Application of machine learning techniques to side-channel analysis on code-based cryptography

Tania Richmond

Application of machine learning techniques to side-channel analysis on code-based cryptography

Short Term Scientific Mission (STSM) Report

Tania RICHMOND

December 12, 2018

1 Project

COST Action IC1403: Cryptanalysis in Ubiquitous Computing Systems (Cryptacus)

1.1 Involved Institutions

1.1.1 Home institution

Univ Rennes, Inria, CNRS, IRISA
Inria Rennes - Bretagne Atlantique
Campus universitaire de Beaulieu
263 Avenue du Général Leclerc
35042 Rennes
France

Web page: `https://www.inria.fr/en/centre/rennes`

1.1.2 Host institution

Faculty of Science
Radboud University
Postbus 9010
6500 GL Nijmegen
The Netherlands

Web page: `https://www.ru.nl/english/`

Supervisor: Prof. Lejla Batina
`https://www.cs.ru.nl/~lejla/`

2 Visit Details

Visiting researcher: Tania Richmond
`https://sites.google.com/site/taniarichmondnc/home/en`
Visiting period: November 18 - 25, 2018

2.1 Mission Statement

2.1.1 Short bibliography

Ms. Tania Richmond is a PostDoc at Univ Rennes, in the laboratory IRISA/Inria Rennes - Bretagne Atlantique. She defended her PhD thesis in October 2016 on "Secure implementation of cryptographic protocols based on error-correcting codes". Her main results are the invention of novel timing attacks and power consumption analysis against code-based cryptography.
Side-channel and fault injection attacks are mainly performed against symmetric cryptography. When it comes to asymmetric cryptography, this is most of the time against elliptic-curve cryptography. Her work is to adapt those techniques to code-based cryptography. Indeed, code-based cryptography is one family of the so-called post-quantum cryptography (with lattice-based, multivariate-based, isogeny-based and hash-based cryptography). Side-channel attacks against code-based cryptography were proposed for the first time ten years ago. With the recent NIST's post-quantum standardization, new schemes need to be analyzed to guarantee security against physical attacks. There are currently eighteen candidates in code-based cryptography.

2.1.2 Subject

The mission of Tania Richmond at Radboud University has at least two objectives:

1. study machine learning techniques for physical attacks (and in the context of the COST action, on embedded systems) against cryptographic protocols based on error-correcting codes,

2. modify existing algorithms in order to make them more resilient against physical attacks by proposing countermeasures.

Prof. Lejla Batina is an expert on side-channel analysis methods, fault attacks and countermeasures for secure embedded devices. She recently has been looking into machine learning for security and privacy and how to use it for side-channel analysis. On the defending side, she is interested in low-cost countermeasures against all kinds of physical attacks. Moreover she is currently supervising a PhD student, M. Pedro Maat C. Massolino, who did this Master's thesis on an hardware implementation of a code-based cryptosystem. These knowledge combined with Tania's research expertise may lead to a promising collaboration, where both partners may contribute from each other.

3 Scientific Mission Details

3.1 Side-channel attacks and machine learning techniques

3.1.1 Side-channel attacks

In a so-called Side-Channel Attack (SCA), an attacker can exploit laws of physical phenomena in order to obtain information contained in channels associated to an implementation (software or hardware). The side-channel does not aim at transmitting intentionally some information. It leaks information that an observer can interpret. The first one was proposed by Kocher in 1996 [9]. The most famous attacks are based on time and power, but not limited to these channels. Indeed, an attacker can try for example to exploit electromagnetic or acoustic leakage. Then, an attacker can use these additional information to drastically reduce the work factor for a key recovery attack. Indeed, one advantage of SCA over traditional cryptanalysis is that SCA can apply a divide-and-conquer approach to recover the message or more efficiently the secret/private key.

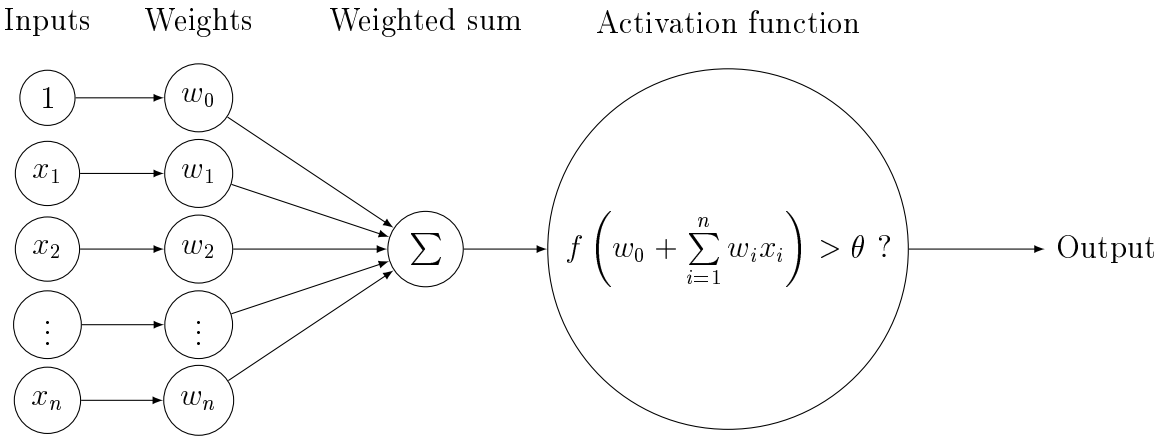

Figure 1: Perceptron (i.e. an artificial neuron)

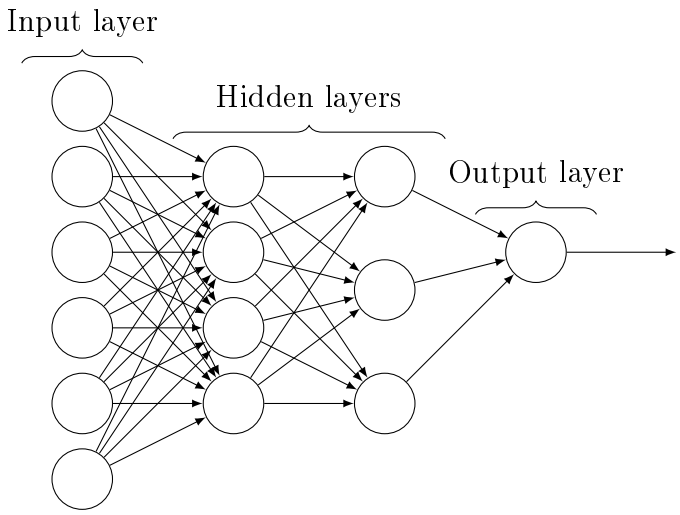

Figure 2: Multi Layer Perceptron (MLP), where circles represent neurons

3.1.2 Machine learning techniques

We give here a brief orverview of machine learning techniques. We do not claim that it is exhaustive. For instance, we are not talking about genetic algorithms in this report.

Neural networks family, or more precisely Artificial Neural Networks (ANNs), is a widely used machine learning family of algorithms. ANNs are inspired from Biological Neural Networks to simulate biological learning systems in the human brain.
A very simple type of a neural network is called perceptron (i.e. an artificial neuron). A perceptron is a linear binary classifier applied to the feature vector. Each vector component has an associated weight w_i and each perceptron has a threshold value θ. The output of a perceptron equals "1" if the direct sum between the feature vector and the weight vector is larger than zero and "-1" otherwise. A perceptron is depicted in Figure 1.
By adding more layers to a perceptron, we arrive to the Multi Layer Perceptron (MLP) algorithm. A neural network (e.g. MLP algorithm) consists of three or more layers: an input lay, one or more hidden layers, and an output layer. Data is used as the input layer, then is modified in the hidden layer(s) and the result is got from the output layer (see Figure 2). The hidden layers in a MLP algorithm must consist of non-linear activating nodes. Notice that more than one hidden layer is considered as a deep learning architecture. To define a neural network, we need to know the parameters: the number of hidden layers, the basic multiplication operation, and the activation functions.

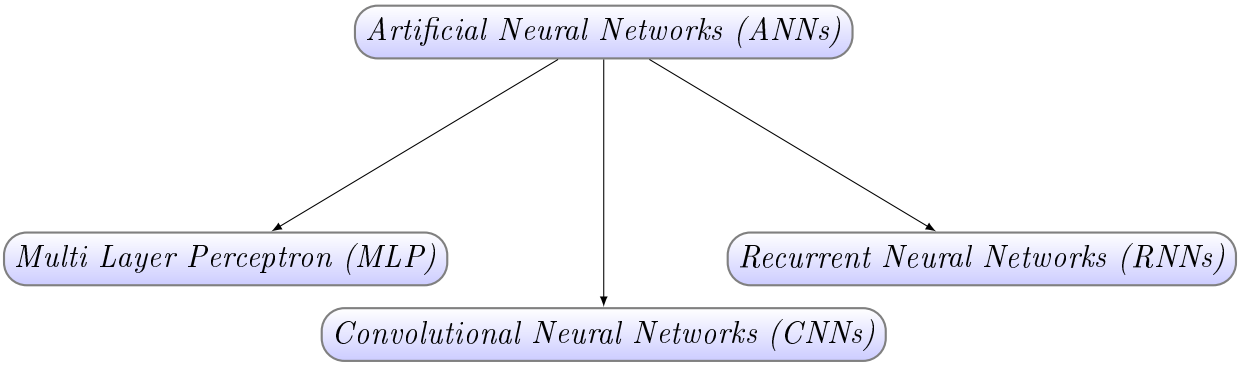

Figure 3: Different Neural Networks

One can see in Figure 3 different architectures of neural networks. MLPs were described above because of their interest in this work. The activation function of a MLP is a linear combination of a given set of inputs. In case of a CNN, the activation function is a convolutional function, and for RNN, a time notion appears because the activation function takes in inputs from previous states.

3.1.3 Related works

Side-channel analysis using machine learning techniques was first introduced by Hospodar *et al.* in [6]. They considered a classification algorithm to attack a software implementation of the Advanced Encryption Standard (AES) through the power consumption. They also compared this new approach to template attacks (in the category of profiling side-channel analysis).
A comparison between template attacks and machine learning in SCA was proposed by Lerman *et al.* in [11]. Then few works followed to demonstrate the effectiveness of machine learning techniques against classical profiled side-channel attacks [8, 17].
In [12], Maghrebi *et al.* extended this work to profiling techniques based on deep learning. They targeted an unprotected hardware AES and a first-order masked AES software implementation. They compared their results in deep learning-based attacks to machine learning-based and template attacks. More recently, neural networks have been applied in SCAs to increase the attacker capabilities [4, 10].

SCAs have been frequently used for cryptanalysis, in particular for key recovery attacks in cryptography and for reverse engineering of cryptographic algorithms. However, usage of side-channel leakage in order to attack machine learning architectures is much less investigated. In [7], Hua *et al.* showed how to reverse engineer AlexNet and SqueezeNet, two CNNs, through memory and timing information leakages. Parashar *et al.* obtained the weight of the neural networks by attacking a specific operation (i.e. the zero pruning) in [16]. More recently, in [23] Wei *et al.* recovered the input image from a power consumption analysis on an FPGA-based CNN.

3.1.4 First direction

In [1] Lejla Batina *et al.* explore the problem of reverse engineering a neural network via a simple power analysis, from a more generic perspective and in a grey to black box setting. Differential Power Analysis is used to recover secret weights from a pre-trained network. Horizontal Power Analysis is used to perform input recovery attack for a known network where they show that it works very well for medium to large sized networks. They conclude that using an appropriate combination of Simple Power Analysis and Differential Power Analysis techniques, all sensitive parameters of the network can be recovered. Moreover, a powerful Horizontal Power Analysis method is used to recover secret inputs from a known network in a single shot side-channel

analysis. As neural network algorithms are often optimized for performance, the presence of such side-channels is often ignored. The idea is to extend this work by reverse engineering a neural network implemented on a FPGA in order to compare our results with [7, 16, 23].

3.2 Code-based cryptography

Cryptography is the art to scramble a message for any undesirable reader. This principle appears in Antiquity. Modern cryptography (also known as public-key cryptography) used in real life is based on number theory, with factorization and discret logarithm problems. But since 1994 [20], the cryptographers' community knows that those problems can be solved in reasonable time with a sufficient quantum computer. An efficient quantum computer has not yet been built but progess is constantly made by IBM/Google. Consequently, NIST has begun to prepare for the transition to quantum-resistant cryptography, with a standardization process started in 2017 [15]. The selection criteria will include security and performance criteria.
Post-quantum cryptography includes any cryptography that is based on hard problems unbroken yet by a quantum algorithm. One solution can be found in coding theory with the so-called syndrome decoding problem [2]. The first code-based cryptosystem was proposed by McEliece in 1978 [13]. Unfortunately, due to large keys, the McEliece cryptosystem did not become as popular as the RSA cryptosystem, based on the facorization problem [19] and proposed the same year. Improvements have been done regarding key sizes. On one hand, different families of codes have been proposed, with interesting propoerties. On the other hand, storage capabilities have been increased in forty years.

3.2.1 Error-correcting codes

Error-correcting codes (codes in short) were first considered, as their name suggests, to correct errors in a message sending through a noisy channel. Since 1978, codes are also considered in cryptography [13]. Linear codes are the most common because all codewords can be generated thanks to a basis (smaller set), so it is easier to describe them. The most promissing families are Goppa codes (originally proposed, but the large keys, in Hamming metric) [5], QC-MDPC codes (codes with Quasi-Cyclic Moderate Density Parity-Check matrices, in Hamming metric) [14] and LRPC codes (codes with Low-Rank Parity-Check matrices, equivalent to QC-MDPC codes in Rank metric) [3].

3.2.2 Related works

Side-channel analysis against code-based cryptographic protocols using Goppa codes were proposed first by Falko Strenzke during his PhD [21], essentially by timing analysis. This work has been improved by Tania Richmond during her PhD [18], for timing attacks and power consumption analysis. In the meantime, Ingo von Maurich focused on QC-MDPC codes during his PhD [22], essentially by power consumption analysis.

3.2.3 Second direction

The main issue today for companies is that post-quantum cryptosystems are too big to fit in a microcontroller. To show them that they should consider more closely post-quantum cryptography and especially code-based cryptography, the objective is to provide some microcontroller implementations based on the C code of certain NIST submissions. Because of their simple keys' representation, QC-MDPC codes seem to be the best option to start with. NIST submissions based on QC-MDPC codes are BIKE[1] and QC-MDPC KEM[2]. Before the NIST standardization process, one implementation was proposed by Tung Chou on a Cortex-M4 microcontroller[3].

[1] `http://bikesuite.org/`

[2] `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/QC_MDPC_KEM.zip`

[3] `http://www.win.tue.nl/~tchou/qcbits/`

This last implementation is already under analysis by Tania Richmond, in collaboration with Annelie Heuser and Benoît Gérard[4].

3.3 Conclusion and perspectives for this collaboration

During this short term scientific mission, we manage to fix ideas about what we could work on together. We explore two possible topics:

1. investigate a neural network implemented in hardware (FPGA) in terms of side-channel resilience:
   (a) find a VHDL implementation of a neural network, in particular a multi layer perceptron;
   (b) integrate it on a FPGA and adapt the attack proposed by Lejla Batina *et al.* in [1] to this type of device;
2. side-channel analysis of NIST proposals:
   (a) integrate the C implementation of BIKE or QC-MDPC KEM in a microcontroller, using for instance the piñata board[5];
   (b) analyze them.

In the future, we expect to solve the debbuging part for the integration in a microcontroller of some code-based submissions to the NIST standardization. More exchanges are expected between our two teams to continue our collaboration.

References

[1] Lejla Batina, Shivam Bhasin, Dirmanto Jap, and Stjepan Picek. CSI neural network: Using side-channels to recover your artificial neural network information. Cryptology ePrint Archive, Report 2018/477, 2018. `https://eprint.iacr.org/2018/477`.

[2] Elwyn R. Berlekamp, Robert James McEliece, and Henk C. A. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Transactions on Information Theory*, 24(3):384–386, May 1978.

[3] Philippe Gaborit, Gaétan Murat, Olivier Ruatta, and Gilles Zémor. Low rank parity check codes and their application to cryptography. pages 168–180, 2013.

[4] R. Gilmore, N. Hanley, and M. O'Neill. Neural network based attack on a masked implementation of AES. In *2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 106–111, May 2015.

[5] Valerii Denisovich Goppa. A new class of linear error-correcting codes. *Problemy Peredachi Informatsii*, 6(3):24–30, September 1970.

[6] Gabriel Hospodar, Benedikt Gierlichs, Elke De Mulder, Ingrid Verbauwhede, and Joos Vandewalle. Machine learning in side-channel analysis: a first study. *Journal of Cryptographic Engineering*, 1(4):293, Oct 2011.

[7] Weizhe Hua, Zhiru Zhang, and G. Edward Suh. Reverse engineering convolutional neural networks through side-channel information leaks. In *Proceedings of the 55th Annual Design Automation Conference*, DAC '18, pages 4:1–4:6, New York, NY, USA, 2018. ACM.

[4] `https://labh-curien.univ-st-etienne.fr/cryptarchi/workshop18/abstracts/richmond.pdf`
[5] `https://www.riscure.com/product/pinata-training-target/`

[8] Dirmanto Jap, Marc Stöttinger, and Shivam Bhasin. Support vector regression: Exploiting machine learning techniques for leakage modeling. In *Proceedings of the Fourth Workshop on Hardware and Architectural Support for Security and Privacy*, HASP '15, pages 2:1–2:8, New York, NY, USA, 2015. ACM.

[9] Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In Neal Koblitz, editor, *Advances in Cryptology (CRYPTO'96)*, volume 1109 of *Lecture Notes in Computer Science (LNCS)*, pages 104–113, Berlin, Heidelberg, 1996. Springer.

[10] Yinan Kong and Ehsan Saeedi. The investigation of neural networks performance in side-channel attacks. *Artificial Intelligence Review*, Jun 2018.

[11] Liran Lerman, Romain Poussier, Gianluca Bontempi, Olivier Markowitch, and François-Xavier Standaert. Template attacks vs. machine learning revisited (and the curse of dimensionality in side-channel analysis). In Stefan Mangard and Axel Y. Poschmann, editors, *Constructive Side-Channel Analysis and Secure Design*, pages 20–33, Cham, 2015. Springer International Publishing.

[12] Houssem Maghrebi, Thibault Portigliatti, and Emmanuel Prouff. Breaking cryptographic implementations using deep learning techniques. In Claude Carlet, M. Anwar Hasan, and Vishal Saraswat, editors, *Security, Privacy, and Applied Cryptography Engineering*, pages 3–26, Cham, 2016. Springer International Publishing.

[13] Robert James McEliece. A public-key cryptosystem based on algebraic coding theory. Technical Report 44, California Inst. Technol., Pasadena, CA, January 1978.

[14] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *IEEE International Symposium on Information Theory Proceedings (ISIT 2013)*, pages 2069–2073, July 2013.

[15] NIST. Post-quantum cryptography standardization. Official webpage: `https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization`.

[16] Angshuman Parashar, Minsoo Rhu, Anurag Mukkara, Antonio Puglielli, Rangharajan Venkatesan, Brucek Khailany, Joel Emer, Stephen W. Keckler, and William J. Dally. SCNN: An accelerator for compressed-sparse convolutional neural networks. *SIGARCH Comput. Archit. News*, 45(2):27–40, June 2017.

[17] S. Picek, A. Heuser, A. Jovic, S. A. Ludwig, S. Guilley, D. Jakobovic, and N. Mentens. Side-channel analysis and machine learning: A practical perspective. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 4095–4102, May 2017.

[18] Tania Richmond. *Secure implementation of cryptographic protocols based on error-correcting codes (in French)*. PhD thesis, Université Jean Monnet, Saint-Étienne, October 2016.

[19] Ronald L. Rivest, Adi Shamir, and Len Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[20] Peter W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, November 1994.

[21] Falko Strenzke. *Efficiency and Implementation Security of Code-based Cryptosystems*. PhD thesis, Technische Universität, Darmstadt, 2013.

[22] Ingo von Maurich. *Efficient Implementation of Code- and Hash-Based Cryptography*. PhD thesis, Ruhr-Universität Bochum, October 2016.

[23] Lingxiao Wei, Bo Luo, Yu Li, Yannan Liu, and Qiang Xu. I know what you see: Power side-channel attack on convolutional neural network accelerators. In *Proceedings of the 34th Annual Computer Security Applications Conference*, ACSAC '18, pages 393–406, New York, NY, USA, 2018. ACM.