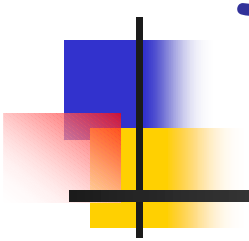
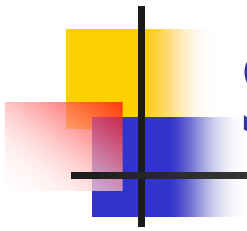


Template attacks

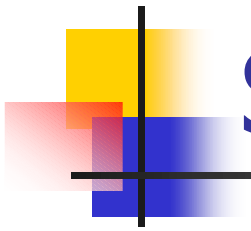
Suresh Chari, Josyula R. Rao,
Pankaj Rohatgi

IBM Research



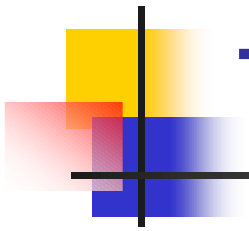
Side channel attacks

- Lots of sources: Power, EM, timing ……..
- Problem one of signal classification
  - Which possible value of key bit/byte/nibble does signal correspond to?
- Many types of attacks(SPA/DPA & variants)
  - Use coarse statistical methods
    - Collect lots of samples to eliminate noise
    - Differentiate based on expected signal.



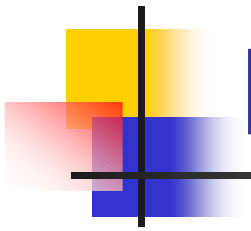
Signal classification

- Technique relies on precise modeling of noise (inherent statistical variations + other ambient noise) AND expected signal
  - Requires experimentation(offline)
- Technique based on Signal Detection and estimation theory
  - Powerful statistical methods
  - Tools to classify a single sample.



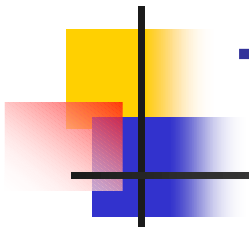
Template attacks: overview

- Need single target sample from device under test
- Need programmable identical device.
- Build *precise model* of noise AND expected signal for all possible values of first portion of key
- Use *statistical characterization* to restrict first portion to **small** subset of possible values.
- Iterate to retain small number of possible values for entire key.
- Strong statistical methods extract ALL information from each target sample.



Plan

- Test case: RC4
- Noise Modeling
- Classification Technique
  - Variants
  - Empirical Results
- Related work



Test case-RC4

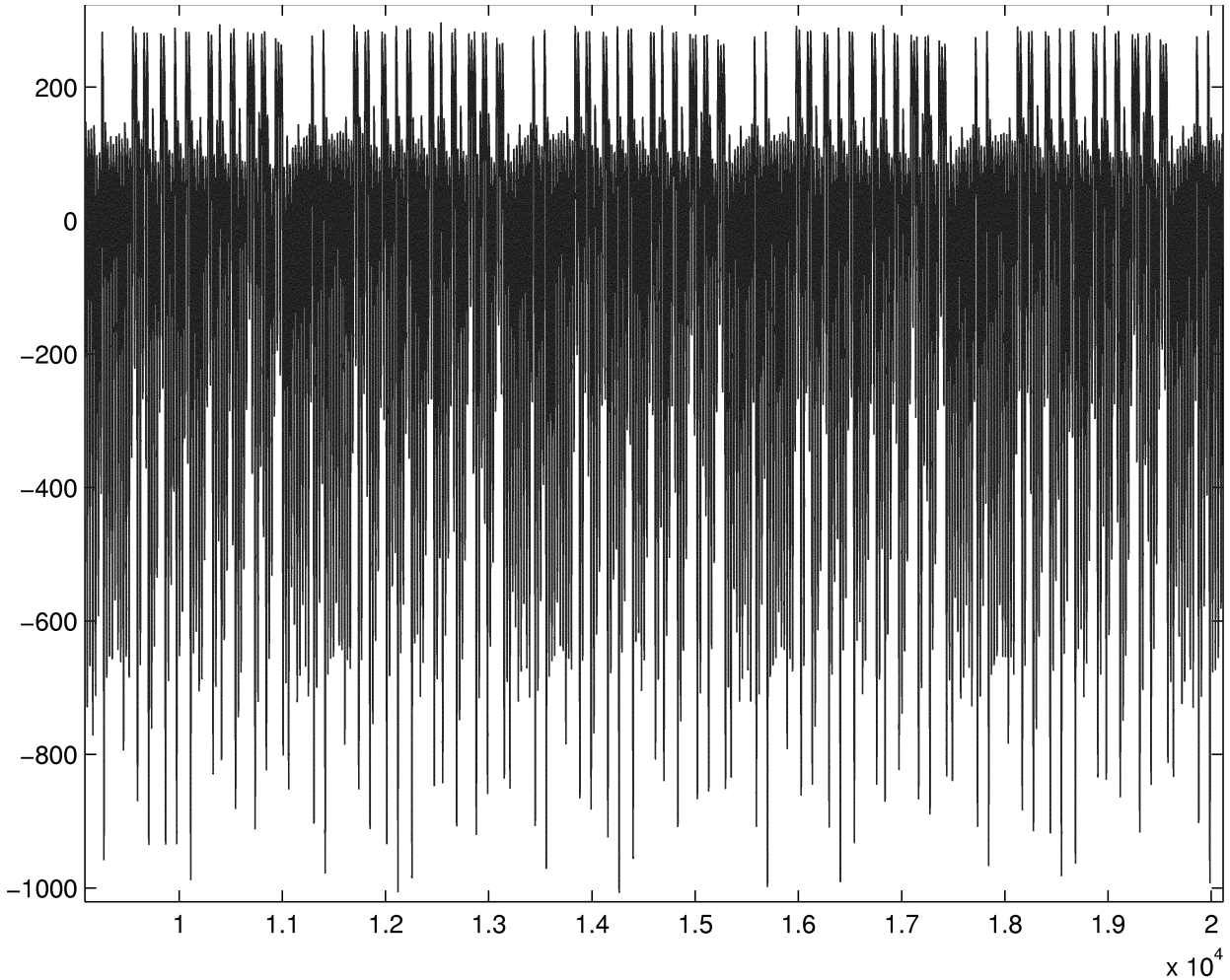
- Implementation: RC4 on smart card.
  - Representative example for template attacks.
    - Single sample of initialization with key.
    - State changes on each invocation.
  - Similar approach for most crypto algorithms.
- Other cases: hardware based DES, EM on SSL accelerators.

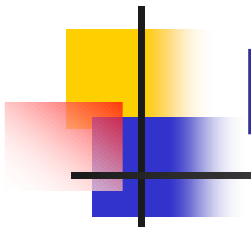
RC4- Initialization with key

```
i= j = 0;
For(ctr=0, ctr < 256, ctr++)
{
    j = key[i] + state[ctr]  + j;
    SwapByte(state[ctr] ,
    state[j] );
     i=i+1;
}
```

- Simple implementation with no key dependent code (No SPA)
- No DPA possible due to single sample.
- Ideal for template attacks: Key byte independently affects iteration.

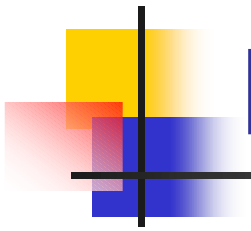
Sample





Methodology

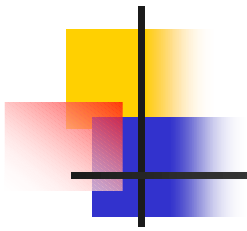
- Collect single sample of key initialization from device under test.
- With experimental device, collect large number(100s) of samples with all values of first key byte.
- Identify points on samples of first iteration directly affected by first key byte.
- For each distribution compute *precise statistical characterization*
- Use to classify target sample



Model

- *Assumption*: Gaussian model for noise.
- *Noise characterization*: Given L-point samples for a particular value of key compute
  - Averages L point average **A**
  - M Noise correlation matrix (L x L)
    - M[i,j] = covariance(T[i]-**A**[i], T[j]-**A**[j]) for samples T
- Compute characterization for each of K values of the key byte.
- Probability of observing noise vector ***n*** for a sample from this distribution is inverse exponential in

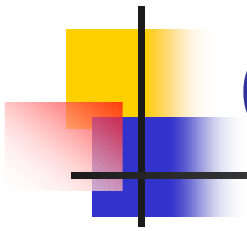
$$\boldsymbol{n}^{\mathbf{T}} M^{-1} \boldsymbol{n}$$



Maximum likelihood

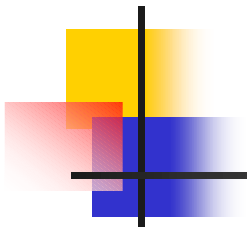
- *Classification*: Among K distributions, classify target sample *S* as belonging to distribution predicting highest probability for noise vector
- "Best" classifier in information theoretic sense.
- For binary hypotheses case, with same noise covariance error is inverse exponential in

$$\text{sqrt}\left((A_1 - A_2)^T N^{-1} (A_1 - A_2) \right)$$



Classification

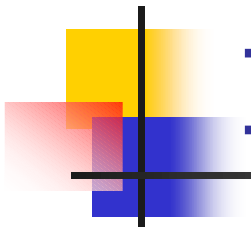
- Univariate Statistics
  - Assume sample at points is independent
  - Good results when keys are very different
  - Not good if keys are close.
- Multivariate statistics:
  - Assume points are correlated.
  - Very low classification errors.
  - Error of not identifying correct hypothesis is less than 5-6 %



Empirical result

Key byte	0xFE	0xEE	0xDE	0xBE	0x7E	0xFD	0xFB	0xF7	0xED	0xEB
	98.62	98.34	99.16	98.14	99.58	99.70	99.64	100	99.76	99.94

- Correct Classification percentage improves dramatically
- Keys chosen to be very close

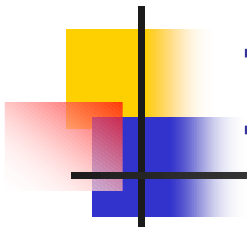


Improvement

- *Maximum Likelihood*: Retain hypothesis predicting max probability for observed noise (P_{max})
- *Approximation*: Retain ALL hypotheses predicting probability at least (P_{max}/c), c constant.
  - Retain more than 1 hypothesis for each byte.
  - Tradeoff between number of hypothesis retained and correctness.

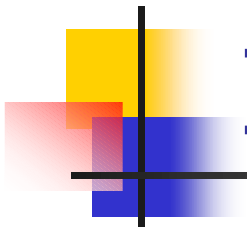
Empirical Results

	Size $c=1$	Size $c=e^{6}$	Size $c=e^{12}$	Size $c=e^{24}$
Success probability	95.02	98.67	99.37	99.65
Avg. number of hypothesis retained	1	1.29	2.11	6.89



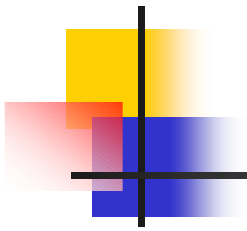
Iteration: Extend and prune

- For each remaining possible value of first byte
  - For each value of second byte
    - Build template independently ONLY for second iteration (less accurate)
    - OR Build template for first 2 iterations together (twice as large)
  - Classify using new template to reduce choices for first 2 bytes



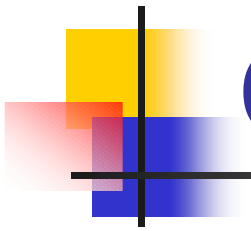
Iteration: Empirical Result

- Using templates independently in each stage reduces entropy in RC4 case to about $(1.5)^k$ for k bytes of key
- Substantially better when templates include sample for all iterations upto now
  - Error rates of not retaining correct hypothesis is almost same as single byte case.
  - Number of retained hypothesis is smaller
  - Able to correct previous bytes: After 2 iterations of attack no hypothesis with wrong first byte.



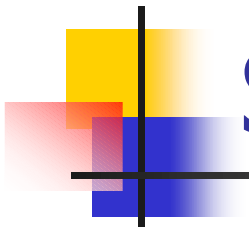
Related work

- [Messerges,Dabbish,Sloan][Walter] Use signal based iterative method based to extract exponent of device implementing RSA.
- [Fahn, Pearson] Use profiling of experimental device before attack on device-under-test.
- Signal based classification methods.



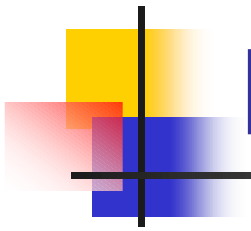
Countermeasures

- Use randomness as much as possible.
  - Blinding/masking of data
- Templates can be built for masked data values
  - Not feasible if lots of entropy.
  - Caveat: Vulnerable if attacker has control of random source in experimental device.



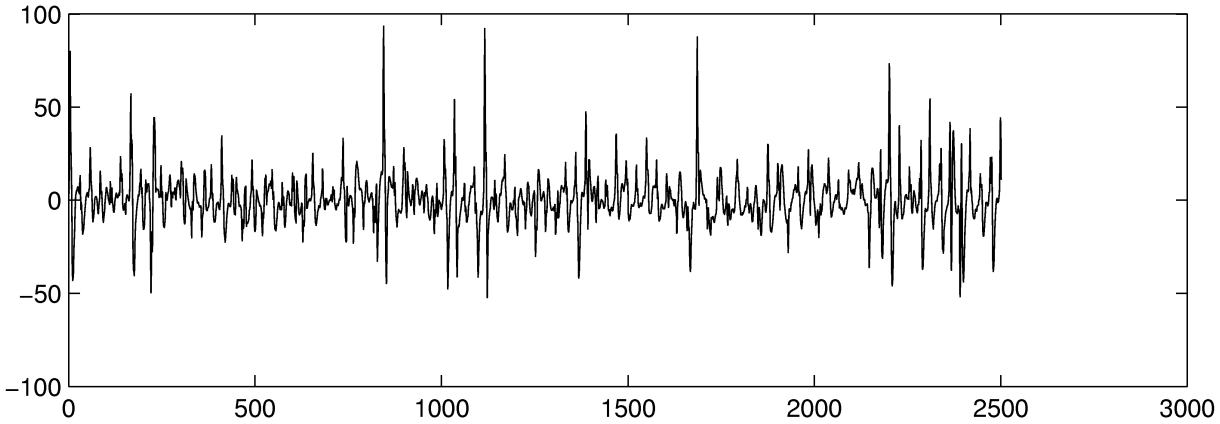
Summary

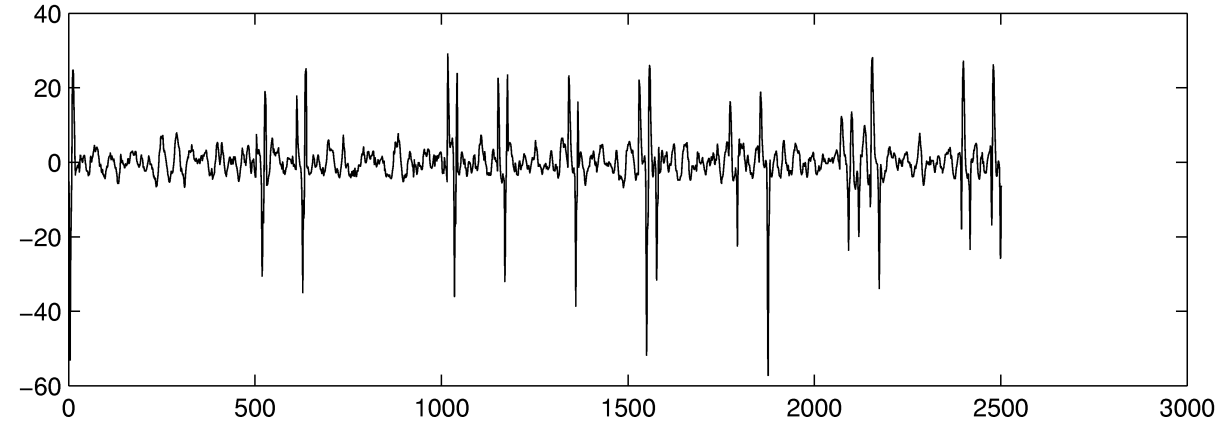
- Formalized new type of attack.
  - Powerful methodology
  - Works with single/few samples
  - Requires *extensive* work with experimental device
- Experimental results
  - Works where SPA/DPA are not feasible



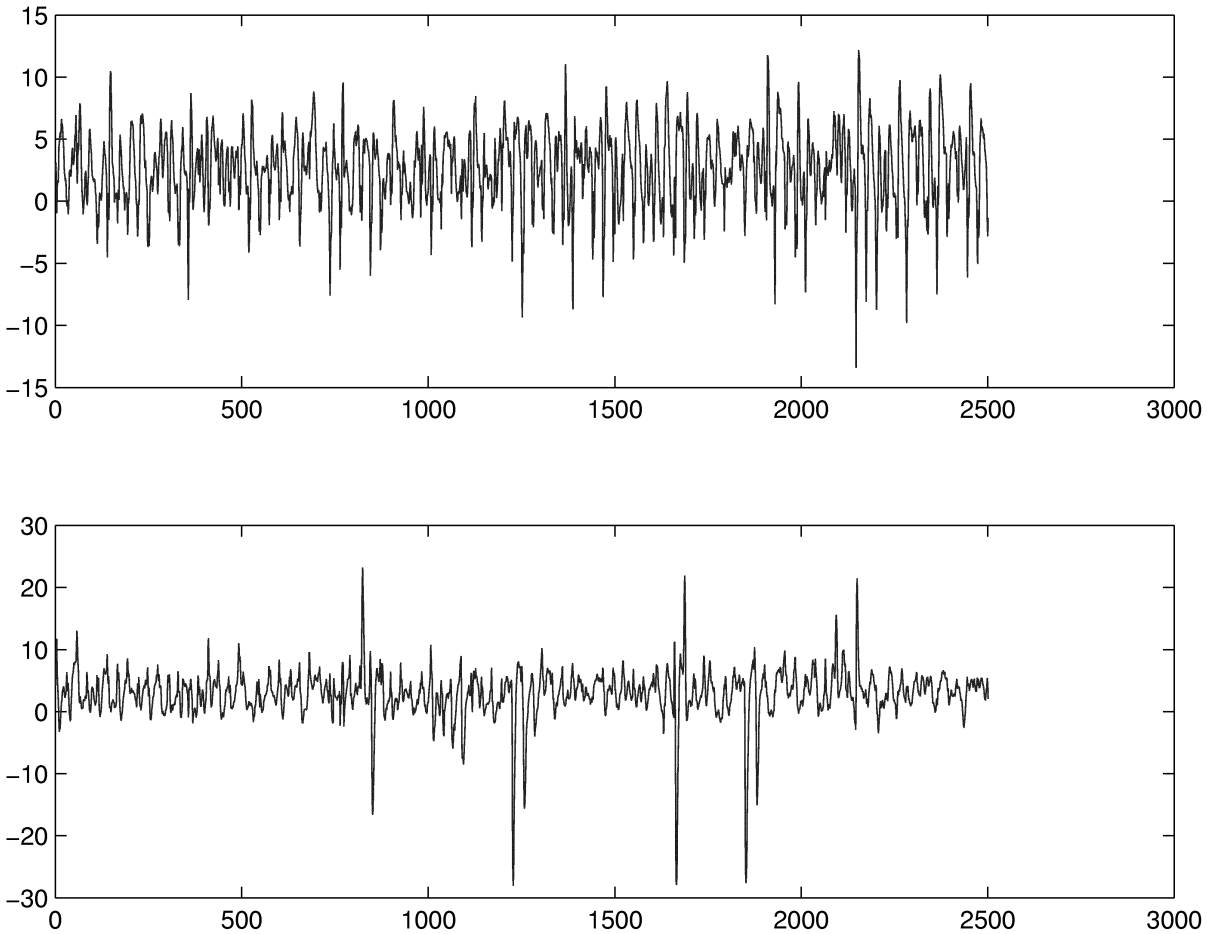
BACKUP

Averaging 5 samples





Averaging 50 samples



Univariate approximation

- Assume that the sample at each point is independent of other points
- Simplifies probability of observing noise
  - Inverse exponential in

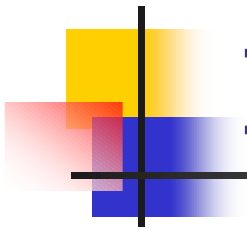
    $\boldsymbol{n}^{T} M^{-1} \boldsymbol{n}$ (which is just sum of squares)
- *Classification*: Use maximum likelihood with simplified characterization
- Classification error is high in some cases but can distinguish very different keys.

Empirical Results

	11111110	11101110	11011110	10111110	00010000
11111110	0.86	0.04	0.07	0.03	0
11101110	0.06	0.65	0.10	0.19	0
11011110	0.08	0.16	0.68	0.09	0
10111110	0.10	0.11	0.08	0.71	0
00010000	0	0	0	0	1.00

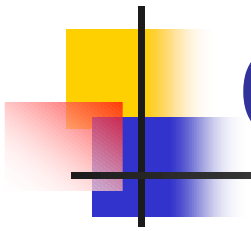
Cross Classification probability.

- Low error if key bytes have very different Hamming weights.
- Possibility of high error in other cases.



Intuition

- Samples and expected signals can be viewed as points in some L dimensional space.
- *Approximation*: Starting from received signal point keep all hypothesis falling in ball around received samples
- For binary case, classification error proportional to sqrt(1/c).



Other cases

- Template attacks verified in other cases
  - EM emanations from hardware SSL accelerators
    - Single sample noisy analogue of earlier work.
  - Hardware based DES
    - Attacking key checksum verification steps
- Other cases under investigation