Is your design leaking keys? Efficient testing for side-channel leakage

Benjamin Jun
Cryptography Research Inc

Pankaj Rohatgi
Cryptography Research Inc
Side-channels: The current state of (in)security

From HSMs to mobile devices, cryptographic implementations easily succumb to side-channel attacks.

RSA: Electromagnetic side-channel information leakage from a modern FIPS 140-2 Level 3 HSM. EM emissions traces from the HSM are different for two different keys.

RSA Private Key Operation:
Computing $M^d \mod N$

For each bit of secret exponent $d$
if bit == 0, perform Square (S)
if bit == 1, perform Square (S) followed by Multiply (M)
EndFor

RSA: Side-channel vulnerability on modern smart phone
EM trace shows Square(S)/Multiply(M) operation sequence during modular exponentiation, revealing secret exponent $d$
Side-channel (in)security: What’s being done

- Side-channel resistance requirements are being added to security standards
  - E.g., FIPS 140-3 Draft

- But testing seen as a challenge
  - Vulnerabilities cross many abstraction layers
  - Countermeasures can’t be applied and verified at a single layer
    - Cannot be validated without physical testing
  - Evaluation-style side-channel testing is the norm
    - E.g.: Common Criterion, EMVCo
    - Costly, time consuming & requires high degree of lab expertise
## Testing styles: Validation vs. Evaluation

**Validation**
- E.g., FIPS 140-2
- Demonstrate conformance to specification
- Structured test/check methodology
  - Defined tasks
  - Lab consistency
  - Cost effective
    - New vulnerabilities not addressed
    - No penetration testing
    - Only as good as spec and test plan coverage

**Evaluation**
- E.g., Common Criteria
- Defined security environment and threat model
- Intrinsic risk assessment
  - Threat based analysis
  - Best use of lab expertise
  - Flexibility
    - Limited by lab expertise
    - Potential inconsistency of evaluations
    - Higher cost

With a good specification and test coverage, validation approach can be low-cost, yet effective.
Effective, low cost, validation-based side-channel testing is possible

- Test vector leakage assessment (TVLA) methodology
- Highlights
  - Pre-specified set of test vectors, chosen by experts
  - Pre-specified set of tests on collected data, designed by experts
  - Standard statistical test of significance, with clear pass/fail criteria
- Main idea: focus on identifying statistically significant information leakage, not key extraction
  - Detecting leakages is much easier
  - With (much) additional effort, leakages lead to key extraction attacks
Core statistical test (Univariate leakage)

- Each test specifies and compares two subsets A & B of collected traces
  - Some sensitive Intermediates will be different in subsets A and B if the implementation not properly protected
  - Statistically significant difference between subsets \( \rightarrow \) sensitive information leakage \( \rightarrow \) device fails
- Statistical test: Welch’s t-test for significance of “difference of means”

\[
t(I) = \frac{X_A(I) - X_B(I)}{\sqrt{\frac{S^2_A(I)}{N_A} + \frac{S^2_B(I)}{N_B}}}
\]

- Test performed twice on two independent data sets
  - Failure must occur at the same time-instant in both tests
AES testing specification: moderate resistance

Data collection:

► Specified number of side-channel traces to collect:
  ► Trace based: “at least 1,000,000 traces”
  ► Time based: “up to 1 day of data collection by attacker”

► Test vectors for AES (AES 128, 192, 256)
  ► Fixed key K
  ► “Random” data set
    ► Successive AES encryptions starting from a fixed plaintext block
  ► “Fixed” data set
    ► Repeated encryptions of the same fixed plaintext block
    ► Selected to trigger special conditions within AES
Tests: Six Categories

- **Non-specific** leakage test: fixed vs. varying data
  - Examine middle third of operation
- Five varying data tests targeting **specific leakages**
  - XOR of round input and output
  - S-box outputs in a round
  - Round output
  - Value of 1\textsuperscript{st} byte of round output
  - Value of 2\textsuperscript{nd} byte of round output

**Pass/Fail criteria:**

- Fail if t-statistic exceeds ±4.5 for two independent data sets at the same point in time
Live Demo: Testing unprotected AES on FPGA

Failure condition reached within in 2 minutes of data collect/analysis
Example: Masked AES on FPGA

► DUT: Hardware AES implementation on FPGA with masking countermeasure
  ► Countermeasure not fully effective

► Automated data collection
  ► DUT supports 20 traces/second
  ► Bulk ECB encryption allows 10000 ops/2 minutes
  ► Overnight data collect using ECB mode: 3 million AES ops

Result is a **definitive FAIL**
- Passed all specific leakage tests
- **Failed non-specific Fixed vs. Random test**
- Less than 24 hours data collect + analysis
**Masked AES: Passing and failing tests**

**T-test traces for two independent data sets for XOR leakage:**

- The t-statistic remains between +/- 4.5 throughout the round: **PASS**

**T-test trace for FIXED vs. RANDOM leakage test:**

- The t-statistic has large excursions beyond +/- 4.5: **FAIL**!
## Test specification for RSA

### Test Vectors Sets
- **Set 1**
  - Constant key, constant ciphertext
  - Baseline
- **Set 2**
  - Same constant Key, varying ciphertext
- **Set 3**
  - Varying key, same constant ciphertext
- **Set 4**
  - Same constant key, ciphertext from a set of “special values” (28 different cases used in our experiments)
- **Set 5**
  - Same constant key, ciphertext corresponding to small messages

### Tests
- Test 1: t-test Set 2 vs. Set 1
- Test 2: t-test Set 3 vs. Set 1
- Test 3: t-test Set 4 vs. Set 1
- Test 4: t-test Set 5 vs. Set 1

### Alignment at multiple points
- start, end, middle (CRT)

### Pass/Fail criteria
- t-statistic exceeds +/- 4.5 for two independent data sets A and B at same time location
Example: DUT implementing RSA exponent and data blinding, but not prime blinding

Average trace for RSA CRT operation, aligned at middle

T-test traces for Test 3

Time
Conclusion

- Low-cost and effective testing for side-channel resistance is possible

- Proposed tests for detecting leakage also useful to product designers implementing countermeasures
  - Specialized attack knowledge not required to perform tests
  - Non-specific tests capture large classes of leakages
  - Quick turn-around
  - Failed tests provide feedback to designers about remaining leakages
Thank You!

Benjamin C Jun
VP Technology
Cryptography Research Inc
415.397.0123 x4323
ben@cryptography.com

Pankaj Rohatgi
Director of Engineering
Cryptography Research Inc
415.397.0123 x4338
rohatgi@cryptography.com