riscure

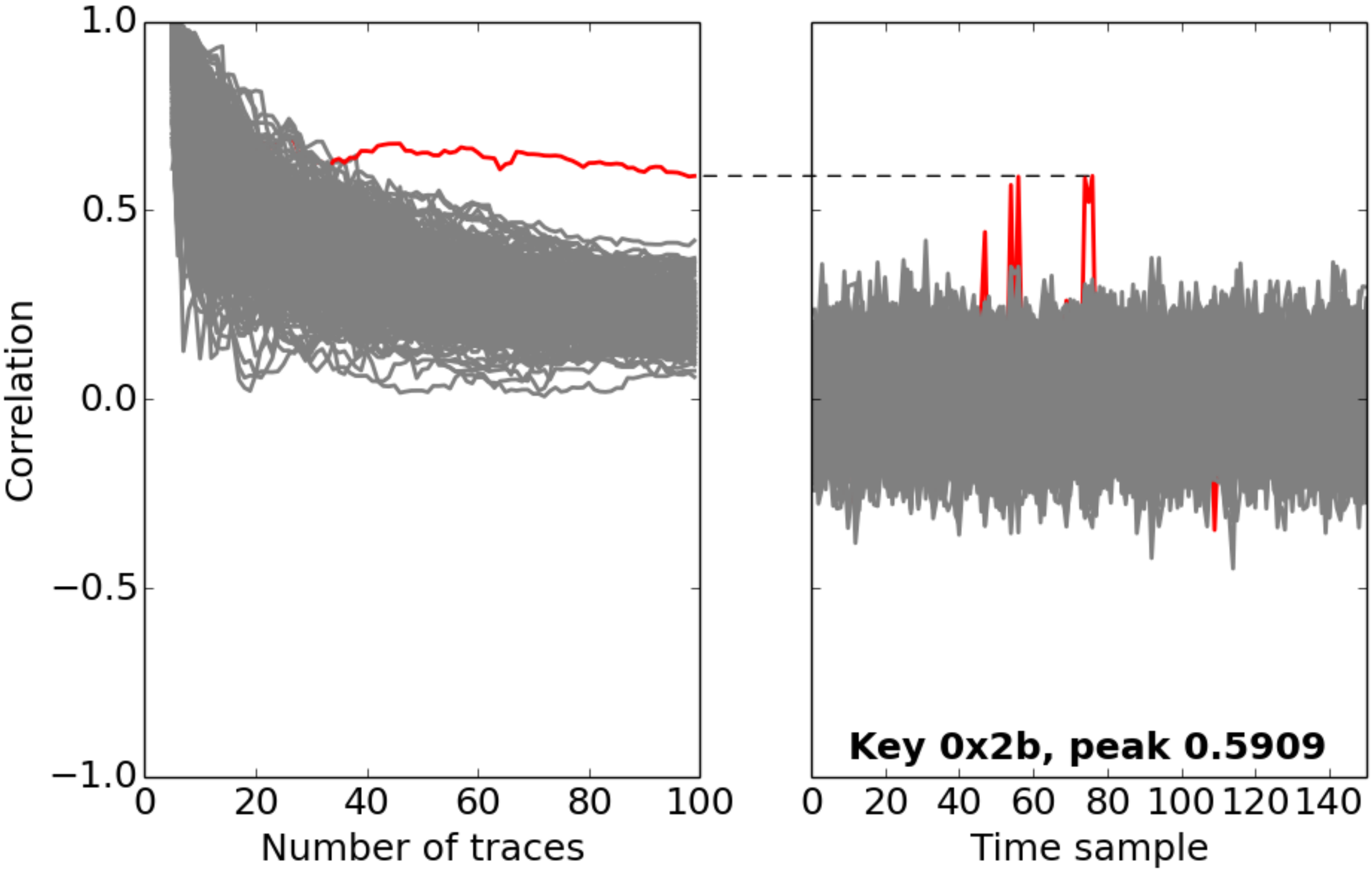
Efficient practical key recovery for side channel attacks

Ilya Kizhvatov and Marc Witteman (Riscure)
Andrey Bogdanov and Kamran Manzoor (DTU)

Uni.Lu 9 December 2014

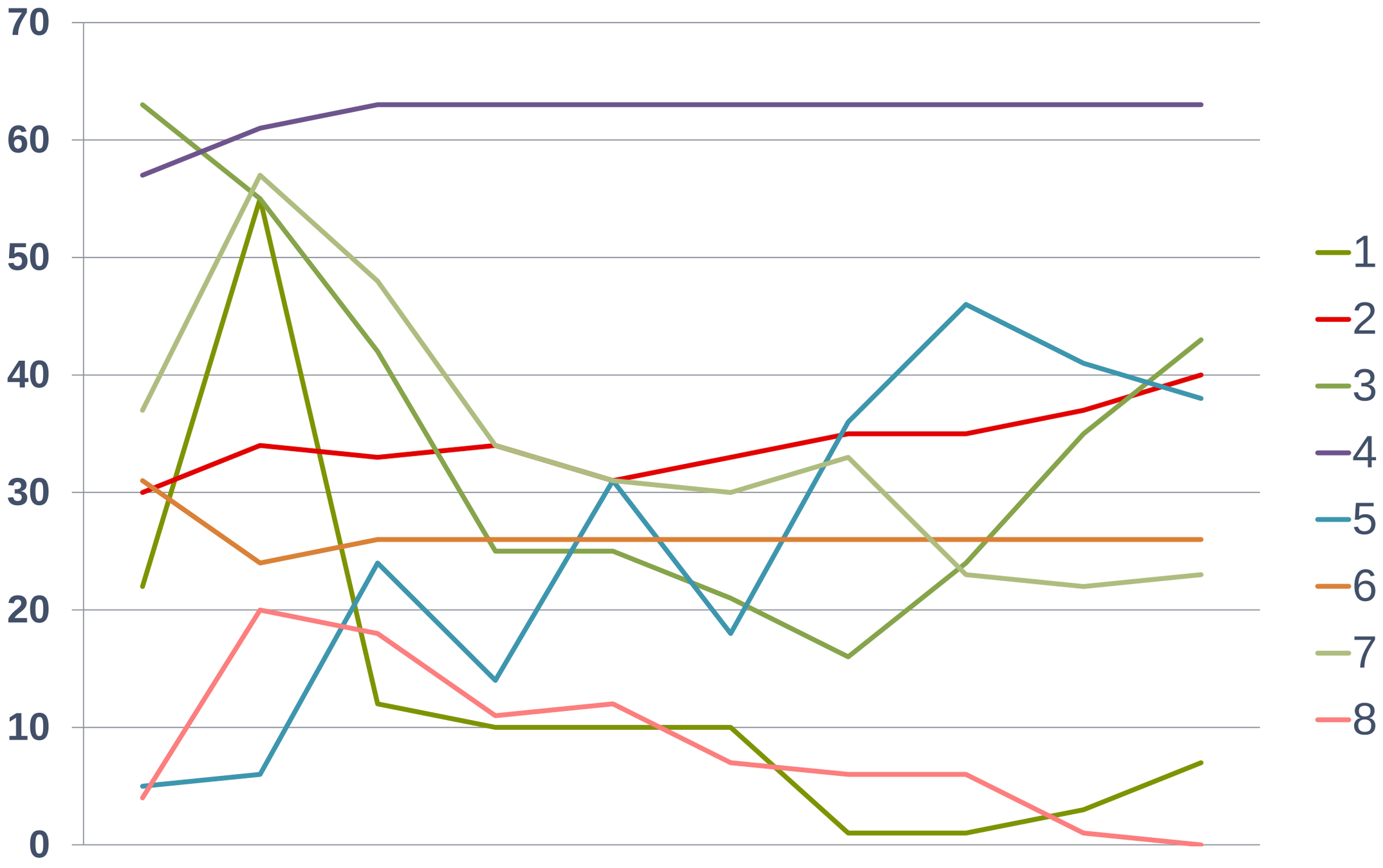
DPA recap

Distinguishing key chunks



DPA recap

Evolution of the rank of correct key chunk candidates



Problem statement

Conquer phase:

- The sub-keys are combined to generate the full-key.
- *Ideally* full-key = the most probable sub-key candidate from each list

k_0	0.0065*	k_0	0.0071	…….	k_0	0.0070
k_1	0.0063	k_1	0.0068		k_1	0.0067
⋮	⋮	⋮	⋮		⋮	⋮
k_n	0.0010	k_n	0.0011		k_n	0.0012

*Probability values for illustration

Problem statement

- How to choose the full key?

k_0	0.0065*
k_1	0.0063
⋮	⋮
k_n	0.0010

k_0	0.0071
k_1	0.0068
⋮	⋮
k_n	0.0011

… … …

k_0	0.0070
k_1	0.0067
⋮	⋮
k_n	0.0012

*Probability values for illustration

Problem statement

- How to choose the full key?

k_0	0.0065*
k_1	0.0063
⋮	⋮
k_n	0.0010

k_0	0.0071
k_1	0.0068
⋮	⋮
k_n	0.0011

… … …

k_0	0.0070
k_1	0.0067
⋮	⋮
k_n	0.0012

*Probability values for illustration

Problem statement

- How to choose the full key?

k_0	0.0065*
k_1	0.0063
⋮	⋮
k_n	0.0010

k_0	0.0071
k_1	0.0068
⋮	⋮
k_n	0.0011

… … …

k_0	0.0070
k_1	0.0067
⋮	⋮
k_n	0.0012

*Probability values for illustration

Problem statement

- How to choose the full key?

k_0	0.0065*
k_1	0.0063
⋮	⋮
k_n	0.0010

k_0	0.0071
k_1	0.0068
⋮	⋮
k_n	0.0011

… … …

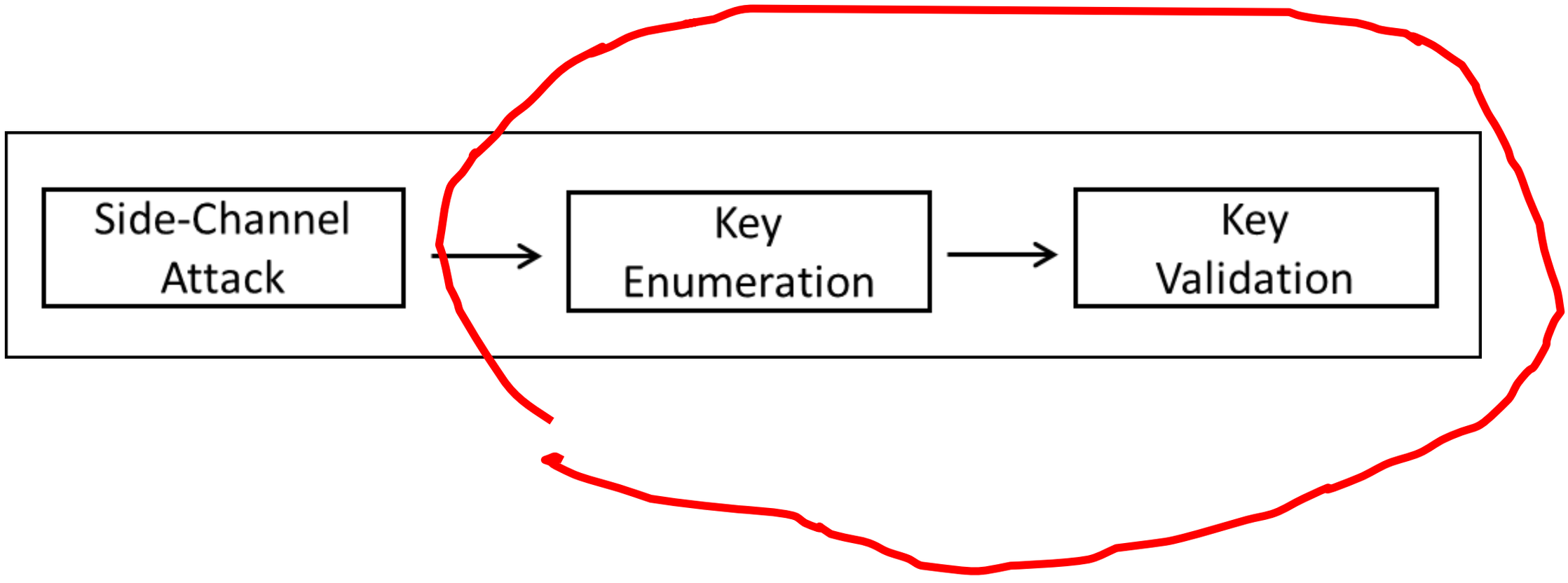
k_0	0.0070
k_1	0.0067
⋮	⋮
k_n	0.0012

- Solution:
  - Key enumeration
  - Concurrent validation of full keys

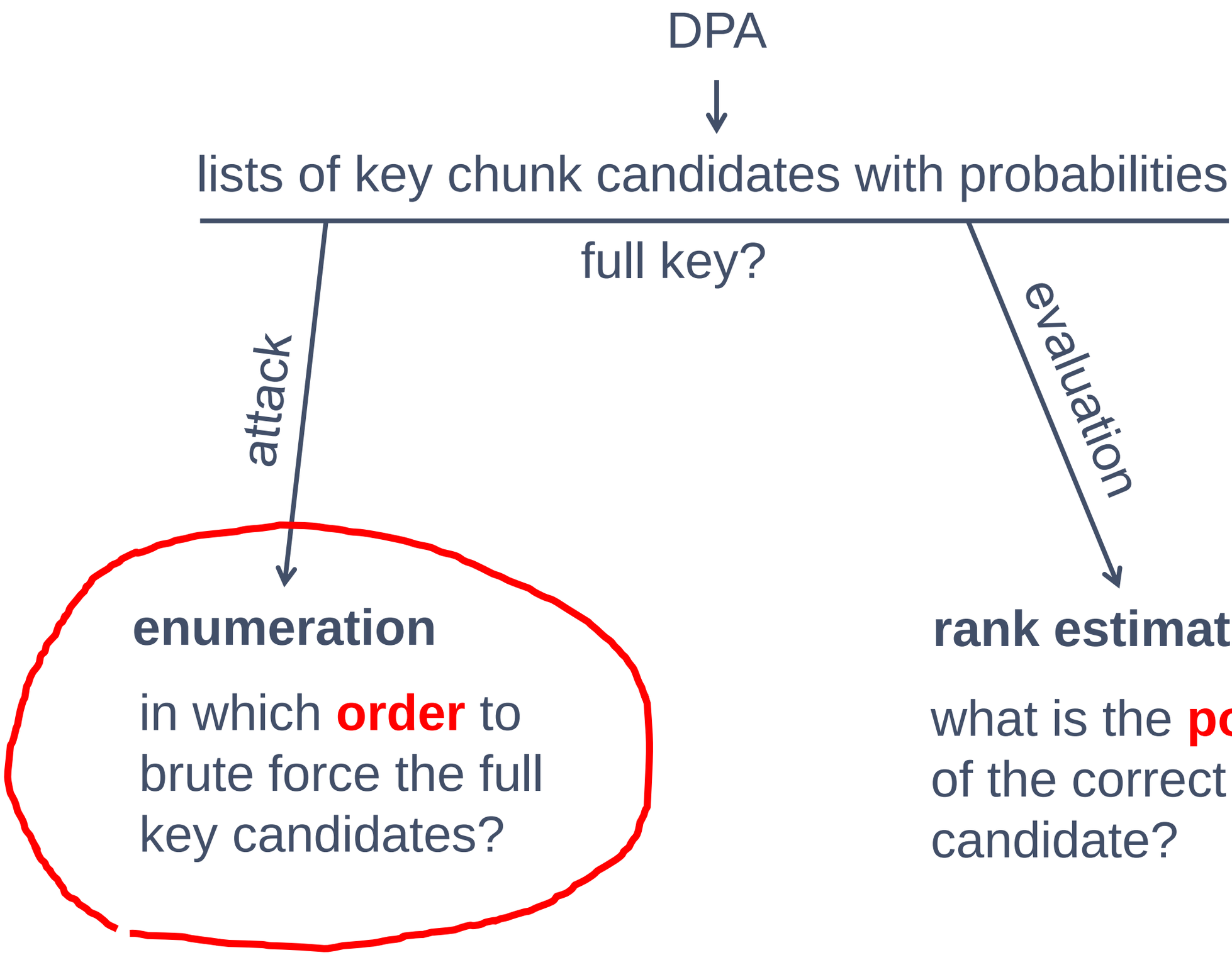
*Probability values for illustration

Problem statement

Find the key after DPA as fast as possible. On a desktop.



Context

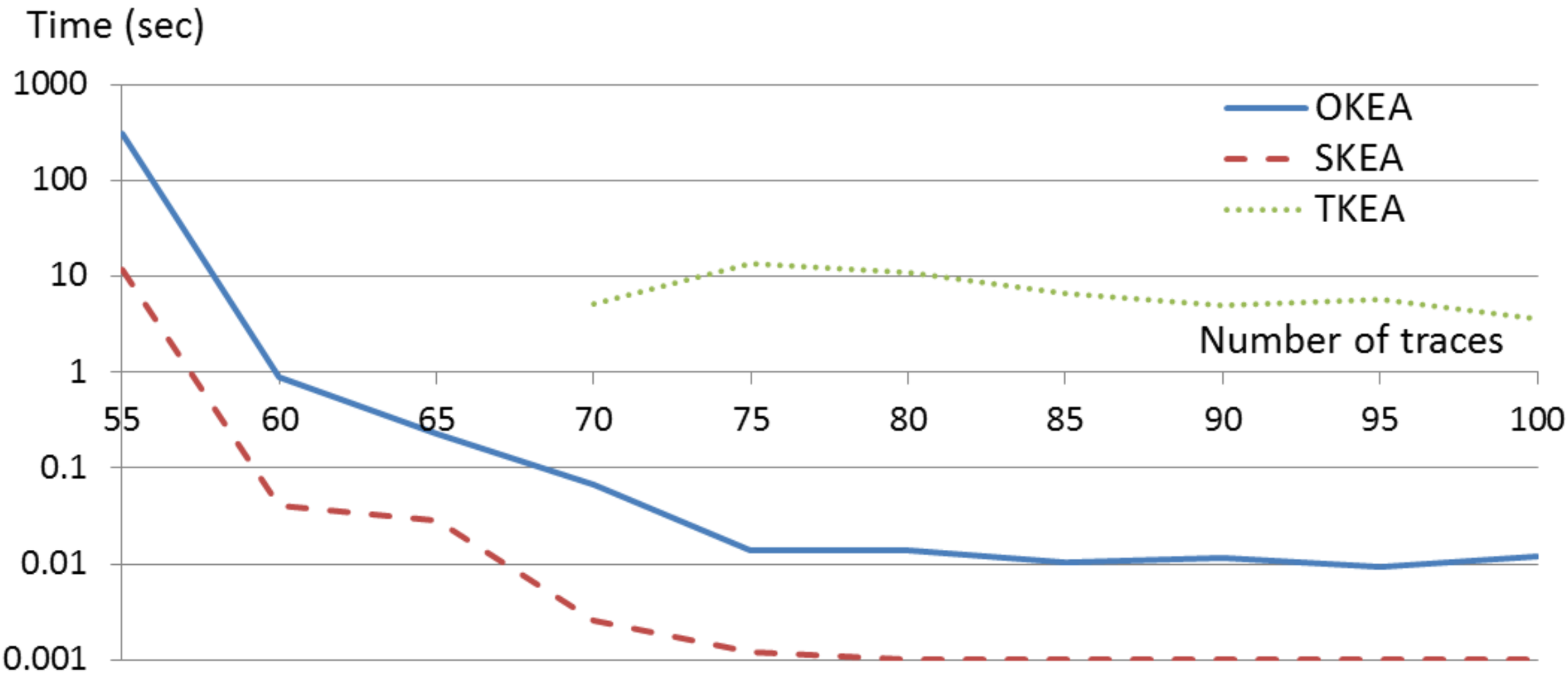

DPA
lists of key chunk candidates with probabilities
full key?
attack
evaluation
enumeration
in which order to brute force the full key candidates?
rank estimation
what is the position of the correct full key candidate?

In this work so far…

- Benchmarked 3 enumeration algorithms:
  - Trivial Key Enumeration Algorithm (TKEA)
  - Optimal Key Enumeration Algorithm (OKEA) [SAC 2012]
  - **Score based Key Enumeration Algorithm (SKEA) [Marc Witteman] – new solution**

- Combined with key validation to get the full solution

Time to find the key

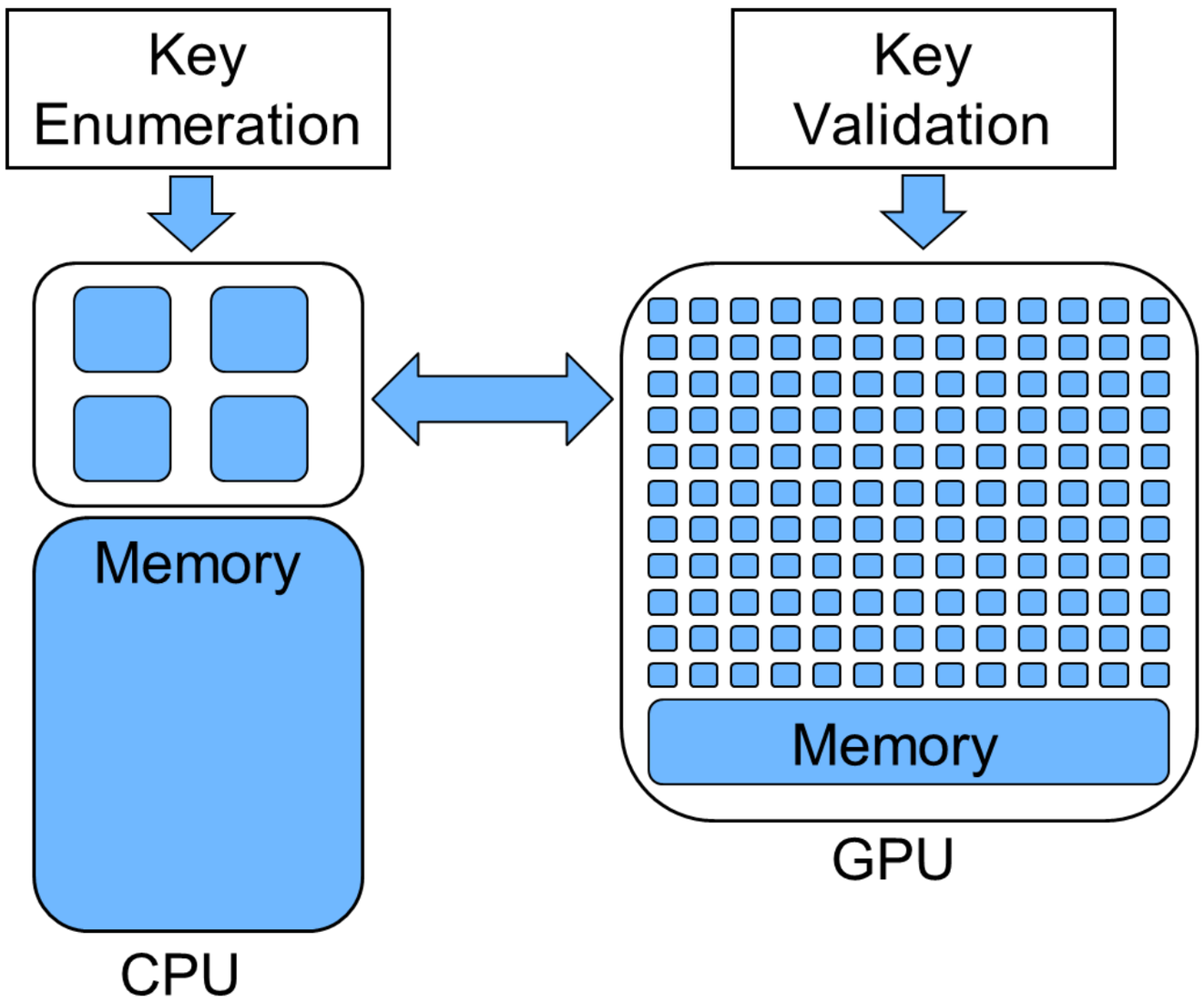


Key Validation

- We propose to deploy SKEA. Maximum throughput of SKEA $\cong 2^{23}\ keys/\text{sec}$
- Throughput of key validation should at least be $\cong 2^{23}\ keys/\text{sec}$
- Utilize the immense parallel processing power of a GPU
- Implemented AES on an NVIDIA GPU using CUDA platform
- Achieved key validation rate of more than $\cong 2^{23}\ \ keys/\text{sec}$

Proposed Solution

- Simultaneous execution of key enumeration (SKEA) on a CPU and key validation on a GPU.



riscure

Challenge your security

Contact: Ilya Kizhvatov
ilya@riscure.com

Riscure B.V.
Frontier Building, Delftechpark 49
2628 XJ Delft
The Netherlands
Phone: +31 15 251 40 90

www.riscure.com

Riscure North America
71 Stevenson Street, Suite 400
San Francisco, CA 94105
USA
Phone: +1 650 646 99 79

inforequest@riscure.com