NPTEL ONLINE CERTIFICATION COURSES

Course Name: Hardware Security

Faculty Name: Prof Debdeep Mukhopadhyay

Department : Computer Science and Engineering

Topic

Lecture 38: Power Analysis-XIV

CONCEPTS COVERED

Concepts Covered:

- Template Attacks
- Multivariate Normal Distribution and Noise
- Correlation in Power Traces
- Maximum Likelihood Decision Rule
- Numerical Instability and Reduced Templates
- Least Square Test (LSQ)

Template Attacks

- Strongest form of side channel attacks
- Require an adversary to have access to an identical experimental device that he can program to his choice.
- The success of these attacks relies on the ability to model noise.
- This helps to extract maximal knowledge from each sample.
- Threatens countermeasures, which rely on the fact that many samples would not be available to the adversary
  - In template attacks the adversary needs few, and ideally a single trace!

Multivariate Normal Distribution and Noise

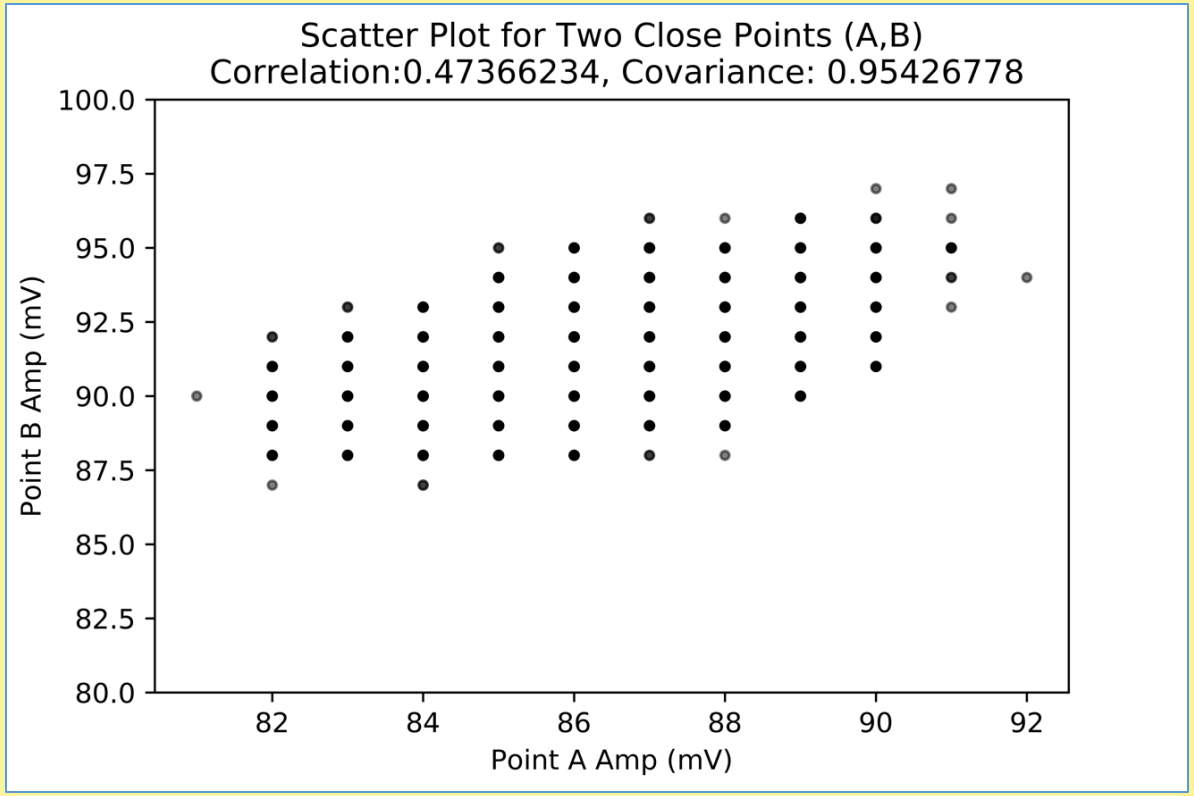
- We have seen that the electronic noise at every point of a power trace is normally distributed.
- However, this model does not take into account the correlation between neighboring points.
- In order to do so, we model the power trace as a multivariate normal distribution.
- This is a generalization of the normal distribution to higher dimensions.
- It can be described by a covariance matrix C and a mean vector m.

Definition of Multivariate Gaussian Model

- The probability density function (pdf.) of the multivariate normal distribution:
  - $f(\mathbb{x}) = \frac{1}{\sqrt{(2\pi)^n \cdot \det(\mathbb{C})}} . exp\left(-\frac{1}{2} \cdot (\mathbb{x} - \mathbb{m})' \cdot \mathbb{C}^{-1} \cdot (\mathbb{x} - \mathbb{m})\right)$
- The covariance matrix $\mathbb{C}$ contains the covariances $c_{ij} = Cov(X_i, X_j)$of the points at time index i and j.
- The mean vector $\mathbb{m}$ lists the mean values. $m_i = E(X_i)$ for all points in the curve.
- When filling $\mathbb{C}$ and $\mathbb{m}$ into the above equation, the pdf for the vector $\mathbb{x}$ is returned.

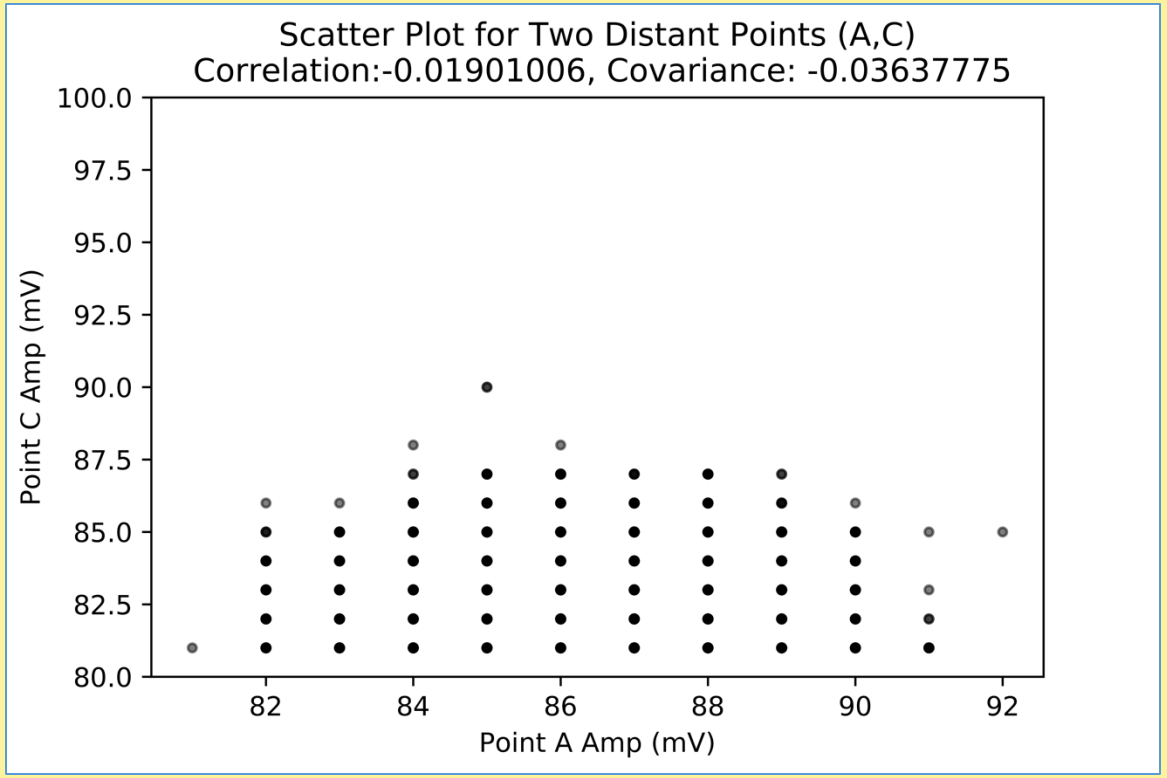
Correlation in Power Traces

- Electrical noise usually does not change significantly from one point of the power trace to the next.
- Hence, the electrical noise that is present in neighboring points is typically related.
- Scatter plot helps to visualize this fact.



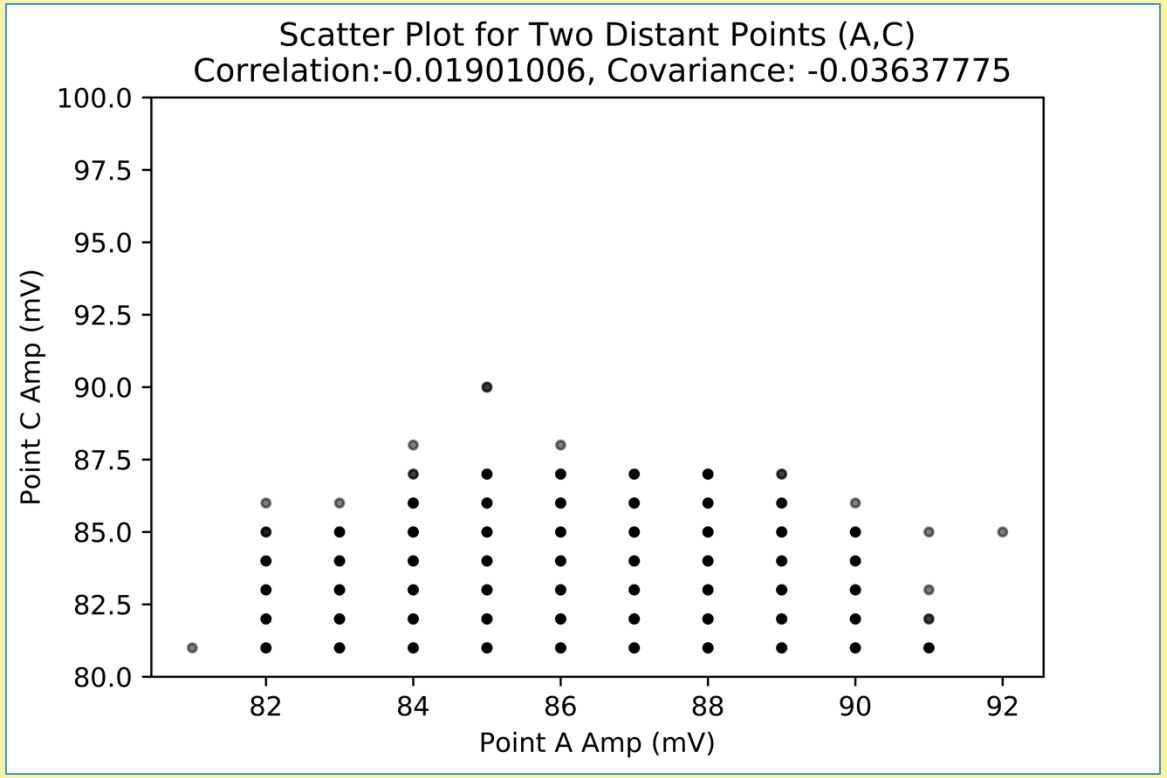
Correlation in Power Traces

- The linear relationship between two points of a trace is based on covariance or correlation.
- Cov(X,Y)=E(XY)-E(X)E(Y)
- Estimation of Covariance:
- Type equation here.

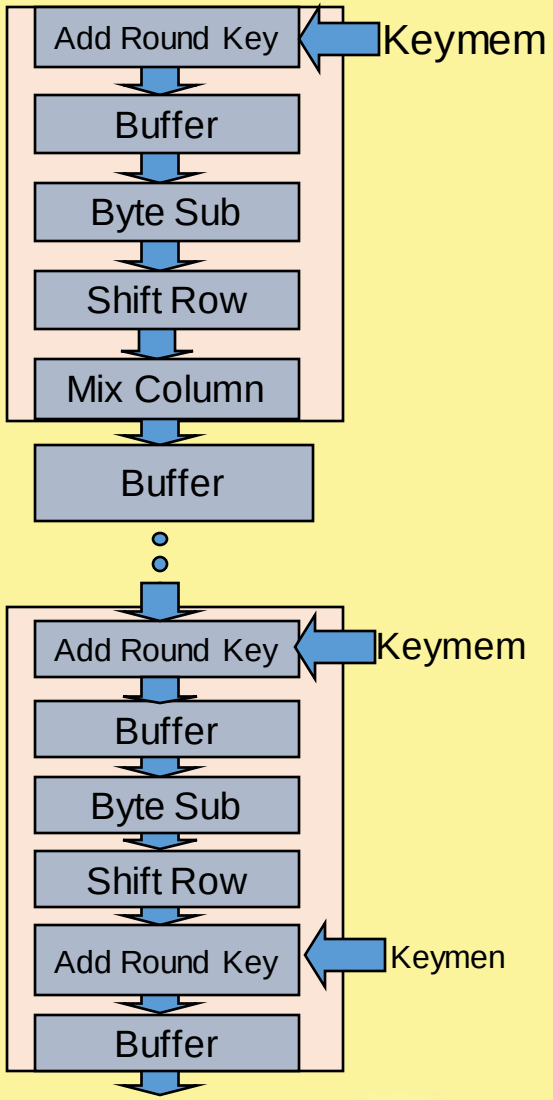


Correlation in Power Traces

- The linear relationship between two points of a trace is based on covariance or correlation.
- Correlation Coefficient is also a measure for the linear relationship between two adjacent points in the trace.
- $\rho(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}} = \frac{\Sigma_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\Sigma_{i=1}^{n}(x_i-\bar{x})^2\Sigma_{i=1}^{n}(y_i-\bar{y})^2}}$



The closer the points more is the correlation!



The AES Architecture and Template building Phase

We keep the plaintext same and vary say the first byte of the key matrix, K_0. Depending on the Hamming Weight of K_0, we create 9 templates.

Note that there are some templates where there are more values possible: Number of instances with Hamming Weight w is: $\binom{8}{w}$

Creating Templates

- We target say the key xoring step, by keeping the plaintext byte, P0 constant and changing the other 15 bytes.
- For each template again we keep K0 from a Hamming class say, w, and vary the other 15 key bytes.
- We target the first part of the power traces when the Add Round Key is participating in the underlying computation.
- We choose 10 time instances and calculate the mean vector $\mathbb{M} = \mathbb{m}$, and Covariance matrix $\mathbb{C} = \mathbb{c}$.
- Thus for every $(P0, K0)$: $T_{P0,K0} = (\mathbb{m}, \mathbb{c})_{P0,\ K0}$ is a template for K0 belonging to the Hamming Class w.

A Tale of Two Templates

- Let us create two templates with Hamming Weight 0 and 4.

```
Mean Vector for Class 0 - Hamming Weight 0
[31.08695652 31.56521739 50.91304348 60.86956522 51.7826087  41.69565217 46.73913043 59.56521739 64.04347826
58.26086957]

Covariance Matrix for Class 0 - Hamming Weight 0
[[2.90118577 2.22134387 2.05335968 1.87549407 1.79249012 2.70948617 3.06916996 2.13043478 1.54150198 1.88537549]
 [2.22134387 4.16600791 4.5513834  2.80434783 1.44664032 2.63438735 4.10869565 3.62055336 2.20158103 1.98221344]
 [2.05335968 4.5513834  6.71936759 3.94268775 2.11660079 2.83596838 5.06719368 4.5513834  3.04940711 2.38735178]
 [1.87549407 2.80434783 3.94268775 3.39130435 2.06126482 2.00395257 3.46442688 3.21343874 2.64229249 1.94466403]
 [1.79249012 1.44664032 2.11660079 2.06126482 3.17786561 1.24901186 2.34980237 2.08300395 2.23715415 1.9229249 ]
 [2.70948617 2.63438735 2.83596838 2.00395257 1.24901186 4.03952569 3.82608696 2.36166008 1.65019763 1.94664032]
 [3.06916996 4.10869565 5.06719368 3.46442688 2.34980237 3.82608696 5.29249012 4.01778656 2.96640316 2.61660079]
 [2.13043478 3.62055336 4.5513834  3.21343874 2.08300395 2.36166008 4.01778656 4.07509881 2.83794466 2.11857708]
 [1.54150198 2.20158103 3.04940711 2.64229249 2.23715415 1.65019763 2.96640316 2.83794466 3.22529644 1.8972332 ]
 [1.88537549 1.98221344 2.38735178 1.94466403 1.9229249 1.94664032 2.61660079 2.11857708 1.8972332  2.29249012]]
```

Template for Hamming Weight 4

```
Mean Vector for Class 1 - Hamming Weight 4
[31.84227642 31.93333333 50.49430894 61.27100271 52.58373984 42.69376694 46.9696477  59.66233062 64.62547425
58.69322493]

Covariance Matrix for Class 1 - Hamming Weight 4
[[7.18172583 4.63698482 2.56824507 3.32747385 5.11652294 5.52488404 3.98544874 2.62838298 3.1274474  4.57197855]
 [4.63698482 6.28893709 4.90607375 3.3087491  3.15607375 4.61677513 5.11836587 4.1461316  2.97230658 3.07281273]
 [2.56824507 4.90607375 6.21214221 3.42182512 1.98136077 3.12217167 4.58062977 4.52220958 3.05345308 2.17721196]
 [3.32747385 3.3087491  3.42182512 4.18899059 3.39974516 3.1611261 2.97731895 3.07257846 3.18072824 3.1314466 ]
 [5.11652294 3.15607375 1.98136077 3.39974516 5.53270726 4.35076627 2.72597041 2.11912685 3.13902615 4.24913056]
 [5.52488404 4.61677513 3.12217167 3.1611261  4.35076627 5.72558477 4.1436285  2.88677701 2.89066863 3.94884603]
 [3.98544874 5.11836587 4.58062977 2.97731895 2.72597041 4.1436285 5.40254895 3.9040621  2.70934195 2.72549924]
 [2.62838298 4.1461316  4.52220958 3.07257846 2.11912685 2.88677701 3.9040621  4.47756615 2.77910546 2.15232057]
 [3.1274474  2.97230658 3.05345308 3.18072824 3.13902615 2.89066863 2.70934195 2.77910546 3.6042314  2.92734188]
 [4.57197855 3.07281273 2.17721196 3.1314466 4.24913056 3.94884603 2.72549924 2.15232057 2.92734188 4.45139117]]
```

The Attack Phase

- Later on we use these template characterizations to identify the unknown key bytes from a power trace from the device under attack.
- This means, we evaluate the probability density function of the multivariate normal distribution $(\mathbb{m}, \mathbb{c})_{P0,\ K0}$ and the power trace of the device under attack.
  - Given a power trace $\boldsymbol{t}$ of the device under attack, and a template $(\mathbb{m}, \mathbb{c})_{P0,\ K0}$ we compute the probability:

$$\boldsymbol{p\left(t; (\mathbb{m}, \mathbb{c})_{P0,K0}\right) = \frac{\exp\left(-\frac{1}{2}(t-\mathbb{m})'\mathbb{c}^{-1}(t-\mathbb{m})\right)}{\sqrt{(2\pi)^n \det(\mathbb{c})}}}$$

Note that t is accumulated by keeping P0 say fixed and choosing arbitary values for the remaining 15 plaintext bytes. The key is uknown here and we have no handle.

In a setting, where we don't have access to the input also, we can target the output of $P0 \oplus K0$ for building the templates.

Maximum Likelihood Decision Rule

- The decision rule which maximizes the probability is a potential candidate for the key byte K0.
- That is, if: $\boldsymbol{p}\left(\boldsymbol{t}; (\mathbb{m}, \mathbb{c})_{\mathbf{P0,K0}}\right) > \boldsymbol{p}\left(\boldsymbol{t}; (\mathbb{m}, \mathbb{c})_{\mathbf{P0,K1}}\right)$, then the key byte is returned as K0, over K1

Case Study for Unknown Key

- Unknown key has a K0 value of Hamming Weight 4
- We compute the probabilities as mentioned wrt. the templates for 0 and 4.
- Hamming Weight 0 Prob: 0.00000725
- Hamming Weight 4 Prob: 0.00001790
- This shows that it more likely that K0 has a Hamming Weight it 4, which is indeed correct!

For Another Few Runs

- Trial 2:
  - Hamming Weight 0 Prob: 0.00000000
  - Hamming Weight 4 Prob: 0.00000336
- Trial 3:
  - Hamming Weight 0 Prob: 0.00000000
  - Hamming Weight 4 Prob: 0.00000228

- In all cases Hamming Weight 4 has a more likelihood from the template analysis

Numeric Problems in Template Analysis

- While performing the template analysis, we can get into 2 important numeric problems:
  - These are related to the covariance matrix.
  - Size of the matrix depends on the number of important points, say n.
  - This must be chosen carefully.
  - There can also be problems arising out of the requirement of the existence of its inverse.
  - Also the exponent tends to be smaller and leading to inaccuracies.

Numeric Manipulations

- Let us take the logarithm of the value $\boldsymbol{p\left(t;(\mathbb{m},\mathbb{c})_{P0,K0}\right)}$
- Thus,

$$\ln \boldsymbol{p\left(t;(\mathbb{m},\mathbb{c})_{P0,K0}\right)}$$
$$= -\frac{1}{2}\left(\boldsymbol{ln\left((2\pi)^{n}det(C)\right)} + (t-\mathbb{m})'\mathbb{c}^{-1}(\mathrm{t}-\mathbb{m})\right)$$

The template that leads to the smallest absolute value of the logarithm of the probability indicates the correct key.

- Again if, $\ln \boldsymbol{p\left(t;(\mathbb{m},\mathbb{c})_{P0,K0}\right)} < \ln \boldsymbol{p\left(t;(\mathbb{m},\mathbb{c})_{P0,Kl}\right), \forall l \neq 0}$, then K0 is the predicted class.

Reduced Templates

- To further avoid problems with the covariance matrix we set covariance to the identity matrix:
  - We are neglecting the effect of the covariances between the points in the window we are observing.
  - This template is called reduced template.
- Thus we have,

$$p(t; \mathbb{m}) = \frac{\exp(-\frac{1}{2}(t - \mathbb{m})'(t - \mathbb{m}))}{\sqrt{(2\pi)^n}}$$

- Again taking logarithms,

$$\ln p(t; \mathbb{m}) = -\frac{1}{2}\left(ln((2\pi)^n) + (t - \mathbb{m})'(\mathrm{t} - \mathbb{m})\right)$$

Least Square Test

- The method using reduced templates is also called Least-Square Test (LSQ)
  - It is because the only relevant term in the probability estimate is now the square of the difference of t and $\mathbb{m}$.
- Thus if

$$\left(\boldsymbol{t}-\mathbb{m}_{\boldsymbol{P0,K0}}\right)'\left(\mathrm{t}-\mathbb{m}_{\boldsymbol{P0,K0}}\right)<\left(\boldsymbol{t}-\mathbb{m}_{\boldsymbol{P0,Kl}}\right)'\left(\mathrm{t}-\mathbb{m}_{\boldsymbol{P0,Kl}}\right),\forall \boldsymbol{l}\neq \mathbf{0},$$

then K0 is the predicted class.

LSQ in our Case Study

- Trial 1:
  - Hamming Weight 0 Squared Difference: 19.07939509
  - Hamming Weight 4 Squared Difference: 32.18028070
- Trial 2:
  - Hamming Weight 0 Squared Difference: 57.86200378
  - Hamming Weight 4 Squared Difference: 36.24640536
- Trial 3:
  - Hamming Weight 0 Squared Difference: 94.03591682
  - Hamming Weight 4 Squared Difference: 66.37973869

Conclusions

Correlation across points in the trace leads to a multivariate normal distribution of noise.

Template attacks try to accurately model this noise.

Template matching is based on the maximum likelihood principle.

Reduced Templates help in performing template analysis with less numerical insgtability.

References:

1. Stefan Mangard, Elisabeth Oswald, Thomas Popp, Power Analysis Attacks: Revealing the Secrets of Smart Cards, Springer.

2. Suresh Chari, Joysula Rao and Pankaj Rohatgi, Template Attacks, CHES 2002.

3. Christian Rechberger, Elisabeth Oswald, Practical Template Attacks.

FREE ONLINE EDUCATION
swayam
शिक्षित भारत, उन्नत भारत

NPTEL ONLINE CERTIFICATION COURSES

Thank you!