Cryptographic Module for BIG-IP®

Module version 12.1.2 HF1

FIPS 140-2 Non-Proprietary Security Policy

Document Version 1.0

Last update: 2017-05-05

Prepared by:

atsec information security corporation

9130 Jollyville Road, Suite 260

Austin, TX 78759

www.atsec.com

Table of Contents

Copyrights and Trademarks

F5® and BIG-IP® are registered trademarks of F5 Networks.

VMware ESXi™ is a trademark of VMware®, Inc.

Intel® Xeon® is a registered trademark of Intel® Corporation.

1. Introduction

This document is the non-proprietary FIPS 140-2 Security Policy of Cryptographic Module for BIG-IP with version 12.1.2 HF1. It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-2 (Federal Information Processing Standards Publication 140-2) for a Security Level 1 module.

2. Cryptographic Module Specification

The following section describes the cryptographic module and how it conforms to the FIPS 140-2 specification in each of the required areas.

2.1. Module Overview

The Cryptographic Module for BIG-IP (hereafter referred to as “the module”) is a software library implementing general purpose cryptographic algorithms.

The module provides cryptographic services to applications through an Application Program Interface (API). The module also interacts with the underlying operating system via system calls.

The software block diagram below shows the module, its interfaces with the operational environment and the delimitation of its logical boundary:

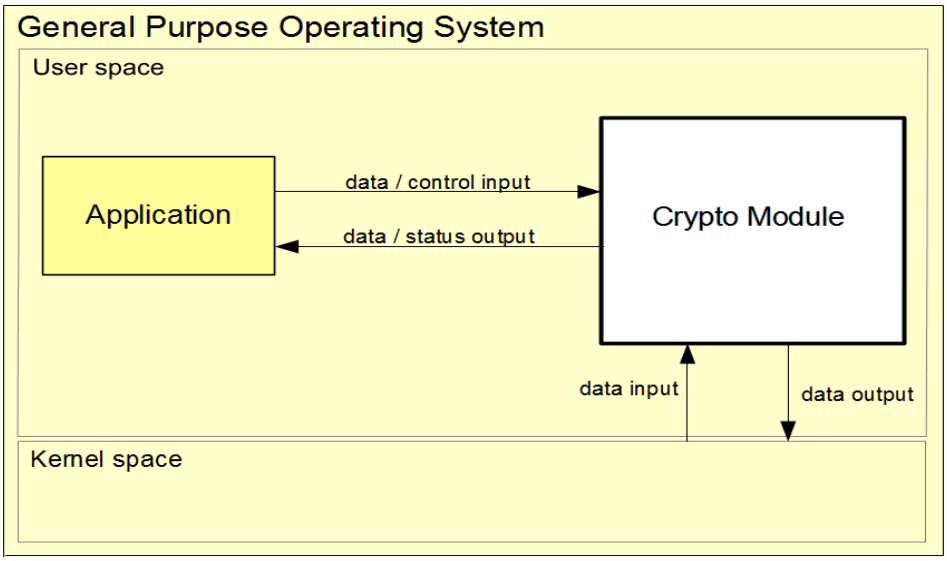

Figure 1 - Software Block Diagram

The module is implemented as a shared library. The cryptographic logical boundary consists of a shared library and the integrity check file used for integrity tests.

Filename	Purpose
libcrypto.so	The binary for cryptographic implementations.
.libcrypto.so.hmac	The integrity check file for libcrypto.so binary.

Table 1 - Cryptographic Module Components

The module is aimed to run on a general purpose computer; the physical boundary is the surface of the case of the target platform, as shown with dotted lines in the diagram below:

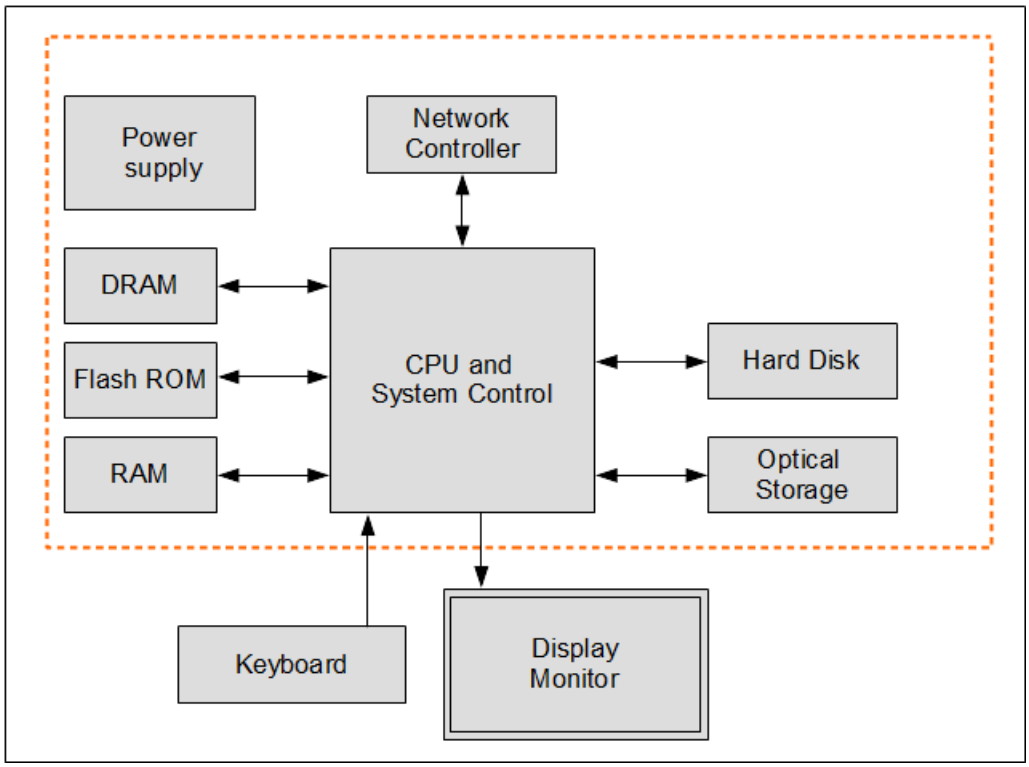

Figure 2 - Cryptographic Module Physical Boundary

2.2. FIPS 140-2 Validation

The module is a software-only, multi-chip standalone cryptographic module validated at overall security level 1. The table below shows the security level claimed for each of the eleven sections that comprise the FIPS 140-2 standard:

FIPS 140-2 Section		Security Level
1	Cryptographic Module Specification	1
2	Cryptographic Module Ports and Interfaces	1
3	Roles, Services and Authentication	1
4	Finite State Model	1
5	Physical Security	N/A
6	Operational Environment	1
7	Cryptographic Key Management	1
8	EMI/EMC	1
9	Self-Tests	1
10	Design Assurance	1
11	Mitigation of Other Attacks	N/A
Overall Level		1

Table 2 - Security Levels

The module has been tested on the following multichip standalone platform with the corresponding module variant and configuration options:

Hardware	Processor	PAA function	Operating System
VMware ESXi™ 5.5 hypervisor running on HP ProLiant BL490c	Intel® Xeon® X5650	with and without AES-NI	BIG-IP 12.1.2 HF1[1]

Table 3 - Tested Platforms

2.3. Modes of operation

The module supports two modes of operation:

- in "FIPS mode" (the FIPS Approved mode of operation) only approved or allowed security functions with sufficient security strength can be used as specified in table 5.
- in "non-FIPS mode" (the non-Approved mode of operation) only non-approved security functions can be used.

The module enters FIPS mode after power-up tests succeed. Once the module is operational, the mode of operation is implicitly assumed depending on the security function invoked and the

[1] BIG-IP consists of a Linux based operating system customized for performance that runs directly on the hardware or in virtual environment.

security strength of the cryptographic keys. Critical Security Parameters (CSPs) used or stored in FIPS mode are not used in non-FIPS mode, and vice versa.

3. Cryptographic Module Ports and Interfaces

As a software-only module, the module does not have physical ports. For the purpose of the FIPS 140-2 validation, the physical ports are interpreted to be the physical ports of the hardware platform on which it runs.

The logical interfaces are the API through which applications request services. The following table summarizes the four logical interfaces:

Logical Interface	Description
Data Input	API input parameters for data.
Data Output	API output parameters for data.
Control Input	API function calls for control.
Status Output	API return codes, error messages.

Table 4 - Ports and Interfaces

The Data Input interface consists of the input parameters of the API functions. The Data Output interface consists of the output parameters of the API functions. The Control Input interface consists of the API function calls used to control the behavior of the module. The Status Output interface includes the return values of the API functions and error messages.

4. Roles, Services and Authentication

4.1. Roles

The module supports the following roles:

- **User role**: performs all services (in both FIPS mode and non-FIPS mode of operation), except module initialization.
- **Crypto Officer role**: performs module initialization.

The User and Crypto Officer roles are implicitly assumed by the entity accessing the module services.

4.2. Services

The module provides services to users that assume one of the available roles. All services are described in detail in the user documentation.

The following table lists the Approved services and the non-Approved but allowed services in FIPS mode of operation, the roles that can request the service, the algorithms involved with their corresponding CAVS certificate numbers (if applicable), the CSPs involved and how they are accessed:

Service	Algorithms	CAVS	Role	CSP	Access
AES encryption and decryption	AES-ECB, AES-CBC, AES-GCM with AES-NI implementation	Cert# 4436	User	128/192/256-bit AES key	Read
	AES-ECB, AES-CBC, AES-GCM with assembler implementation	Cert# 4437	User		
Random Number Generation	NIST SP800-90A CTR_DRBG with AES-256 using AES-NI	Cert# 1435	User	Entropy input string, V and Key values	Read, Write
	NIST SP800-90A CTR_DRBG with AES-256 assembler	Cert# 1436			
	NDRNG used to seed module's DRBG. Allowed in FIPS mode	N/A			Read
RSA key pair generation	FIPS186-4 Appendix B.3.3 RSA key generation	Cert# 2418	User	RSA public and private key pair with 2048/3072-bit modulus size	Write
RSA signature generation	PKCS#1 v1.5 RSA signature generation with SHA-256 and SHA-384			RSA private key with 2048/3072-bit modulus size	Read
RSA signature verification	PKCS#1 v1.5 RSA signature verification with SHA-256 and SHA-384			RSA public key with 2048/3072-bitmodulus size	Read

Service	Algorithms	CAVS	Role	CSP	Access
ECDSA key pair generation / ECDH key pair generation	FIPS186-4 Appendix B.4.2 ECC key pair generation	Cert# 1076	User	ECDSA/ECDH public/private key pair for P-256 and P-384 curves	Write
ECDSA key verification	FIPS186-4 Public Key Validation (PKV)			ECDSA public key for P-256 and P-384 curves	Read
ECDSA signature generation	ECDSA signature generation with SHA-256 and SHA-384			ECDSA private key according to P-256 and P-384 curves	Read
ECDSA signature verification	ECDSA signature verification with SHA-256 and SHA-384			ECDSA public key according to P-256 and P-384.	Read
EC Diffie-Hellman key agreement without KDF (shared secret computation)	SP800-56A KAS ECC except KDF, Scheme: Ephemeral Unified, Section 5.7.1.2 ECC CDH Primitive	CVL Cert# 1144	User	EC Diffie-Hellman public and private Key with P-256 and P-384 curves	Read, Write
Message digest	SHA-1 with SSSE3 implementation	Cert# 3655	User	n/a	n/a
	SHA-1, SHA-256, SHA-384 with assembler implementation	Cert# 3656			
Message authentication	HMAC-SHA-1 with SSSE3 implementation	Cert# 2948	User	At least 112-bit HMAC key	Read
	HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384 with assembler implementation	Cert# 2949			
Show Status	n/a	n/a	User	n/a	
Self-Tests	n/a	n/a	User	HMAC-SHA-256 key for module integrity test	Read
Zeroization	n/a	n/a	User	All aforementioned Keys/CSPs	Zeroize
Module initialization	n/a	n/a	CO	n/a	n/a

Table 5 - Services in FIPS mode of operation

The following table lists the services only available in non-FIPS mode of operation.

Service	Role	Usage/Notes
AES encryption and decryption	User	With OFB, CFB, CTR, XTS, CCM, KW modes

Service	Role	Usage/Notes
Message digest	User	SHA-224, SHA-512, MD4, MD5, MDC2, RIPEMD, Whirlpool
Message authentication	User	HMAC-SHA224, HMAC-SHA512, CMAC with AES, CMAC with Triple-DES
Key generation	User	RSA with key sizes other than 2048 and 3072 bit.
		ECDSA/ECDH with public/private key pair for curves other than P-256 and P-384
RSA signature generation and verification	User	Using PKCS #1 v1.5 scheme with key sizes other than 2048 and 3072 bit.
	User	Using PSS, X9.31 schemes
	User	Using PKCS #1 v1.5 scheme with SHA-1, SHA-224 and SHA-512
ECDSA signature generation & verification	User	Using curves other than P-256 and P-384
		Using curves P-256 and P-384 with SHA-1, SHA-224 and SHA-512
RSA encrypt/decrypt	User	With modulus sizes up to 16384 bits
DSA domain parameter generation, domain parameter verification, key pair generation, signature generation and verification	User	With all key and SHA sizes
Random Number Generation	User	Using HMAC_DRBG and Hash_DRBG for all SHA sizes
	User	CTR_DRBG with AES-128 or AES-192
	User	ANSI X9.31 RNG
Key Agreement	User	Diffie-Hellman Key agreement without KDF, J-PAKE, SRP
		EC Diffie-Hellman with curves other than P-256 and P-384 without KDF
Encryption and Decryption	User	Blowfish, Camellia, CAST, DES, IDEA, RC2, RC4, SEED, Triple-DES

Table 6 - Services in non-FIPS mode of operation

4.3. Operator Authentication

The module does not implement authentication. The role is implicitly assumed based on the service requested.

5. Physical Security

The module is comprised of software only and therefore this security policy does not make any claims on physical security.

6. Operational Environment

6.1. Applicability

The module operates in a modifiable operational environment per FIPS 140-2 level 1 specifications. The module runs on a BIG-IP 12.1.2 HF1 operating system executing on the hardware and hypervisor specified in section 2.2.

6.2. Policy

The operating system is restricted to a single operator; concurrent operators are explicitly excluded.

The application that requests cryptographic services is the single user of the module.

7. Cryptographic Key Management

The following table summarizes the CSPs that are used by the cryptographic services implemented in the module:

Name	Generation	Storage	Zeroization
AES Key	N/A. Input as API parameter	RAM	Zeroized by FIPS_cipher_ctx_cleanup()
HMAC Key	N/A. Input as API parameter	RAM	Zeroized by HMAC_CTX_cleanup()
RSA Key Pair	Generated using FIPS 186-4 Key generation method, and the random value used in the key generation is generated using SP800-90A DRBG.	RAM	Zeroized by FIPS_rsa_free()
ECDSA Key Pair		RAM	Zeroized by EC_KEY_free()
EC Diffie-Hellman Key pair	Generated using 186-4 Key generation method and the random value used in the key generation is generated using SP800-90A DRBG	RAM	Zeroized by EC_KEY_free()
DRBG entropy input string	Obtained from NDRNG.	RAM	Zeroized by FIPS_drbg_free()
DRBG V and Key values	Derived from entropy string as defined by [SP800-90A]	RAM	Zeroized by FIPS_drbg_free ()

Table 7 - Life cycle of CSPs

The following sections describe how CSPs, in particular cryptographic keys, are managed during its life cycle.

7.1. Key Generation

For generating RSA and ECDSA/ECDH keys, the module implements asymmetric key generation services compliant with [FIPS186-4], and using DRBG compliant with [SP800-90A]. A seed (i.e. the random value) used in asymmetric key generation is obtained from [SP800-90A] DRBG. The module does not implement symmetric key generation. In accordance with FIPS 140-2 IG D.12, the cryptographic module performs Cryptographic Key Generation (CKG) for asymmetric keys as per SP800-133 (vendor affirmed).

7.2. Key Establishment

The module implements key agreement scheme based on SP800-56A without KDF. The module provides EC Diffie-Hellman shared secret computation with curves P-256 or P-384, providing 128 or 192 bit equivalent security strength, respectively.

7.3. Key Entry / Output

The module does not support manual key entry or intermediate key generation key output. In addition, the module does not produce key output outside its physical boundary. The keys can be entered or output from the module in plaintext form via API parameters, to and from the calling application only. This is allowed by FIPS 140-2 IG 7.7 Table 1, according to the “CM Software to/from App Software via GPC INT Path” entry which refers to keys communicated within the physical boundary of the GPC.

7.4. Key / CSP Storage

Public and private keys are provided to the module by the calling process, and are destroyed when released by the appropriate API function calls.

The module does not perform persistent storage of keys. The only exception is the HMAC-SHA-256 key used for integrity test, which is stored in the module and relies on the operating system for protection.

7.5. Key / CSP Zeroization

The memory occupied by keys is allocated by regular memory allocation operating system calls. The application is responsible for calling the appropriate destruction functions provided in the module's API. The destruction functions overwrite the memory occupied by keys with “zeros” and deallocate the memory with the regular memory deallocation operating system call.

7.6. Random Number Generation

The module employs a Deterministic Random Bit Generator (DRBG) based on [SP800-90A] for the generation of random value used in asymmetric keys, and for providing an RNG service to calling applications.

The Approved DRBG provided by the module is the CTR_DRBG with AES-256. The DRBG is initialized during module initialization.

The module uses a Non-Deterministic Random Number Generator (NDRNG) to seed the DRBG. The NDRNG provides at least 256 bits of entropy to the DRBG during initialization (seed) and reseeding (reseed). The NDRNG is outside of the module’s logical boundary but within its physical boundary.

8. Self-Tests

8.1. Power-Up Tests

The module performs power-up tests automatically when the module is loaded into memory; power-up tests ensure that the module is not corrupted and that the cryptographic algorithms work as expected.

While the module is executing the power-up tests, services are not available, and input and output are inhibited. The module does not return control to the calling application until the power-up tests are completed. On successful completion of the power-up tests, the module enters operational mode and cryptographic services are available. If the module fails any of the power-up tests, it will return an error code and enter into the Error state to prohibit any further cryptographic operations. The module must be re-loaded in order to clear the error condition.

8.1.1. Integrity Tests

The integrity of the module is verified by comparing an HMAC-SHA-256 value calculated at run time with the HMAC value stored in the module that was computed at build time.

8.1.2. Cryptographic algorithm tests

The module performs self-tests on all FIPS-Approved cryptographic algorithms supported in the approved mode of operation, using the Known Answer Test (KAT) and Pair-wise Consistency Test (PCT) as shown in the following table:

Algorithm	Test
CTR_DRBG	• KAT with AES 256 bits with and without derivation function
AES	• KAT of AES encryption with ECB mode and 128 bit key • KAT of AES decryption with ECB mode and 128 bit key
RSA	• KAT of RSA PKCS#1 v1.5 signature generation with 2048 bit key and SHA-256 • KAT of RSA PKCS#1 v1.5 signature verification with 2048 bit key and SHA-256
ECDSA	• PCT of ECDSA signature generation and verification with P-256 curve
EC Diffie-Hellman	• KAT of primitive “Z” computation with P-256 curve
SHA-1, SHA-256, SHA-384	• KAT of SHA-1 • KAT of SHA-256 • KAT of SHA-384 is covered by KAT for HMAC-SHA-384
HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384	• KAT of HMAC-SHA-1 • KAT of HMAC-SHA-256 • KAT of HMAC-SHA-384

Table 8- Self-Tests

8.2. On-Demand self-tests

The module provides the Self-Test service to perform self-tests on demand. On demand self-tests can be invoked by powering-off and reloading the module. This service performs the same cryptographic algorithm tests executed during power-up. During the execution of the on-demand self-tests, crypto services are not available and no data output or input is possible.

8.3. Conditional Tests

The module performs conditional tests on the cryptographic algorithms shown in the following table. If the module fails any of these tests, it will enter into the Error state to prohibit any further cryptographic operations. The module must be re-loaded in order to clear the error condition.

Algorithm	Test
CTR_DRBG	• Continuous random number generator test
RSA key generation	• PCT using SHA-256
ECDSA/ECDH key generation	• PCT using SHA-256

Table 9 - Conditional Tests

9. Guidance

9.1. Delivery

The module is distributed as a part of BIG-IP product in the form of the 12.1.2-HF1 ISO. The installation will require the install of 12.1.2 base ISO and 12.1.2-HF1 ISO. The module i.e. libcrypto.so binary gets installed together with the product. The FIPS validated module activation requires installation of the 'FIPS 140-2 Compliant Mode' add-on license.

9.2. Crypto Officer Guidance

On the BIG-IP product the Crypto Officer should run following command to ensure the version shown is 12.1.2.HF1.

tmsh show sys version

Sys::Version

Main Package

Product BIG-IP

Version 12.1.2

Edition Hotfix HF1

The Crypto Officer should also verify the FIPS validated module license activation by running the command: 'tmsh show sys license' which should list 'BIG-IP VE, FIPS 140-2 Compliant Mode' in the list of 'Active Modules'. After the FIPS validated module license is installed, the command prompt will change to 'REBOOT REQUIRED'. The Crypto Officer must reboot the BIG-IP for all FIPS-compliant changes to take effect.

9.3. User Guidance

The module supports two modes of operation. Table 5 lists the FIPS approved services. Using the services in Table 6 will put the module in non-FIPS mode implicitly.

Usage:

- AES-GCM shall be used in the context of TLS version 1.2. In case the module's power is lost and then restored, the AES GCM key shall be re-distributed.

10. Mitigation of Other Attacks

The module does not implement security mechanisms to mitigate other attacks.

Appendix A. Glossary and Abbreviations

AES	Advanced Encryption Standard
AES-NI	Advanced Encryption Standard New Instructions
CBC	Cipher Block Chaining
CFB	Cipher Feedback
CSP	Critical Security Parameter
CTR	Counter Mode
CVL	Component Validation List
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DRBG	Deterministic Random Bit Generator
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
FIPS	Federal Information Processing Standards Publication
GCM	Galois Counter Mode
HMAC	Hash Message Authentication Code
J-PAKE	Password Authentication Key exchange by Juggling
KAS	Key Agreement Scheme
KAT	Known Answer Test
MAC	Message Authentication Code
NIST	National Institute of Science and Technology
NDRNG	Non-Deterministic Random Number Generator
OFB	Output Feedback
PAA	Processor Algorithm Acceleration
PSS	Probabilistic Signature Scheme
RNG	Random Number Generator
RSA	Rivest, Shamir, Adleman
SHA	Secure Hash Algorithm
SSSE3	Supplemental Streaming SIMD Extensions 3
XTS	XEX-based Tweaked-codebook mode with cipher text stealing

Appendix B. References

FIPS140-2 **FIPS PUB 140-2 - Security Requirements For Cryptographic Modules**
May 2001
http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

FIPS140-2_IG **Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program**
February 2017
http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf

FIPS180-4 **Secure Hash Standard (SHS)**
March 2012
http://csrc.nist.gov/publications/fips/fips180-4/fips 180-4.pdf

FIPS186-4 **Digital Signature Standard (DSS)**
July 2013
http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

FIPS197 **Advanced Encryption Standard**
November 2001
http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

FIPS198-1 **The Keyed Hash Message Authentication Code (HMAC)**
July 2008
http://csrc.nist.gov/publications/fips/fips198 1/FIPS-198 1_final.pdf

PKCS#1 **Public Key Cryptography Standards (PKCS) #1: RSA Cryptography**
Specifications Version 2.1
February 2003
http://www.ietf.org/rfc/rfc3447.txt

SP800-38A **NIST Special Publication 800-38A - Recommendation for Block Cipher Modes of Operation Methods and Techniques**
December 2001
http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf

SP800-38D **NIST Special Publication 800-38D - Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC**
November 2007
http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

SP800-56A **NIST Special Publication 800-56A - Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised)**
March 2007
http://csrc.nist.gov/publications/nistpubs/800-56A/SP800-56A_Revision1_Mar08-2007.pdf

SP800-90A **NIST Special Publication 800-90A - Recommendation for Random Number Generation Using Deterministic Random Bit Generators**
January 2012
http://csrc.nist.gov/publications/nistpubs/800-90A/SP800-90A.pdf

SP800-131A **NIST Special Publication 800-131A - Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths**
November 2015
http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar1.pdf