FIPS 140-2 Security Policy Working Group (SP WG)

CMVP Managers Meeting

May 16th & 17th, 2016

Objectives of SP WG

Mission Statement:

- The purpose of the Security Policy Working Group is to develop a Security Policy template for cryptographic modules validated against FIPS 140-2 with the goal of improving the understanding of the cryptographic module for users and improving the efficiency of the validation process.

Objectives of SP WG

Current Scope of focus:

1. Approved, Allowed and Non-Approved Algorithms Tables
2. Keys and CSPs Table
3. Approved and Non-Approved Services Tables
4. Creating a mapping between these tables

… the hard stuff first!

Contributors

- Mark Hanson (Intel Security)
- Matt Keller (Corsec Security)
- Yi Mao (atsec)
- Bob Pittman (HPE)
- Ryan Thomas (CGI)
- Richard Wang (Gossamer)
- CMVP

Looking for others … let us know if you want to join!

Audiences to consider

- Product Vendor
  - Demonstrate module conformance
- CST Lab
  - Validate module to FIPS 140-2
- Cryptographic Module Validation Program
  - Conformance review of module
- End User
  - Product procurement decisions, implement module
- Security Assessor/Auditor
  - Review configuration of implemented module

Draft Definition of Terms

- Standard, IG and NIST Special Publication documents may have different definitions
- Intention is to provide the Security Policy author with guidance and one-stop point of reference for important terms

Hat Tip to Yi Mao ☺

Draft Definition of Terms

Example terms include:

- Cryptographic Module
- Logical vs. Physical Boundaries
- Key Generation
- Key Derivation
- Key Entry
- Key Wrapping
- Key Transport
- Key Encapsulation

Draft Definition of Terms

Term	Proposed Definition	Definition found in FIPS 140-2 and/or IG (dated on January 11, 2016)	Notes
Key derivation	Key derivation refers to deriving keys from a pre-shared key between two parties (see NIST SP 800-108), or from a shared secret using Application Specific Key Derivation Functions (see NIST SP 800-135v1), or derive a key for storage applications only, in compliance with SP 800-132.	IG 7.10 states, "When a key is shared between two entities, it may be necessary to derive additional keying material using the shared key." IG D.2 states, "Key derivation is a method for deriving keys from the certain parameters using the Approved key derivation functions. One possibility is to derive a key from an already existing related key as described in SP 800-108. Another is to derive a key for storage applications only, in compliance with SP 800-132."	The differences of key generation and key derivation are: (1) Key generation happens entirely within the module while key derivation can derive a key from an externally entered key (2) Key generation has dependency on an Approved RNG while key derivation does not have this dependency
Key entry	Key entry refers to cryptographic keys being manually or electronically entered into a cryptographic module in plain text or encrypted form.	Per IG D.2, Key entry is a method for key establishment where the key is manually or electronically entered into the module. The term "key entry" refers to both plaintext and encrypted entry of the key using a key transport method.	Key import is a method of key entry, but a key may be entered into a cryptographic module as one of the input parameter of a cryptographic function (e.g. AES encryption)
Key output	Key output refers to cryptographic keys being manually or electronically output from a cryptographic module in plain text or encrypted form.	Section 4.7.4 of FIPS 140-2 states, "Cryptographic keys may be entered into or output from a cryptographic module. If cryptographic keys are entered into or output from a cryptographic module, the entry or output of keys shall be performed using either manual (e.g., via a keyboard) or electronic methods (e.g., smart cards/tokens, PC cards, or other electronic key loading devices)."	Key export is a method of key output, but keys may be output through other means than module's key export service. For example, key may be output from a cryptographic module as one of the input parameter of a cryptographic function that the module calls from a bound module.
Key import	Key import is an electronic method of key entry. It is a service provided by a cryptographic module that is by design to bring a key or a key pair from outside of the module into the logical boundary of the module.	According to IG 1.2, a module may not electronically import keys in plaintext in a non-Approved mode of operation and then switch to an Approved mode of operation and use those keys for Approved services.	The term "key import" is not used in FIPG 140-2. However, it is a frequently used term. The vendor tend to use it interchangeably with "key entry" but the former has a narrower meaning in the sense that a cryptographic module may not have key import service, but still have key entry through other means.

Instructions, Caveats & Assumptions

- Important to communicate how SP tables and eventual template should be used
- Expectations from the CMVP
  - Required information (CAVP certs, modes, what is used by the module, linking/references between the tables)

  vs.

  - Optional information (presentation-related options like layout and format)
- Intention is not to go beyond the original requirements
- Still very much a work in progress

Order of the Tables

1. Services Table

2. Keys/CSPs Table

3. Algorithms (Approved/Allowed/Non-Approved) Table(s)

Approved Algorithms Table

CAVP Cert	Algorithm	Standard	Mode/ Method	Key Lengths, Curves or Moduli	Use	Ref to Services (Table X)
Vendor Affirmation	EC-DH	SP 800-56Arev2	ECC	P-521, B-571	Key Agreement	
519	HMAC	FIPS 180-4	HMAC-SHA-256	128	Message Authentication	
907	HMAC	FIPS 198-1	HMAC-SHA-1	112	Message Authentication	
64	KAS EC-DH	SP 800-56A	ECC	P-256, P-384, P-521	Key Agreement	
51	KAS DH	SP 800-56A	FFC	(2048, 224), (2048, 256)	Key Agreement	
61	KAS MQV	SP 800-56A	FFC	(2048, 224)	Key Agreement	
Vendor Affirmation	PBKDF	SP 800-132			Deriving Keys for Storage Applications	
1724	RSA	FIPS 186-4		2048	Signature Verification	
1987	RSA	FIPS 186-4	SHA-1, SHA-224, SHA-256, SHA-384, SHA-512 PKCS1 v1.5	1024, 2048, 3072	Digital Signature Verification	
Vendor Affirmation	RSA	SP 800-56B		2048, 3072	Key Agreement or Key Transport	
912	SHS	FIPS 180-4	SHA-1, SHA-256		Message Digest	
1378	SHS	FIPS 180-4	SHA-1		Message Digest	
745	Triple-DES	SP 800-67	TCBC		Data Encryption/ Decryption	

Approved Algorithms Table

- CAVP algorithm certificate considered the key (index) for the table
- Indicate only modes/functions utilized by the module
- Multiple algorithm implementations – should they be in the same table? Separate table?

Allowed Algorithms Table

Algorithm	Applicable Caveat	Use	Ref to Services (Table X)
Diffie-Hellman (CVL Cert. #533)	Provides 112 or 128 bits of encryption strength.	Key establishment	
Elliptic Curve Diffie-Hellman Supported curves: P-256, P-384, P-521	Provides between 128 and 256-bits of encryption strength.	Key establishment	
MD5		For use in TLS 1.0/1.1	
NDRNG – entropy token external to the module's cryptographic boundary	No assurance can be made for the correct operation of the entropy token nor is there a guarantee of stated entropy.	Seeding for the DRBG	
RSA Key Wrapping	Provides 112 or 128 bits of encryption strength.	Key establishment	

Allowed Algorithms Table

- Used to address key establishment techniques
- Entropy source (NDRNG) – does it reside internal or external to module boundary?
- MD5 when used in TLS 1.0/1.1
- SHA-1 for Digital Signature Generation affirmed for use with TLS as per footnote in NIST SP 800-52

Non-Approved Algorithms Table

Algorithm	Use	Ref to Services (Table X)
AES (non-compliant)	Encryption / Decryption	
DES	Encryption / Decryption	
Diffie-Hellman (CVL Cert. #533)	Key Establishment - Non-compliant less than 112 bits of encryption strength	
DSA (FIPS 186-2)	Digital Signature Generation	
ECDSA (FIPS 186-2)	Digital Signature Generation	
ECDSA (FIPS 186-4; non-compliant)	Digital Signature	
HMAC-SHA-256, HMAC-SHA-512 (SSSE3/AVX/AVX2 implementation; non-compliant)	Keyed Hash	
MD4		
RC2		
RNG (ANS X9.31)	Random Number Generation	
RSA (FIPS 186-2)	Asymmetric Key Generation	
RSA	Key Transport – non-compliant less than 112 bits of encryption strength	
SHA-256, SHA-512 (SSSE3/AVX/AVX2 implementation; non-compliant)	Hashing	
Triple-DES (non-compliant)	2-Key Encryption	

Mapping the Tables

- Intention is to map the Algorithms Tables to the Keys/CSPs Table and Services Tables
- Still very much a work in process

Challenges

- Applying tables to different “real world” modules
- Populating the tables the first time through = a lot of effort ! (Bob P ☺)
- Find balance and list the proper amount of detail for all audiences
- Finding a common point of reference for terminology

Current Drafts

- Will be posted to the Cryptographic Module User Forum (CMUF) website in the SP WG Forum
- Almost ready to share …

Security Policy Do's ~~and Don'ts~~

- Make sure SP is marked non-proprietary
- Ensure all URL links are valid and working
- Use official algorithm names (NIST Standard or SP)
- If AES-GCM, explain how it complies with IG A.5

Security Policy Do's ~~and Don'ts~~

- AES-XTS – state it is only Approved for storage applications
- List the correct standard for RSA (FIPS 186-2 and/or FIPS 186-4)
- Indicate all Algorithm modes supported by the module (AES ECB, CBC and CFB etc.)
- DH private key lengths are expected to be shorter than that of the public key

Security Policy Do's ~~and Don'ts~~

- Make sure you describe the module cryptographic boundary
- Algorithm Tables must be consistent with the postings on the CAVP website
- Coordinate with your lab – make sure the Company Name, HW/FW/SW version in SP are consistent with the lab's submission

Want to Help?

- Contact me:

Ryan Thomas, CGI ITSETF

ry.thomas@cgi.com