Cryptographic Algorithm Validation Program:

Roadmap to Testing of New Algorithms

Sharon Keller, CAVP Program Manager
NIST

November 6, 2015

Overview

- Process of developing validation tests for cryptographic algorithms
- Types of validation tests
- Cryptographic algorithm standards
  - With validation testing
  - For which the CAVP is currently developing validation testing
  - Published but don't have validation tests yet
  - In draft

Mission

- To provide federal agencies—in the United States and Canada—with assurance that a cryptographic algorithm has been implemented according to the specifications in the applicable standard.

Algorithm Validation Testing Development Process

- Review the requirements and algorithmic specifications in the cryptographic algorithm documents (SP, FIPS)
  - Identify the algorithm's:
    - Components
    - Functionality
    - Mathematical formulas

Algorithm Validation Testing Development Process

- Develop and implement the algorithm validation test suite
  - Test the requirements addressable at the CAVP level
  - Develop the test metrics for testing the algorithm
  - Exercise all mathematical elements of the algorithm
  - Assure the specifications in the standard have been implemented correctly
  - If deviates, validation test will fail indicating implementation flaw

Algorithm Validation Testing Development Process

- Develop User Documentation and Guidance
  - Called Validation System Document (VS)
  - Documents test suite
  - Provides instructions on implementing validation tests

Types of validation tests

- **Known Answer Tests (KAT) - designed to verify the components of algorithms (Sboxes, permutation tables, etc)**

- **Multi-block Message Test (MMT)– designed to test the ability of the implementation to process multi-block messages, which may require chaining of information from one block to the next. Test supplies the IUT with messages that are integral numbers of blocks in length**

Types of validation tests

- **Monte Carlo Test (MCT)- designed to exercise entire implementation. Purpose: to detect the presence of flaws in the IUT that were not detected with controlled input of KATs**

Types of validation tests

Positive testing

- The testing process where the implementation is validated against valid input data. In this testing, only valid sets of values are supplied and the results are checked to see if the expected results are generated.
- Goal: to prove that a given implementation has adhered to the specifications and requirements in the standard
- Used for PQG generation, signature generation, etc. where some values are given and the IUT has to compute the answer.

Types of validation tests

Negative testing

- The testing process where the implementation is validated against invalid input data.
  A negative test checks if an implementation behaves as expected with its negative inputs.
- Goal: to test the ability of the IUT to recognize valid and invalid values.
- Usually used for PQG Verification, Signature Verification, etc. where all values are provided and the IUT is verifying the correct result is achieved. Errors are introduced into the different parameters to assure the IUT can recognize the errors

Available Validation Testing

- Symmetric Algorithms
  - AES (FIPS 197)
  - Triple DES (ANSI X9.52-1998)

- Asymmetric algorithms
  - Both FIPS 186-4 and Legacy functions of 186-2
    - DSA
    - ECDSA
    - RSA

- Modes of Operation
  - CMAC (SP 800-38B)
  - CCM (SP 800-38C)
  - GCM/GMAC(SP 800-38D)
  - XTS-AES(SP 800-38E)
  - Methods for Key Wrapping (SP800-38F)

- SHS (FIPS 180-4)
- RNG (FIPS186-2, ANSI X9.31 Appx A.2.4, ANSI X9.62 Appx A. 4)
- DRBG (SP 800-90A)
- Key Agreement Schemes (SP 800-56A)
- HMAC (FIPS 198)
- SP800-108 KDF

- Testing of Components
  - All SP 800-56A except KDF
  - SP800-56A ECCCDH Primitive
  - RSASP1 - Mod Exp for Sig Gen (PKCS1.5 and PKCS-PSS)
  - RSADP - Basic Decrypt Operation
  - KDFs in SP800-135-IKE, TLS, SSH, SNMP
  - ECDSA Sign Gen Component

Example of where tests used

- **Known Answer Tests**
  - Example:
    - AES
      - GFSbox
      - KeySbox
      - Variable Key
      - Variable Text
    - TDES
      - Variable Plaintext
      - Variable Ciphertext
      - Inverse Permutation
      - Variable Key
      - Permutation Operation
      - Substitution Table
- **Multi-block Message Test**
- **Monte Carlo Test**

Example of where tests used

- **RSA**

Signature Generation

- X9.31
  - Mod 2048 with SHA224 256 384 512 512/224 512/256
  - Mod 3072 with SHA224 256 384 512 512/224 512/256
  - Mod 4096(for module revalidation) with SHA 256 384 512

PKCS#1 1.5

- Same as above except add SHA224 for use with Mod 4096

PKCS#1 PSS

- Same as PKCS#1 1.5 except add:
- Salt length for each SHA specified

Key Generation Test (FIPS 186-4)

Random Primes:

Provable primes (Appendix B.3.2)

Probable primes (Appendix B.3.3)

Primes with Conditions:

Primes p1, p2, q1,q2, p and q shall all be provable primes (Appendix B.3.4)

Primes p1, p2, q1, and q2 shall be provable primes and p and q shall be probable primes (Appendix B.3.5)

Primes p1, p2, q1,q2, p and q shall all be probable primes (Appendix B.3.6)

186-4 Signature Verification

•X9.31

- Mod 1024, 2048, 3072
- SHA 1 224 256 384 512 512/224 512/256
- Fixed or random pubkey e

•PKCS#1 1.5

- Same as X9.31

•PKCS#1 PSS

- Same as X9.31 with addition of
- Salt lengths and salt values (optional)

Example of where tests used

- SHA
- Options tested
  - SHA1, 224, 256, 384, 512, 512/224, 512/256
  - Bit or Byte Orientation
  - Support zero length messages or not

Test suite:

- Short Messages Test
- Selected Long Messages Test
- Monte Carlo Test

Cryptographic algorithms for which NIST CAVP is currently in the process of developing validation testing

- FIPS 202: SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions
- SP 800-56C Key Derivation through Extraction-then-Expansion
- SP 800-132 Password-Based Key Derivation Part 1: Storage Applications

Future Algorithm Testing (Published Standards)

- SP 800-56B (Rev 1): Key Agreement Schemes with RSA – June 2014
- SP 800-106 Randomized Hashing for Digital Signatures
- SP800-38A Addendum Block Cipher Mode 3 variants of CT Stealing for CBC Mode
- SP800-56A (Rev 2) Key Agreement Schemes with DSA and ECDSA
- SP800-90A (Rev 1)DRBG

Future Algorithm Testing (Draft Standards)

- SP800-90B Draft : Entropy Sources
- SP800-90C Draft: construction of RBGS

Questions??

Sharon Keller
sharon.keller@nist.gov
(301)975-2910