Comparison of Side-Channel countermeasures

Sonia Belaïd, Vincent Grosso, Romain Poussier, François-Xavier Standaert.

Journées C2, 27th March, 2014

Outline

Background

Results

SPRF

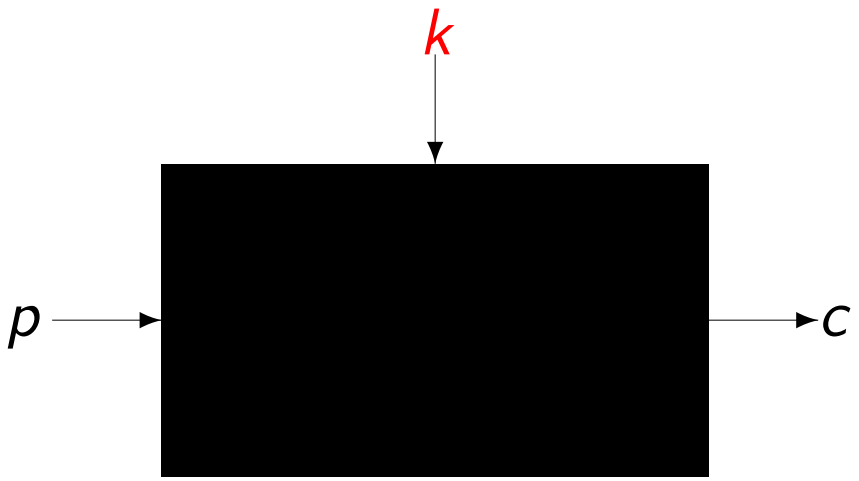
Outline

Background

Results

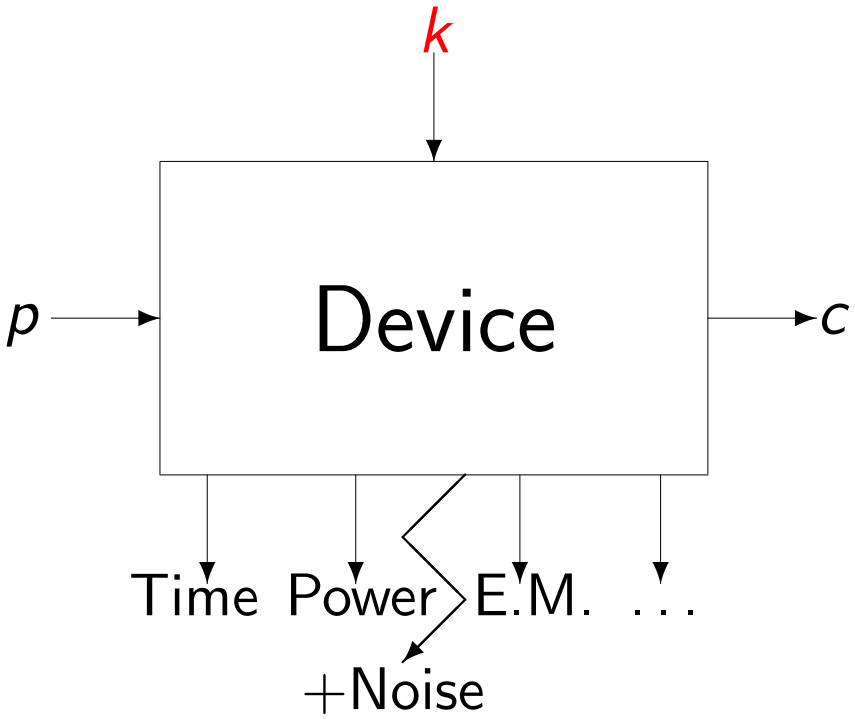
SPRF

Physical model



The adversary uses public information that could be compute by an oracle who knows the secret.

Physical model



The adversary can use additional information dues to computation of secret operations.

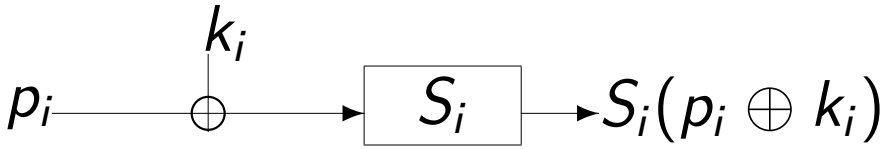
Side Channel Attack

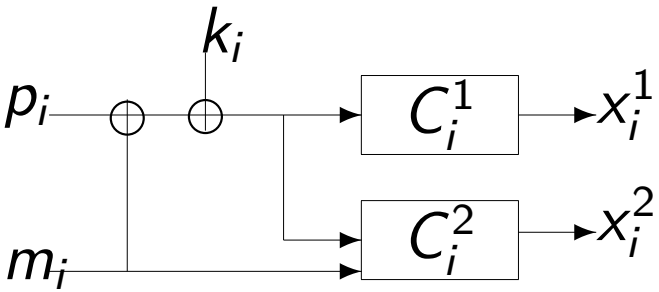
- A probability for each subkey k_i^*, according to the leakages $\mathbf{l}$, and public information $\mathbf{p}$.

$$Pr[\mathrm{K}_i = k_i^* | \mathrm{L} = \mathbf{l}, \mathrm{P} = \mathbf{p}]$$

- Recover the full key $K = (k_0, \ldots, k_{15})$, according to the previous probability (some enumeration could be done).
- A Side Channel Attack is evaluated according to the enumeration required to recover the key in function of the data complexity.

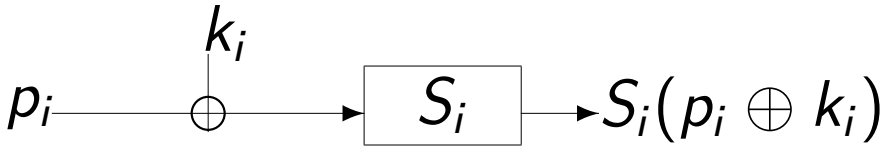
Masking (a.k.a. secret sharing)

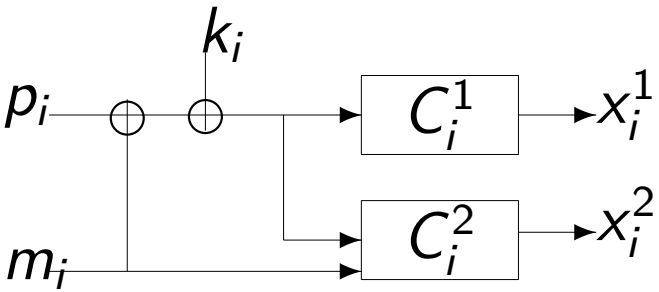




For unprotected device basic computations depend on the secret.

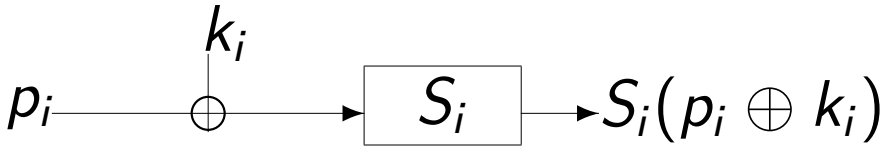
Masking (a.k.a. secret sharing)

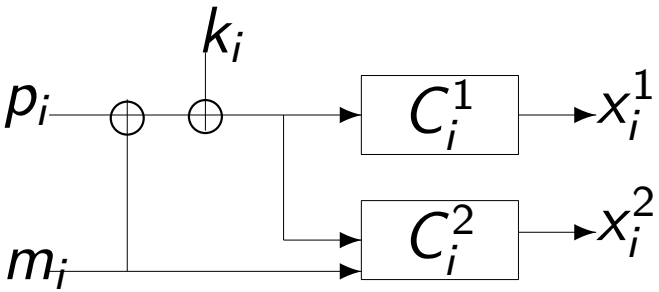




When masking is implemented basic computations are independent of the secret.

Masking (a.k.a. secret sharing)



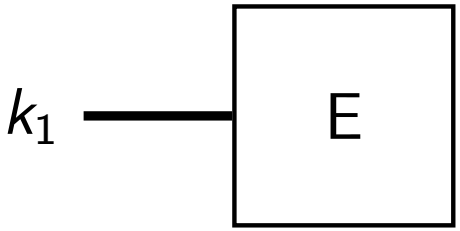


The number of measurements required for an attack grows up exponentially in the number of shares with noise as basis.

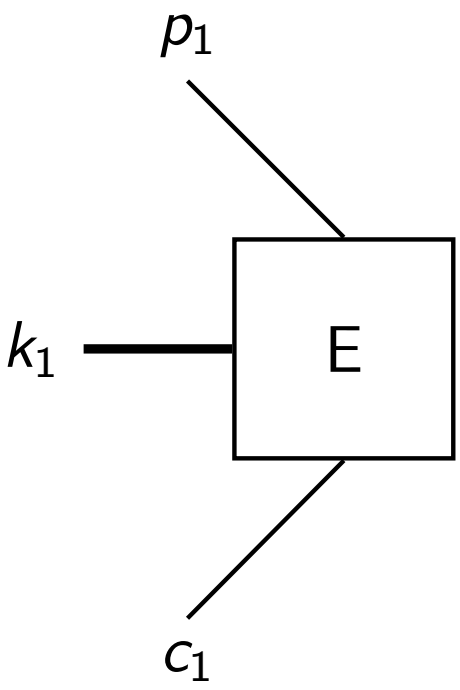
Re-keying (a.k.a. leakage resilient)

- Limit the number of use of a key (n).
- The number of observations is limited.
- The information leaked on secret is limited.

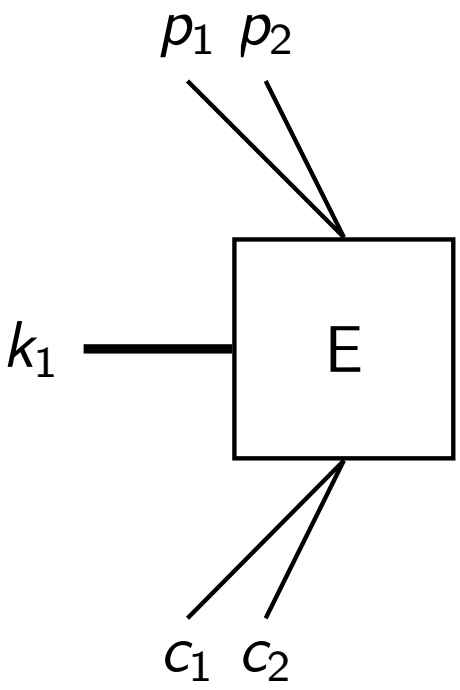
PRG



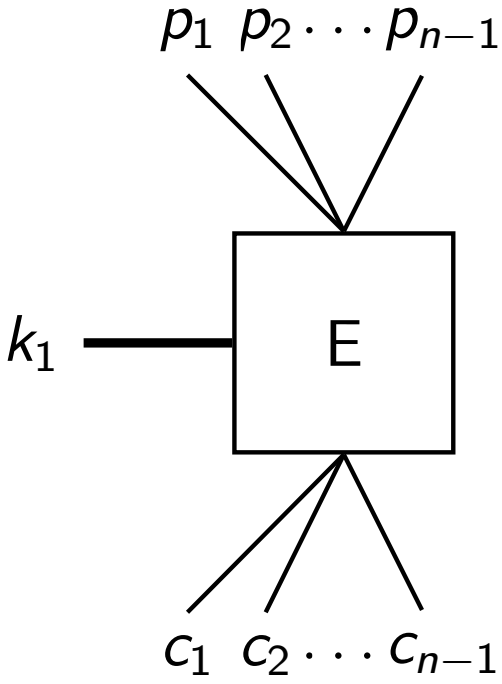
PRG



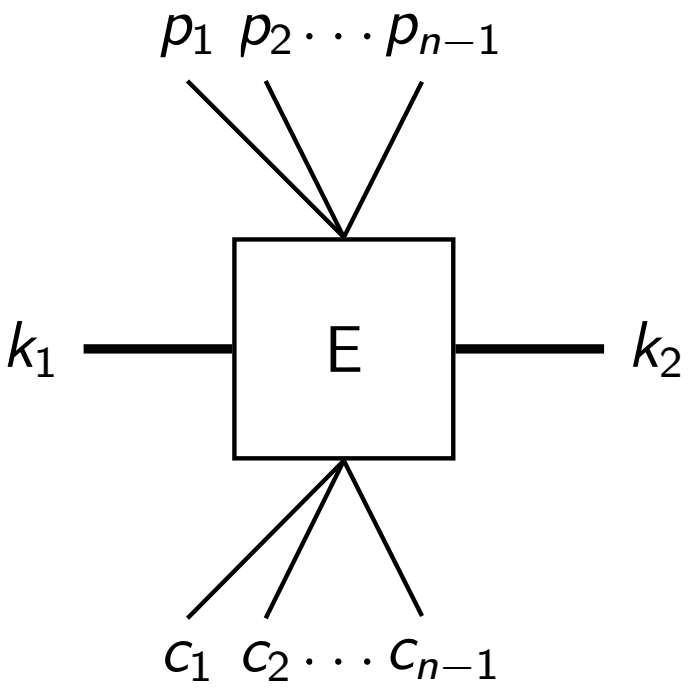
PRG



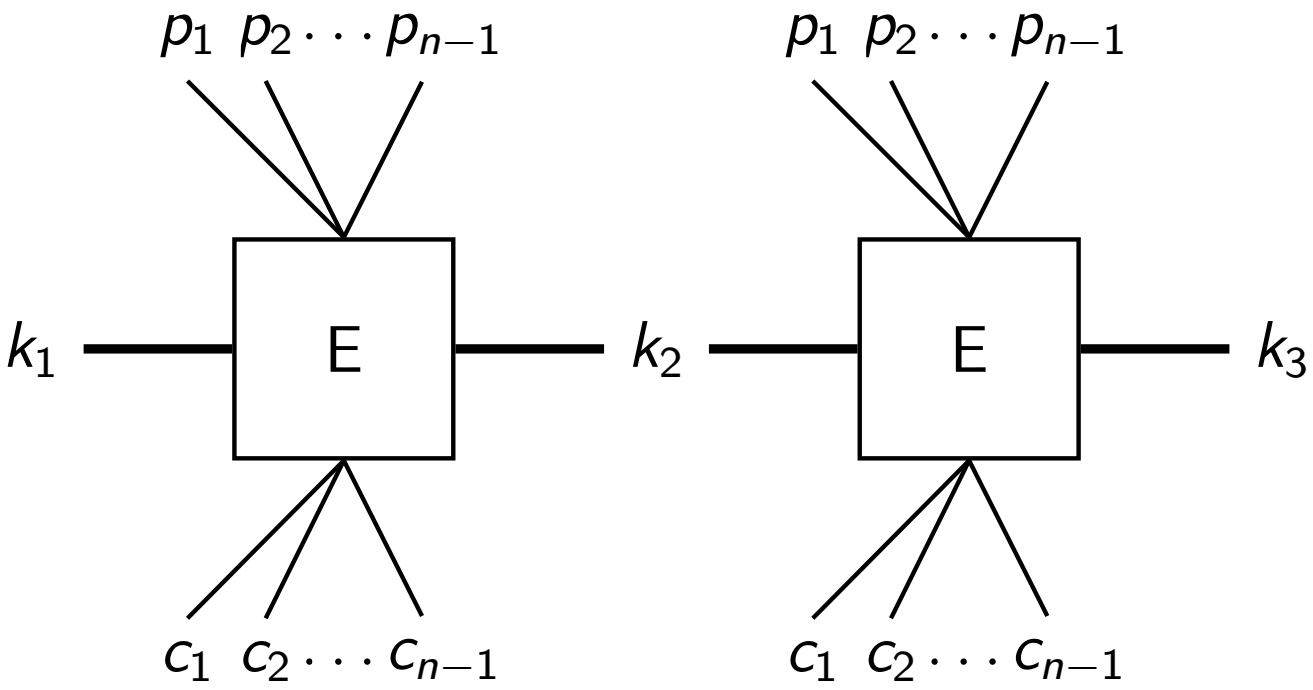
PRG



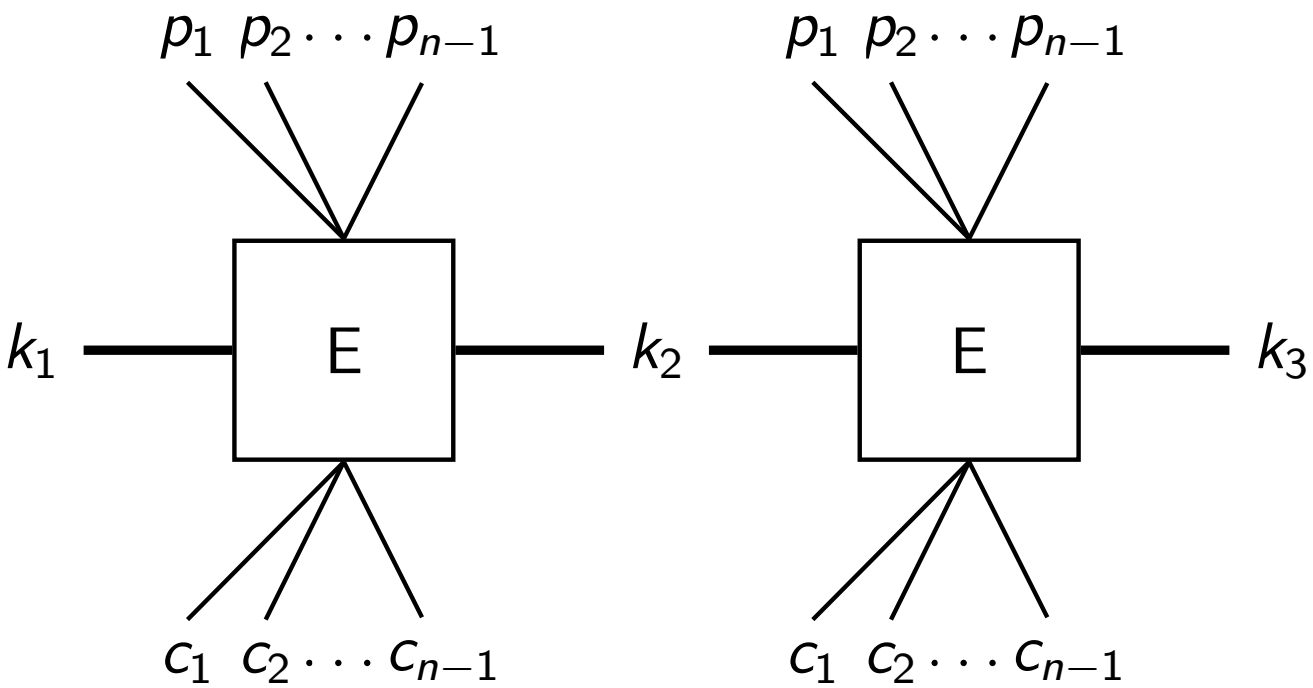
PRG



PRG

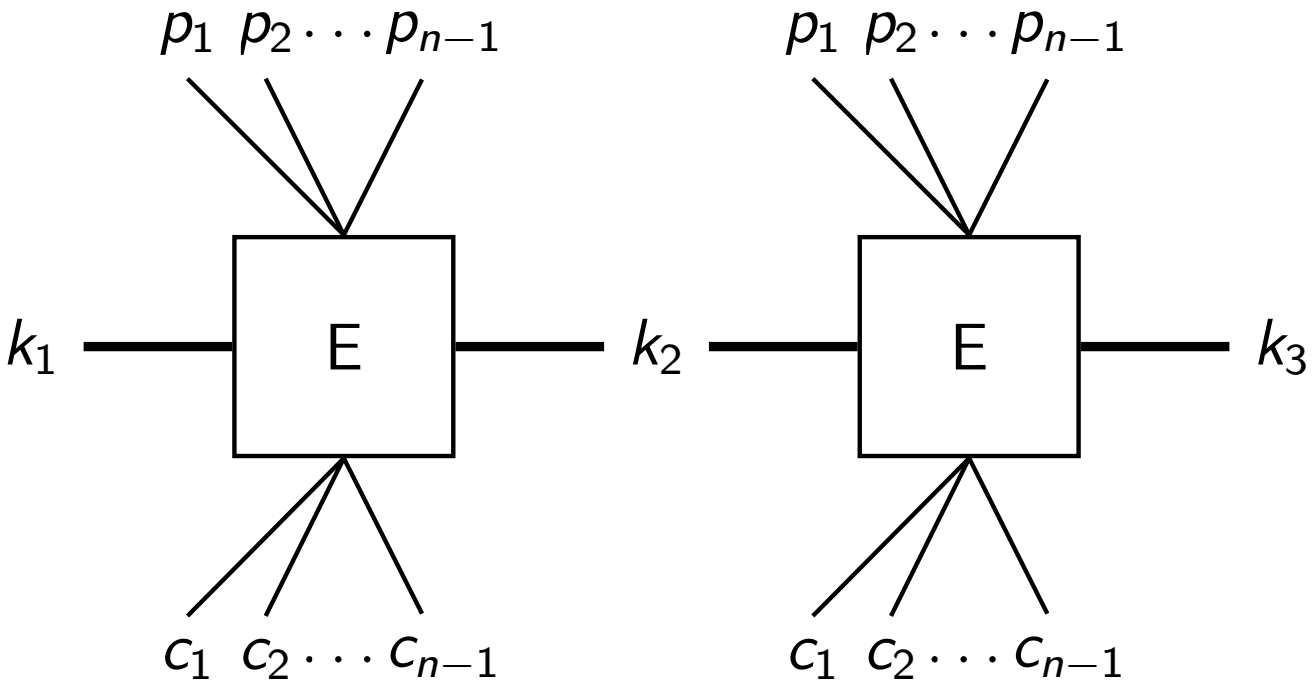

p_1 $p_2 \cdots p_{n-1}$
k_1
E
k_2
c_1 $c_2 \cdots c_{n-1}$
p_1 $p_2 \cdots p_{n-1}$
E
k_3
c_1 $c_2 \cdots c_{n-1}$

PRG



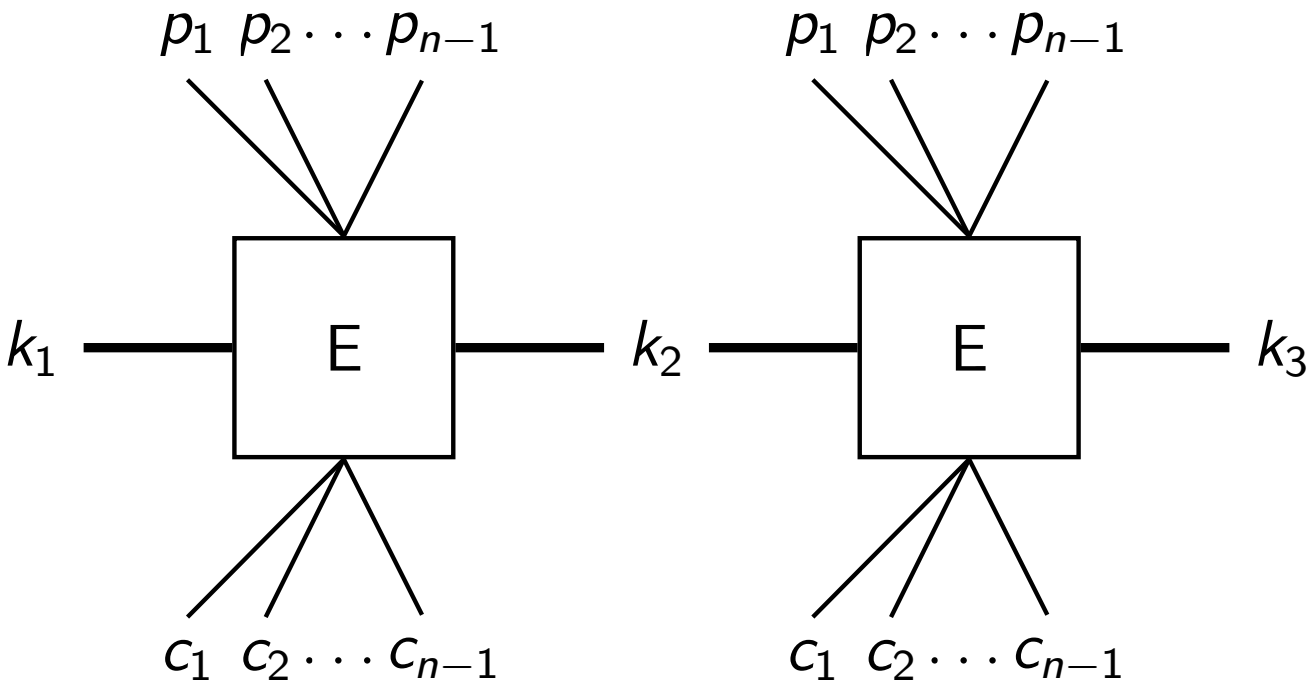
The construction limits the number of measurements for 1 key.

PRG



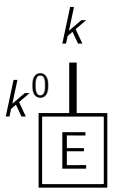
Efficient $\frac{n-1}{n}$ useful evaluation.

PRG

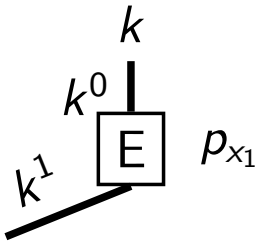


But 2 parties need to be synchronized (statefull).

PRF

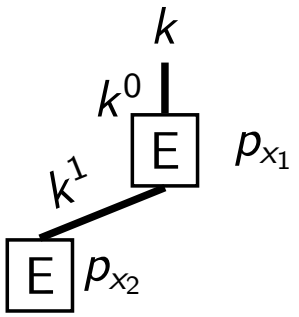


PRF

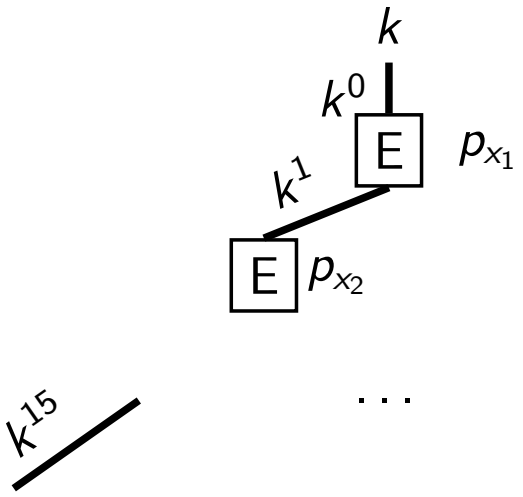


The 2 parties had shared different input p_i that are index by a part of the input of the PRF x_j.

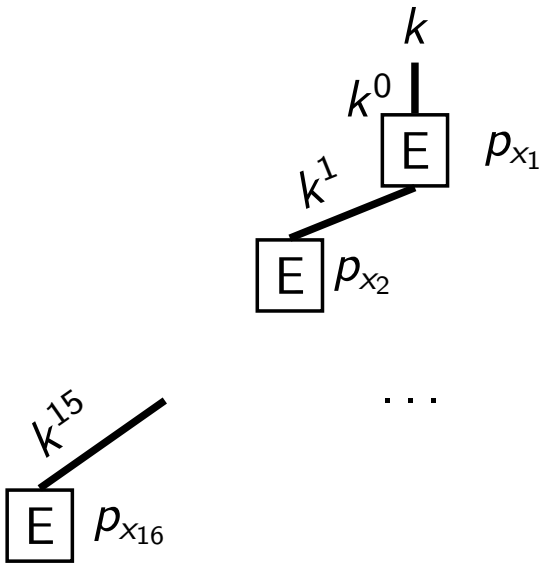
PRF



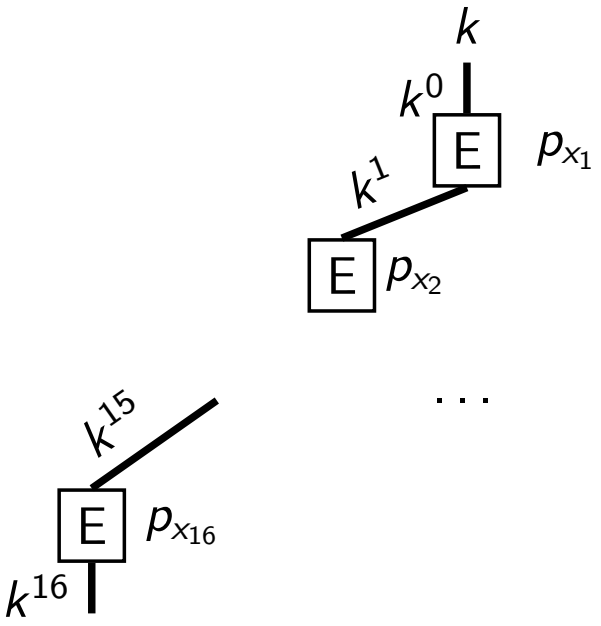
PRF



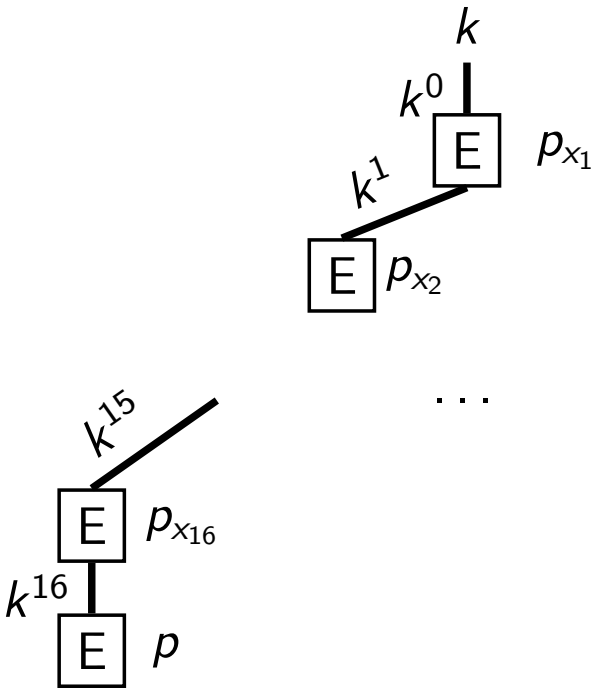
PRF



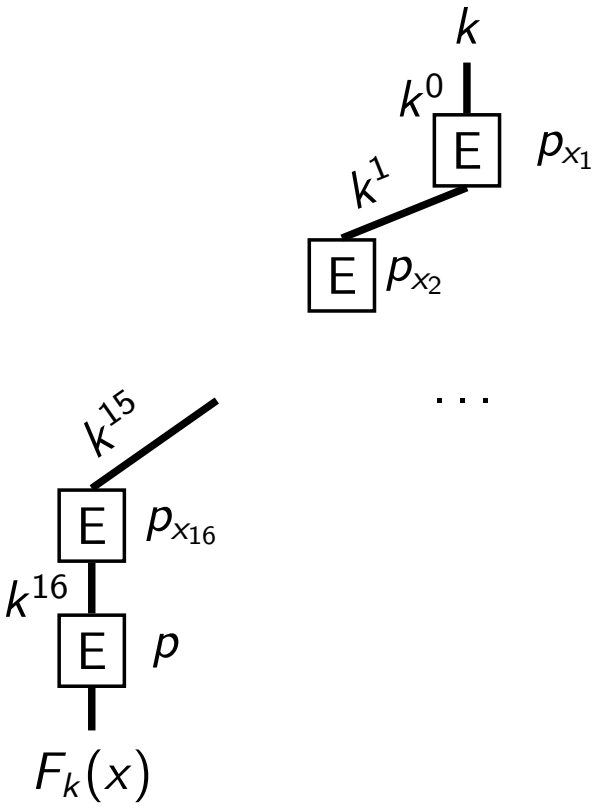
PRF



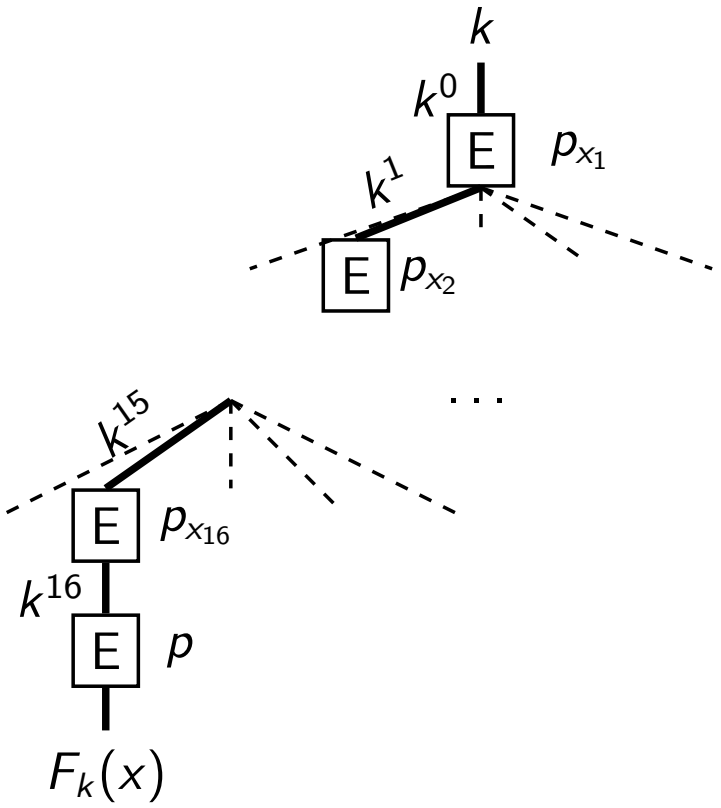
PRF



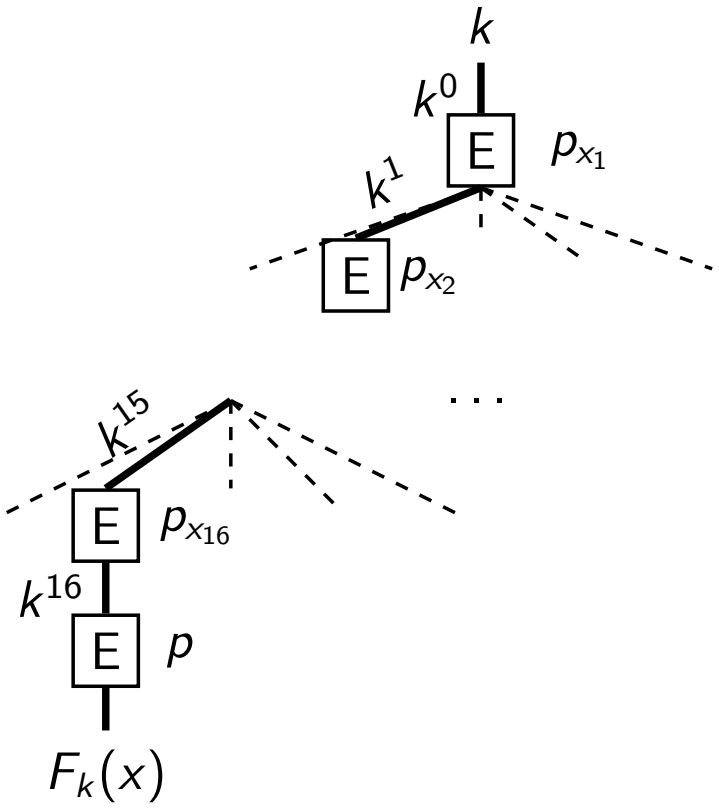
PRF



PRF

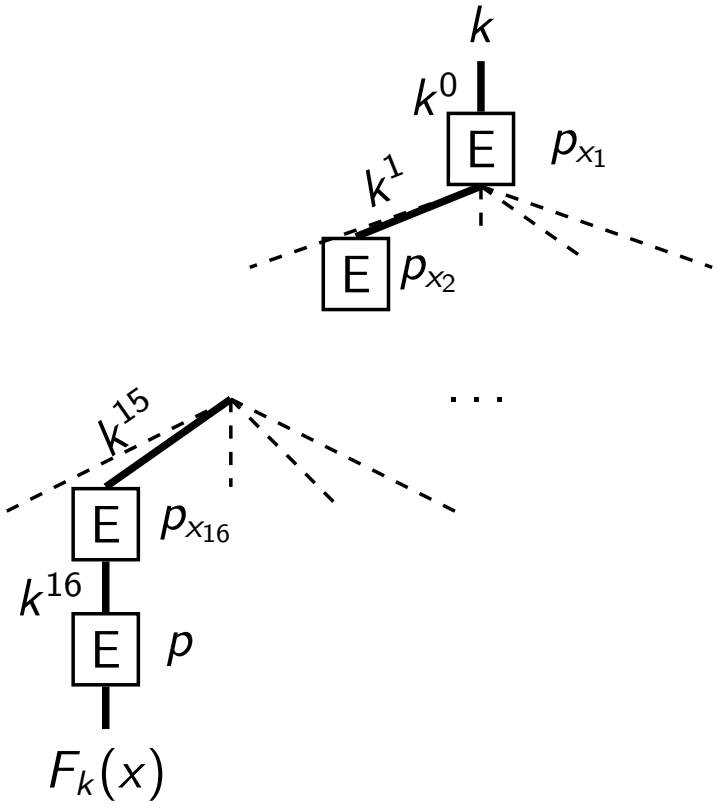


PRF



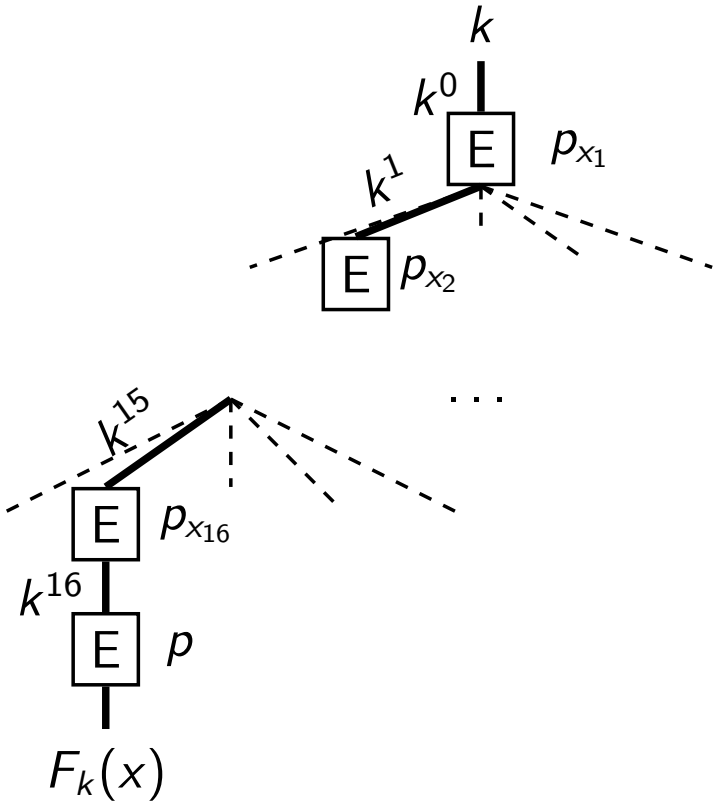
The 2 parties don't need to be synchronized (stateless).

PRF



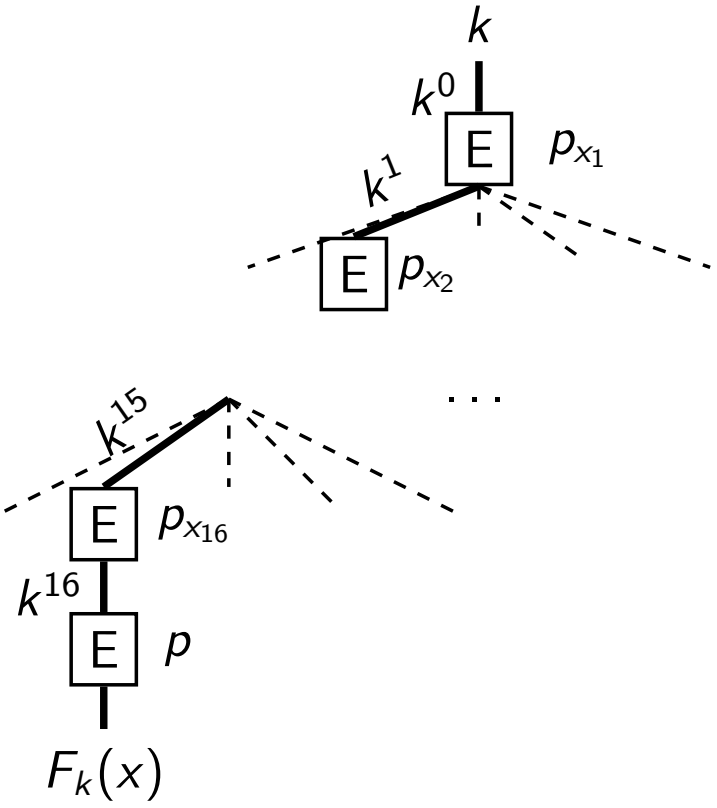
But only the data complexity is limited not the number of measurements.

PRF



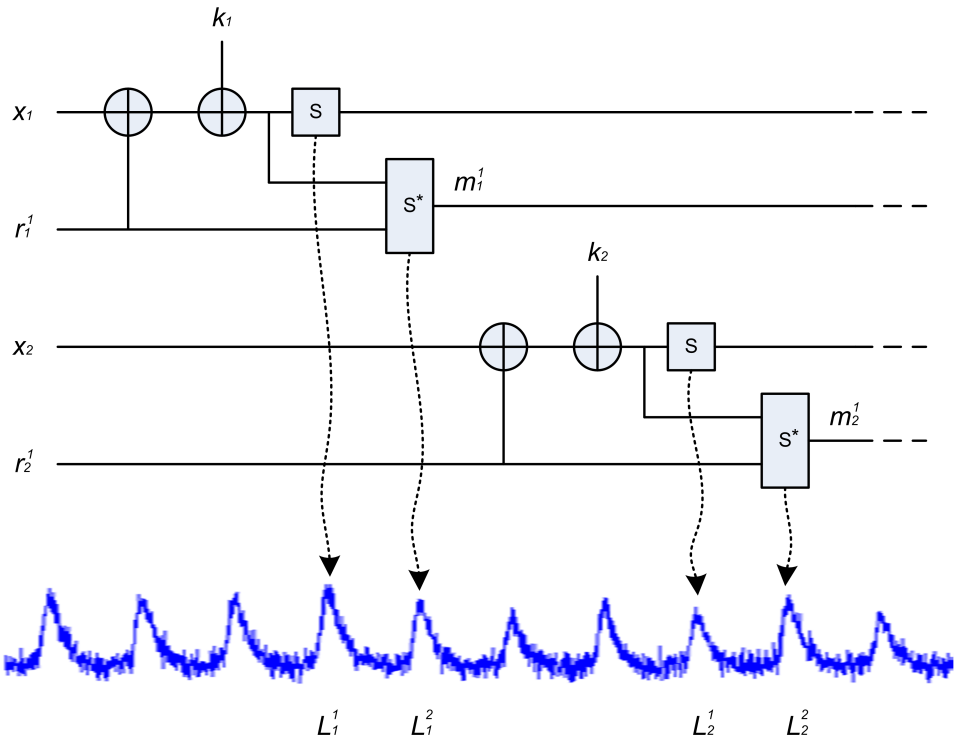
Each node had at most n sons.

PRF



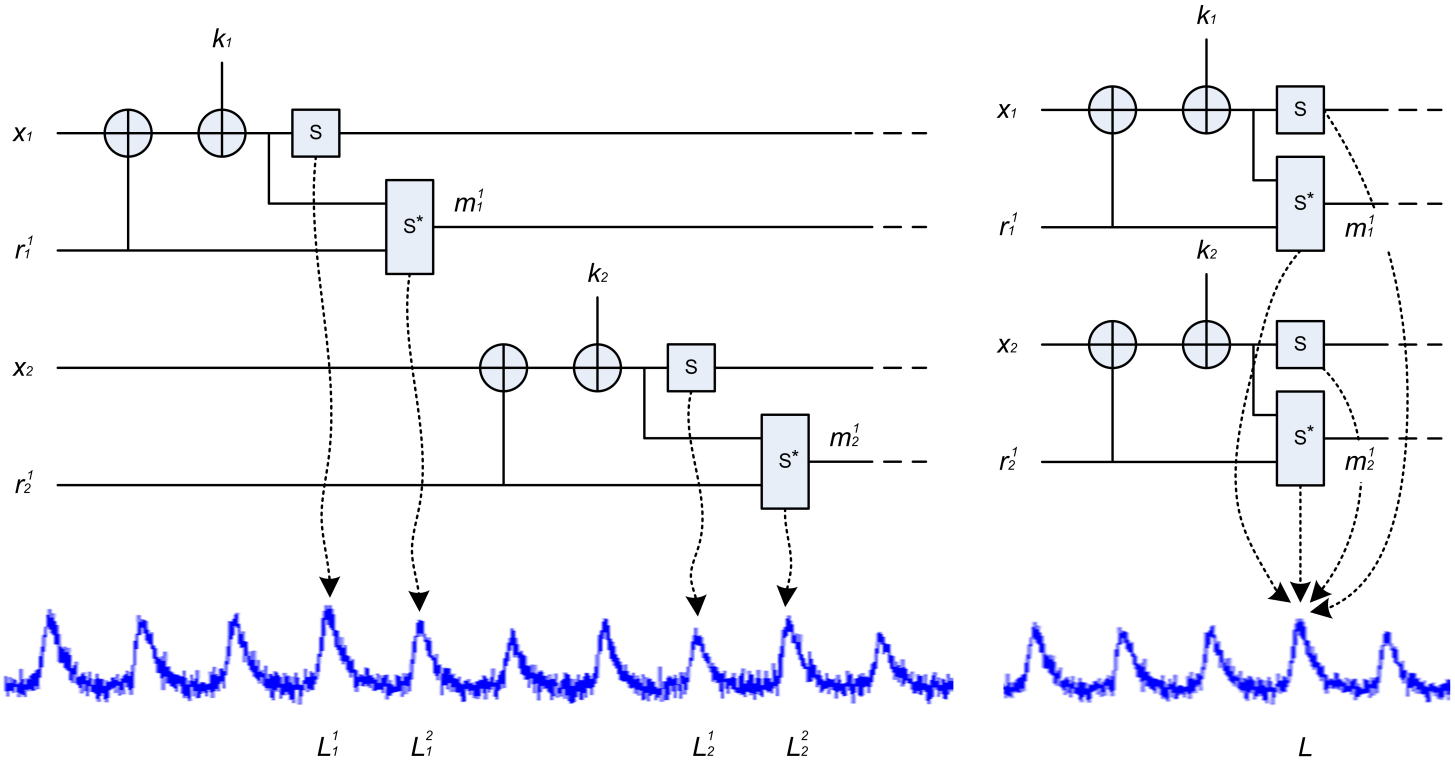
Expensive: depth of the tree call for one PRF evaluation.

Simulated leaking implementations



For software, the operations are done sequentially. At each time, the trace depends only on one share.

Simulated leaking implementations



For hardware, the operations are done in parallel. The leakages on different shares are at the time in the trace.

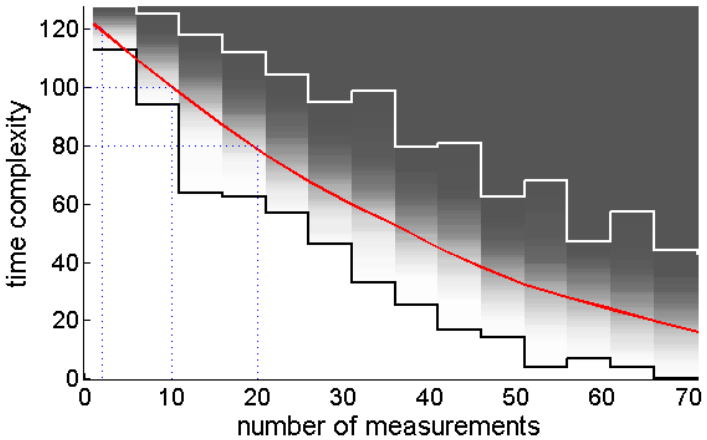
Outline

Background

Results

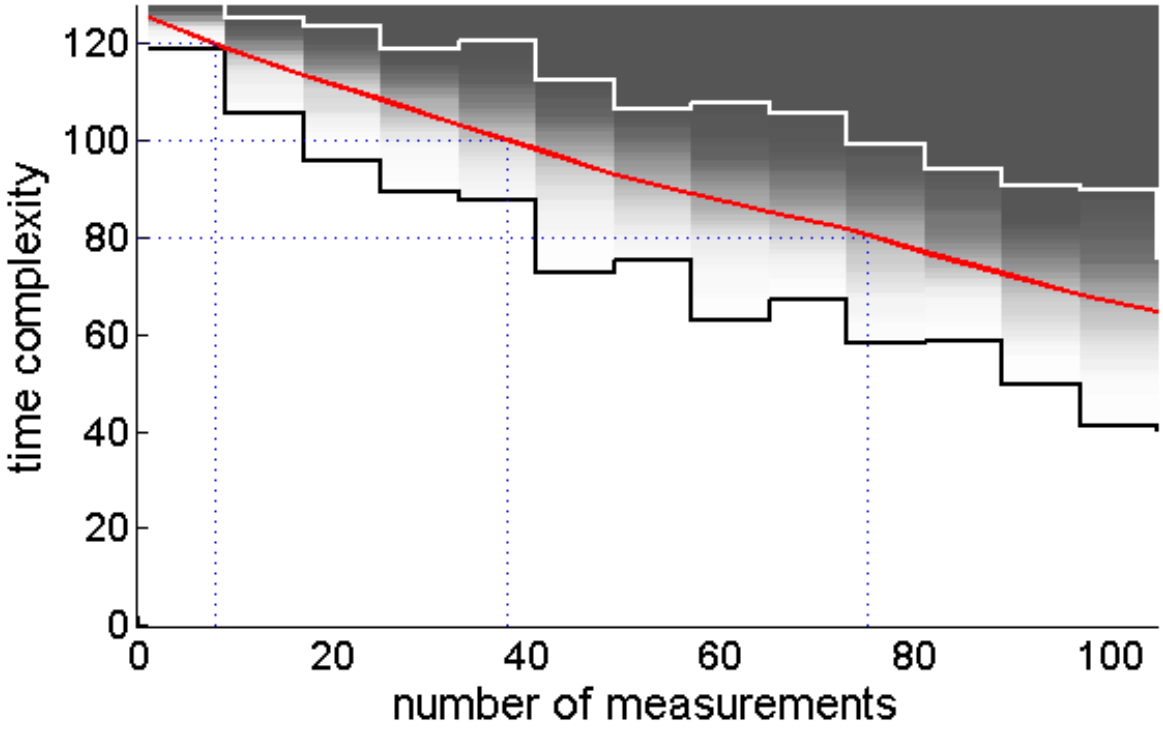
SPRF

Security graph [VCGS13]



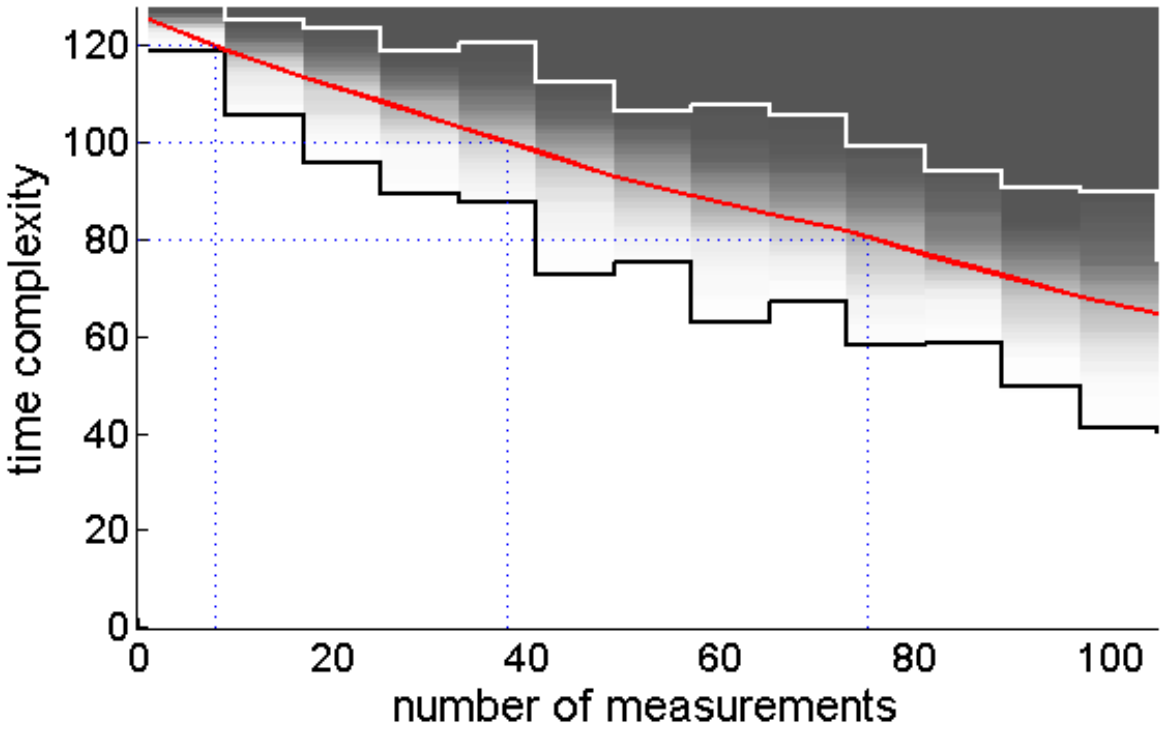
- x-axis corresponds to the number of measurements.
- y-axis corresponds to the rank of the key ($\log_2$ scale).
- E=AES.
- Template attacks, with perfect leakage function.
- Several experiments for a number of measurements.
- Maximal number of measurements for a fixed enumeration.

PRG



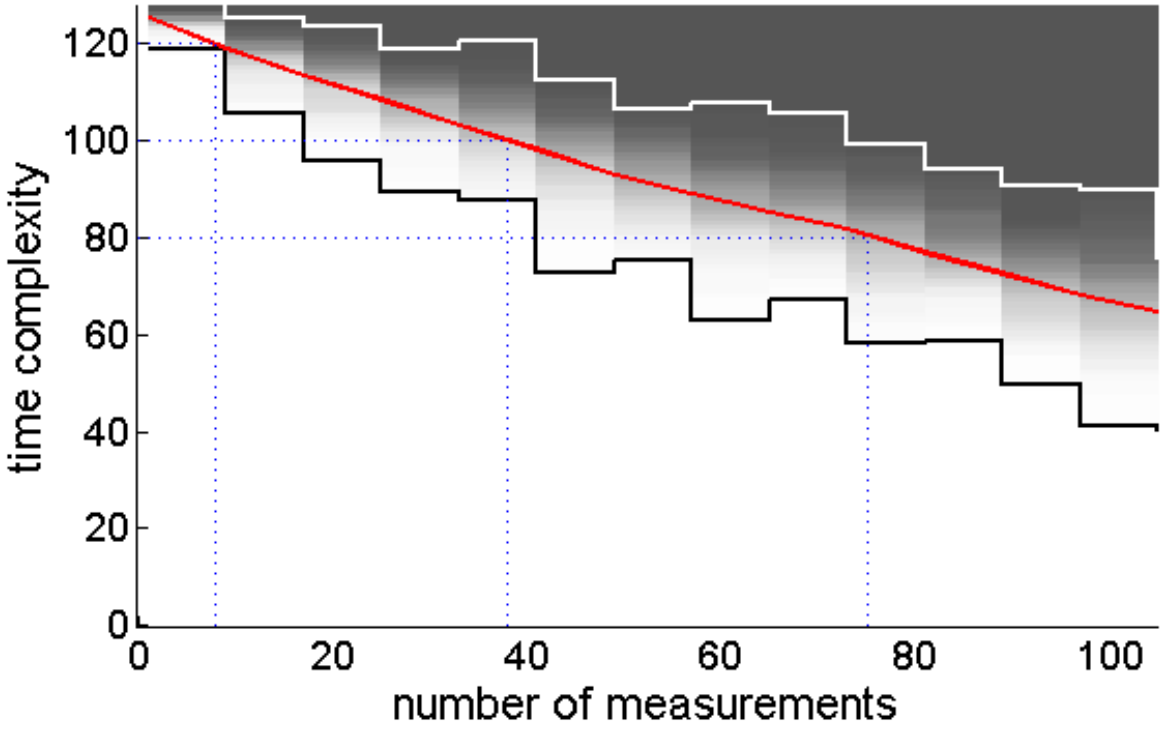
Hardware case, with an SNR of 0.2.

PRG



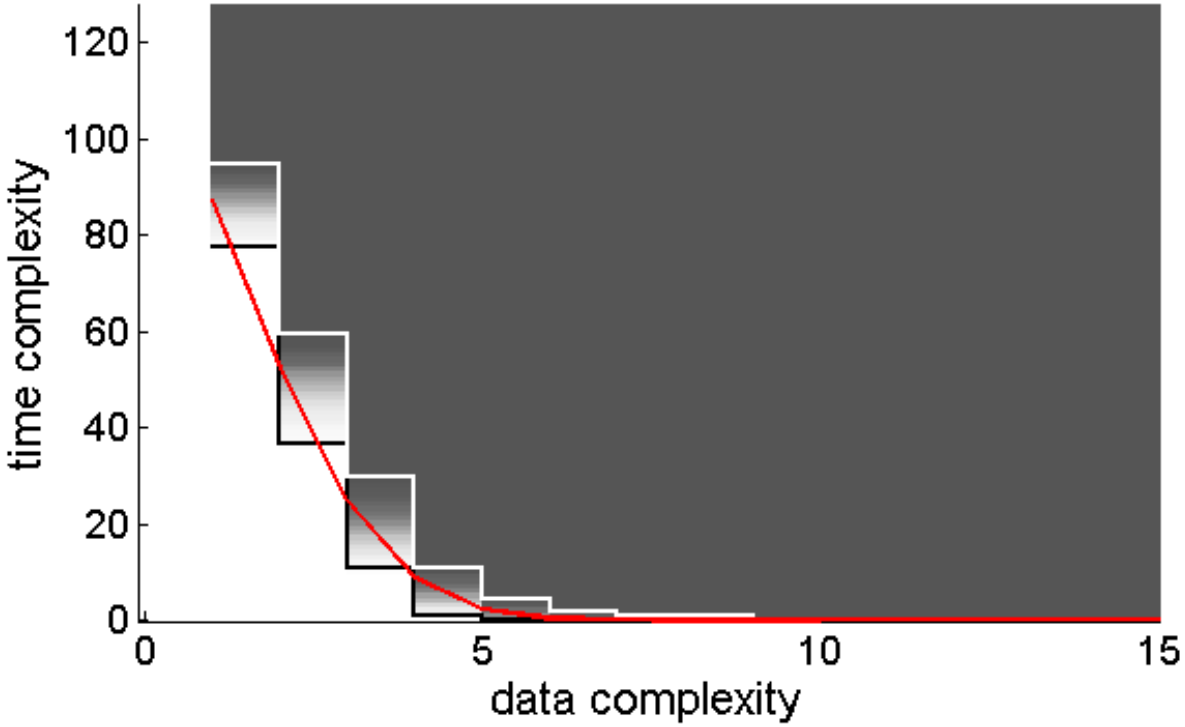
For different security level (i.e. the enumeration to perform) we can extract different number of measurements, before re-keying:-)

PRG



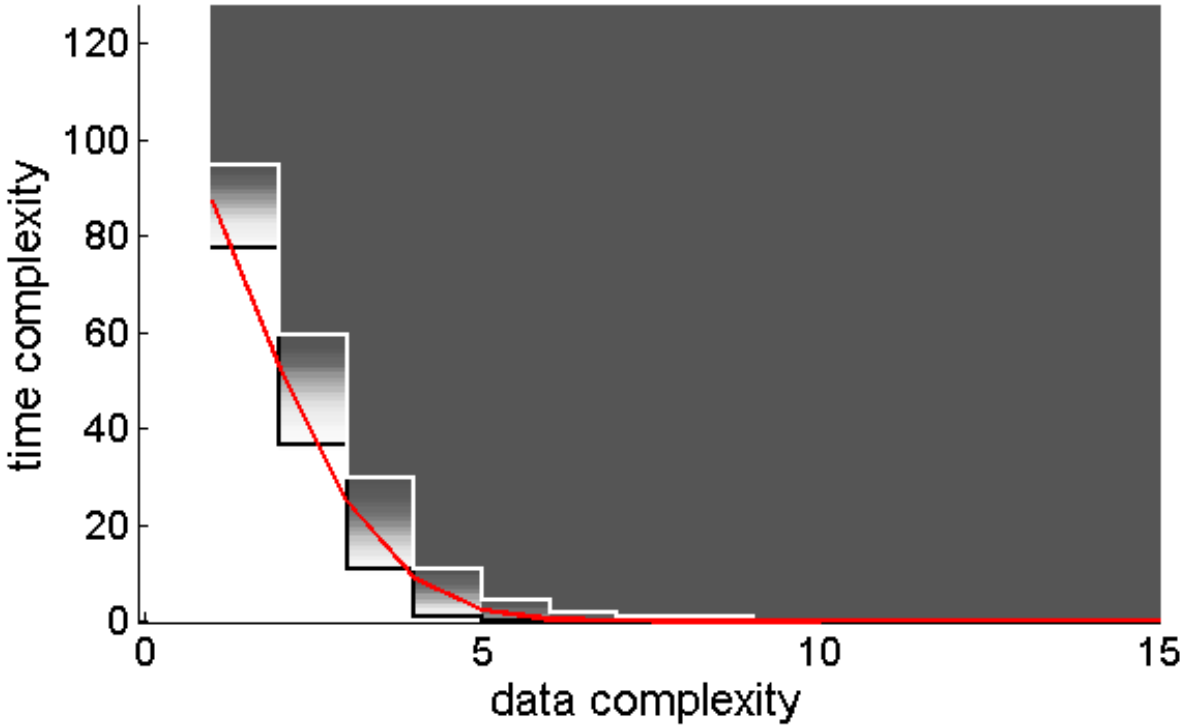
The masking does not improve the bounds, with respect to the cost of masking.

PRF



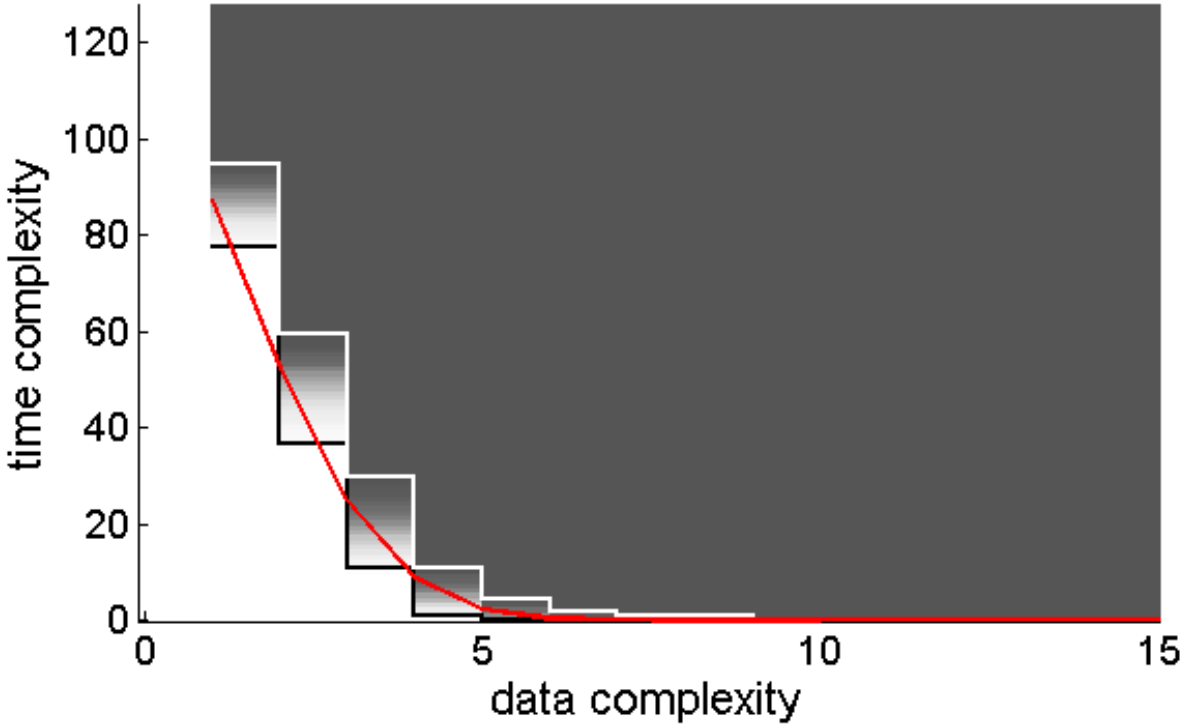
In that case the SNR is ∞. Indeed, the noise can be removed by performing again and again the same measurement.

PRF



No bound could be reached :-(

PRF

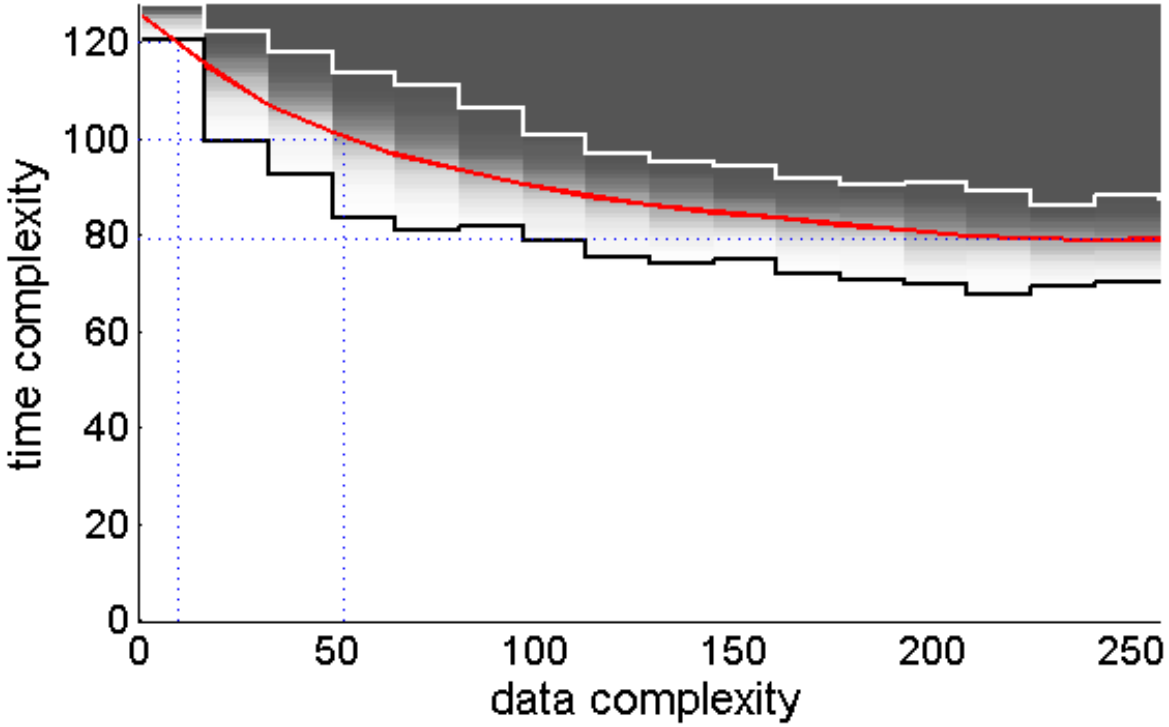


The masking increase the number of measurements required to eliminate the noise.

PRF hardware case

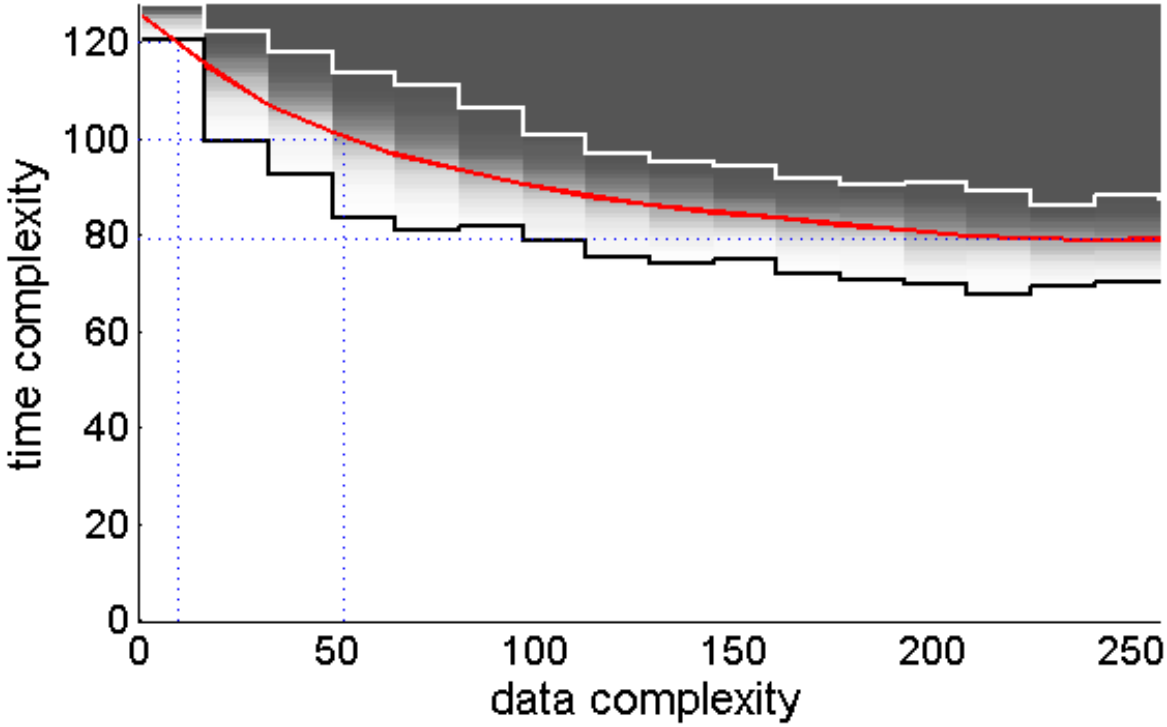
- The parallelize of hardware implementation could be used to realize noise key dependent [MSJ12].
- Use the same byte of plaintext, in the tree construction ($p_{x_1} = (x_1, \ldots, x_1)$).
- $L = \sum_i S(x_1 \oplus k_i)$.
- Even if an adversary recover the key she still needs to perform an enumeration to recover the order for 16 subkey 2^{44}.

Hardware PRF + tweak



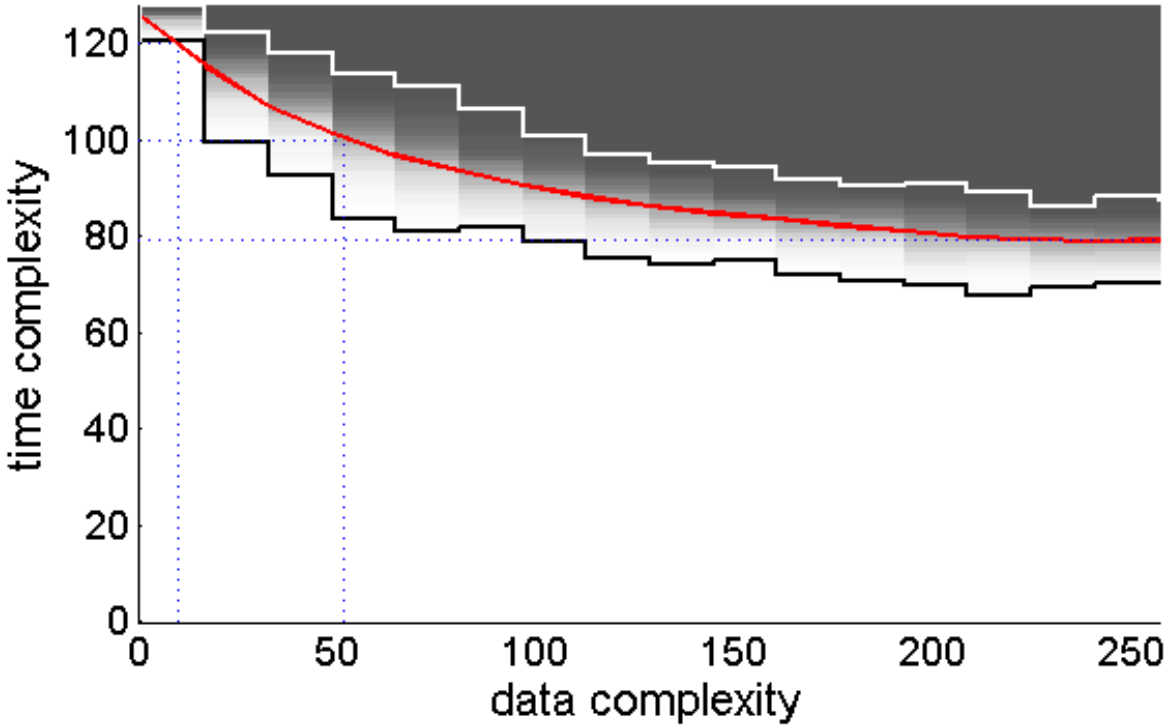
The data complexity is limited to 256.

Hardware PRF + tweak



The physical noise can be averaged, but the key dependent noise is still present (improve the bound).

Hardware PRF + tweak



A bound could be reached :-)

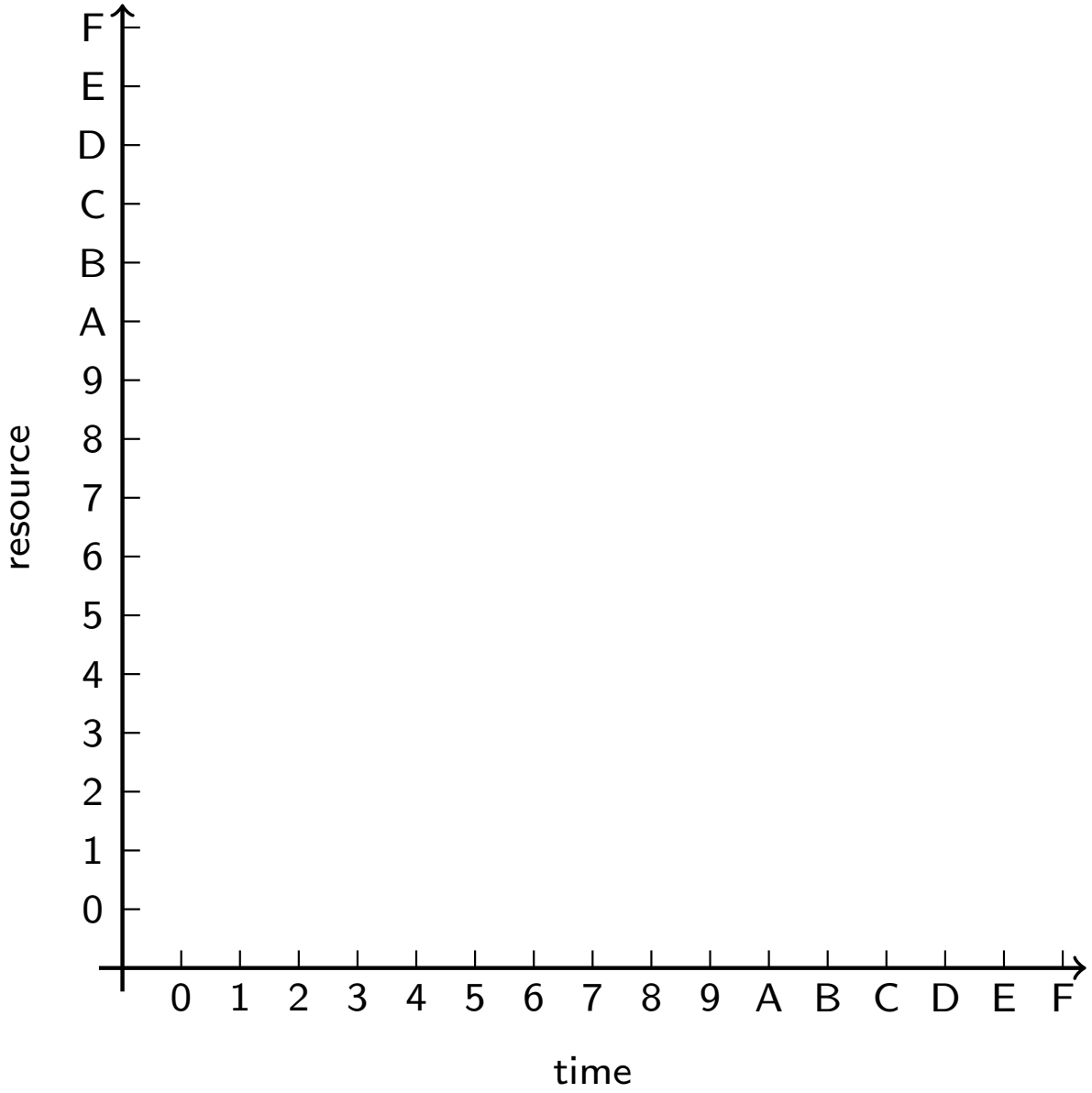
Outline

Background

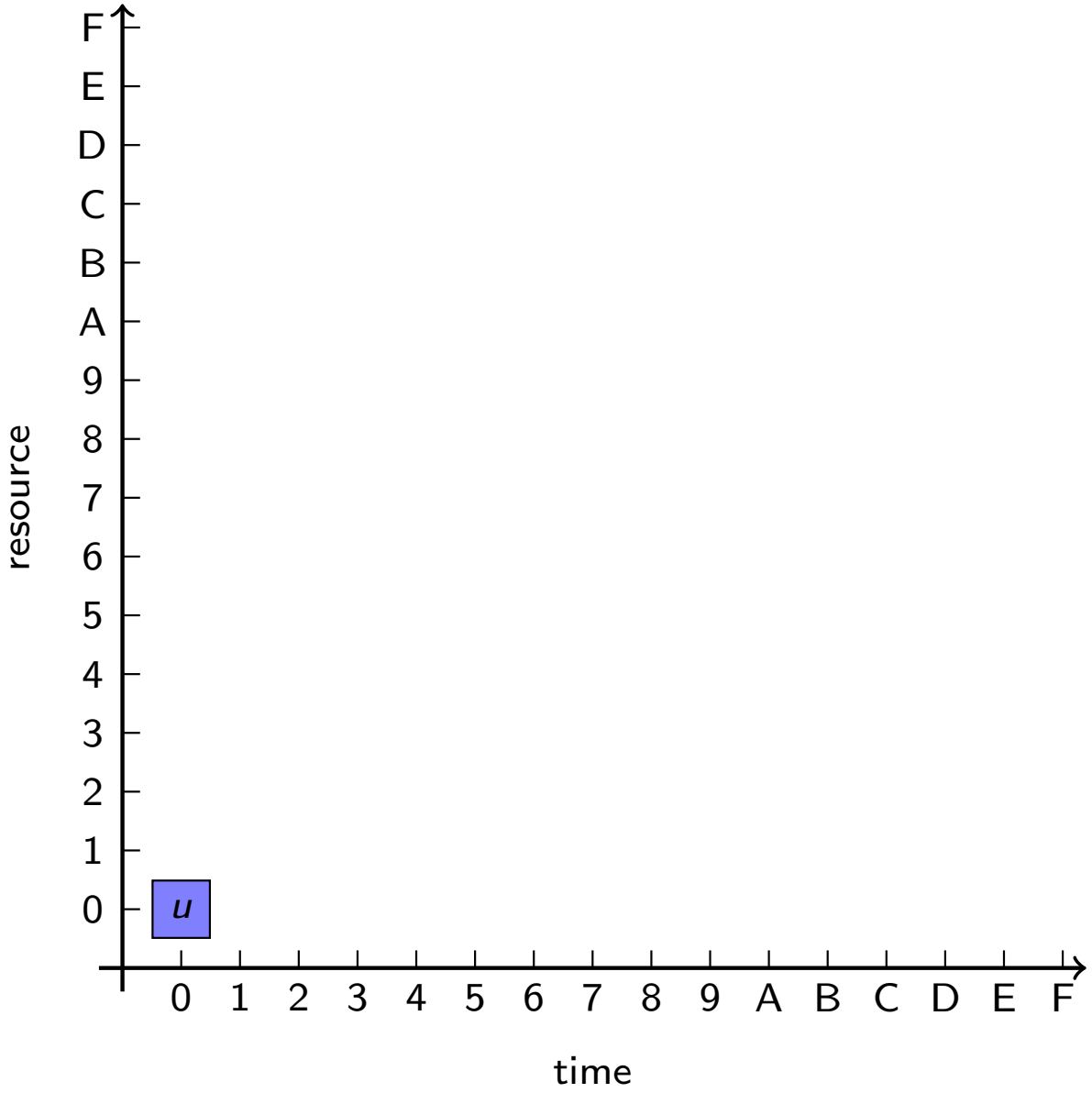
Results

SPRF

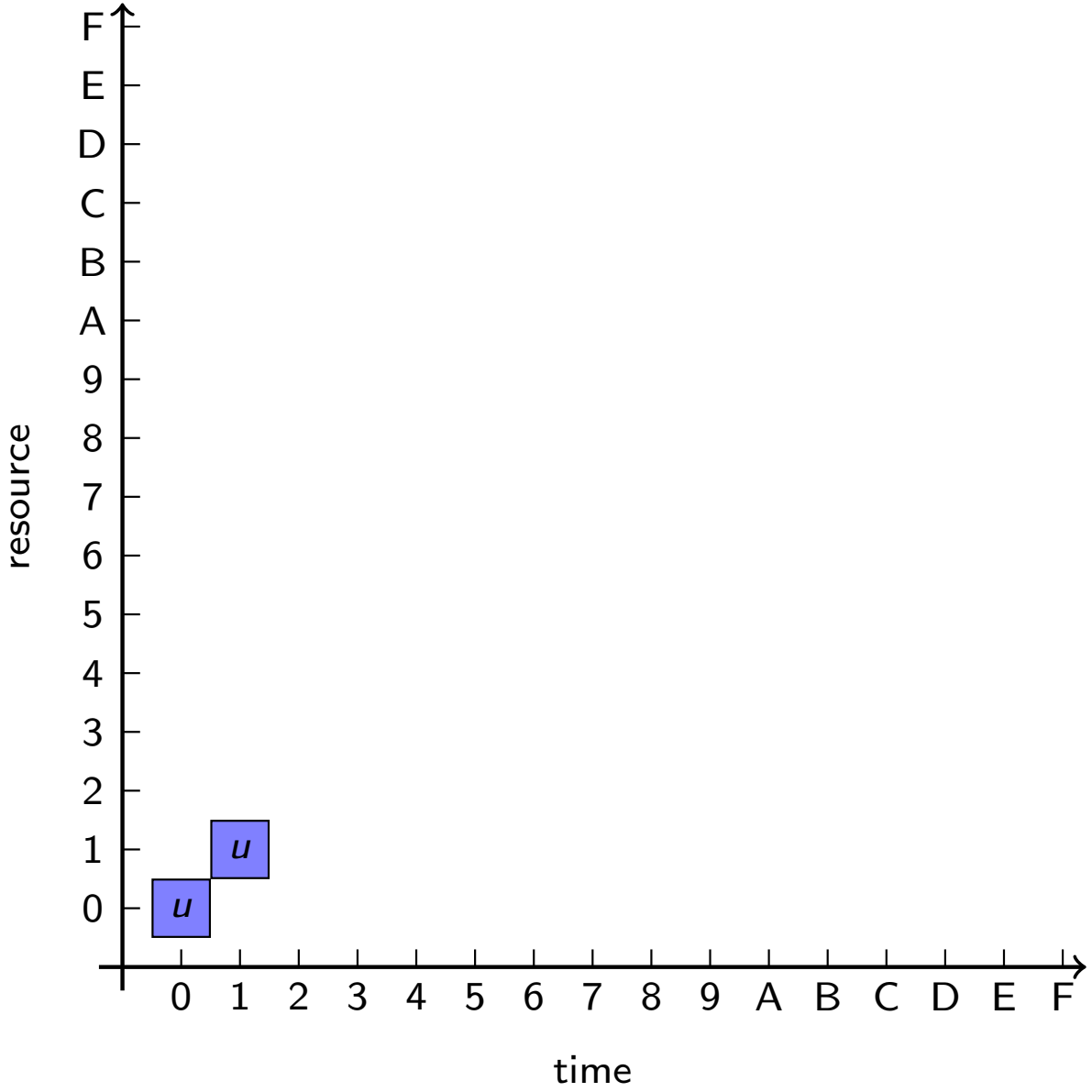
Shuffling



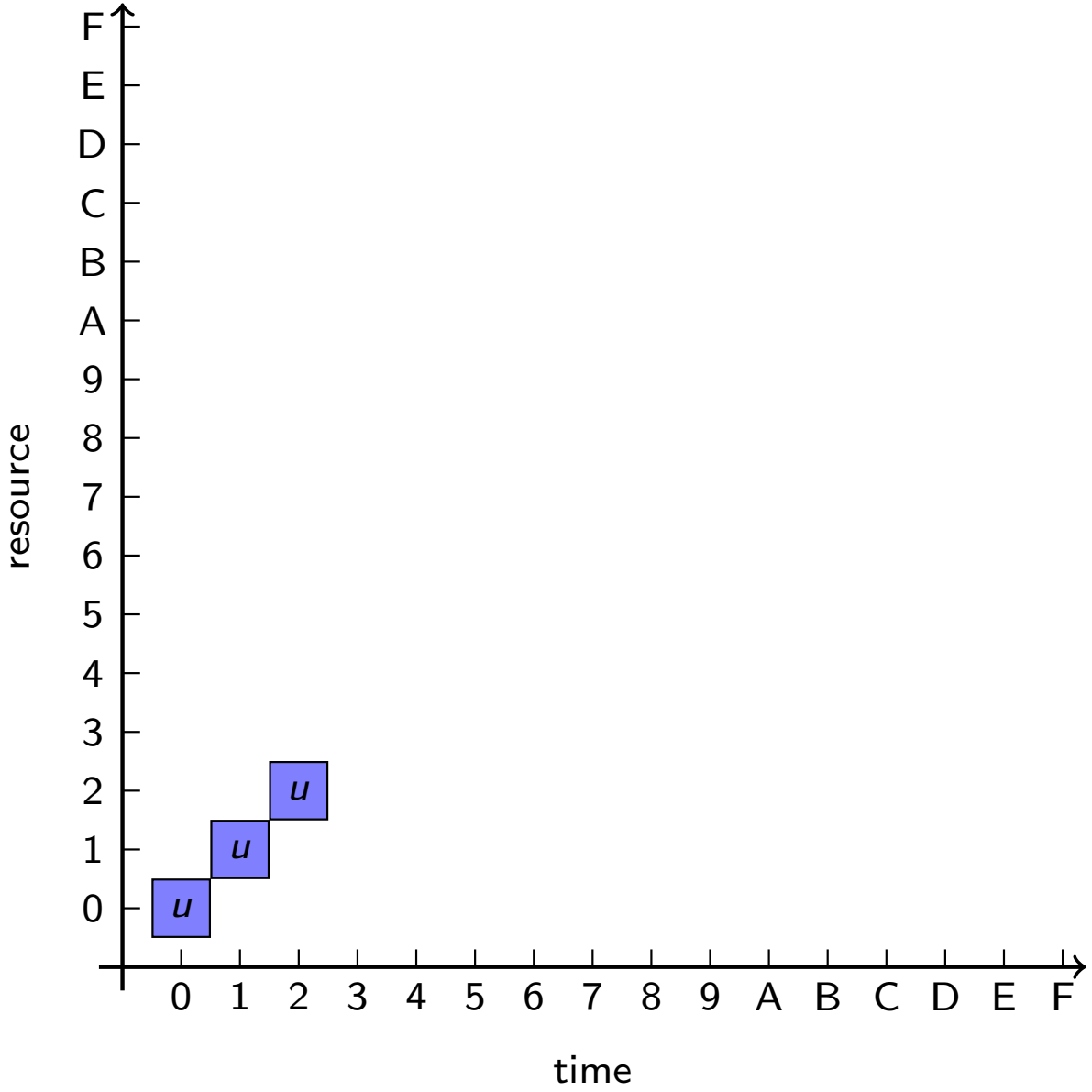
Shuffling



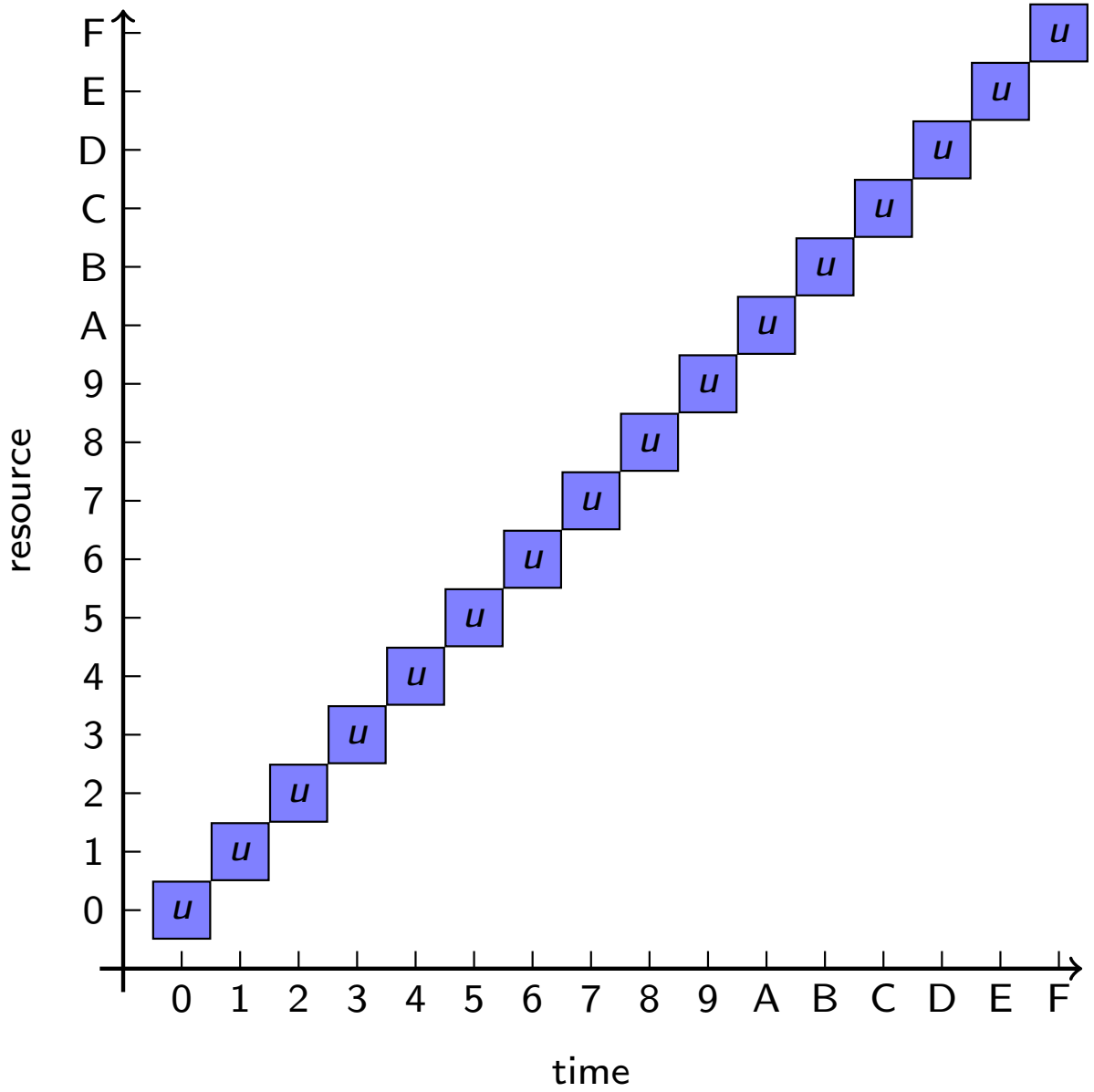
Shuffling



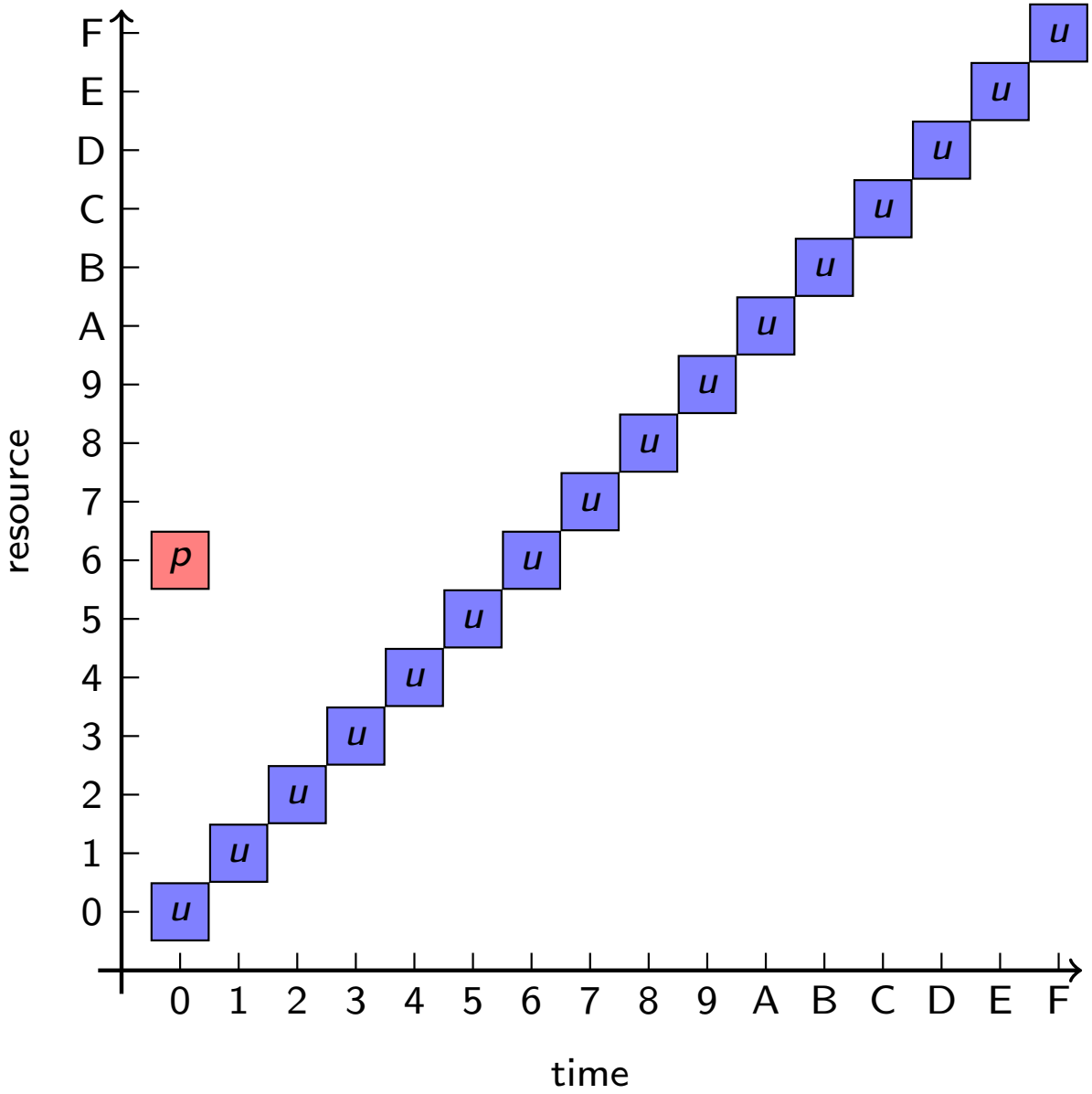
Shuffling



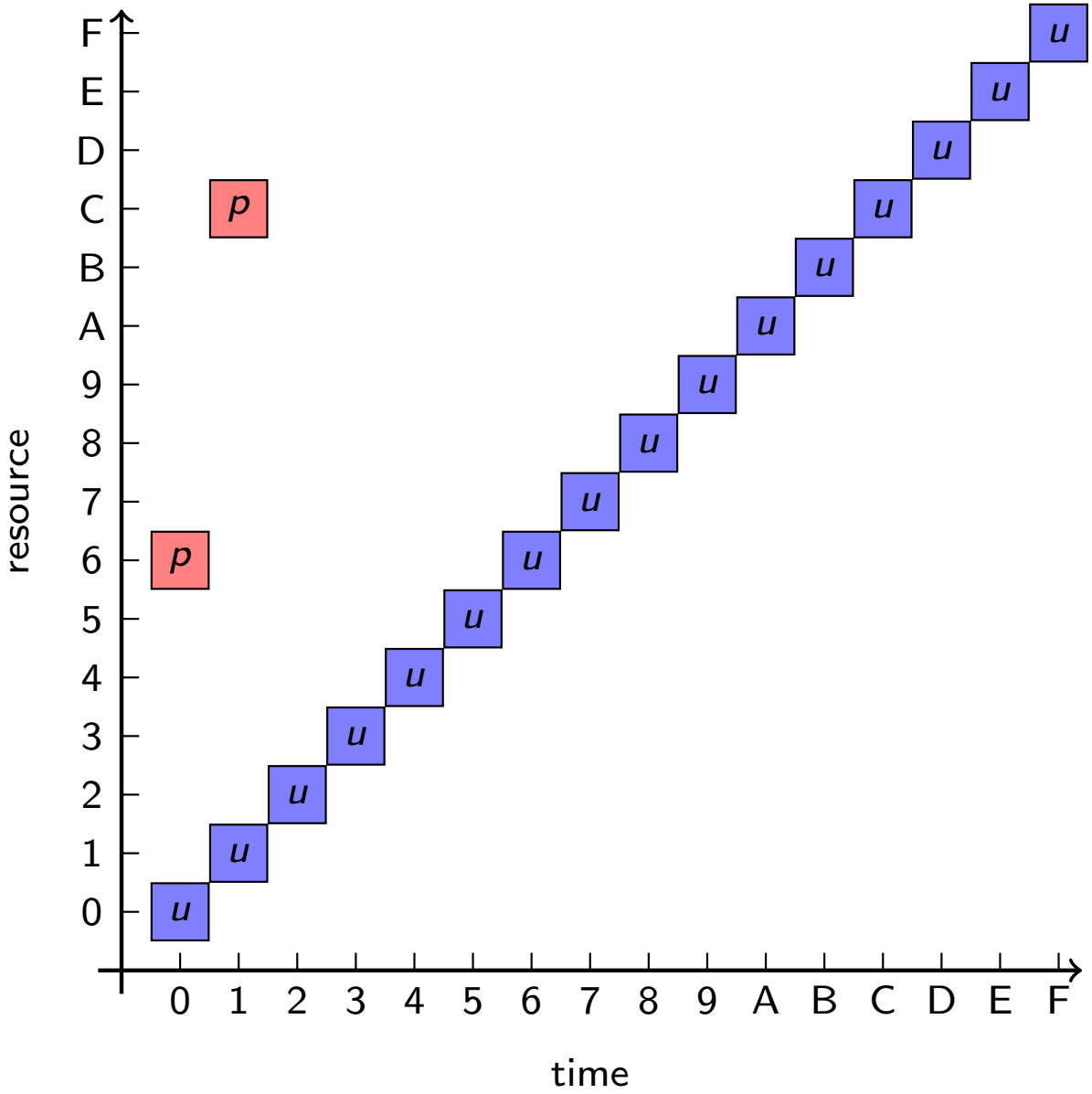
Shuffling



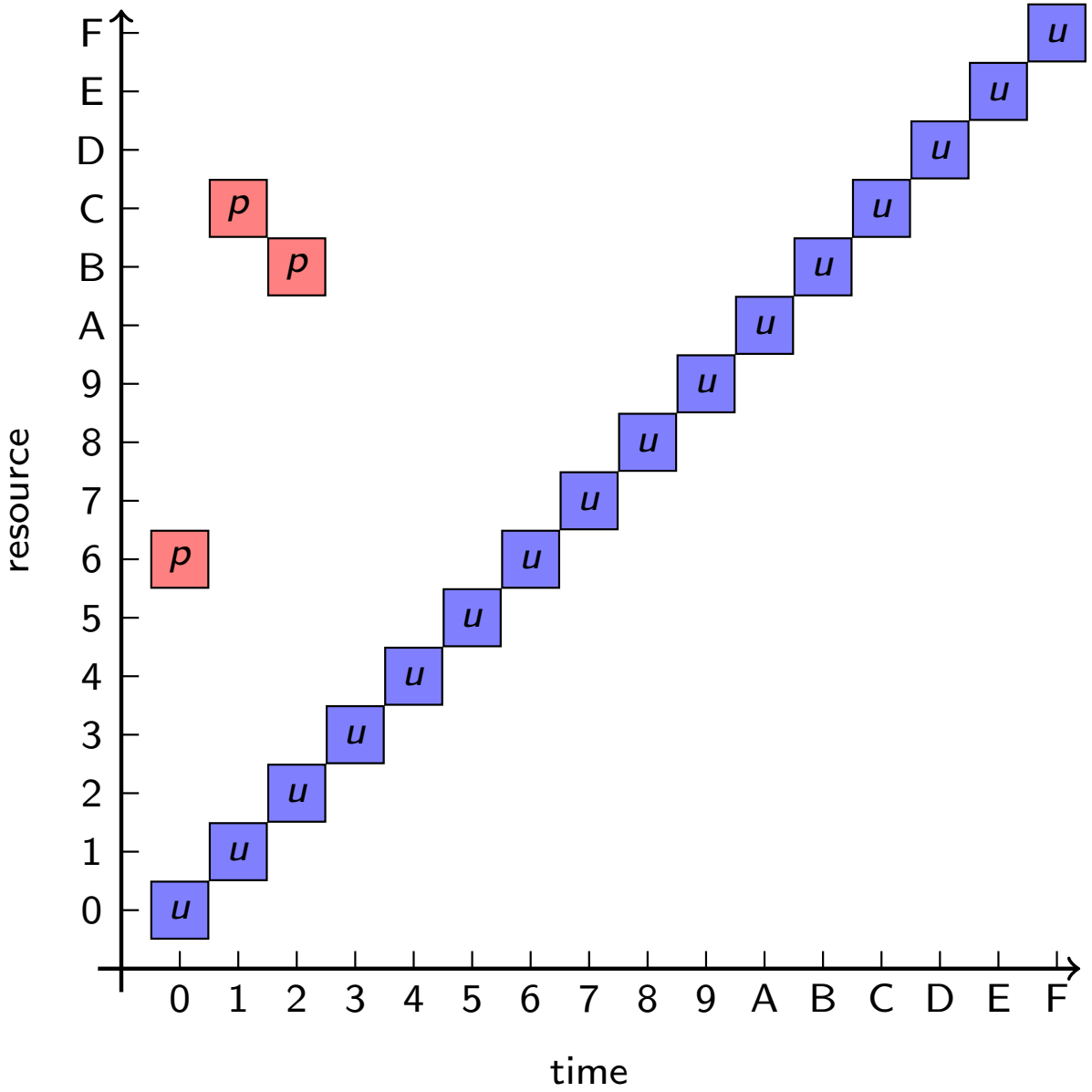
Shuffling



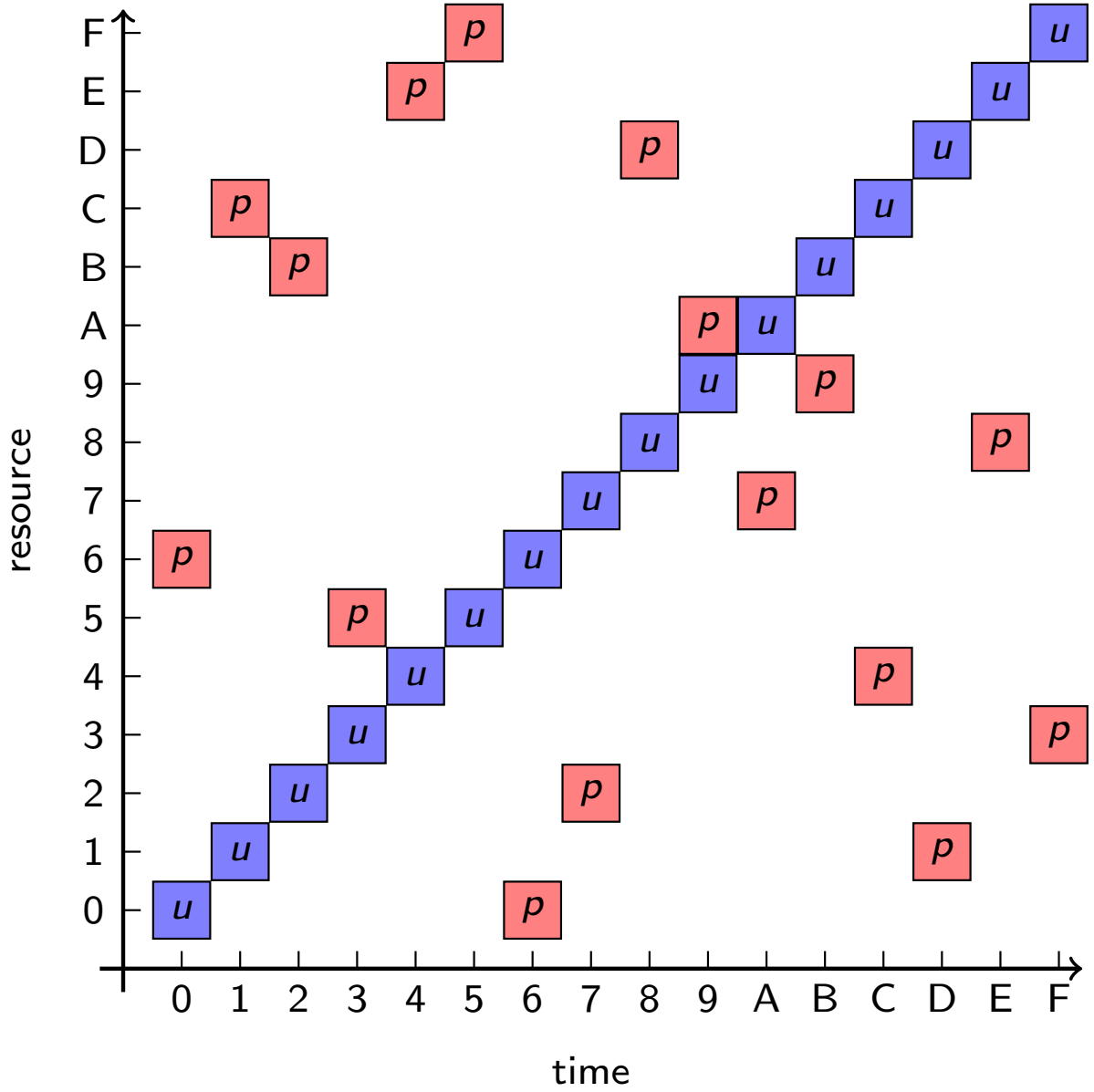
Shuffling



Shuffling



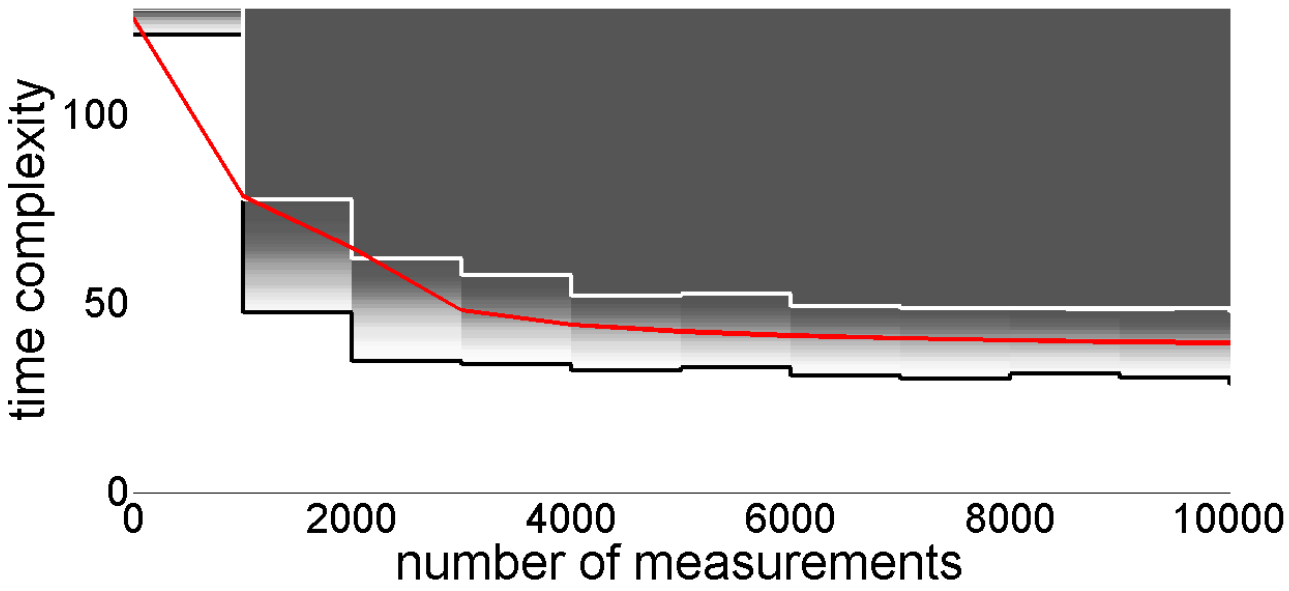
Shuffling



PRF shuffling case

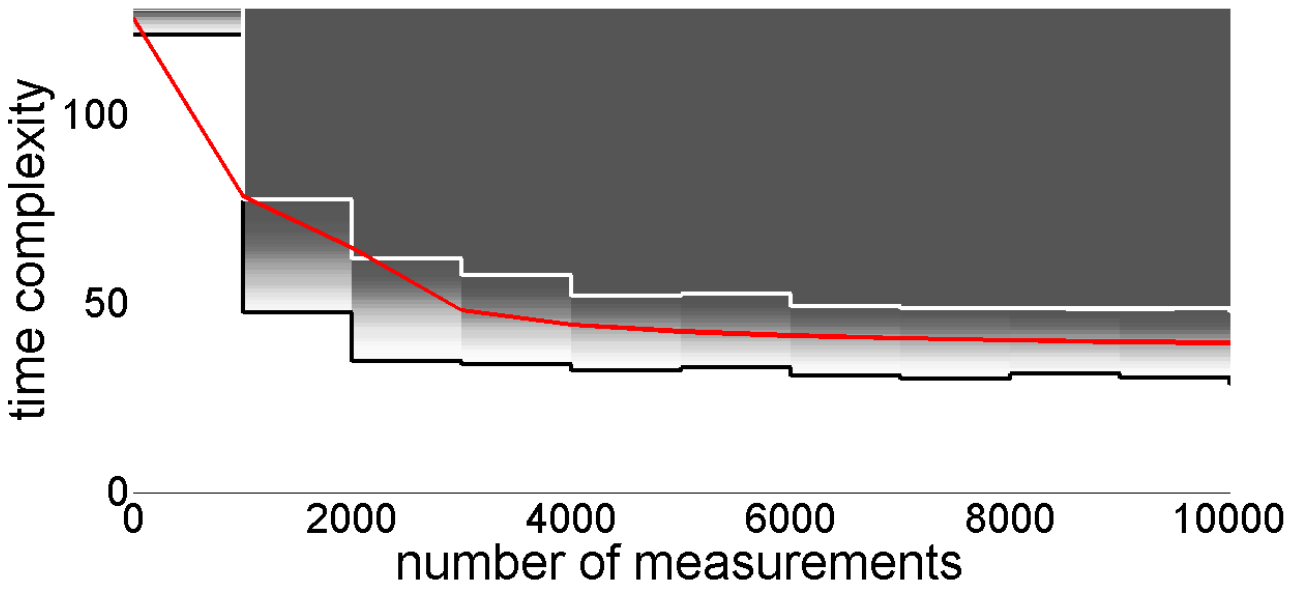
- The shuffling emulates the hardware.
- Some leakage on the permutation used could be additionally obtained by the adversary.

SPRF + tweak



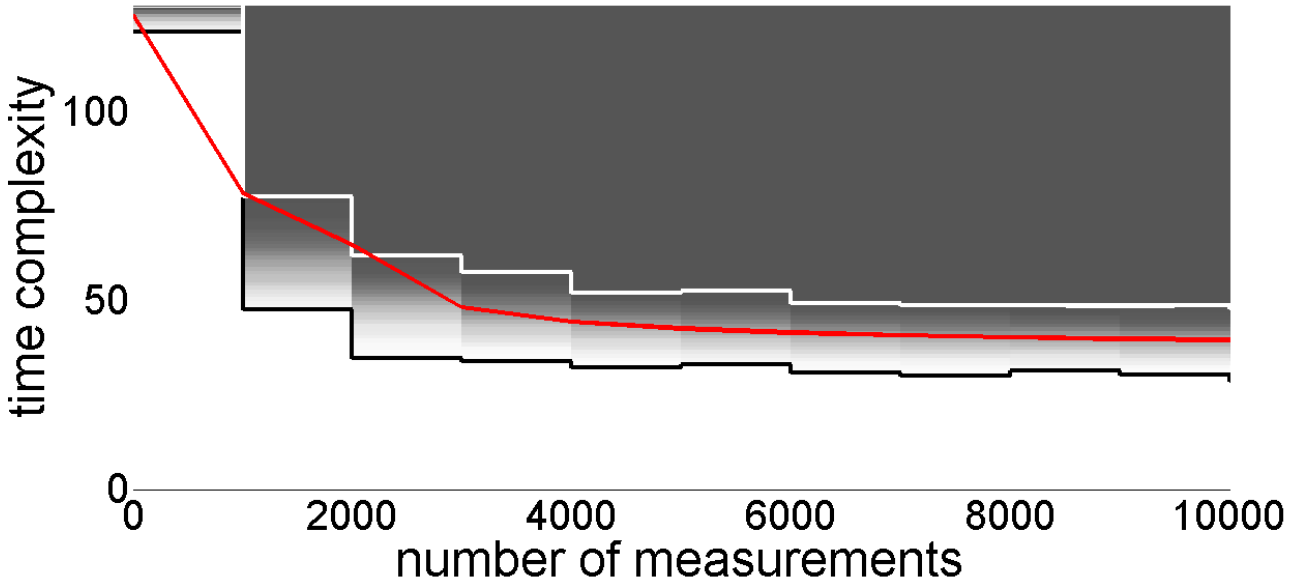
The data complexity is limited to 256. But averaging is not the best option to attack shuffling, hence we consider number of measurements.

SPRF + tweak



The leakage on the permutation give some additional information to reduce the enumeration part.

SPRF + tweak



A bound could be reached :-)

Conclusion

- For PRG bounds are easy to reach.
- For PRF no bound is reachable.
  - Masking increase the number of measurements.
  - Tweak in hardware allows to reach bound.
  - Shuffling + tweak allow to reach bound in software.

References I

- Marcel Medwed, François-Xavier Standaert, and Antoine Joux, *Towards super-exponential side-channel security with efficient leakage-resilient prfs*, CHES (Emmanuel Prouff and Patrick Schaumont, eds.), Lecture Notes in Computer Science, vol. 7428, Springer, 2012, pp. 193–212.
- Nicolas Veyrat-Charvillon, Benoît Gérard, and François-Xavier Standaert, *Security evaluations beyond computing power*, EUROCRYPT (Thomas Johansson and Phong Q. Nguyen, eds.), Lecture Notes in Computer Science, vol. 7881, Springer, 2013, pp. 126–141.