

# Contents

<b>I</b>	<b>Context and State of the Art</b>	<b>1</b>
<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Introduction to Cryptography . . . . .	3
1.1.1	Secret-Key Cryptography . . . . .	3
1.1.2	Public-Key Cryptography . . . . .	3
1.2	Secure Hardware and Embedded Cryptography . . . . .	3
1.2.1	The Example of the Smart Card . . . . .	3
1.2.2	Certification of a Secure Hardware . . . . .	3
1.2.3	Modern More Complex Devices to Certify . . . . .	3
1.2.4	Embedded Cryptography Vulnerabilities . . . . .	3
<b>2</b>	<b>Introduction to Side-Channel Attacks</b>	<b>5</b>
2.1	Introduction to Side-Channel Attacks . . . . .	6
2.1.1	Historical Overview . . . . .	6
2.1.2	Terminology and Generalities . . . . .	6
	Target and Leakage Model . . . . .	6
	Points of Interest . . . . .	6
	Simple vs Advanced SCAs . . . . .	6
	Vertical vs Horizontal SCAs . . . . .	6
	Profiled vs Non-Profiled SCAs . . . . .	6
	Side-Channel Algebraic Attacks . . . . .	6
	Distinguishers . . . . .	6
	SCA Metrics . . . . .	6
2.2	Main Side-Channel Countermeasures . . . . .	6
2.2.1	Random Delays and Jitter . . . . .	6
2.2.2	Shuffling . . . . .	6
2.2.3	Masking . . . . .	6
2.3	Higher-Order Attacks . . . . .	6
2.3.1	Higher-Order Moments Analysis and Combining Functions . . . . .	6
2.3.2	Profiling Higher-Order Attacks . . . . .	6
	Profiling with Masks Knowledge . . . . .	6
	Profiling without Masks Knowledge . . . . .	6
2.4	Thesis Contribution and Organization . . . . .	6
2.4.1	Foreword of this Thesis: Research of Points of Interest . . . . .	6
2.4.2	Dimensionality Reduction Approach . . . . .	6
	Linear Methods for First-Order Attacks . . . . .	6
	Kernel Methods for Higher-Order Attacks . . . . .	6
2.4.3	Neural Network Approach . . . . .	6
	Toward Getting Rid of Information-Loosing Preprocessing . . . . .	6

<b>3</b>	<b>Introduction to Machine Learning</b>	<b>7</b>
3.1	Basic Concepts of Machine Learning	7
3.1.1	The Task, the Experience and the Performance	7
3.1.2	Supervised, Semi-Supervised, Unsupervised Learning	7
3.1.3	Training, Validation and Test Sets	7
3.1.4	Underfitting, Overfitting and Regularization	7
3.1.5	Data Augmentation	7
3.1.6	No Free Lunch Theorem	7
3.2	Machine Learning Applications in Side-Channel Context	7
3.2.1	Profiled Attack as a Classification Problem	7
	Support Vector Machine	7
	Random Forest	7
	Neural Networks	7
<b>II</b>	<b>Contributions</b>	<b>9</b>
<b>4</b>	<b>Points of Interest</b>	<b>11</b>
4.1	Motivations	11
4.1.1	The Curse of Dimensionality	11
4.2	Selection on Points of Interest: Classical Statistics	11
4.3	Related Issues: Leakage Detection and Leakage Assessment	11
4.4	Generalized SNR for Multi-Variate Attacks	11
4.5	Observations Leading to Take a Dimensionality Reduction Approach	11
<b>5</b>	<b>Linear Dimensionality Reduction</b>	<b>13</b>
5.1	Introduction	13
5.1.1	Principal Component Analysis	13
5.1.2	Linear Discriminant Analysis	13
5.1.3	Projection Pursuits	13
5.2	Principal Component Analysis	13
5.2.1	Statistical Point of View	13
5.2.2	Geometrical Point of View	13
5.3	Application of PCA in SCAs	13
5.3.1	Original vs Class-Oriented PCA	13
5.3.2	The Choice of the Principal Components	13
5.4	Linear Discriminant Analysis	13
5.4.1	Statistical Point of View	13
5.4.2	Geometrical Point of View	13
5.5	Application of LDA in SCAs	13
5.5.1	The Small Sample Size problem	13
<b>6</b>	<b>Kernel Dimensionality Reduction</b>	<b>15</b>
6.1	Motivation	15
6.1.1	Higher-Order Attacks	15
	Higher-Order Version of Projection Pursuits	15
6.2	Kernel Function and Kernel Trick	15
6.2.1	Local Kernel Functions as Similarity Metrics	15
6.3	Kernel Discriminant Analysis	15
6.4	Experiments over Atmega328P	15
6.4.1	The Regularization Problem	15

6.4.2	The Multi-Class Trade-Off . . . . .	15
6.4.3	Multi-Class vs 2-class Approach . . . . .	15
6.4.4	Asymmetric Preprocessing/Attack Approach . . . . .	15
	Comparison with Projection Pursuits . . . . .	15
6.5	Drawbacks of Kernel Methods . . . . .	15
	Misalignment Effects . . . . .	15
	Memory Complexity and Actual Number of Parameters . . . . .	15
	Two-Phases Approach: Preprocessing-Templates . . . . .	15
<b>7</b>	<b>Convolutional Neural Networks against Jitter-Based Countermeasures</b>	<b>17</b>
7.1	Moving from Kernel Machines to Neural Networks . . . . .	17
7.2	Misalignment of Side-Channel Traces . . . . .	17
7.2.1	The Necessity and the Risks of Applying Realignment Techniques . . . . .	17
7.2.2	Analogy with Image Recognition Issues . . . . .	17
7.3	Convolutional Layers to Impose Shift-Invariance . . . . .	17
7.4	Data Augmentation for Misaligned Side-Channel Traces . . . . .	17
7.5	Experiments against Software Countermeasures . . . . .	17
7.6	Experiments against Artificial Hardware Countermeasures . . . . .	17
7.7	Experiments against Real-Case Hardware Countermeasures . . . . .	17
<b>8</b>	<b>KDA vs Neural Networks Approach for HO-Attacks</b>	<b>19</b>
8.1	Simulated Experiment for Profiled HO-Attacks . . . . .	19
8.1.1	The Simulations . . . . .	19
8.1.2	Comparison between KDA and MLP . . . . .	19
8.2	Real-Case Experiments over ARM Cortex-M4 . . . . .	19
<b>9</b>	<b>Siamese Neural Networks for Collision Attacks</b>	<b>21</b>
9.1	Introduction . . . . .	21
9.2	Siamese Neural Networks . . . . .	21
9.2.1	Distances and Loss Functions . . . . .	21
9.2.2	Relation with Kernel Machines . . . . .	21
9.3	Collision Attacks with Siamese NNs . . . . .	21
9.3.1	Experimental Results . . . . .	21
<b>10</b>	<b>Conclusions and Perspectives</b>	<b>23</b>
10.1	Summary . . . . .	23
10.2	Strengthen Embedded Security: the Main Challenge for Machine Learning Applications . . . . .	23