Information Security – Theory vs. Reality

0368-4474, Winter 2013-2014

Lecture 6:
Machine Learning Approach to Side-Channel Attacks

Guest lecturer: Guy Wolf
Machine Learning Approach to Side-Channel Attacks

Guy Wolf
Tel Aviv University
guy.wolf@cs.tau.ac.il
2013
Outline

1. Introduction
   - Side-Channel Leaks
   - Machine Learning

2. Quick Recap: Correlation Power Analysis

3. Alternative Attack: Template Power Analysis
   - Dimensionality Reduction
   - Classification
   - Power Trace Alignment

4. Information & Activity Leaks
   - Hidden Markov Model
   - Acoustic Analysis

5. Conclusion
Side-Channel Leaks
Side-Channel Leaks

Power Consumption
Side-Channel Leaks

- Power Consumption
- Probing

Machine Learning & Side-Channels

Guy Wolf (TAU)
Side-Channel Leaks

Power Consumption

Probing

Response Times
Side-Channel Leaks

- Power Consumption
- Probing
- Response Times
- Acoustics
Side-Channel Leaks

- Power Consumption
- Probing
- Response Times
- Acoustics
- Electromagnetic Radiation
Side-Channel Leaks

- Power Consumption
- Probing
- Response Times
- Temperatures
- Acoustics
- Electromagnetic Radiation
Processing Leaked Data
Processing Leaked Data

leaked data

⊆ R

Guy Wolf (TAU)

Machine Learning & Side-Channels
Processing Leaked Data

leaked data

\[ \subseteq \mathbb{R}_O(100+\ldots) \]
Processing Leaked Data

leaked data

⊆ \mathbb{R}^{100}
Processing Leaked Data

leaked data

⊆ \mathbb{R}^{O(100+)}
Processing Leaked Data

\[ \text{leaked data} \subseteq \mathbb{R}^{O(100^+)} \]
Processing Leaked Data

leaked data \subseteq \mathbb{R}^{O(100^+)}
Processing Leaked Data
Processing Leaked Data
Machine Learning

Machine learning encompasses tools that perform smart analysis of data, such as:

- Discovery of useful, possibly unexpected, patterns in data
- Non-trivial extraction of implicit, previously unknown and potentially useful information from data
- Exploration & analysis, by automatic or semi-automatic means, of large quantities of data in order to discover meaningful patterns

Common tasks include:

- Dimensionality reduction
- Clustering & Classification
- Regression & Out-of-sample extension
Quick Recap:
Correlation Power Analysis
Quick Recap: Correlation Power Analysis

Key $k$ → Internal value $y = f(k, p)$ → Ciphertext

Plaintext $p$ →

Typical assumption:

power consumption correlates with Hamming Weight of $y$
Quick Recap: Correlation Power Analysis

Plaintexts: Traces: Internal Vals.: Hamm. Wts.:

$p_1$ → 

\[ \begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array} \]

→ 00101001 → 3

$p_n$ → 

\[ \begin{array}{c}
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\vdots \\
\end{array} \]

→ 10011011 → 5
Quick Recap: Correlation Power Analysis

Traces matrix:

<table>
<thead>
<tr>
<th>plaintexts</th>
<th>times</th>
</tr>
</thead>
<tbody>
<tr>
<td>trace $x_1$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>trace $x_n$</td>
<td></td>
</tr>
</tbody>
</table>

Hamming Weight matrix:

<table>
<thead>
<tr>
<th>key candidates</th>
</tr>
</thead>
<tbody>
<tr>
<td>plaintexts</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
Quick Recap: Correlation Power Analysis

Traces matrix:

<table>
<thead>
<tr>
<th>plaintexts</th>
<th>times</th>
<th>plaintexts</th>
<th>times</th>
</tr>
</thead>
<tbody>
<tr>
<td>trace $x$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>trace $x_n$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hamming Weight matrix:

<table>
<thead>
<tr>
<th>plaintexts</th>
<th>times</th>
<th>key candidates</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hamming weight of internal value for designated plaintext and key candidate

Pearson Coefficient: linear correlation

Mutual Information: general statistical correlation
Quick Recap: Correlation Power Analysis

Traces matrix:

Hamming Weight matrix:

Correlation indicates the correct time and key
Quick Recap: Correlation Power Analysis

Traces matrix:

<table>
<thead>
<tr>
<th>plaintexts</th>
<th>times</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Hamming Weight matrix:

<table>
<thead>
<tr>
<th>key candidates</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Pearson Coefficient:
linear correlation \( \frac{\text{cov}(x, y)}{\text{std}(x)\text{std}(y)} \)

Mutual Information:
general statistical correlation \( (\text{Entropy}(x) + \text{Entropy}(y) - \text{Entropy}(x, y)) \)

Correlation indicates the correct time and key
Alternative Attack: Template Power Analysis
Alternative Attack: Template Power Analysis

Training: Learn traces of many plaintexts & keys

<table>
<thead>
<tr>
<th>plaintexts &amp; keys</th>
<th>times</th>
</tr>
</thead>
<tbody>
<tr>
<td>~~~~~~~~~ trace</td>
<td>~~~~~</td>
</tr>
<tr>
<td>~~~~~~~~~ trace</td>
<td>~~~~~</td>
</tr>
<tr>
<td>~~~~~~~~~ trace</td>
<td>~~~~~</td>
</tr>
<tr>
<td>~~~~~~~~~ trace</td>
<td>~~~~~</td>
</tr>
<tr>
<td>~~~~~~~~~ trace</td>
<td>~~~~~</td>
</tr>
<tr>
<td>~~~~~~~~~ trace</td>
<td>~~~~~</td>
</tr>
</tbody>
</table>

Online: Attack using a single trace
Alternative Attack: Template Power Analysis

Training: Learn traces of many plaintexts & keys

plaintext = p_1

times

<table>
<thead>
<tr>
<th>trace</th>
<th>trace</th>
<th>trace</th>
<th>trace</th>
<th>trace</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

key = k_0

key = k_1

Online: Attack using a single trace
Alternative Attack: Template Power Analysis

Training: Learn traces of many plaintexts & keys

times

plaintext = \( p_1 \)

\( = k_0 \)

Online: Attack using a single trace
Alternative Attack: Template Power Analysis

Representing power traces as vectors

Essentially, power traces are high-dimensional vectors with
\( m = O(100+) \) of measurements.

\[
\text{trace} \times \text{trace} \equiv (x[1], x[2], \ldots, x[m]) \in \mathbb{R}^m
\]

Traces can be analyzed as vectors in the Euclidean space \( \mathbb{R}^m \) with \( \ell_2 \) norms & distances

- Somewhat arbitrary representation, but effective and convenient
Alternative Attack: Template Power Analysis

Learning & analyzing trace vectors

Template assumption: the positions of traces in $\mathbb{R}^m$ (with $\ell_2$ norm) correlate with the plaintext & key values that generated them.

- Correlation is either directly seen or via some internal value
- Ideally, correlation is expressed as clustering by plaintext & key

Theoretically, machine learning tools can be used to detect clustering...
Alternative Attack: Template Power Analysis

Example: classic TPA attack using Gaussian statistical modeling

Model power consumption of encrypting plaintext $p$ with key $k$ as a random variable $\vec{X}_{(k,p)} \sim \mathcal{N}(\vec{\mu}_{(k,p)}, \Sigma_{(k,p)})$

$\vec{X}_{(k,p)}$ is drawn from a multidimensional normal (Gaussian) distribution

$\vec{\mu}_{(k,p)} \in \mathbb{R}^m$ is the mean power consumption for $k$ & $p$

$\Sigma_{(k,p)}$ is the $m \times m$ noise covariance matrix for $k$ & $p$

The likelihood of a trace $\vec{x}$ originating from $k$ & $p$ is

$$
\mathcal{L}_{(k,p)}(\vec{x}) = \left( (2\pi)^m |\Sigma_{(k,p)}| \right)^{-1/2} \exp \left( -\frac{1}{2} (\vec{x} - \vec{\mu}_{(k,p)})^T \Sigma_{(k,p)}^{-1} (\vec{x} - \vec{\mu}_{(k,p)}) \right)
$$

Attack single trace (with known plaintext $p$):

- Compute likelihood of it originating from each key candidate (with $p$)
- Choose key candidate with maximum likelihood

---

$^1$“Template Attacks”, 2003, by S. Chari, J.R. Rao, and P. Rohatgi
Unfortunately, directly analyzing high-dimensional vectors is usually unfeasible due to the “Curse of Dimensionality”.

**Curse of Dimensionality**

A general term for various phenomena that arise when analyzing/organizing high-dimensional data.

- Common theme - difficult/impractical/impossible to obtain statistical significance due to sparsity of the data in high-dimensions
- Causes poor performance/results of statistical methods compared to low-dimensional data

Common solution - use **dimensionality reduction** methods and analyze their resulting embedded space.

Example: only use the $\ell < m$ time indices that provide the highest differences between mean power consumptions of different key-plaintext pairs.
Dimensionality Reduction
with
Principal Component Analysis
Dimensionality Reduction
Principal Component Analysis (PCA)

Assume: \( \text{avg} = 0 \)
Find: max variance directions

3D space
Dimensionality Reduction

Principal Component Analysis (PCA)

Assume: $\text{avg} = 0$

Find: max variance directions
Dimensionality Reduction

Principal Component Analysis (PCA)

Assume:
\[ \text{avg} = 0 \]

Find:
max variance directions

3D space
Dimensionality Reduction
Principal Component Analysis (PCA) - covariance matrix

$$\text{cov}(t_1, t_2) \triangleq \sum_i \text{trace}_i[t_1] \cdot \text{trace}_i[t_2]$$
Spectral theorem applies to cov. matrices:

SVD (Singular Value Decomposition)

Spectral Theorem: \( \text{COV}(t_1, t_2) = \sum_i \lambda_i \phi_i(t_1) \phi_i(t_2) \)
Dimensionality Reduction
Principal Component Analysis (PCA) - truncated SVD

Many datasets (incl. power traces) have a decaying cov. spectrum
Dimensionality Reduction

Principal Component Analysis (PCA) - truncated SVD

Many datasets (incl. power traces) have a decaying cov. spectrum

Covariance matrix

Eigenvectors

principal components

Eigenvalues

Approximate cov. matrix by truncating small eigenvalues from SVD
Dimensionality Reduction
Principal Component Analysis (PCA) - example

Consider simple case of traces that are all on the same high dimensional line

- Straight line is defined by a unit vector $\|\vec{\psi}\| = 1$
- Points on the line are defined by multiplying $\vec{\psi}$ by scalars
- The traces can be formulated as $x_i = c_i \vec{\psi}$
- Covariance: $\text{cov}(t_1, t_2) = \sum x_i[t_1] x_i[t_2] = \sum c_i \vec{\psi}[t_1] c_i \vec{\psi}[t_2] = (\sum_i c_i^2) \vec{\psi}[t_1] \vec{\psi}[t_2] = \|\vec{c}\|^2 \vec{\psi}(t_1) \vec{\psi}(t_2) \quad \vec{c} \triangleq (c_1, c_2, \ldots)$
Consider simple case of traces that are all on the same high dimensional straight line.

\[ \| \vec{c} \| \vec{\psi} = 1 \]

Points on the line are defined by multiplying \( \vec{\psi} \) by scalars.

The traces can be formulated as

\[ x_i = c_i \vec{\psi} \]

Covariance:

\[ \text{cov}(t_1, t_2) = \sum_i x_i [t_1] x_i [t_2] = \sum_i c_i \vec{\psi} [t_1] c_i \vec{\psi} [t_2] = (\sum_i c_i^2) \vec{\psi} [t_1] \vec{\psi} [t_2] = \| \vec{c} \|^2 \vec{\psi} (t_1) \vec{\psi} (t_2) \]

\[ \vec{c} \equiv (c_1, c_2, \ldots) \]

Covariance matrix has a single eigenvalue \( \| \vec{c} \|^2 \) and a single eigenvector \( \vec{\psi} \), which defines the principal direction of the trace vectors.
Dimensionality Reduction

Principal Component Analysis (PCA) - example

Consider simple case of traces that are all on the same high dimensional line

- Straight line is defined by a unit vector $\|\vec{\psi}\| = 1$
- Points on the line are defined by multiplying $\vec{\psi}$ by scalars
- The traces can be formulated as $x_i = c_i\vec{\psi}$
- Covariance: $\text{cov}(t_1, t_2) = \sum x_i[t_1]x_i[t_2] = \sum c_i\vec{\psi}[t_1]c_i\vec{\psi}[t_2] = (\sum c_i^2)\vec{\psi}[t_1]\vec{\psi}[t_2] = \|\vec{c}\|^2 \vec{\psi}(t_1)\vec{\psi}(t_2)$
  \[\vec{c} \triangleq (c_1, c_2, \ldots)\]

Covariance matrix has a single eigenvalue $\|\vec{c}\|^2$ and a single eigenvector $\vec{\psi}$, which defines principal direction of the trace-vectors
Dimensionality Reduction
Principal Component Analysis (PCA) - example

3D space

$\phi_1 = \psi_1$

Length: eigenvalues
Direction: eigenvectors

Principal components $\Rightarrow$ max var directions
Dimensionality Reduction
Principal Component Analysis (PCA) - example

3D space

$\phi_1 = \psi$

Length: eigenvalues
Direction: eigenvectors
principal components $\Rightarrow$ max var directions
Dimensionality Reduction
Principal Component Analysis (PCA) - example

3D space

Length: eigenvalues
Direction: eigenvectors

Principal components ⇒ max var directions

Guy Wolf (TAU)
Machine Learning & Side-Channels
2013 22 / 61
Dimensionality Reduction
Principal Component Analysis (PCA) - example

**Length**: eigenvalues
**Direction**: eigenvectors

3D space

principal components $\Rightarrow$ max var directions
Projection on principal components:

\[
\begin{align*}
\text{Traces} & \mapsto \text{1D space} \\
\text{Principal components} & \mapsto \text{Traces}
\end{align*}
\]
Dimensionality Reduction
Principal Component Analysis (PCA) - projection

Projection on principal components:

3D space

$\lambda_1 \phi_1$

1D space
Dimensionality Reduction

PCA algorithm

**PCA algorithm:**

1. Centering
2. Covariance
3. Eigendecomposition
4. Projection

**Alternative method:** Multi-Dimensional Scaling (MDS) - preserve distances/inner-products with minimal set of coordinates.

Short tutorial on PCA & MDS:

www.cs.haifa.ac.il/~rita/uml_course/lectures/PCA_MDS.pdf
Dimensionality Reduction

Summary

Traces (high-dim. vectors):

Projected traces (low-dim. vectors):

PCA
Dimensionality Reduction

Summary

Traces (high-dim. vectors):

Projected traces (low-dim. vectors):

Next task: how to find keys from the low-dimensional vectors?
Clustering & Classification with Support Vector Machine
Cluster analysis

*Clustering* - the task of grouping objects such that objects in the same cluster are more similar to each other than to those in other clusters.
Classification

Clustering

Cluster analysis

Clustering - the task of grouping objects such that objects in the same cluster are more similar to each other than to those in other clusters.

Learning types

Unsupervised learning: Trying to find hidden structures in unlabeled data.

Supervised learning: Inferring functions from labeled training data.
Classification
Clustering & classification approaches

Classic TPA using Gaussian statistical models:

- The analysis considers many clusters (one for each key-plaintext pair)
- Clusters are assumed to correlate with normally distributed random variables \( \tilde{X}_{(k,p)} \sim \mathcal{N}(\tilde{\mu}_{(k,p)}, \Sigma_{(k,p)}) \)
- Requires many traces for each key-plaintext pair to compute \( \tilde{\mu}_{(k,p)} \) & \( \Sigma_{(k,p)} \)

Simplified bit clustering with Support Vector Machine (SVM):

- Classify each bit separately - only two classes are considered for each bit
- Requires less training traces than classic TPA - traces are grouped by bit values, not by the key value
- No statistical assumptions required - geometric classification using a separating hyperplane
Classification

Simplified bit clustering - only two classes

\[ (p_0, k_0) \]
\[ (p_1, k_1) \]
\[ (p_2, k_2) \]
\[ (p_3, k_3) \]
Classification
Support Vector Machine (SVM) - separation with hyperplane
Classification

Support Vector Machine (SVM) - separation with hyperplane

\[ b = 0 \]

\[ b = 1 \]
Classification
Support Vector Machine (SVM) - separation with hyperplane

\[ b = 0 \]

\[ b = 1 \]
Classification

Support Vector Machine (SVM) - separation with hyperplane

\[ b = 0 \]

\[ b = 1 \]
Classification

Support Vector Machine (SVM) - separation with hyperplane

\[ b = 0 \]

\[ b = 1 \]
Classification
Support Vector Machine (SVM) - separation with hyperplane

\[ b = 0 \]

\[ b = 1 \]
Classification
Support Vector Machine (SVM) - quantifying robustness with margins

\[ b = 0 \]

\[ b = 1 \]
Classification
Support Vector Machine (SVM) - quantifying robustness with margins

\[ b = 0 \]

\[ b = 1 \]
Classification

SVM formulation - hyperplane

\[ \vec{w} \cdot \vec{x} > 0 \]

\[ \vec{w} \cdot \vec{x} < 0 \]
Classification
SVM formulation - hyperplane with margin

\[ \mathbf{w} \cdot \mathbf{x} \geq \alpha \]

\[ \mathbf{w} \cdot \mathbf{x} \leq -\alpha \]
Classification

SVM formulation - shifted hyperplane with margin

\[ c = \vec{w} \cdot \vec{u} \]

\[
\begin{align*}
\vec{w} \cdot \vec{x} - c & \geq \alpha \\
\vec{w} \cdot \vec{x} - c & \leq -\alpha
\end{align*}
\]
SVM training

Input:
- Points \( \{\vec{x}_i\} \) from PCA of the traces
- Labels \( \{b_i\} \) according to attacked bit:
  \[
  b_i = \begin{cases} 
  1 & \text{bit is 0} \\
  -1 & \text{bit is 1}
  \end{cases}
  \]

Solve the quadratic program (e.g., using Lagrange multipliers):

Find \( \max \alpha \) such that \( \vec{w} \cdot \vec{x}_i - c \geq b_i \alpha \)

Output: the solution \( (\vec{w}, c, \alpha) \)
Classification

SVM algorithm

**SVM classifier**

**Input:**
- New point $\vec{x}$ from PCA projection of attacked trace
- The solution $(\vec{w}, c, \alpha)$ from SVM training

Classify by value of $\vec{w} \cdot \vec{x} - c$:

- **attacked bit is 0**
  \[ -\alpha \]
  attacked bit is probably 0

- **attacked bit is probably 1**
  \[ \alpha \]

- **attacked bit is probably 1**
  \[ \alpha \]
  attacked bit is 1
Power Analysis with PCA & SVM
based on recent paper²

Use single PCA & multiple SVMs (one per bit) to learn traces (in training phase) and attack a key byte

²“Side channel attack: an approach based on machine learning”, 2011, by L. Lerman, G. Bontempi, and O. Markowitch
Power Analysis with PCA & SVM
based on recent paper

PCA results (colored by specified bit values):

Effective SVM attack:  

SVM attack will fail:

Empirical results (on 3DES): desc. success rate (by bit position)

7th bit success rate: $\sim 95\% \rightarrow$ 1st bit success rate: $\sim 50\%$

2“Side channel attack: an approach based on machine learning”, 2011, by L. Lerman, G. Bontempi, and O. Markowitch
Power Trace Alignment with DTW
Power traces can be misaligned for several reasons, such as

- Synchronization issues between the sampling devices and the tested hardware
- Clock variabilities and instabilities
- Intentional countermeasures such as delays and modulations

Misaligned traces $\Rightarrow$ incorrect/inaccurate correlations $\Rightarrow$ wrong classification and useless attacks
Power Trace Alignment

Naïve approach: static alignment by time offset

Theoretically:
Power Trace Alignment

Naïve approach: static alignment by time offset

Theoretically:
Use time offset to align traces
Power Trace Alignment

Naïve approach: static alignment by time offset

Realistically:
Power Trace Alignment
Naïve approach: static alignment by time offset

Realistically:

Which offset to use?
Power Trace Alignment
Machine-learning approach: adaptive alignment by Dynamic Time Warp (DTW)
Power Trace Alignment
Machine-learning approach: adaptive alignment by Dynamic Time Warp (DTW)
Power Trace Alignment

Machine-learning approach: adaptive alignment by Dynamic Time Warp (DTW)
Power Trace Alignment
Using pairwise alignment in an attack

Training:
1. Acquire power traces
2. Choose reference trace (e.g., arbitrarily or use mean of all traces)
3. Align each trace to the reference trace using the pairwise alignment
4. Apply training algorithm (e.g., PCA & SVM) to the aligned traces

Online:
1. Acquire trace from attacked hardware
2. Align trace to the reference trace (from the training) using pairwise alignment
3. Apply classification algorithm (e.g., PCA & SVM)
Power Trace Alignment

Pairwise alignment

Trace $x$

Trace $y$

Pairwise diff. matrix: each cell holds difference between two trace entries

Pairwise alignment

$\mathbf{x}[i] - \mathbf{y}[j]$
Power Trace Alignment

Pairwise alignment

\[\text{Pairwise diff. matrix: each cell holds difference between two trace entries}\]

\[\text{Alignment path: get from start to end of both traces}\]

\[\text{1:1 alignment: trivial - nothing modified by the alignment}\]

\[\text{Aligned distance: } \sum (\|x - y\|_2)\]

\[\text{Time offset: works sometimes, but not always optimal}\]

\[\text{Aligned distance: } \sum (\|x\|_2 + \|y\|_2)\]

\[\text{Extreme offset: complete misalignment - worst alignment alternative}\]

\[\text{Optimal alignment: Optimize alignment by minimizing aligned distance}\]

\[\text{Aligned distance: } \sum (\|x - y\|_2) = \min\]
Power Trace Alignment

Pairwise alignment

Trace $x$  

Trace $y$

1:1 alignment: trivial - nothing modified by the alignment

Aligned distance:

$$\sum (\text{cell})^2 = \|x - y\|^2$$

Time offset: works sometimes, but not always optimal

Extreme offset: complete misalignment - worst alignment

Optimal alignment: Optimize alignment by minimizing aligned distance
Power Trace Alignment

Pairwise alignment

Trace $x$     

Trace $y$

Time offset: 
works sometimes, but 
not always optimal

Aligned distance:

$$\sum (\textcolor{green}{\text{cell}})^2 = ?$$
Power Trace Alignment

Pairwise alignment

Trace $x$  

Trace $y$

Pairwise diff. matrix: each cell holds difference between two trace entries

Alignment path: get from start to end of both traces

1:1 alignment: trivial - nothing modified by the alignment

Aligned distance:

$\sum (\bar{\text{1}})^2 = \|x\|^2 + \|y\|^2$

Extreme offset: complete misalignment - worst alignment alternative

Aligned distance:

$\sum (\bar{\text{2}})^2 = \|x\|^2 + \|y\|^2$

Optimal alignment: Optimize alignment by minimizing aligned distance

$\sum (\bar{\text{3}})^2 = \min$

Guy Wolf (TAU)
Power Trace Alignment

Pairwise alignment

Pairwise diff. matrix: each cell holds difference between two trace entries

Alignment path: get from start to end of both traces

1:1 alignment: trivial - nothing modified by the alignment

Aligned distance: \[ \sum (\text{difference})^2 \]

Time offset: works sometimes, but not always optimal

Aligned distance: \[ \sum (\text{difference})^2 = \min \]

Extreme offset: complete misalignment - worst alignment alternative

Aligned distance: \[ \sum (\text{difference})^2 = \|x\|_2 + \|y\|_2 \]

Optimal alignment: Optimize alignment by minimizing aligned distance

Aligned distance: \[ \sum (\text{difference})^2 = \min \]
Power Trace Alignment
Finding optimal pairwise alignment

Dynamic Programming

- A method for solving complex problems by breaking them down into simpler subproblems.
- Applicable to problems exhibiting the properties of overlapping subproblems and optimal substructure.
- Better performances than naive methods that do not utilize the subproblem overlap.
Power Trace Alignment
Dynamic Time Warp (DTW)

**Basic DTW Algorithm:**

For each trace-time $i$ and for each trace-time $j$:

- Set $cost \leftarrow (x[i] - y[j])^2$
- Set the optimal distance at stage $[i, j]$ to:

\[
DTW_{[i,j]} \leftarrow cost + \min \left\{ \begin{array}{l}
DTW_{[i,j-1]} \\
DTW_{[i-1,j-1]} \\
DTW_{[i-1,j]}
\end{array} \right. 
\]

Optimal distance: $DTW_{[m,n]}$ (where $m$ & $n$ are lengths of traces).

Optimal alignment: backtracking the path leading to $DTW_{[m,n]}$ via min-cost choices of the algorithm.
Use coarse-grained matrices to avoid bad/unreasonable portions:

Drill down by fine graining to approximate the optimal alignment with quasi-linear time & space requirements

---

Power Analysis with DTW-based Alignment
based on recent paper

Experimental results:

Compare correlation DPA using 3 alignment methods:

**Static**: Simple static alignment by time offset

**SW**: Replace trace entries with avg. of sliding window
  - Not strictly an alignment method, but simple & sometimes effective

**DTW**: Elastic alignment with DTW

---

Power Analysis with DTW-based Alignment

based on recent paper\textsuperscript{3}

DES with stable clock

\textsuperscript{3}“Improving Differential Power Analysis by Elastic Alignment”, 2011, by J.G.J. van Woudenberg, M.F. Witteman, and B. Bakker
Power Analysis with DTW-based Alignment
based on recent paper³

DES with unstable clock

Analyzing Non-Cryptographic Leaks with Hidden Markov Model
Consider secret sequence of activities and leaked information with the following properties:

- Contains information about the secret sequence
- Contains noise
- Insufficient for directly recovering the secret information

If activities follow *known statistical patterns*, then an attacker can “guess” secret sequence from noisy leaks.

Attack: find best hypothesis such that:

1. It matches the leaked data
2. Has high probability according to statistical distribution of activity sequences
Information & Activity Leaks

Can it work?

Leaked information can be used for more than cryptographic purposes:

- Users are predictable - most activities are similar & repetitive
  - Internet - common websites and surfing routines
  - Emails/documents - linguistic models
  - Passwords - most common password is “password”
    - News services often publish lists of most common passwords of the year/month

- Guess activities/information by detecting “reasonable” usage patterns from leaked data

A statistical model of user activity profile can be used for this task.
Markov Chain

Stochastic Process:

Transition probabilities:

\[ \text{Pr} [q_{i+1} = ? | q_i, q_{i-1}, \ldots, q_2, q_1] \]
Transition probabilities (no history):

\[
\Pr [q_{i+1} = ? | q_i] = \Pr [q_{i+1} = ? | q_i, q_{i-1}, \ldots, q_2, q_1]
\]
Markov Chain

Keyboard structure & text auto-complete
Markov Chain

Keyboard structure & text auto-complete
Markov Chain

Keyboard structure & text auto-complete
Markov Chain

Keyboard structure & text auto-complete
Hidden Markov Model

Transition probabilities:
\[
\Pr[h_{i+1} = \cdot | h_i]
\]

Leak probabilities:
\[
\Pr[o_i = \cdot | h_i]
\]
Hidden Markov Model

Transition probabilities:
$$P[r_{i+1} = o | h_i]$$

Leak probabilities:
$$P[o_i = o | h_i]$$
Hidden Markov Model

Transition probabilities:
\[ \Pr[h_{i+1} = ? | h_i] \]

Leak probabilities:
\[ \Pr[o_i = ? | h_i] \]
Hidden Markov Model

Viterbi Algorithm
A dynamic programming algorithm for finding the most likely sequence of hidden states, especially in the context of Hidden Markov models.
Acoustic Analysis of Keyboards

based on paper\textsuperscript{4}

\textsuperscript{4}“Keyboard Acoustic Emanations Revisited”, 2005, by L. Zhuang, F. Zhou, and J.D. Tygar
Acoustic Analysis of Keyboards
based on paper 4

(a) Training Phase: Build keystroke classifier using unsupervised learning

(b) Recognition Phase: Recognize keystrokes using the classifier from (a).

the big money fight has drawn the support of dozens of companies in the entertainment industry as well as attorneys generals in states, who fear the file sharing software will encourage illegal activity, stem the growth of small artists and lead to lost jobs and diminished sales tax revenue.

HMM only:

the big money fight has drawn the shoporo
od dosens of companies in the entertainment
industry as well as attorneys gnnerals on
states, who fear the fild shading softwate
will encourage illegal acyivitt, srem the
grosth of small arrists and lead to lost
cobs and dimished sales tas revenue.

---

Zhou, and J.D. Tygar
Acoustic Analysis of Keyboards

based on paper$^4$

HMM & spelling corrections:

the big money fight has drawn the support of dozens of companies in the entertainment industry as well as attorneys generals in states, who fear the film sharing software will encourage illegal activity, stem the growth of small artists and lead to lost jobs and finished sales tax revenue.

---

Acoustic Analysis of Keyboards
based on paper\textsuperscript{4}

Password retrieval

\textsuperscript{4}“Keyboard Acoustic Emanations Revisited”, 2005, by L. Zhuang, F. Zhou, and J.D. Tygar
Acoustic Analysis of Printers

based on recent paper


(Pictures taken from URL: mindmachine.co.uk/book/print_06_dotmatrix_overview01.html)

(Picture taken from URL: flylib.com/books/en/2.374.1.27/1/)
Acoustic Analysis of Printers

based on recent paper\textsuperscript{5}

\textsuperscript{5}“Acoustic Side-Channel Attacks on Printers”, 2010, by M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, C. Sporleder
Acoustic Analysis of Printers
based on recent paper\(^5\)

**Training:**
- Feature extraction (split into words, noise reduction, etc.)
- Construct DB with (word, sound) pairs

**Online:**
- Feature extraction (same as in training)
- For each word:
  - Sort DB by similarity/difference from recorded sound
  - Reorder DB by n-gram/word distribution using HMM
  - Guess printed word as the top candidate from reordered DB

\(^5\)“Acoustic Side-Channel Attacks on Printers”, 2010, by M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, C. Sporleder
# Acoustic Analysis of Printers

based on recent paper\(^5\)

<table>
<thead>
<tr>
<th></th>
<th>Text 1</th>
<th>Text 2</th>
<th>Text 3</th>
<th>Text 4</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Top 1 (Top 3)</td>
<td>60.5 % (75.1 %)</td>
<td>66.5 % (79.2 %)</td>
<td>62.8 % (78.7 %)</td>
<td>61.5 % (77.9 %)</td>
<td>62.9 % (78.0 %)</td>
</tr>
<tr>
<td>HMM 3-gram</td>
<td>66.7 %</td>
<td>71.8 %</td>
<td>71.2 %</td>
<td>69.0 %</td>
<td>69.9 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Declaration 1</th>
<th>Declaration 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Top 1 (Top 3)</td>
<td>59.5 % (77.8 %)</td>
<td>57.5 % (72.6 %)</td>
</tr>
<tr>
<td>HMM 3-gram (using general-purpose corpus)</td>
<td>68.3 %</td>
<td>60.8 %</td>
</tr>
<tr>
<td>HMM 3-gram (using domain-specific corpus)</td>
<td>95.2 %</td>
<td>72.5 %</td>
</tr>
</tbody>
</table>

---

\(^5\)“Acoustic Side-Channel Attacks on Printers”, 2010, by M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, C. Sporleder
Further Reading I

Side-channel attacks using machine learning tools:

- “Further hidden Markov model cryptanalysis” (2005) by P.J. Green, R. Noad, N.P. Smart
- “Analyzing side channel leakage of masked implementations with stochastic methods” (2007) by K. Lemke-Rust & C. Paar
- “Theoretical and practical aspects of mutual information based side channel analysis” (2009) by E. Prouff & M. Rivain
- “Side channel attack: an approach based on machine learning” (2011) by L. Lerman, G. Bontempi, O. Markowitch
- “Side channel cryptanalysis using machine learning” (2012) by H. He, J. Jaffe, & L. Zou
- “PCA, eigenvector localization and clustering for side-channel attacks on cryptographic hardware devices” (2012) by D. Mavroeidis, L. Batina, T. van Laarhoven, E. Marchiori
Further Reading II

Trace alignment:

- “Recovering secret keys from weak side channel traces of differing lengths” (2008) by C.D. Walter
- “Improving differential power analysis by elastic alignment” (2011) by J.G.J. van Woudenberg, M.F. Witteman, B. Bakker
- “A general approach to power trace alignment for the assessment of side-channel resistance of hardened cryptosystems” (2012) by Q. Tian & S.A. Huss

Information retrieval from leaked data:

- “Acoustic side-channel attacks on printers” (2010) by M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, C. Sporleder
- “Building a side channel based disassembler” (2010) by T. Eisenbarth, C. Paar, Björn Weghenkenel
- “Automated black-box detection of side-channel vulnerabilities in web applications” (2011) by P. Chapman & D. Evans
- “Engineering statistical behaviors for attacking and defending covert channels” (2013) by V. Crespi, G. Cybenko, A. Giani
Conclusion

Machine learning
Retrieve meaningful information from vast amounts of leaked data.

Machine learning tools/concepts:
- Training/testing scheme
- Dimensionality reduction with PCA
- Clustering/classification with SVM
- Alignment with DTW
- Predicting/guessing usage patterns with HMM

Side channel applications
- Template based power analysis & power trace alignment
- Acoustic analysis of keyboards & printers
Conclusion

Machine learning
Retrieve meaningful information from vast amounts of leaked data.

Machine learning tools/concepts:
- Training/testing scheme
- Dimensionality reduction with PCA
- Clustering/classification with SVM
- Alignment with DTW
- Predicting/guessing usage patterns with HMM

Side channel applications
- Template based power analysis & power trace alignment
- Acoustic analysis of keyboards & printers
Big Data Will Play Key Role In Security's Future, Study Says

'Intelligence-driven security' will enable enterprises to deeply analyze security data and assess risk more accurately, RSA report says

Jan 17, 2013 | 06:21 AM | 0 Comments

By Tim Wilson
Dark Reading

Big data is coming to security, RSA says, and it will change the face of today's security technology and practices.
**Computerworld**

### Applications

<table>
<thead>
<tr>
<th>Topics</th>
<th>News</th>
<th>In Depth</th>
<th>Reviews</th>
<th>Blogs</th>
<th>Opinion</th>
<th>Share</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applications</td>
<td>App Development</td>
<td>Big Data</td>
<td>Business Intelligence/Analytics</td>
<td>Content/CMO</td>
<td>Emerging Technologies</td>
<td>Enterprise Architecture</td>
</tr>
</tbody>
</table>

**Home > Applications > Big Data**

**News Analysis**

**Applying big data approaches to information security a challenge**

Data integration and correlation a hard thing to do, say security experts at RSA Conference

*By Jaikumar Vijayan*  
February 28, 2013 06:00 AM ET  
**Add a comment**

---

Computerworld - SAN FRANCISCO -- Applying big data approaches to information security can help enterprises build better situational awareness capabilities, but implementation could prove to be a major challenge, security experts said at the RSA Conference 2013 being held here this week.
IBM Takes a Big Data Approach to Security

Companies will spend an estimated $50 billion on computer security this year, but they are not feeling particularly secure these days.

Blame innovation, if you like. Every big digital advance opens the door to both opportunity and mischief. Smartphones, cloud computing and the data explosion promise a revolution in communications, cost-savings and knowledge discovery. But those three trends in technology also create security headaches.
Big Data

IBM Takes a Big Step in Security

Applying big data to information security

Data integration and correlation

RSA Conference

By Jaikumar Vijayan
February 28, 2013 06:00 AM ET

Companies will spend an estimated $50 billion on computer security this year, but they are not feeling particularly secure these days.

Blame innovations both opportunistically and at a data explosion.

Knowledge discovery in security is on the rise.

Computerworld - SAN FRANCISCO

Information security can help enterprise capabilities, but implementation requires more rigor.

Experts said at the RSA Conference 2013 being held here this week that big data is changing today's security technology and practices.

RSA, IBM Bet On Big Data Analytics To Boost Security

RSA and IBM's turning to big data analytics to improve security monitoring mark what some analysts say could be the wave of the future.

Jan 31, 2013 | 05:12 PM |

By Brian Prince, Contributing Writer
Dark Reading

IBM and EMC's RSA security division are placing a big bet on big data playing an important role in security.

With an eye toward using analytics to improve threat detection, both companies released products this week that industry watchers say may herald the coming of a new set of technologies for identifying and stopping security threats.
Big Data
Not just from security leaks...

Big Data is produced and collected everywhere:
- Web data
- E-commerce (Amazon, Ebay, etc..)
- Purchases at department/grocery stores
- Bank/Credit Card transactions
- Netflix/Blockbuster/VOD
- Social networks (Facebook, Google, etc.)
Big Data
What can be done with it?

Processing Big Data involves several aspect such as:

- Aggregation and Statistics
  - Data warehouse and OLAP
- Indexing, Searching, and Querying
  - Keyword based search
  - Pattern matching (XML/RDF)
- Knowledge discovery
  - Data Mining
  - Statistical Modeling