | KAT | CAST | FIPS_OK | HMAC tag Generation | Performed prior to the software integrity test |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| KAS-ECC-SSC Sp800-56Ar3 (A9077) | Scheme: Ephemeral Unified, Curve: P-256 | KAT | CAST | FIPS_OK | Key Agreement - Shared Secret Computation | On reloading the module |
| KAS-FFC-SSC Sp800-56Ar3 (A9077) | Scheme: dhEphem; Modulus: L = 2048 bits, N = 256 bits | KAT | CAST | FIPS_OK | Key Agreement - Shared Secret Computation | On reloading the module |
| Safe Primes Key Generation (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Key Agreement (KAS FFC-SSC) |
| KAS-IFC-SSC (A9077) | Schemes: Basic, CRT, Modulus: L = 2048 bits | KAT | CAST | FIPS_OK | Key Agreement - Shared Secret Computation | On reloading the module |
| KDF SP800-108 (A9077) | Mode: Counter, PRF: HMAC-SHA2-256 | KAT | CAST | FIPS_OK | Counter Mode (HMAC-SHA2-256) | On reloading the module |
| KDA OneStep SP800-56Cr2 (A9077) | Auxiliary Function, H = SHA2-256 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| KDA TwoStep SP800-56Cr2 (A9077) | Auxiliary Function, H = HMAC-SHA2-256 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| KTS-IFC (A9077) | Schemes: Basic, CRT for decrypt, Modulus: L = 2048 bits | KAT | CAST | FIPS_OK | Encrypt and Decrypt | On reloading the module |
| PBKDF (A9077) | Derivation of the Master Key (MK), PRF: SHA2-256 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| RSA SigGen (FIPS186-5) (A9077) | Scheme: PKCS#1, Modulus: L = 2048, Hash: SHA2-256 | KAT | CAST | FIPS_OK | Sign | On reloading the module |
| RSA SigVer (FIPS186-5) (A9077) | Scheme: PKCS#1, Modulus: L = 2048, Hash: SHA2-256 | KAT | CAST | FIPS_OK | Verify | On reloading the module |
| SHA-1 (A9077) | SHA-1 | KAT | CAST | FIPS_OK | Hash | On reloading the module |
| SHA2-512 (A9077) | SHA2-512 | KAT | CAST | FIPS_OK | Hash | On reloading the module |
| SHA3-256 (A9077) | SHA3-256 | KAT | CAST | FIPS_OK | Hash | On reloading the module |
| KDF ANS 9.42 (A9077) | PRFs: AES KW (128 bits), SHA-1 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| KDF ANS 9.63 (A9077) | PRF: SHA2-256 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| KDF SSH (A9077) | PRF: SHA-1 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| TLS v1.2 KDF RFC7627 (A9077) | PRF: SHA2-256 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| TLS v1.3 KDF (A9077) | PRF: SHA2-256 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| KDF KMAC Sp800-108r1 (A9077) | KMAC128 | KAT | CAST | FIPS_OK | Key Derivation | On reloading the module |
| RSA KeyGen (FIPS186-5) (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Key Transport (KTS IFC)/Key Agreement (KAS IFC)/Signature Generation/Signature Verification |
| ECDSA KeyGen (FIPS186-5) (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Key Agreement (KAS ECC)/Signature Generation/Signature Verification |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| ECDSA SigGen (FIPS186-5) (A9077) | Curve: P-224, B-233; Hash: SHA2-256 | KAT | CAST | FIPS_OK | Sign | On reloading the module |
| ECDSA SigVer (FIPS186-5) (A9077) | Curve: P-224, B-233; Hash: SHA2-256 | KAT | CAST | FIPS_OK | Verify | On reloading the module |
| EDDSA KeyGen (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Signature Generation/Signature Verification |
| EDDSA SigGen (A9077) | Curve: Edwards25519, Edwards448 | KAT | CAST | FIPS_OK | Sign | On reloading the module |
| EDDSA SigVer (A9077) | Curve: Edwards25519, Edwards448 | KAT | CAST | FIPS_OK | Verify | On reloading the module |
| ML-KEM KeyGen (A9077) | Parameter set; ML-KEM-512 | KAT | CAST | FIPS_OK | Key Generation | On reloading the module |
| ML-KEM EncapDecap (A9077) | Parameter set; ML-KEM-512. Key Decapsulation for the rejection and normal path as indicated in IG 10.3.A Resolution 14 | KAT | CAST | FIPS_OK | Key Encapsulation and Decapsulation | On reloading the module |
| ML-KEM KeyGen PCT (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Key Encapsulation/Decapsulation |
| ML-DSA KeyGen (A9077) | Parameter set; ML-DSA-65 | KAT | CAST | FIPS_OK | Key Generation | On reloading the module |
| ML-DSA SigGen (A9077) | Parameter set; ML-DSA-65 | KAT | CAST | FIPS_OK | Signature generation covering the applicable | On reloading the module |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Properties | Test Method | Test Type | Indicator | Details | Conditions |
|---|---|---|---|---|---|---|
| | | | | | rejection sampling loop paths as indicated in IG 10.3.A. Resolution 15 | |
| ML-DSA SigVer (A9077) | Parameter set; ML-DSA-65 | KAT | CAST | FIPS_OK | Signature Verification | On reloading the module |
| ML-DSA KeyGen PCT (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Signature Generation/Signature Verification |
| SLH-DSA KeyGen (A9077) | Parameter set; SLHL-DSA-SHA2-128f | KAT | CAST | FIPS_OK | Key Generation | On reloading the module |
| SLH-DSA SigGen (A9077) | Parameter set; SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128f | KAT | CAST | FIPS_OK | Signature Generation | On reloading the module |
| SLH-DSA SigVer (A9077) | Parameter set; SLH-DSA-SHA2-128f, SLH-DSA-SHAKE-128f | KAT | CAST | FIPS_OK | Signature Verification | On reloading the module |
| SLH-DSA KeyGen PCT (A9077) | Performed post key generation | PCT | PCT | FIPS_OK | Key Generation | On generating keys for Signature Generation/Signature Verification |

Table 21: Conditional Self-Tests

Each time the Module is powered up it tests that the cryptographic algorithms still operate correctly and that sensitive data has not been damaged. On instantiation, the Module performs the pre-operational self-tests and all CASTs listed above. All KATs must complete successfully prior to any other use of cryptography by the Module.

The module implements PCT on key import for RSA, ECDSA, and EDDSA for signature generation and verification, and for ECDSA and Safe primes for KAS ECC and KAS FFC as per FIPS 140-3 IG 10.3.A Additional Comment 1.

## 10.3 Periodic Self-Test Information

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Method | Test Type | Period | Periodic Method |
|---|---|---|---|---|
| HMAC-SHA2-256 (A9077) | Message Authentication over the complete module file image | SW/FW Integrity | On Demand | Manually by reloading the module or calling the fips_self_test function |

Table 22: Pre-Operational Periodic Information

| Algorithm or Test | Test Method | Test Type | Period | Periodic Method |
|---|---|---|---|---|
| AES-ECB (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| AES-GCM Encrypt | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| AES-GCM Decrypt | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| TDES-ECB (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| Counter DRBG (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| DSA SigVer (FIPS186-4) (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| Hash DRBG (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| HMAC DRBG (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| HMAC-SHA2-256 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KAS-ECC-SSC Sp800-56Ar3 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KAS-FFC-SSC Sp800-56Ar3 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| Safe Primes Key Generation (A9077) | PCT | PCT | On Demand | On generation of keys |
| KAS-IFC-SSC (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Method | Test Type | Period | Periodic Method |
|---|---|---|---|---|
| KDF SP800-108 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KDA OneStep SP800-56Cr2 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KDA TwoStep SP800-56Cr2 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KTS-IFC (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| PBKDF (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| RSA SigGen (FIPS186-5) (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| RSA SigVer (FIPS186-5) (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| SHA-1 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| SHA2-512 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| SHA3-256 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KDF ANS 9.42 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KDF ANS 9.63 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KDF SSH (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| TLS v1.2 KDF RFC7627 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| TLS v1.3 KDF (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| KDF KMAC Sp800-108r1 (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

| Algorithm or Test | Test Method | Test Type | Period | Periodic Method |
|---|---|---|---|---|
| RSA KeyGen (FIPS186-5) (A9077) | PCT | PCT | On Demand | On generation of keys |
| ECDSA KeyGen (FIPS186-5) (A9077) | PCT | PCT | On Demand | On generation of keys |
| ECDSA SigGen (FIPS186-5) (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ECDSA SigVer (FIPS186-5) (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| EDDSA KeyGen (A9077) | PCT | PCT | On Demand | On generation of keys |
| EDDSA SigGen (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| EDDSA SigVer (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ML-KEM KeyGen (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ML-KEM EncapDecap (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ML-KEM KeyGen PCT (A9077) | PCT | PCT | On Demand | On generation of keys |
| ML-DSA KeyGen (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ML-DSA SigGen (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ML-DSA SigVer (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| ML-DSA KeyGen PCT (A9077) | PCT | PCT | On Demand | On generation of keys |
| SLH-DSA KeyGen (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| SLH-DSA SigGen (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| SLH-DSA SigVer (A9077) | KAT | CAST | On Demand | Manually by reloading the module or calling the fips_self_test function |
| SLH-DSA KeyGen PCT (A9077) | PCT | PCT | On Demand | On generation of keys |

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

Table 23: Conditional Periodic Information

## 10.4 Error States

| Name | Description | Conditions | Recovery Method | Indicator |
|---|---|---|---|---|
| ERROR STATE | The error state is persistent and no services are available; All attempts to use the Module's services result in the return of a non-zero error code, PROV_R_FIPS_MODULE_IN_ERROR_STATE | If one of the KATs or if the Software Integrity Test fails, the Module enters the self-test failure error state | To recover from an error state, reload the Module into memory | PROV_R_FIPS_MODULE_IN_ERROR_STATE |

Table 24: Error States

## 10.5 Operator Initiation of Self-Tests

The operator can reload the module or the *fips_self_test* function (inclusive of software integrity verification) can also be called on demand, fulfilling AS05.11.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 11 Life-Cycle Assurance

## 11.1 Installation, Initialization, and Startup Procedures

The Module is provided to vendors who integrate it into their product, typically in a manufacturing environment, and is not provided directly to US or Canadian Federal agencies. Adherence to the instructions in this document maintains security throughout the distribution, build, installation and configuration processes.

An authorized Cryptographic Officer is required to perform these steps on each platform where it is intended to be used. The config file output contains information about the Module (such as the self-test status and the Module checksum) and must not be manually modified without using the openssl fipsinstall command.

### Crypto Officer Guidance

a. Installation and Usage Guidance

The Module is installed as part of the OpenSSL 3.5.9 library. The source distribution package is located at https://www.openssl.org/source/openssl-3.5.9.tar.gz.

The OpenSSL FIPS Provider can be installed on the Tested Configurations listed in Table 3 by performing the following steps:

1. Build and install OpenSSL 3.5.9 to the default location:

   The OpenSSL FIPS Provider (i.e., the Module) does not get built and installed automatically. To install the Module automatically during the normal OpenSSL 3.5.9 installation process it must be enabled by configuring OpenSSL using the 'enable-fips' option.

   FreeBSD/Linux/macOS:

   $ ./Configure enable-fips

   $ make

   $ make install

   Windows:

   $ perl Configure enable-fips

   $ nmake

   $ nmake install

   The 'install_fips' make target can also be invoked explicitly to install the FIPS Provider independently, without installing the rest of OpenSSL:

   $ make install_fips

   Note: The instructions for building and installing OpenSSL 3.5.9 on other platforms can be found in the platform-specific guidance provided in INSTALL.md and README-FIPS.md in the OpenSSL 3.5.9 distribution package. Please see Appendix A for further information on porting the Module to platforms apart from the Tested Configurations in Table 3.

2. Verify the version:

   $ openssl version -v

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

The Installation of the OpenSSL FIPS Provider that occurs as a result of Step 1 above ensures that the shared library and the configuration file containing information about the Module (e.g., the Module checksum) is copied to its installed location.

To install the configuration file to a non-default location, this can be achieved by running the 'fipsinstall' command line application manually:

$ openssl fipsinstall -pedantic

Please see fipsinstall.html /docs/man3.5.9/man1/openssl-fipsinstall.html for options supported for the 'openssl fipsinstall' command.

Note: The software integrity check (per Section 5 of this document) is performed using HMAC-SHA2-256 on the Module file to validate that the Module has not been modified. The integrity value is compared to a value written to the config file during installation.

b. CVEs

The publication of a CVE does not require immediate re-validation or maintenance in the CMVP process. The module may be updated in the field as needed depending on the severity or consequences of the CVE. The Module will be kept up to date with re-validation and maintenance as required, generally bundling fixes for known CVEs in a next release.

The OpenSSL organization maintains a Vulnerabilities page which describes known vulnerabilities and potential resolution. These are reported to the NVD, where they are independently assessed. The OpenSSL group publishes fixes for these vulnerabilities according to their triage process.

c. Miscellaneous

The module performs run-time checks related to enforcement of security parameters such as the minimum-security strength of keys, valid key sizes, and usage of approved curves. These checks shall not be disabled (by using OPENSSL_NO_FIPS_SECURITYCHECKS or any other method).

Validation of domain parameters prior to generating keys using functions provided by the module is the responsibility of the Cryptographic Officer and not enforced by the module itself.

## 11.2 Administrator Guidance

No additional guidance applies for the operation of the module apart from that specified in Sections 2, 3 of this document and other subsections under this section.

## 11.3 Non-Administrator Guidance

No additional guidance applies for the operation of the module apart from that specified in Sections 2, 3 of this document and other subsections under this section.

## 11.4 Design and Rules

No additional rules apply for the operation of the module apart from those specified in the remainder of this section and Section 2.4 of this document.

## 11.5 Maintenance Requirements

No maintenance requirements apply for operation of the module in the Approved/non-Approved modes as defined above.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

## 11.6 End of Life

### Module Sanitization and Destruction

Sanitization is defined in [ISO19790] as "... the process of removing sensitive information (e.g. SSPs, user data, etc.) from the module, so that it may either be distributed to other operators or disposed."

The Module itself does not manage persistent SSPs, authentication data or any user data. The Module may be securely sanitized by deletion of the folder in which the Module was located.

There are no additional procedures required for secure destruction of the Module.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.

# 12 Mitigation of Other Attacks

## 12.1 Attack List

The Module implements mitigations for some types of attacks using constant-time implementations and blinding.

Constant-time implementations protect cryptographic implementations in the Module against timing analysis since such attacks exploit differences in execution time depending on the cryptographic operation, and constant-time implementations ensure that the variations in execution time cannot be traced back to the key, CSP or secret data.

Numeric blinding protects the RSA, DSA and ECDSA algorithms from timing attacks. These algorithms are vulnerable to such attacks since attackers can measure the time of signature operations or RSA decryption. To mitigate this, the Module generates a random blinding factor which is provided as an input to the decryption/signature operation and is discarded once the operation has completed and resulted in an output. This makes it difficult for attackers to attempt timing attacks on such operations without the knowledge of the blinding factor, and therefore the execution time cannot be correlated to the RSA/DSA/ECDSA key.

The OpenSSL Corporation. 2026
This document may be reproduced and distributed only in its original entirety without revision.