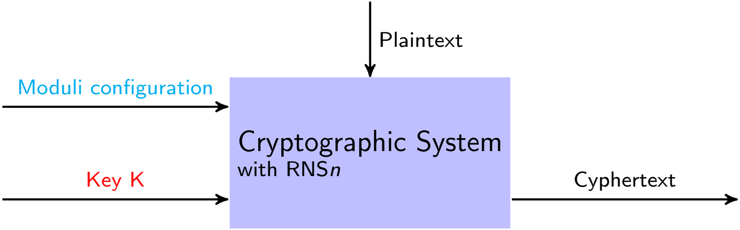
Resilience of randomized RNS arithmetic with respect to side-channel leaks of cryptographic computation

Jérôme Courtois

jerome.courtois@lip6.fr

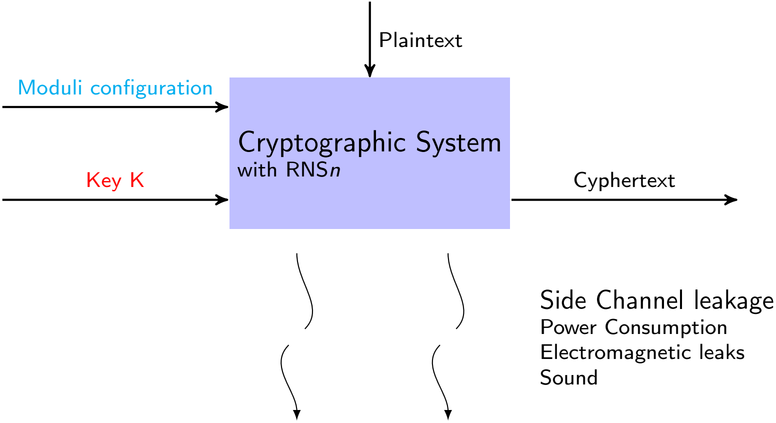
May 5, 2019

In collaboration with Lokmane Abbas-Turki and Jean-Claude Bajard

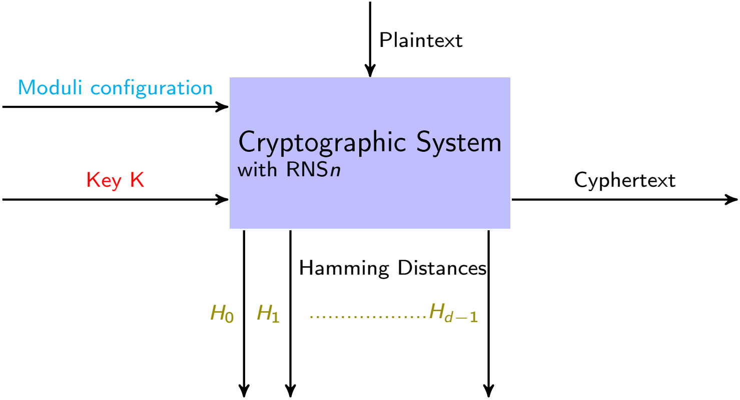


- $\mathcal{B}_n = \{m_1, ..., m_n\}$, m_i pairwise coprime.
- Chinese Remainder theorem
  $\rightarrow$ unique representation of integers in [0;M[, $M = \prod_{i=1}^{n} m_i$, with theirs residues in $\mathcal{B}_n$
- X is denoted $\{x_1, ..., x_n\}$ in $\mathcal{B}_n$ with $x_i = X \bmod m_i$

Find K from leakage

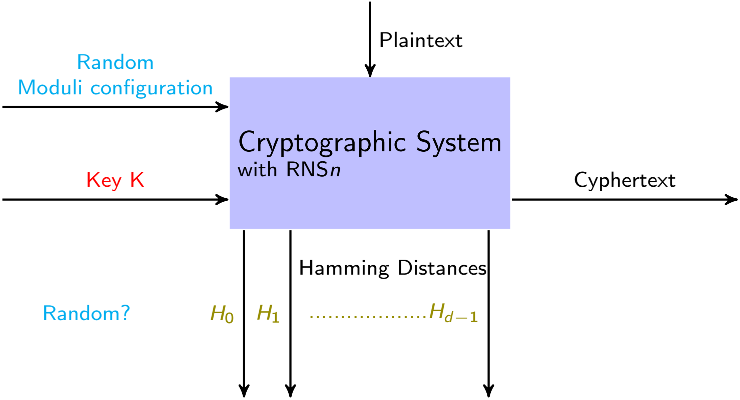


Find K from Hamming distances



Side Channel Leakage proportional to Hamming distances.

Find K from Hamming distances



J.C. Bajard & al.(2004) "Leak Resistant Arithmetic".

Scalar Multiplication on ECC

Denote RNSn an RNS representation with n moduli.

Algorithm Montgomery Powering Ladder (MPL) for ECC in RNSn

Require: A point G in RNSn representation
A key K with a binary representation $K = 2^{d-1}b_0 + 2^{d-2}b_1 + ... + 2b_{d-2} + b_{d-1}$
Ensure:
$A_0 = [K]G$
$(H_i)_{i\in\{0,..,d-1\}}$, the Hamming distances
function

$A_1 = [2]A_0$

for i=1 to d-1 **do**
$A_{\overline{b_i}} = A_{\overline{b_i}} + A_{b_i}$
$A_{b_i} = [2]A_{b_i}$

end for
end function

Denote RNSn an RNS representation with n moduli.

Algorithm Montgomery Powering Ladder (MPL) for ECC in RNSn

Require: A point G in RNSn representation
A key K with a binary representation $K = 2^{d-1}b_0 + 2^{d-2}b_1 + ... + 2b_{d-2} + b_{d-1}$
Ensure:
$A_0 = [K]G$
$(H_i)_{i \in \{0,..,d-1\}}$, the Hamming distances
function
Random Moduli configuration C
$A_1 = [2]A_0$

for i=1 to d-1 **do**
$A_{\overline{b_i}} = A_{\overline{b_i}} + A_{b_i}$
$A_{b_i} = [2]A_{b_i}$

end for
end function

Scalar Multiplication on ECC

Denote RNSn an RNS representation with n moduli.

Algorithm Montgomery Powering Ladder (MPL) for ECC in RNSn

Require: A point G in RNSn representation
A key K with a binary representation $K = 2^{d-1}b_0 + 2^{d-2}b_1 + \ldots + 2b_{d-2} + b_{d-1}$
Ensure:
$A_0 = [K]G$
$(H_i)_{i\in\{0,..,d-1\}}$, the Hamming distances
function
Random Moduli configuration C
$A_1 = [2]A_0$
$H_0 =$ **Hamming Weight of** (A_0, A_1)
for i=1 to d-1 **do**
$A_{\overline{b_i}} = A_{\overline{b_i}} + A_{b_i}$
$A_{b_i} = [2]A_{b_i}$
$H_i =$ **Hamming distance between actual** (A_0, A_1) **and previous** (A_0, A_1)
end for
end function

We obtain a vector of Hamming distances $H = (H_0, \ldots, H_{d-1})$.

Question!

Can we find K if we know the sequence H?

Modular multiplication Montgomery Algorithm

Algorithm RNS modular multiplication

Require:

A base $\mathcal{B}_n = \{m_1, ..., m_n\}$ where $M = \prod_{i=0}^{n} m_i$

A base $\widetilde{\mathcal{B}}_n = \{\widetilde{m}_1, ..., \widetilde{m}_n\}$ where $\widetilde{M} = \prod_{i=0}^{n} \ \widetilde{m}_i$

N in $\mathcal{B}_n$ and $\widetilde{\mathcal{B}}_n$ with gcd(N,M)=1 and 0<2N<M

$A, B \in \mathbb{Z}$ in $\mathcal{B}_n$ and $\widetilde{\mathcal{B}}_n$ with $A \times B < NM$

function

$Q \leftarrow (-A \times B) \times N^{-1}$ in base $\mathcal{B}_n$

Extension 1 of Q, from $\mathcal{B}_n$ to $\widetilde{\mathcal{B}}_n$

$R \leftarrow (A \times B + Q \times N) \times M^{-1}$ in base $\widetilde{\mathcal{B}}_n$

Extension 2 of R, from $\widetilde{\mathcal{B}}_n$ to $\mathcal{B}_n$

end function

Ensure: $R \equiv ABM^{-1}$ mod N with R<2N

J.C. Bajard & al.(2004) "Leak Resistant Arithmetic".

- Choose $2n$ fixed moduli $\{\mu_1, .., \mu_{2n}\}$ pairwise coprime.
- Draw $\{m_1, ..., m_n\}$ among $\{\mu_1, .., \mu_{2n}\}$ for $\mathcal{B}_n$, the remaining $\{\widetilde{m}_1, ..., \widetilde{m}_n\}$ for $\widetilde{\mathcal{B}}_n$.

Algorithm RNS modular multiplication

Require:

A base $\mathcal{B}_n = \{m_1, ..., m_n\}$ where $M = \prod_{i=0}^{n} m_i$

A base $\widetilde{\mathcal{B}}_n = \{\widetilde{m}_1, ..., \widetilde{m}_n\}$ where $\widetilde{M} = \prod_{i=0}^{n} \widetilde{m}_i$

N in $\mathcal{B}_n$ and $\widetilde{\mathcal{B}}_n$ with gcd(N,M)=1 and 0<2N<M

$A, B \in \mathbb{Z}$ in $\mathcal{B}_n$ and $\widetilde{\mathcal{B}}_n$ with $A \times B < NM$

function

$Q \leftarrow (-A \times B) \times N^{-1}$ in base $\mathcal{B}_n$

Extension 1 of Q, from $\mathcal{B}_n$ to $\widetilde{\mathcal{B}}_n$

$R \leftarrow (A \times B + Q \times N) \times M^{-1}$ in base $\widetilde{\mathcal{B}}_n$

Extension 2 of R, from $\widetilde{\mathcal{B}}_n$ to $\mathcal{B}_n$

end function

Ensure: $R \equiv ABM^{-1}$ mod N with R<2N

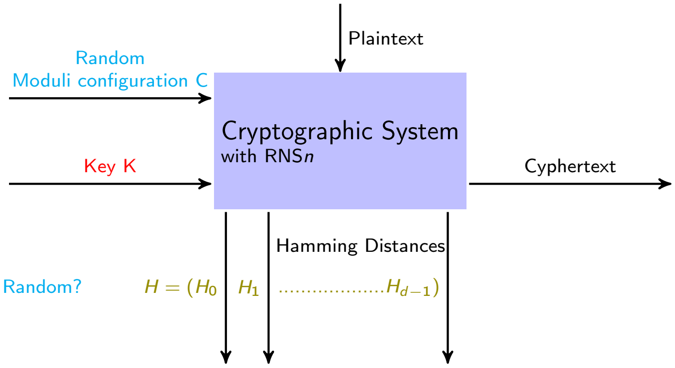
J.C. Bajard & al.(2004) "Leak Resistant Arithmetic".

- Choose $2n$ fixed moduli $\{\mu_1, .., \mu_{2n}\}$ pairwise coprime.
- Draw $\{m_1, ..., m_n\}$ among $\{\mu_1, .., \mu_{2n}\}$ for $\mathcal{B}_n$, the remaining $\{\widetilde{m}_1, ..., \widetilde{m}_n\}$ for $\widetilde{\mathcal{B}}_n$.

Question

What is the level of protection ensured by random moduli?

Perfect Noise



- $L(H, K)$ the joint distribution of (H, K),
- $L(H|K)$ the conditional distribution of H given K,
- $L(H)$ and $L(K)$ the marginal distributions of H and K.

The perfect noise must fulfill $L(H, K) = L(H|K)L(K) = L(H)L(K)$.

Said differently

$$L(H) - L(H|K) = 0$$

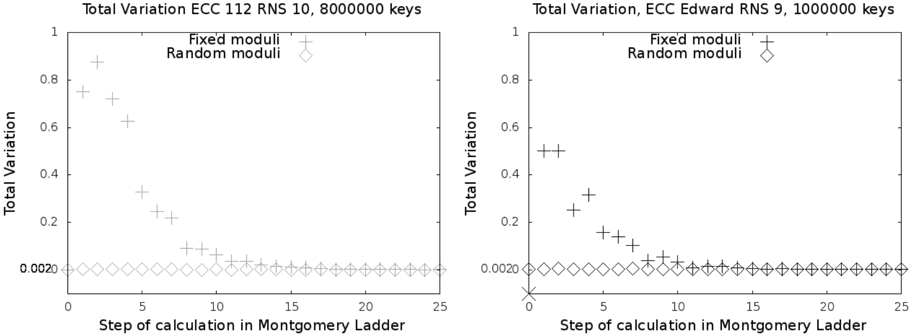
.

Total Variation to Independence (TVI) with Monte Carlo Method

Evaluation of the distance between $L(H)$ and $L(H|K)$

$$I = [0, 2^p[= \bigcup_{k=0}^{2^{p'}-1} I_k \text{ and } \mathcal{H}^i = [min(H_i), max(H_i)] = \bigcup_{j=0}^{q-1} \mathcal{H}^i_j$$

$$\text{TVI}_i = \frac{1}{2} \sum_{k=0}^{2^{p'}-1} \sum_{j=0}^{q-1} \left| P\left(H_i \in \mathcal{H}^i_j\right) - P\left(H_i \in \mathcal{H}^i_j | K \in I_k\right) \right|.$$



Total Variation as a function of the calculation step.

Testing tools

Given values of $H = (H_0, \ldots, H_{d-1})$, what can be done to evaluate the quality of randomization?

1. Nist Statistical Tests
   Issue: the vector H has a multivariate Gaussian distribution.
2. Leakage Analysis
   - Total Variation to Independence (TVI).
   - Mutual Information Analysis (MIA).
   - Differential Power Analysis (DPA).
   - Correlation Power Analysis (CPA).
   - Maximum Likelihood Estimator (MLE) used for Template Attack.

Mutual Information Analysis (MIA) for randomized moduli

$$MIA_i = \sum_{k=0}^{2^{p'}-1} P(K \in I_k) \sum_{j=0}^{q-1} P(H_i \in \mathcal{H}_j^i | K \in I_k) \log \left(\frac{P(H_i \in \mathcal{H}_j^i | K \in I_k)}{P(H_i \in \mathcal{H}_j^i)} \right).$$

- Using Mean Square Error $MSE = variance(P)$

$$MSE_{P\left(H_i \in \mathcal{H}_j^i | K \in I_k\right)} \approx \frac{\sigma^2 \left(\mathbf{1}_{\{H_i \in \mathcal{H}_j^i | K \in I_k\}} \right)}{S}.$$

Mutual Information Analysis (MIA) for randomized moduli

$$MIA_i = \sum_{k=0}^{2^{p'}-1} P(K \in I_k) \sum_{j=0}^{q-1} P(H_i \in \mathcal{H}_j^i | K \in I_k) \log \left(\frac{P(H_i \in \mathcal{H}_j^i | K \in I_k)}{P(H_i \in \mathcal{H}_j^i)} \right).$$

- Using Mean Square Error $MSE = variance(P)$

$$MSE_{P\left(H_i \in \mathcal{H}_j^i | K \in I_k\right)} \approx \frac{\sigma^2 \left(\mathbf{1}_{\{H_i \in \mathcal{H}_j^i | K \in I_k\}} \right)}{S}.$$

- $\log\left(P\left(H_i \in \mathcal{H}_j^i\right)\right)$ and $\log\left(P\left(H_i \in \mathcal{H}_j^i | K \in I_k\right)\right)$ have biased Monte Carlo estimators.

- Using Mean Square Error $MSE = bias^2(\log(P)) + variance(\log(P))$

$$MSE_{\log\left(P\left(H_i \in \mathcal{H}_j^i\right)\right)} \approx \frac{\sigma^2 \left(\mathbf{1}_{\{H_i \in \mathcal{H}_j^i\}} \right)}{SP^2(H_i \in \mathcal{H}_j^i)} \quad \text{and} \quad MSE_{\log\left(P\left(H_i \in \mathcal{H}_j^i | K \in I_k\right)\right)} \approx \frac{\sigma^2 \left(\mathbf{1}_{\{H_i \in \mathcal{H}_j^i | K \in I_k\}} \right)}{SP^2(H_i \in \mathcal{H}_j^i | K \in I_k)}.$$

Conclusion

For quantities smaller than one, the logarithm increases the distances but amplifies significantly the variance. **It becomes difficult to use MIA_i as a distinguisher**.

DPA for randomized moduli

Denote

$$\overline{H}_i(K,C) = \frac{1}{S}\sum_{l=1}^{S} H_i(K,C^l) \quad \text{and} \quad \overline{H}_i(K'_j,C') = \frac{1}{S}\sum_{l=1}^{S} H_i(K'_j,C^{l+S}).$$

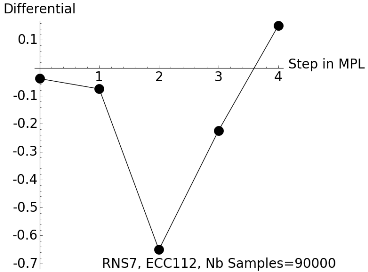
We use the difference:

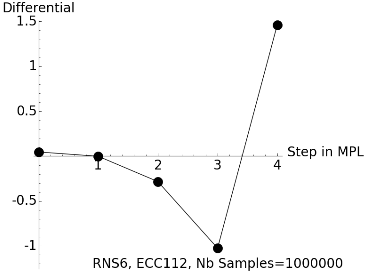
$$\mathrm{DIFF}_i = \overline{H}_i(K,C) - \overline{H}_i(K'_j,C').$$

For example, when $K = 110111101110_2$:

- We get 1^{st} zero from $K = 110111101110_2$ and $K'_1 = 111111111111_2$.
- We get 2^{de} zero from $K = 110111101110_2$ and $K'_2 = 110111111111_2$.
- We get 3^{rd} zero from $K = 110111101110_2$ and $K'_3 = 110111101111_2$.

DPA for randomized moduli





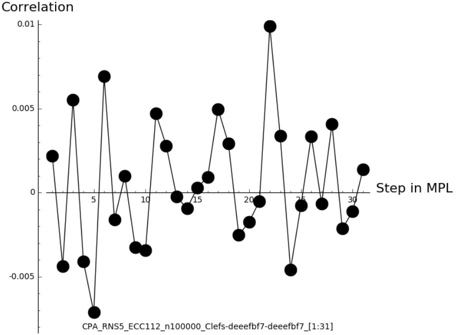
RNS6 and RNS7: DPA between $0xfffffff$ and $0xdeeefbf7$ with respectively a sample of size $S = 1000000$ and $S = 90000$.

$0xdeeefbf7 = 11\textcolor{red}{0}11110111011101111101111110111_2$

CPA for randomized moduli

CPA use the correlation at step i between observations $H_i(K, C^l)$ and simulations $H_i(K', C^{l+S})$.

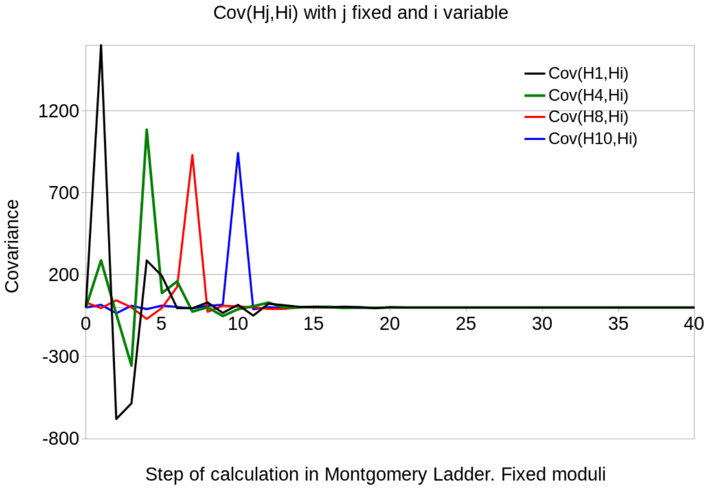
$$\xi_i = \frac{\frac{1}{S}\sum_{l=1}^{S}\left[H_i(K, C^l) - \overline{H}_i(K, C)\right]\left[H_i(K', C^{l+S}) - \overline{H}_i(K', C)\right]}{\sqrt{\frac{1}{S}\sum_{l_1=1}^{S}\left[H_i(K, C^{l_1}) - \overline{H}_i(K, C)\right]^2 \frac{1}{S}\sum_{l_2=1}^{S}\left[H_i(K', C^{l_2+S}) - \overline{H}_i(K', C)\right]^2}}$$



RNS5, Correlation between $0 \times deeefbf7$ and $0 \times deeefbf7$ for a sample of size $S = 100000$.

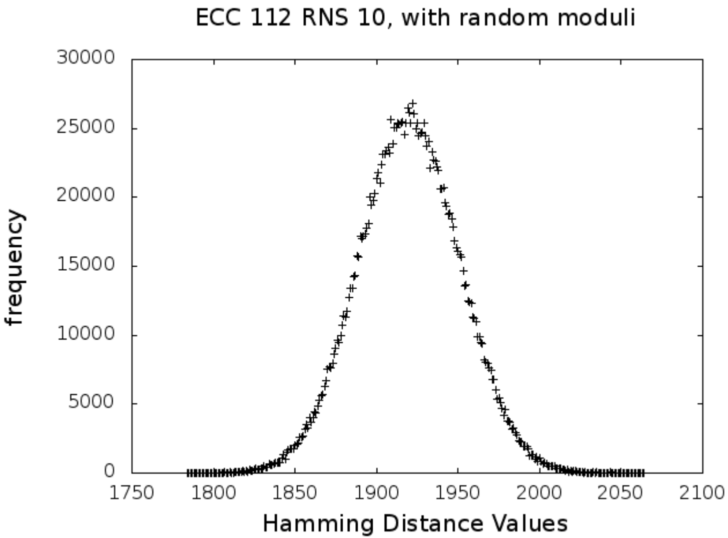
Cross Information

CPA and DPA do not consider cross information between calculation steps.



RNS10, $Cov(H_j, H_i)_{j=1,4,8,10}$.

Marginal Distribution of Hamming distances



Frequency of H_{10}, S=2×10^6.

Assume $H^i = (H_0, ..., H_i)$ has a multivariate Gaussian distribution

Maximum Likelihood Estimator (MLE)

Assume $H^i = (H_0, ..., H_i)$ has a multivariate Gaussian distribution with a density

$$p_{k,i}(x^i) = \frac{1}{\left(\sqrt{2\pi}\right)^{i+1}\sqrt{\det(\Gamma_{k,i})}} \exp\left(-\frac{{}^t(x^i - m^{k,i})\Gamma_{k,i}^{-1}(x^i - m^{k,i})}{2}\right),$$

where $x^i = (x_0, ..., x_i)$ and $(m^{k,i}, \Gamma_{k,i})$ are the mean and the covariance matrix of $H^i = (H_0, ..., H_i)$.

Maximum Likelihood Estimator (MLE)

Assume $H^i = (H_0, ..., H_i)$ has a multivariate Gaussian distribution with a density

$$p_{k,i}(x^i) = \frac{1}{\left(\sqrt{2\pi}\right)^{i+1}\sqrt{\det(\Gamma_{k,i})}} \exp\left(-\frac{{}^t(x^i - m^{k,i})\Gamma_{k,i}^{-1}(x^i - m^{k,i})}{2}\right),$$

where $x^i = (x_0, ..., x_i)$ and $(m^{k,i}, \Gamma_{k,i})$ are the mean and the covariance matrix of $H^i = (H_0, ..., H_i)$.

- Learning Phase

Maximum Likelihood Estimator (MLE)

Assume $H^i = (H_0, ..., H_i)$ has a multivariate Gaussian distribution with a density

$$p_{k,i}(x^i) = \frac{1}{\left(\sqrt{2\pi}\right)^{i+1}\sqrt{\det(\Gamma_{k,i})}} \exp\left(-\frac{{}^t(x^i - m^{k,i})\Gamma_{k,i}^{-1}(x^i - m^{k,i})}{2}\right),$$

where $x^i = (x_0, ..., x_i)$ and $(m^{k,i}, \Gamma_{k,i})$ are the mean and the covariance matrix of $H^i = (H_0, ..., H_i)$.

- Learning Phase

- Estimation Phase

Maximum Likelihood Estimator (MLE)

Assume $H^i = (H_0, ..., H_i)$ has a multivariate Gaussian distribution with a density

$$p_{k,i}(x^i) = \frac{1}{\left(\sqrt{2\pi}\right)^{i+1}\sqrt{\det(\Gamma_{k,i})}} \exp\left(-\frac{{}^t(x^i - m^{k,i})\Gamma_{k,i}^{-1}(x^i - m^{k,i})}{2}\right),$$

where $x^i = (x_0, ..., x_i)$ and $(m^{k,i}, \Gamma_{k,i})$ are the mean and the covariance matrix of $H^i = (H_0, ..., H_i)$.

- Learning Phase

  Learning of $(m^{k,i}, \Gamma_{k,i})$ with a sample of size L.

- Estimation Phase

Maximum Likelihood Estimator (MLE)

Assume $H^i = (H_0, ..., H_i)$ has a multivariate Gaussian distribution with a density

$$p_{k,i}(x^i) = \frac{1}{\left(\sqrt{2\pi}\right)^{i+1} \sqrt{\det(\Gamma_{k,i})}} \exp\left(-\frac{{}^t(x^i - m^{k,i})\Gamma_{k,i}^{-1}(x^i - m^{k,i})}{2}\right),$$

where $x^i = (x_0, ..., x_i)$ and $(m^{k,i}, \Gamma_{k,i})$ are the mean and the covariance matrix of $H^i = (H_0, ..., H_i)$.

- Learning Phase

  Learning of $(m^{k,i}, \Gamma_{k,i})$ with a sample of size L.

- Estimation Phase

  We observe S realizations $\left(x_j^i\right)_{1 \leq j \leq S}$ of $H^i = (H_0, ..., H_i)$.

Maximum Likelihood Estimator (MLE)

Assume $H^i = (H_0, \ldots, H_i)$ has a multivariate Gaussian distribution with a density

$$p_{k,i}(x^i) = \frac{1}{\left(\sqrt{2\pi}\right)^{i+1}\sqrt{\det(\Gamma_{k,i})}} \exp\left(-\frac{{}^t(x^i - m^{k,i})\Gamma_{k,i}^{-1}(x^i - m^{k,i})}{2}\right),$$

where $x^i = (x_0, \ldots, x_i)$ and $(m^{k,i}, \Gamma_{k,i})$ are the mean and the covariance matrix of $H^i = (H_0, \ldots, H_i)$.

- Learning Phase

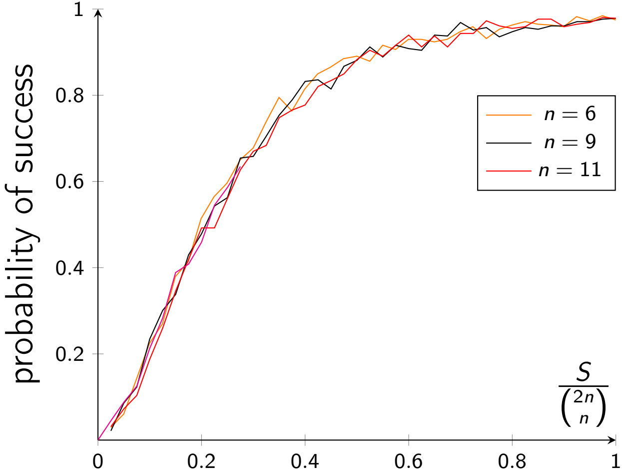
  Learning of $(m^{k,i}, \Gamma_{k,i})$ with a sample of size L.

- Estimation Phase

  We observe S realizations $\left(x_j^i\right)_{1 \leq j \leq S}$ of $H^i = (H_0, \ldots, H_i)$.

  We choose $K = \arg\max_k \left\{ \prod_{j=1}^{S} p_{k,i}(x_j^i) \right\}$.

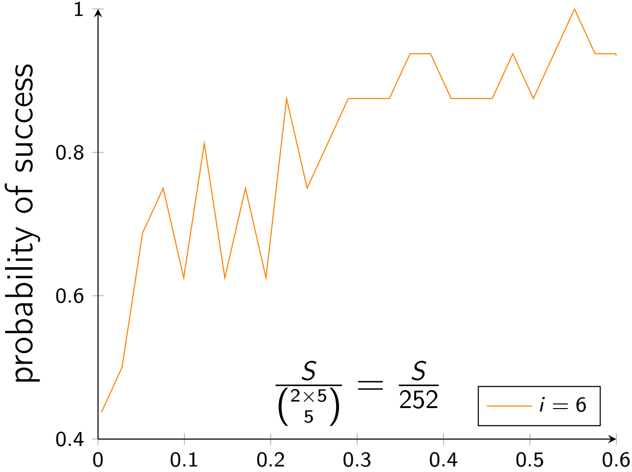
Comparaison between different RNSn with $i = 10$ i.e. $H^{10} = (H_0, \dots, H_{10})$.



Probability of success to find a 10-bits key with MLE on ECC 112 Montgomery in Jacobian coordinates.

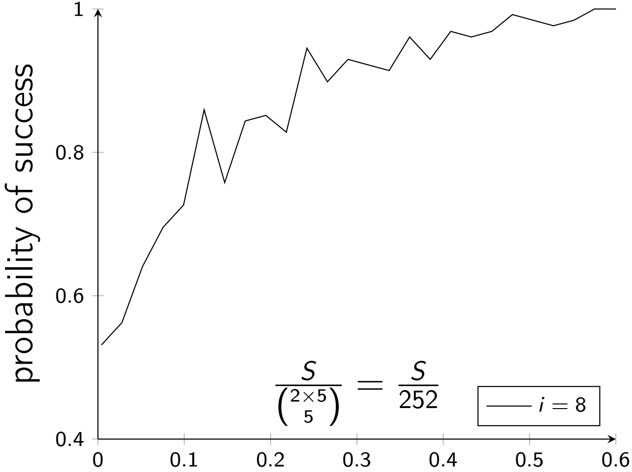
Maximum Likelihood Estimator (MLE)

What happen when $i < 11$ in $H^i = (H_0, H_1, H_2, H_3, H_4, H_5, H_6)$?



Probability of success to find the second bit of the key with MLE on ECC 112 in RNS5.

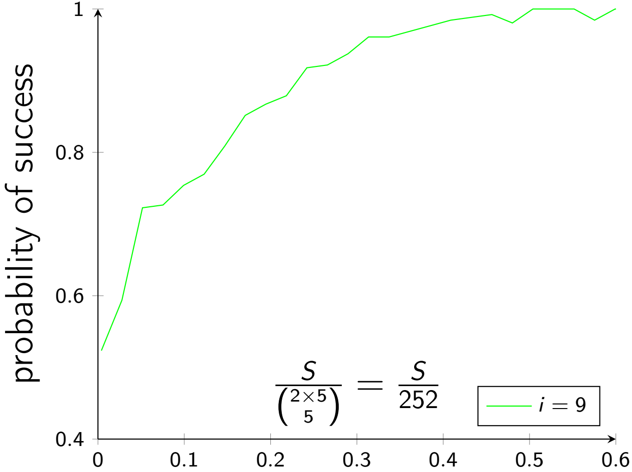
What happen when $i < 11$ in $H^i = (H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8)$?



Probability of success to find the second bit of the key with MLE on ECC 112 in RNS5.

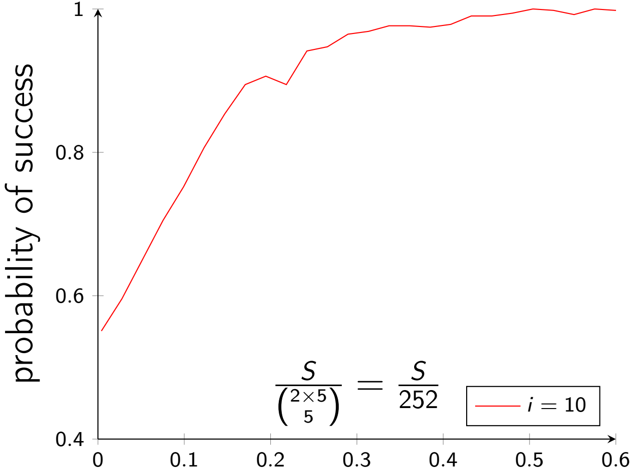
Maximum Likelihood Estimator (MLE)

What happen when $i < 11$ in $H^i = (H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_9)$?



Probability of success to find the second bit of the key with MLE on ECC 112 in RNS5.

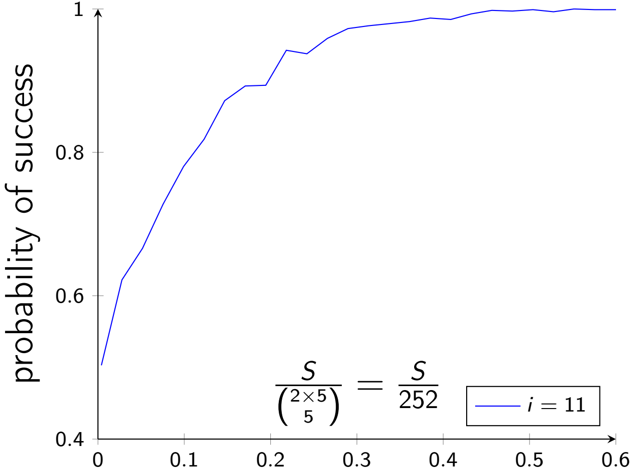
What happen when $i < 11$ in $H^i = (H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_{10})$?



Probability of success to find the second bit of the key with MLE on ECC 112 in RNS5.

Maximum Likelihood Estimator (MLE)

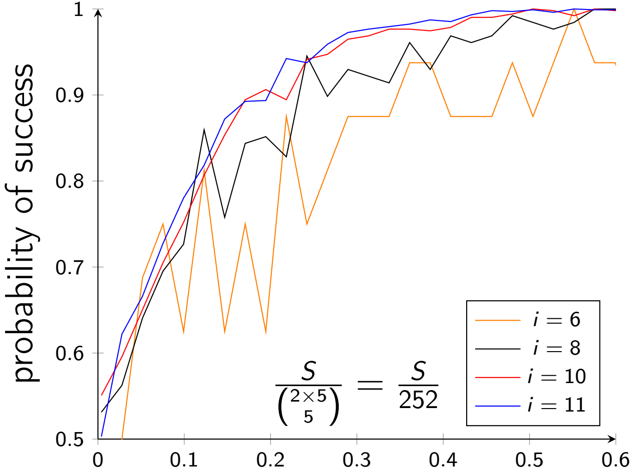
What happen when $i < 11$ in $H^i = (H_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7, H_8, H_{10}, H_{11})$?



Probability of success to find the second bit of the key with MLE on ECC 112 in RNS5.

Maximum Likelihood Estimator

What happen when $i < 11$ in $H^i = (H_0, ..., H_i)$?



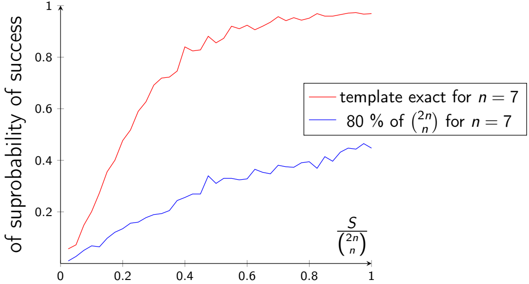
Probability of success to find the second bit of the key with MLE on ECC 112 in RNS5.

Maximum Likelihood Estimator

- Considering success rate < 0.1, what is the minimum n to protect an attack based on S traces?

	#ECC			
Number of traces S	112	256	384	521
2^{30}	16	15	15	18

- The learning phase costs more than the estimation phase even with Monte Carlo.



- From which level we loose random behaviour?
  We have to use $n > 7$ to avoid an attack with a single trace
  With a 95% prediction interval for an error<0.1%.

Conclusion and future work

Conclusion

- Maximum Information in ten first steps of calculation.
- DPA is possible but inconsistent.
- CPA is unreliable.
- MIA is difficult to be used as distinguisher.
- MLE give strong information on leakage.
  Modelisation of success as a function of $\frac{S}{\binom{2n}{n}}$ invariant with n.

Future Work

- Is there sufficient information in only one trace? Few traces?
- A template with conditional desintegration could give more information on the key?
- Can we find a better template with the Monte Carlo method using variance reduction?

Thanks for your attention.
Do you have any questions?

jerome.courtois@lip6.fr

Maximum Likelihood Estimator

$S \times \frac{\#ECC-1}{9}$	#ECC 112	256	384	521
2^{10}	6	9	13	18
2^{15}	8	9	13	18
2^{20}	11	10	13	18
2^{25}	13	13	13	18
2^{30}	16	15	15	18
2^{35}	19	18	18	18
2^{40}	21	20	20	20
2^{45}	24	23	23	22
2^{50}	26	26	25	25

Table: Minimum n to protect the whole key till $S \times \frac{\#ECC-1}{9}$ traces of the target key.$(m^{k,\mathbf{10}}, \Gamma_{k,\mathbf{10}})$ is the exact value. $p_t = 0.1$.

Elliptic Curves for Cryptography (ECC)

Elliptic Curves for Cryptography(ECC)

The domain of an ECC denoted $E(F_p)$ is defined by:

- A finite field F_p with p a prime number
- Two elements a and b belonging to F_p
- An equation E : $y^2 \equiv x^3 + ax + b$ mod p
- $G(x_G, y_G)$ a base point of $E(F_p)$ and n prime number is the order of G on $E(F_p)$
- Four types of curve are implemented: 112, 256, 384 et 521 bits
- Implementation in Jacobian coordinates.
- Scalar Multiplication with Montgomery or Co-Z Scale.

In addition we test on an Edward curve 25219 in affine coordinates.

Extensions

1. Raw method, only for first extension.
   But we obtain $\widetilde{X} = X + \alpha \times M$.
2. Shenoy-Kamuresan for the second extension.
   Correction of the error with using an extra modulo and large choice of moduli.
3. Mix-Radix to have an exact computation.

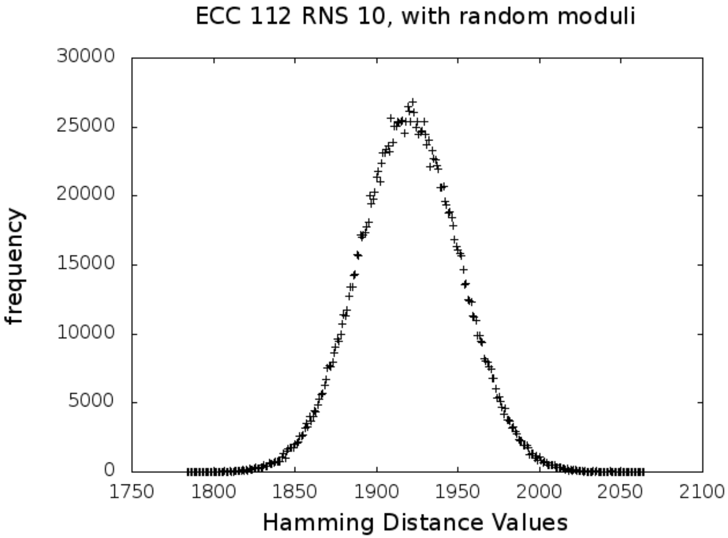


Figure: Frequency of $H_{\mathbf{10}}$, $2 \times 10^{\mathbf{6}}$ computations.

Evaluation of moduli MLE

Not hollow moduli=as many as 1 as 0
Hollow moduli=a maximum of 1
$2^{32} - \epsilon$= many 1 as most significant bit

moduli type	size	special	succes	9 and 10 bits found
Not hollow moduli	≤ 32	random	62.89%	77.53%
Hollow moduli	=32	$2^{32} - \epsilon$	62.30% (61.32%)	74.6% (75.78%)
Not hollow moduli	=32	$2^{32} - \epsilon$	59.57%	73.82%
Any	=27	random	58.98%	72.85%
Not hollow moduli	=32	random	52.73% (60.93%)	68.75% (73.4%)
Any	≤ 32	random	62.5.50 % (54.10%)	75.78% (70.31%)
Any	$= 32$	random	54.29%	69.53%

ECC 112, RNS5, 1000 for template, 100 for MLE

Let us denote the null hypothesis
H_0 : "We obtain 10 bits of the key with a probability equal to 2^{-9}"
We calculate the 95% prediction interval with $p = 2^{-9}$:

$$\mathcal{I}_p = \left[p - 1.96\sqrt{\frac{p(1-p)}{SE}}; p + 1.96\sqrt{\frac{p(1-p)}{SE}} \right].$$

SE is a sample size. If $f \in \mathcal{I}_p$, we do not reject **H_0** otherwise we reject **H_0** .
We can notice in Table that we have to use $n > 7$ to avoid an attack with a single trace. This confirms the suggestion of [**?**].

n	5	6	7	8	9	10	11
S	1	1	1	5	7	16	130

Minimum size to reject **H_0** with a sample size

$SE = 32256$ ($error < 0.1\%$ for a 95% prediction interval)