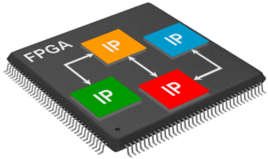
Support Vector Machines for Improved IP Detection with Soft Physical Hash Functions.

Ludovic-Henri Gustin, François Durvaux, Stéphanie Kerckhof, François-Xavier Standaert, Michel Verleysen

UNIVERSITE CATHOLIQUE DE LOUVAIN

COSADE - April 2014

Context

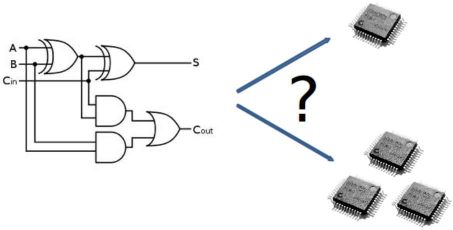


Integrated circuit design and fabrication:

- More and more complex hardware designs
- Designs sold as Intellectual Property (IP)
- IP market growing

Problem

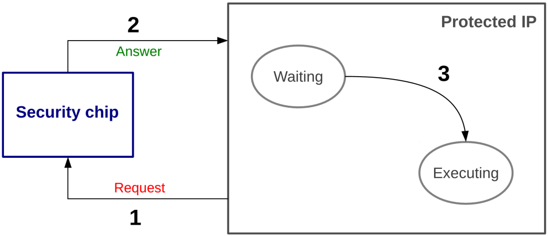
- Counterfeiting



Existing Solutions

Permission-based protections (e.g. security chip, PUFs):

- Key needed to use the IP
- A *priori* solution



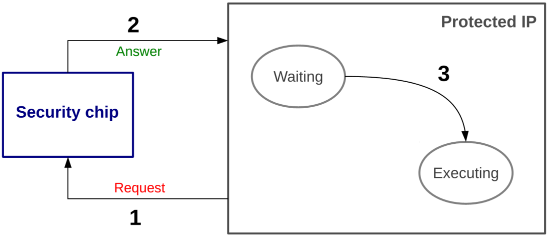
Existing Solutions

Permission-based protections (e.g. security chip, PUFs):

- Key needed to use the IP
- A *priori* solution

Limitation:

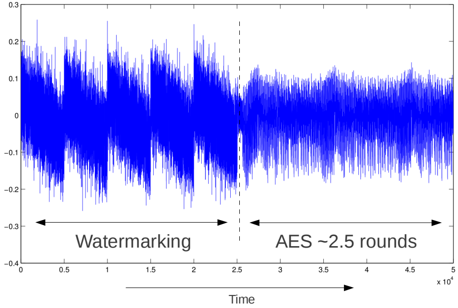
- Difficulty to integrate in customers' products



Existing Solutions

Watermarking (e.g. temperature, power consumption):

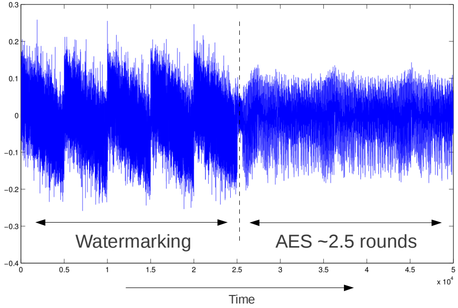
- Specific piece of information inserted
- A *posteriori* solution



Existing Solutions

Watermarking (e.g. temperature, power consumption):

- Specific piece of information inserted
- A *posteriori* solution

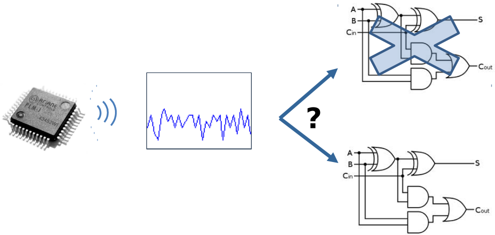


Limitations of both solutions :

- Early integration in design process
- May be removed

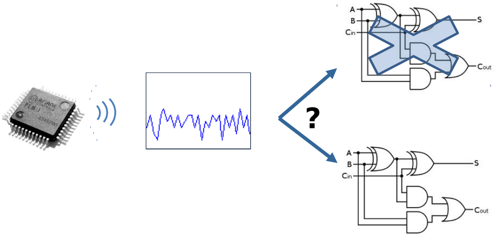
Use of side-channel leakage as an IP signature

- A *posteriori* solution
- Hash (IP "signature") extracted from power traces
- Cannot be removed since intrinsic to the IP execution
- No chip modification required



Use of side-channel leakage as an IP signature

- A *posteriori* solution
- Hash (IP "signature") extracted from power traces
- Cannot be removed since intrinsic to the IP execution
- No chip modification required

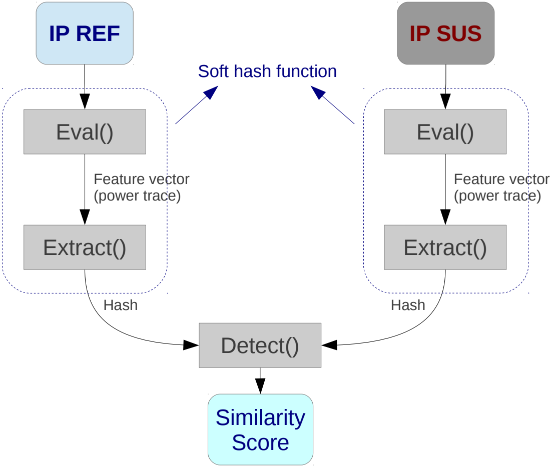


This work :

- Soft Physical Hash (**SPH**) based framework
- Support Vector Machines (**SVM**) detection tool

Outline

Generic detection framework - Soft Physical Hash Function (SPH)



Soft Physical Hash Function (SPH) properties

Perceptual robustness

- Same IPs $\Rightarrow$ high similarity scores
- Linked to the non-detection error probability

Content sensitivity

- Different IPs $\Rightarrow$ low similarity scores
- Linked to false-alarm error probability

Previous experiment :

- **FPGA Designs** : `Xilinx Virtex-II Pro` FPGA, 6 block ciphers
- Promising experimental results
- Essentially, correlation-based statistics (Pearson's correlation coefficient)

Soft Physical Hash Functions (SPH)

Previous experiment :

- **FPGA Designs** : `Xilinx Virtex-II Pro` FPGA, 6 block ciphers
- Promising experimental results
- Essentially, correlation-based statistics (Pearson's correlation coefficient)

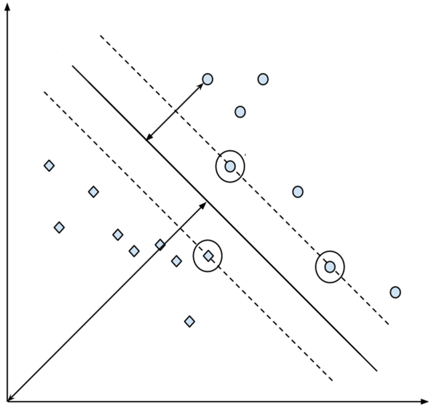
This study : Usage of machine learning's **Support Vector Machines (SVM)** to extract information from power traces (FGPA's block ciphers studied in [2])

- Proven to effectively solves detection/classification tasks in various areas of application
- Learn automatically arbitrary complex functions
- Handle large dimensionality

Estimation of classification fonction(s) of **hash vectors** $\hat{f}_c : \mathbf{x} \rightarrow \{-1, +1\}$:

Support Vector Machines - Binary classification

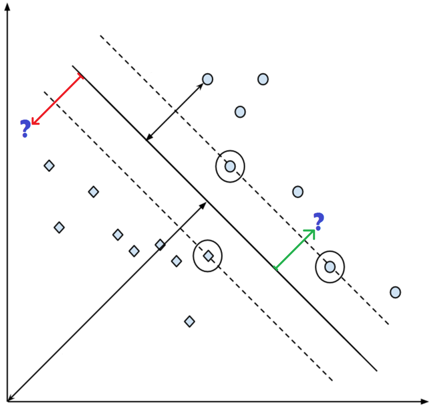
Estimation of classification fonction(s) of **hash vectors** $\hat{f}_c : \mathbf{x} \rightarrow \{-1, +1\}$:



- *m training* vectors $\mathbf{x_i} \in \mathbb{R}^n$ and class $y_i \in \{-1, +1\}$ $(i = 1..m)$

Support Vector Machines - Binary classification

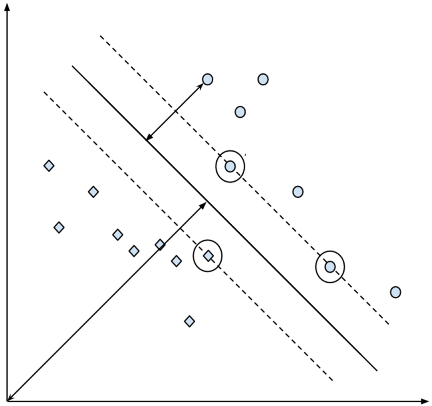
Estimation of classification fonction(s) of **hash vectors** $\hat{f}_c : \mathbf{x} \rightarrow \{-1, +1\}$:



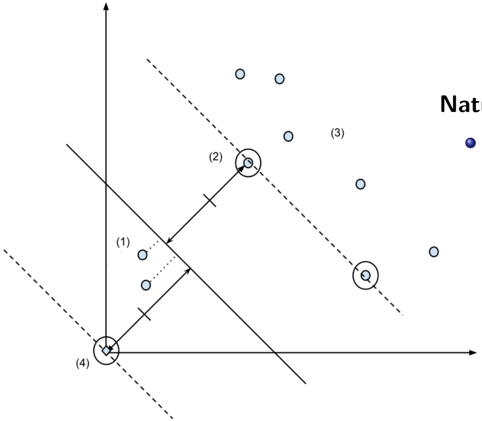
- *m training* vectors $\mathbf{x_i} \in \mathbb{R}^n$ and class $y_i \in \{-1, +1\}$ $(i = 1..m)$
- Predict y_i for **unseen observation**, with *separating hyperplane*

Support Vector Machines - Binary classification

Estimation of classification fonction(s) of **hash vectors** $\hat{f}_c : \mathbf{x} \to \{-1, +1\}$:



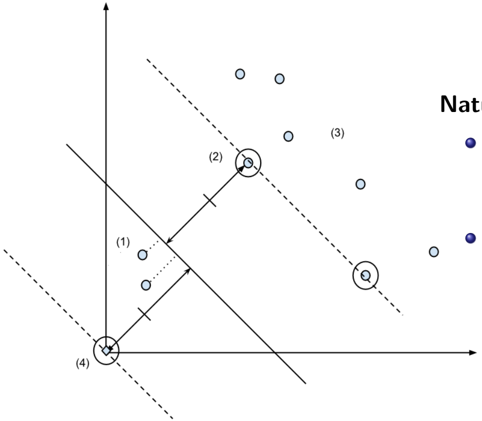
- *m training* vectors $\mathbf{x_i} \in \mathbb{R}^n$ and class $y_i \in \{-1, +1\}$ $(i = 1..m)$
- Predict y_i for **unseen observation**, with *separating hyperplane*
- Non-linear frontiers are possible



Natural extension of the binary case

- No assumption on the negative population : hyperplane H separates most of the data from the origin

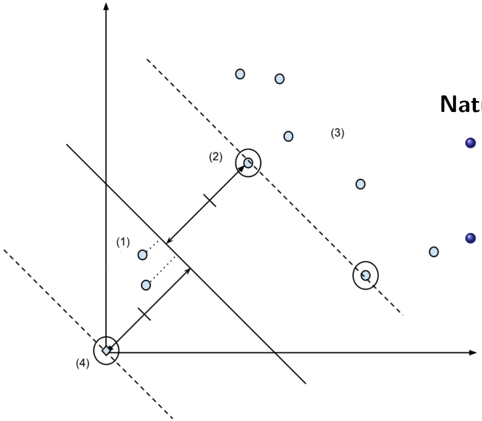
One-class SVM (OSVM)



Natural extension of the binary case

- No assumption on the negative population : hyperplane H separates most of the data from the origin
- Penalty cost for outliers (Adjustable trade-off)

One-class SVM (OSVM)



Natural extension of the binary case

- No assumption on the negative population : hyperplane H separates most of the data from the origin
- Penalty cost for outliers (Adjustable trade-off)

Similarity score $\sim$ *distance* of a classified vector to the hyperplane

Outline

Specification of the detection framework

Object to protect

5 lightweight block ciphers : `HIGHT`, `ICEBERG`, `KATAN`, `NOEKEON`, `PRESENT` running on a `Xilinx Virtex-II Pro FPGA` .

Specification of the detection framework

Object to protect

5 lightweight block ciphers : `HIGHT`, `ICEBERG`, `KATAN`, `NOEKEON`, `PRESENT` running on a `Xilinx Virtex-II Pro FPGA` .

Evaluation

Feature vectors : voltage variation measured around a shunt resistor on the Sasebo-G board.

Specification of the detection framework

Object to protect

5 lightweight block ciphers : `HIGHT`, `ICEBERG`, `KATAN`, `NOEKEON`, `PRESENT` running on a `Xilinx Virtex-II Pro FPGA` .

Evaluation

Feature vectors : voltage variation measured around a shunt resistor on the Sasebo-G board.

Extraction

Reference : construction of an OSVM model based on about 1300 traces (parameters output).
Hypothese of work : Construction of models based solely on *one* measurement context.

Suspicious : no particular processing.

Specification of the detection framework

Object to protect

5 lightweight block ciphers : `HIGHT`, `ICEBERG`, `KATAN`, `NOEKEON`, `PRESENT` running on a `Xilinx Virtex-II Pro FPGA` .

Evaluation

Feature vectors : voltage variation measured around a shunt resistor on the Sasebo-G board.

Extraction

Reference : construction of an OSVM model based on about 1300 traces (parameters output).
Hypothese of work : Construction of models based solely on *one* measurement context.

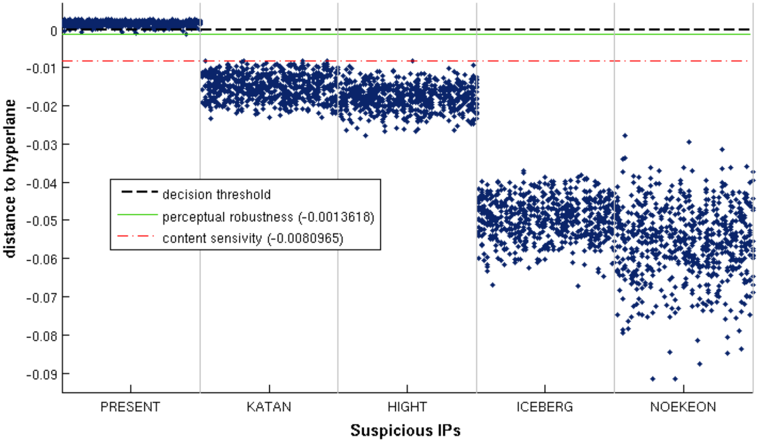
Suspicious : no particular processing.

Detection

Distance metrics to the hyperplane.

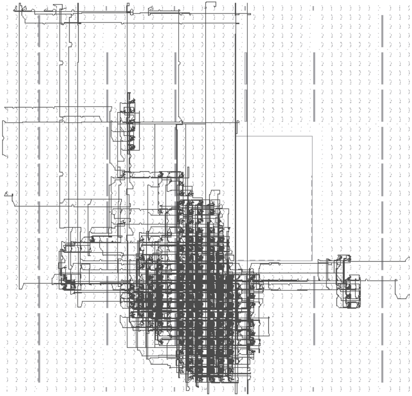
Outline

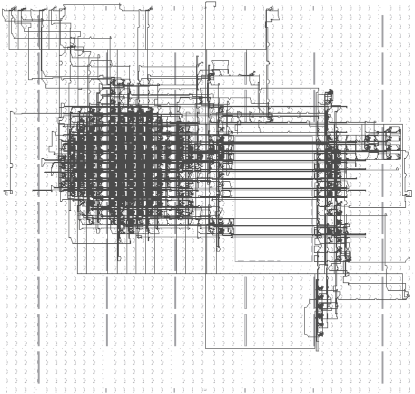
(1) Ref. `PRESENT` - Susp. *standalone*



- Classification outcome (binary output) vs distance metrics
- Green threshold : **min** score for `PRESENT` traces (protected IP).
- Red threshold : **max** score for traces from other IPs.

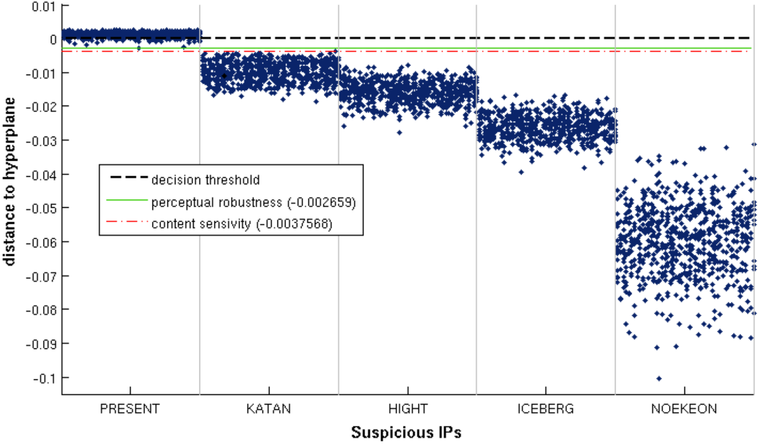
(2) Ref. PRESENT - Susp. *resynthesized*





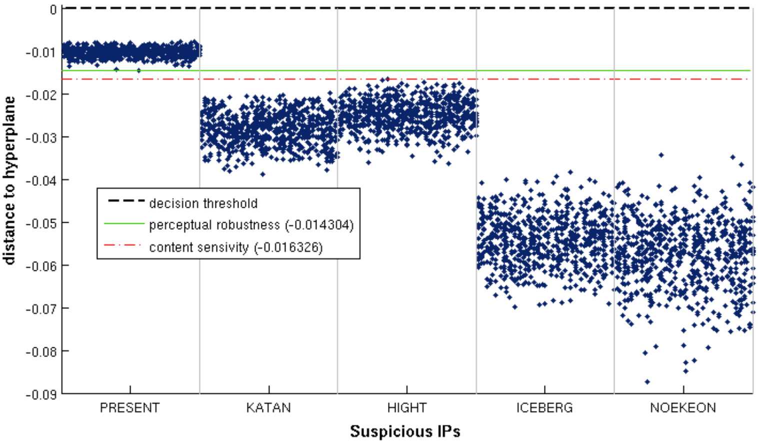
Resynthesized : New placement and routing, under area optimisation constraints.

(2) Ref. PRESENT - Susp. *resynthesized*



Resynthesized : New placement and routing, under area optimisation constraints.

(3) Ref. `PRESENT` - Susp. *with parasitic IP*



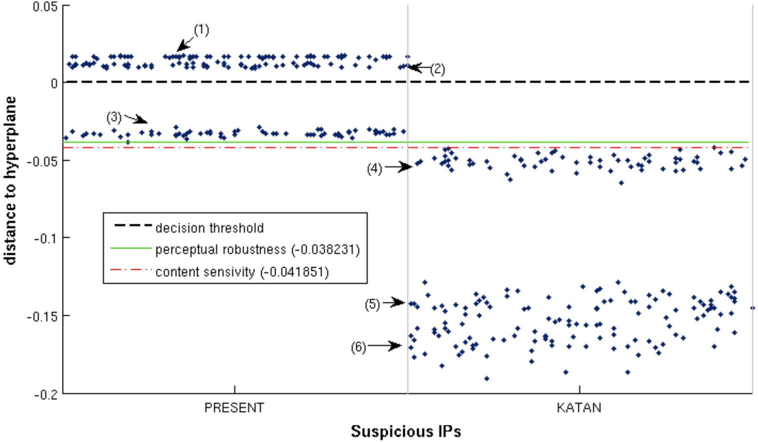
Parasitic IP : Linear feedback shift register (LFSR) of 2048 bits.
Below the 0 threshold : Failure of the classifier, but still a detection area.

Identified problem

We can't find a correct decision threshold when combining cases :

- lowest `PRESENT` (parasited) $<$ highest `KATAN` (re-synthetized)
  - $\Rightarrow$ no detection gap
  - misclassification(s) can occur
- Two tweaks are needed to enhance detection : exploiting **data dependencies** and **noise reduction**

(4) Ref. `PRESENT` - Susp. *all combined*, 5x avg. traces & known inputs



Known inputs : Takes advantage of data dependencies
Averaging : Reduce algorithmic noise

Outline

OSVM

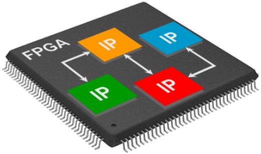
Pros

- Can handle large dimensionality
- Realistic model : no assumption made on negative examples
- Better results than previously reached :
  - Only the more complex case required to exploit data dependencies and noise reduction
  - But more measurement traces needed (1300)

Cons

- Unsupervised : difficulty to build good heuristics to select model's parameters
- Failure of the classifier on datasets with parasitic algorithm noise.
  - new threshold choice
- Necessary rejection of outliers (intrinsic bias).
  - new threshold choice

Going further



More complex and richer set of IPs and transformations of IPs.

Improving detection quality :

- *Evaluation*, other feature vectors potentially interesting

Thank you for your attention !