Lowering the Bar: Deep Learning for Side Channel Analysis

Guilherme Perin, Baris Ege, Jasper van Woudenberg @jzvw
Blackhat, August 9, 2018
Before

Signal processing → Leakage modeling → Key

BLACKHAT 2018
After BLACKHAT 2018
Power / EM side channel analysis
Power analysis

Some crypto algorithm
Example (huge) leakage
Signal processing (demo)

Raw trace

Processed trace
Misalignment (demo)
AES-128 first round attack

Key Addition

Unknown

i_0  i_1  i_2  i_3
i_4  i_5  i_6  i_7
i_8  i_9  i_{10}  i_{11}
i_{12}  i_{13}  i_{14}  i_{15}

Known

k_0  k_1  k_2  k_3
k_4  k_5  k_6  k_7
k_8  k_9  k_{10}  k_{11}
k_{12}  k_{13}  k_{14}  k_{15}

x_0  x_1  x_2  x_3
x_4  x_5  x_6  x_7
x_8  x_9  x_{10}  x_{11}
x_{12}  x_{13}  x_{14}  x_{15}

Leakage model, Power prediction

s_0  s_1  s_2  s_3
s_4  s_5  s_6  s_7
s_8  s_9  s_{10}  s_{11}
s_{12}  s_{13}  s_{14}  s_{15}
Points of interest selection

Correlation, T-test, Difference of Means

Samples showing **statistical dependency** between intermediate (key-related) data and power consumption.
Concept of Template Analysis

Ciphertext Keys

Open Sample

Learn (Profiling) Phase

Leakage Model

Input

Closed Sample

Attack (Exploitation) Phase

Analysis

Fixed Key
Key recovery
The actual process

- Setup
- Analysis
- Processing
- Acquisition
Deep learning background
Deep Learning

Data with labels

cat

dog
Deep Learning

Train a machine to classify these data

Data with labels

Machine

Cat (%)
Dog (%)

Error function

BACK-PROPAGATION ALGORITHM
Deep Learning

Data with labels

Train a machine to classify these data

Test the machine on new data

Trained machine

Cat (%) Dog (%)
Deep Learning

Data with labels

Train a machine to classify these data

Test the machine on new data

Is classification accuracy good enough?

No

Change parameters

Trained machine

Machine = Deep Neural Network

Cat

We are done!
Convolutional Neural Networks (CNNs)

- **Input Layer** (the size is equivalent to the number of samples)

- **Conv. Layers** (feature extractor + encoding)

- **Output Layer** (the size is equivalent to the number of classes)

- **Dense Layers** (classifiers)

  The **convolutional layers** are able to detect the features independently of their positions.
Creating training/test/validation data sets

**features**

- HW = 5
- HW = 7
- HW = 3
- ... (repeated)
- HW = 4

**label**

- HW = 5
- HW = 7
- HW = 3
- ... (repeated)
- HW = 4

**Leakage model**
Classification

Trained Model

Trace (samples)

Key enumeration using output probabilities (Bayes)

Softmax ($\sum p_i = 1$)

- HW = 4: 0.05
- HW = 5: 0.15
- HW = 6: 0.65
- HW = 7: 0.08

0.02
0.02
0.02
0.01
0.01
0.02
0.02
Deep learning on side channels in practice
Step 1: Define initial hyper-parameters (demo)
Step 2: Make sure it’s capable of learning

- Increase the number of training traces and observe the training and validation accuracy
- Overfitting too fast?
  - Training accuracy: 100% | Validation accuracy: low
  - Neural network is too big for the number of traces and samples
Step 3: Make it generalize

Make sure the training accuracy/recall is increasing

NN is learning from its training set

Validation recall stays above the minimum threshold value = model is generalizing

0.111 = 1/9 (9 is the number of classes – HW of a byte)
Step 3: Make it generalize

Regularization techniques:

- L1, L2 (penalty applied to the weights)
- Dropout
- Data Augmentation (+traces)
- Early Stopping

Low Training Accuracy
Low Validation Accuracy

Good Training Accuracy
Good Validation Accuracy

High Training Accuracy
Low Validation Accuracy
Step 4: Key Recovery

In this analysis, we only need slightly-above coin flip accuracy!
Getting keys from the thingz!
Piñata AES-128 with misalignment (demo)
**Bypassing Misalignment with CNNs**

**Neural Network:** Input Layer \(\rightarrow\) ConvLayer \(\rightarrow\) 36 \(\rightarrow\) 36 \(\rightarrow\) 36 \(\rightarrow\) Output Layer

**Training/validation/test sets:** 90000/5000/5000 traces of 500 samples

**Leakage Model:** HW of S-Box Out (Round 1) \(\rightarrow\) 9 classes

---

Use Data Augmentation as regularization technique to improve generalization

**Results for key byte 0:**

---

**Number of traces**
Breaking protected ECC on Piñata

Supervised deep learning attack:
- Curve25519, Montgomery ladder, scalar blinding
- Messy signal
- Brute-force methods for ECC are needed if test accuracy < 100%
- Need to get (almost) all bits from one trace!
Breaking protected ECC

Unsupervised/Supervised Horizontal Attack: 60% success rate
Deep learning: 90% success rate
Deep learning ( + data augmentation): 99.4% success rate
Data augmentation: 25k → 200k traces.

255 Montgomery iterations

Scalar multiplication trace

Misaligned traces

Input (4000) 3 Conv Layers (10 filters) 4 Dense Layers (100 Neurons) Output (2 Classes)

RELU TANH SOFTMAX
Breaking AES with First-Order Masking (demo)

- 40k traces available
- AES-256 (Atmel ATMega-163 smart card)
- Countermeasure: Rotating S-box Masking (RSM)
Breaking AES with First-Order Masking

\[ x \oplus m \]
\[ y \oplus m \]

Remove the relationship between power consumption (EM) and predictable data

Combine data: \((x \oplus m) \oplus (y \oplus m) = x \oplus y\)

Combine samples: \(t[i] \times t[j]\)

Brute-force \(i, j \rightarrow Quadratic\) complexity
Breaking AES with First-Order Masking

Challenge: Training key == validation key

Correct key byte candidate (good generalization)

Wrong key byte candidate (poor generalization)
Breaking AES with First-Order Masking

Overfitting can be verified by checking where the NN is learning

**Correct** key byte candidate
(CNN learns from specific and leaky samples)

**Wrong** key byte candidate
(CNN overfits because it can’t distinguish leaky samples from noise)
Breaking AES with First-Order Masking

Neural Network: Input Layer > ConvLayer > 50 > 50 > 50 > Output Layer
Training/validation/test sets: 36000/2000/2000 traces
Leakage Model: HW of S-Box Out (Round 1) → 9 classes

Results for key byte 0:

The processing of 8 traces is sufficient to recover the key
1\textsuperscript{st} cool thing

This shouldn’t work
2\textsuperscript{nd} cool thing

DL is up there with dozens of SCA research teams
Wrapping up
I want to learn more!

Deeplearningbook.org  introtodeeplearning.com  bookstores  nostarch
Key takeaways

- DL does SCA art + science and scales
- DL requires network fiddling, the bar is low, not yet at 0
- Automation needed to put a dent in embedded insecurity
References

• http://www.deeplearningbook.org/
• S. Haykin, “Neural Networks and Learning Machines”.
• H. Maghrebi et al, “Convolutional Neural Networks with Data Augmentation Against Jitter-Based Countermeasures” https://eprint.iacr.org/2017/740
• Zhang et al, “Understanding deep learning requires re-thinking generalization” https://arxiv.org/abs/1611.03530