

CAVP Testing

Current Algorithm Validation Testing vs. Entropy Source Validation Testing


Sharon Keller



Purpose of CAVP Algorithm Testing

- ▶ To provide assurance that the algorithm has been implemented correctly


Algorithm Validation Test Development

- **Analyze the specifications in the special publications**
 - **Identify all elements to be tested**
 - Identify all mathematical calculations within the elements
 - Identify the requirements identified by “shall” statements addressable at the algorithm level
 - Design a suite of algorithm validation tests that address the above specifications and challenge each specification with allowed and non allowed values to assure they are handled properly.
- 

Example of elements of SP800-90A

- Functions are
 - Instantiate function
 - Reseed Function
 - Generation Function
 - Uninstantiate Function
- For 4 different DRBG mechanisms
 - Hash DRBG
 - HMAC DRBG
 - Counter DRBG
 - Dual EC DRBG

Algorithm Validation Test Development

- Analyze the specifications in the special publications
 - Identify all elements to be tested
 - **Identify all mathematical calculations within the elements**
 - Identify the requirements identified by “shall” statements addressable at the algorithm level
 - Design a suite of algorithm validation tests that address the above specifications and challenge each specification with allowed and non allowed values to assure they are handled properly.
- 

Mathematical Calculations

- ▶ Possible branches within the algorithm
 - Equations
 - If..Then..Else statements
 - While statements
 - Etc.
- ▶ Example – Step 7 of 9.3.1
 - ▶ 7. If *reseed_required_flag* is set, or if *prediction_resistance_request* is set, then
 - ▶ 7.1 *status* = **Reseed_function** (*state_handle*, *prediction_resistance_request*, *additional_input*).
 - ▶ 7.2 If *status* indicates an **ERROR**, then return *status*.
 - ▶ 7.3 Using *state_handle*, obtain the new internal state.
 - ▶ 7.4 *additional_input* = the *Null* string.
 - ▶ 7.5 Clear the *reseed_required_flag*.

Algorithm Validation Test Development

- Analyze the specifications in the special publications
 - Identify all elements to be tested
 - Identify all mathematical calculations within the elements
 - **Identify the requirements identified by “shall” statements addressable at the algorithm level**
- Design a suite of algorithm validation tests that address the above specifications and challenge each specification with allowed and non allowed values to assure they are handled properly.

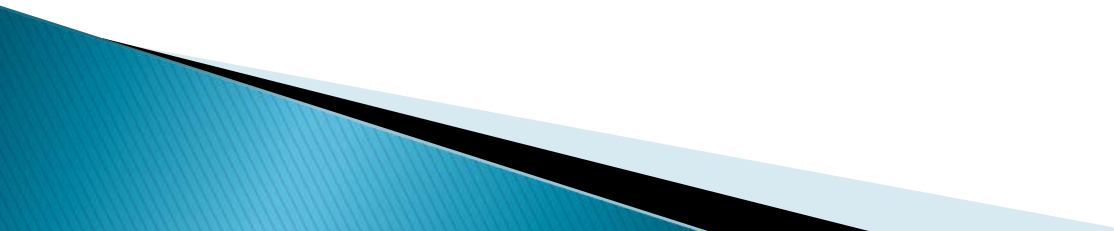
Requirements identified by “shall”

- ▶ **9.3.1 The Generate Function**
- ▶ The following or an equivalent process **shall** be used to generate pseudorandom bits.
- ▶ **Generate_function** (state_handle, requested_number_of_bits, requested_security_strength, prediction_resistance_request, additional_input):
- ▶ ...
- ▶ *The CAVS DRBG tests verify the correct operation of the Generate_function.*

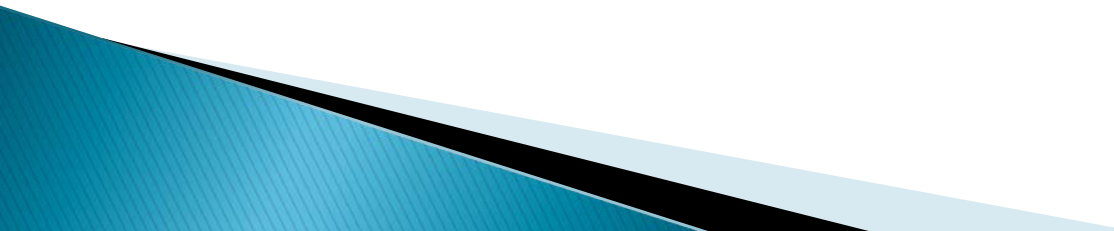
Algorithm Validation Test Development

- Analyze the specifications in the special publications
 - Identify all elements to be tested
 - Identify all mathematical calculations within the elements
 - Identify the requirements identified by “shall” statements addressable at the algorithm level
- **Design a suite of algorithm validation tests that address the above specifications and challenge each specification with allowed and non allowed values to assure they are handled properly.**

Current Algorithm Validation Testing Process

- ▶ Cryptographic Algorithm Validation System (CAVS) Tool
 - uniform validation testing for all approved algorithms
 - Automated, objective tests
 - ▶ Distributed to all accredited laboratories
 - ▶ Testing done by vendor or laboratory at vendor or laboratory site (laboratory not required to be present)
- 

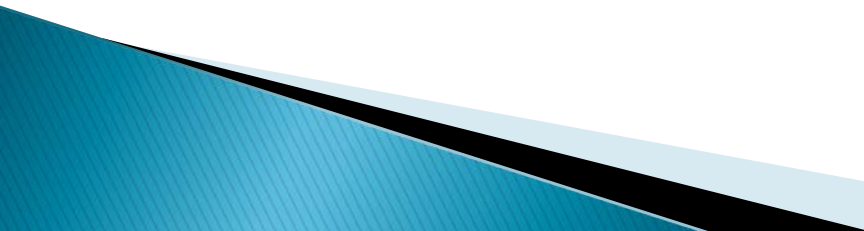
Current Algorithm Validation Testing Process

- ▶ Results are sent to laboratory where results are confirmed
 - ▶ Once results are correct, lab sends validation request to CAVP where it is confirmed
- 

Differences between Algorithm Testing and Entropy Testing (SP800-90B)

- ▶ Algorithm vs. entropy source
 - Deterministic vs. non-deterministic (random)
- ▶ SP800-90B – series of statistical testing provided to estimate the amount of entropy
 - Entropy testing will return a score indicating the amount of entropy provided instead of Pass or Fail.

List of Discussion Points to Assist In Determining Differences in Validation Testing for SP800–90B Entropy Sources

- ▶ Location of testing
 - ▶ How is data collected from the entropy source?
 - ▶ Must the laboratories be present during testing?
 - What does this mean for labs? Staffing, travel costs, etc.
- 

List of Discussion Points to Assist In Determining Differences in Validation Testing for SP800–90B Entropy Sources

- ▶ Section 6.0 Entropy Source Development Requirements
 - Provides required documentation for entropy source validation
 - ▶ Section 7.0 Validation Data And Documentation Requirements
- 