National Institute of Standards and Technology
Technology Administration, U.S. Department of Commerce

The Cryptographic Algorithm Validation Program

Sharon Keller

National Institute of Standards and Technology

14 September 2004

Communications Security Establishment
Centre de la sécurité des télécommunications

Canada

Presentation Outline

- Implementation of Cryptography
- Cryptographic Algorithm Validation Program's relationship to CMVP
- Algorithm Validation Process - CAVS
- Currently Validation Testing Exists for these Algorithms
- Validation Testing Details
- Future Algorithms

Implementation of Cryptography

- Very complex
- 25% of algorithm implementations ready for marketing are incorrect
- NIST's answer: Develop validation testing for *Approved** cryptographic algorithms
- Validation testing provided by NVLAP accredited testing laboratories

* *Approved* – FIPS Approved and/or NIST Recommended

Relationship with CMVP

- Cryptographic Algorithm Validation Program (CAVP) is a prerequisite to the Cryptographic Module Validation Program (CMVP)
  - All underlying algorithms must be validated prior to the module being FIPS 140 validated

The Cryptographic Algorithm Validation System (CAVS)

- Designed and developed by NIST
- Supplied to NVLAP accredited testing laboratories
- Provides uniform validation testing for *Approved* cryptographic algorithms
- Provides thorough testing of the implementation
- Types of errors found by CAVS range from pointer problems to incorrect behavior of the algorithm implementation.

Validation Process

Vendor notifies the laboratory that they want to test their algorithm implementation(s) .

Accredited Laboratory

Laboratory uses vendor information and the CAVS tool to generate input vectors to validate all algorithm implementations.

Vendor

Vendor uses input vectors and documentation to perform validation tests and generate results.

Laboratory uses the CAVS tool to validate the vendor's results.

If the validation passes successfully . . .

If there are errors, laboratory works with vendor to correct the problem.

Validation Process (cont.)

CAVP

Laboratory sends the CAVS results to the CAVP where the CAVP:

Accredited Laboratory

1. Reviews the results to assure that everything has passed successfully;

- ENCRYPTION
- HASHING
- AUTHENTICATION
- SIGNATURE
- KEY MGT.

Validation Process (cont.)

CAVP

ENCRYPTION
HASHING
AUTHENTICATION
SIGNATURE
KEY MGT.

2. Creates an algorithm validation certificate which is signed by NIST and CSE and sent to the vendor via the laboratory;

Advanced Encryption Standard
Algorithm Validation Certificate

The National Institute of Standards and Technology of the United States of America

Certificate No. 800

The Communications Security Establishment of the Government of Canada

The National Institute of Standards and Technology (NIST) and the Communications Security Establishment (CSE) hereby validate the Advanced Encryption Standard (AES) algorithm testing results of the implementation identified as:

Encryption At It's Best, Version 1.0

and supplied by:

Testing Corporation

in accordance with the specifications of the *Advanced Encryption Standard (AES)* (FIPS 197) and *Recommendations for Block Cipher Modes of Operation* (SP800-38A 2001 ED) as indicated on the reverse of this certificate. Implementations bearing the same identification and manufactured to the same specifications as the validated implementation may be labeled as complying with FIPS 197 for the modes, states, and key sizes identified in this certificate. No reliability test has been performed and no warranty of the implementation is either expressed or implied.

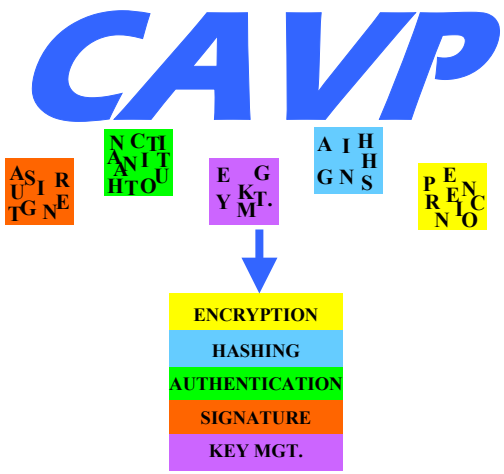
The validated implementation was tested using the following operating environment (for software implementations, operating environment includes processor and operating system; for firmware implementations, operating environment includes processor only; for hardware implementations, operating environment is not applicable):

Pentium III w/ Windows 2000 test

The vendor should be contacted to obtain a list of operating environments that support the validated implementation. Likewise, the vendor should be contacted to obtain a list of products/applications that use the validated implementation.

This certificate must include the following page that details the scope of conformance and includes the validation authorities' signatures.

Validation Process (cont.)



3. Updates the algorithm validation list accessible via http://csrc.nist.gov/cryptval to make this information accessable to the world.

Advanced Encryption Standard (AES) Algorithm Validated Implementations

Cert#	Vendor	Implementation	Operational Environment	Val. Date	Modes/States/Key sizes/ Description
800	Testing Corporation 123 ABC Street, Gaithersburg, MD 20899 USA	Encryption At It's Best Version 1.0	Pentium III w/ Windows 2000 test	8/31/2004	ECB(e/d; 128,192,256); CBC(e/d; 128,192,256); CFB1(e/d; 128,192,256); CFB8(e/d; 128,192,256); CFB128(e/d; 128,192,256); OFB(e/d; " This is a test "

CAVS Testing

- Currently provides validation testing for
  - Symmetric Algorithms
    - Data Encryption Standard (DES) (FIPS 46-3; FIPS 81)
    - Triple Data Encryption Standard (TDES) (FIPS 46-3)
    - Advanced Encryption Standard (AES) (FIPS 197)
  - Asymmetric Algorithms
    - Digital Signature Standard (DSS) (FIPS 186-2 + Change Notice)
    - RSA Signature Algorithm (FIPS 186-2;PKCS #1 v2.1)
    - Elliptic Curve Digital Signature Algorithm (ECDSA) (FIPS 186-2)

CAVS Testing (cont.)

- Currently provides validation testing for
  - Hashing algorithms
    - SHA1, SHA224, SHA256, SHA384, SHA512 (FIPS 180-2)
  - Random Number Generator (RNG) (ANSI X9.31, FIPS 186-2+change notice, ANSI X9.63)
  - MACing algorithms
    - Keyed Hash Message Authentication Code (HMAC) (FIPS 198)
    - Counter w/ Cipher Block Chaining (CBC) MAC (CCM) (Special Pub 800-38C)

Validation Testing Details

- Designed to
  - Determine the correctness of the algorithm contained in the implementation
  - Detect implementation flaws from pointer problems to incorrect behavior of the algorithm implementation
- **NOT** DESIGNED to
  - detect intentional attempts to misrepresent conformance

Validation Testing Details

- Found at http://csrc.nist.gov/cryptval
  - Documentation describing testing requirements for each algorithm.

AES Validation Testing Details

- Designed to test
  - ECB, CBC, OFB, CFB1, CFB8, CFB128, Counter
  - Key Sizes 128-bit, 192-bit, 256-bit
  - Encryption, Decryption
- Types of validation tests
  - 4 Known Answer Tests – tests the components of the algorithm
    - GFSbox
    - KeySbox
    - Variable Key
    - Variable Text
  - Multi-block Message Test – tests the ability to process multi-block messages
  - Monte Carlo Test – designed to exercise the entire implementation.

TDES Validation Testing Details

- Designed to test
  - TECB, TCBC, TCBC-I, TOFB, TOFB-I, TCFB (1, 8, 64 bit), TCFB-P (1, 8, 64 bit), Counter
  - Key Options 1-Key, 2-Key, 3-Key
  - Encryption, Decryption
- Types of validation tests
  - 5 Known Answer Tests – tests the components of the algorithm
    - Variable Plaintext
    - Inverse Permutation
    - Variable Key
    - Permutation Operation
    - Substitution Table
  - Multi-block Message Test – tests the ability to process multi-block messages
  - Monte Carlo Test – designed to exercise the entire implementation.

DES Validation Testing Details

- CURRENTLY FOR LEGACY SYSTEMS ONLY
- PROPOSED WITHDRAW OF DES ANNOUNCED
- Designed to test
  - ECB, CBC, OFB, CFB1, CFB8, CFB64
  - Encryption, Decryption
- Types of validation tests
  - 5 Known Answer Tests – tests the components of the algorithm
    - Variable Plaintext
    - Inverse Permutation
    - Variable Key
    - Permutation Operation
    - Substitution Table
  - Multi-block Message Test – tests the ability to process multi-block messages
  - Monte Carlo Test – designed to exercise the entire implementation.

DSA Validation Testing Details

- Designed to test modulus sizes 512-1024 bits
- Five separate tests - one to validate each of the various algorithm components
  - Domain Parameter Generation
  - Domain Parameter Verification
  - Key Pair Generation
  - Signature Generation
  - Signature Verification

ECDSA Validation Testing Details

- Designed to test
  - NIST recommended curves
- Four separate tests - one to validate each of the various algorithm components
  - Key Pair Generation Test
  - Public Key Validation Test
  - Signature Generation Test
  - Signature Verification Test

RSA Validation Testing Details

- Prerequisite – SHA validation testing
- Designed to test
  - Modulus sizes 1024 – 4096
  - Use with SHA1, SHA224, SHA256, SHA384, and SHA512
- Three variations of the RSA algorithm
  - RSA algorithm specified in FIPS186-2, *Digital Signature Standard (DSS)*, January 27, 2000[1], and
  - Two signature schemes with appendix specified in *Public Key Cryptography Standards(PKCS) #1 v2.1: RSA Cryptography Standard-2002* [2]. These two signature schemes with appendix are
    - RSASSA-PSS
    - RSASSA-PKCS1-v1_5.

RSA Validation Testing Details (Continued)

- Three validation tests - one to validate each of the various algorithm components
  - Key Generation Test
  - Signature Generation Test
  - Signature Verification Test

SHA Validation Testing Details

- Designed to test
  - byte and bit oriented implementations
  - 5 SHA algorithms
    - SHA-1, SHA-224, SHA-256, SHA-384, SHA512
- Three Categories of Tests
  - Short Messages Test- tests the ability to correctly generate message digests for messages of arbitrary length.
  - Long Messages Test – tests the ability to correctly generate message digests for messages that span multiple message blocks.
  - Pseudorandomly Generated Messages Test - tests the correctness of message digests generated from pseudorandomly generated messages.

RNG Validation Testing Details

- Designed to test
  - G functions
  - Seed key bytes sizes
  - RNG Generators / NIST Recommended Curves
- Testing for the following RNG algorithms
  - DSA – FIPS 186-2 Appendix 3.1 – Algorithm for Computing *m* values of *x* (using SHA-1and/or DEA) (Regular and General Purpose),
  - DSA – FIPS 186-2 Appendix 3.2 – Algorithm for Precomputing One or More *k* and *r* Values (using SHA-1 and/or DEA),
  - ECDSA – ANSI X9.62-1998 Appendix A.4 - Pseudorandom Number Generation (using SHA-1 and/or DEA), and
  - rDSA – ANSI X9.31-1998 - Generating Pseudo Random Numbers Using the DEA.

RNG Validation Testing Details (Continued)

- Two Validation tests
  - the Variable Seed Test (VST)
  - Monte Carlo Test (MCT)

HMAC Validation Testing Details

- Prerequisite – SHA Validation Tests
- Designed to test
  - The underlying hash algorithm(s) supported
  - For each of the underlying hash algorithms specified above, supported key sizes, K, related to the block size, B, of the underlying hash algorithm. That is, does the IUT support key sizes of K<B, K=B, or K>B.
  - For each of the underlying hash algorithms supported, the length(s), t, in bytes, of the MAC the IUT is able to produce.
- Validation Test: The Random Message Test
  - provides 15 sets of messages and keys for each hash algorithm/key size/MAC size combination supported by the IUT.

CCM Validation Testing Details

- Prerequisite – AES validation tests
- Designed to test
  - the AES key sizes supported: 128, 192, and/or 256;
  - a range of data lengths supported;
  - a range of payload lengths supported;
  - the nonce lengths supported: 7, 8, 9, 10, 11, 12, and/or 13;
  - the tag lengths supported: 4, 6, 8, 10, 12, 14, and/or 16..
- Types of validation tests
  - Variable Associated Data Test
  - Variable Payload Test
  - Variable Nonce Test
  - Variable Tag Test
  - Decryption-Verification Process Test

Validated Algorithm Implementations (FY04)

- AES – 75
- DES – 36
- TDES – 63
- SHA – 69
- DSA – 16
- RNG – 22 (available since Mar04)
- RSA – 17 (available since Mar04)

- Total 298 algorithm validations compared to 254 in FY03

Future Algorithms to Validate

- CMAC
- Key Establishment Schemes
- Key Wrap Algorithm
- Parallelizable Authenticated Encryption Algorithm
- Additional Random Number Generators

Making a Difference

- CAVP and CMVP isolate and correct security flaws before products go to market.
- Algorithm Validations (during testing)
- (DES, TDES, DSA and SHA-1)
  - 26.5% Security Flaws discovered
  - 65.1% Documentation Errors found
- Cryptographic Modules (during testing)
  - 48.8% Security Flaws discovered
  - 96.3% Documentation Errors found