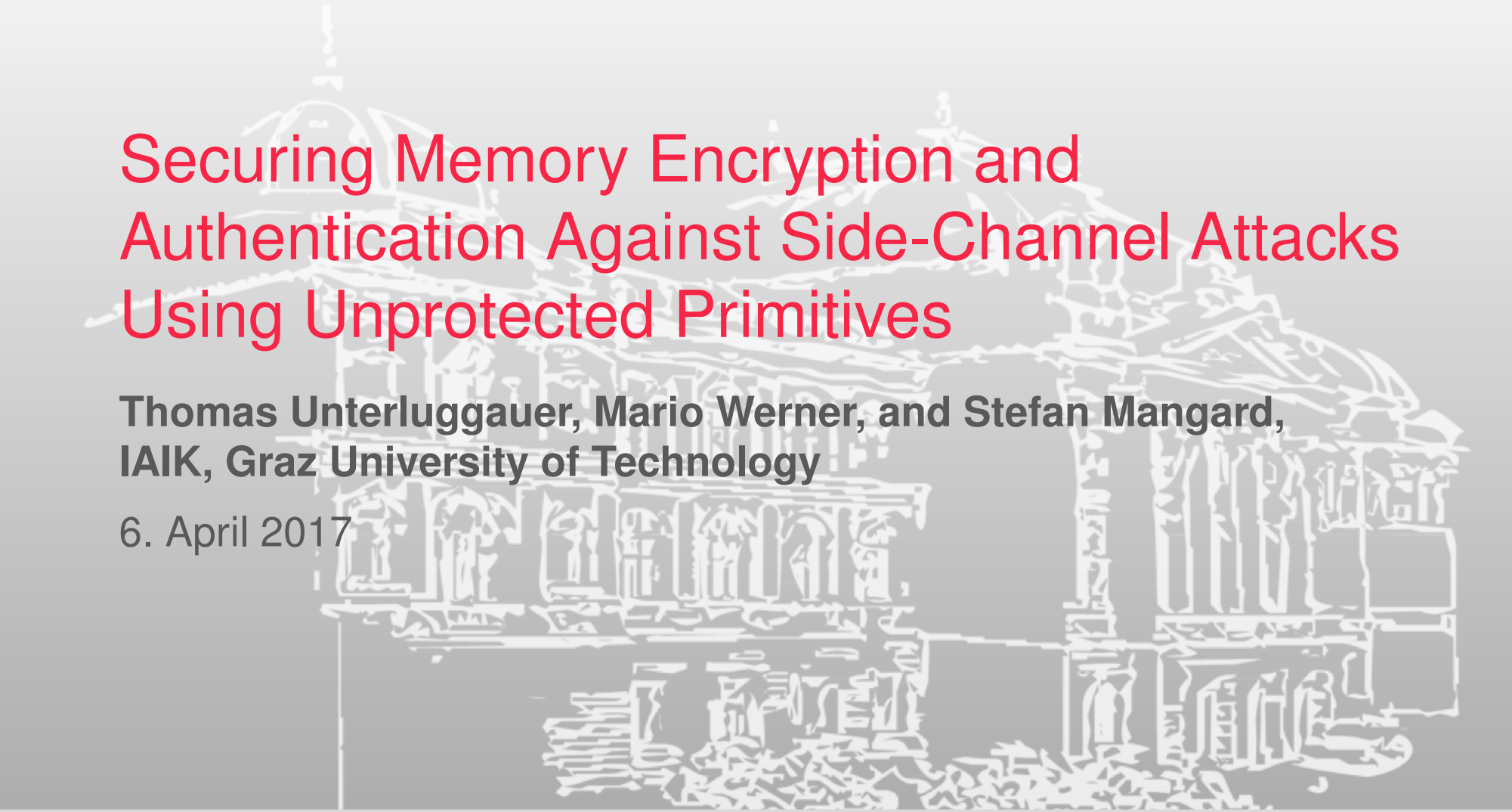


Securing Memory Encryption and Authentication Against Side-Channel Attacks Using Unprotected Primitives

Thomas Unterluggauer, Mario Werner, and Stefan Mangard,
IAIK, Graz University of Technology

6. April 2017

Content

- Memory encryption to protect from physical access
- Differential power analysis (DPA) practically feasible

Content

- Memory encryption to protect from physical access
- Differential power analysis (DPA) practically feasible
- MEAS: Memory Encryption and Authentication secure against DPA
  - Combines re-keying and authentication trees
    - Limit inputs processed under one key by two

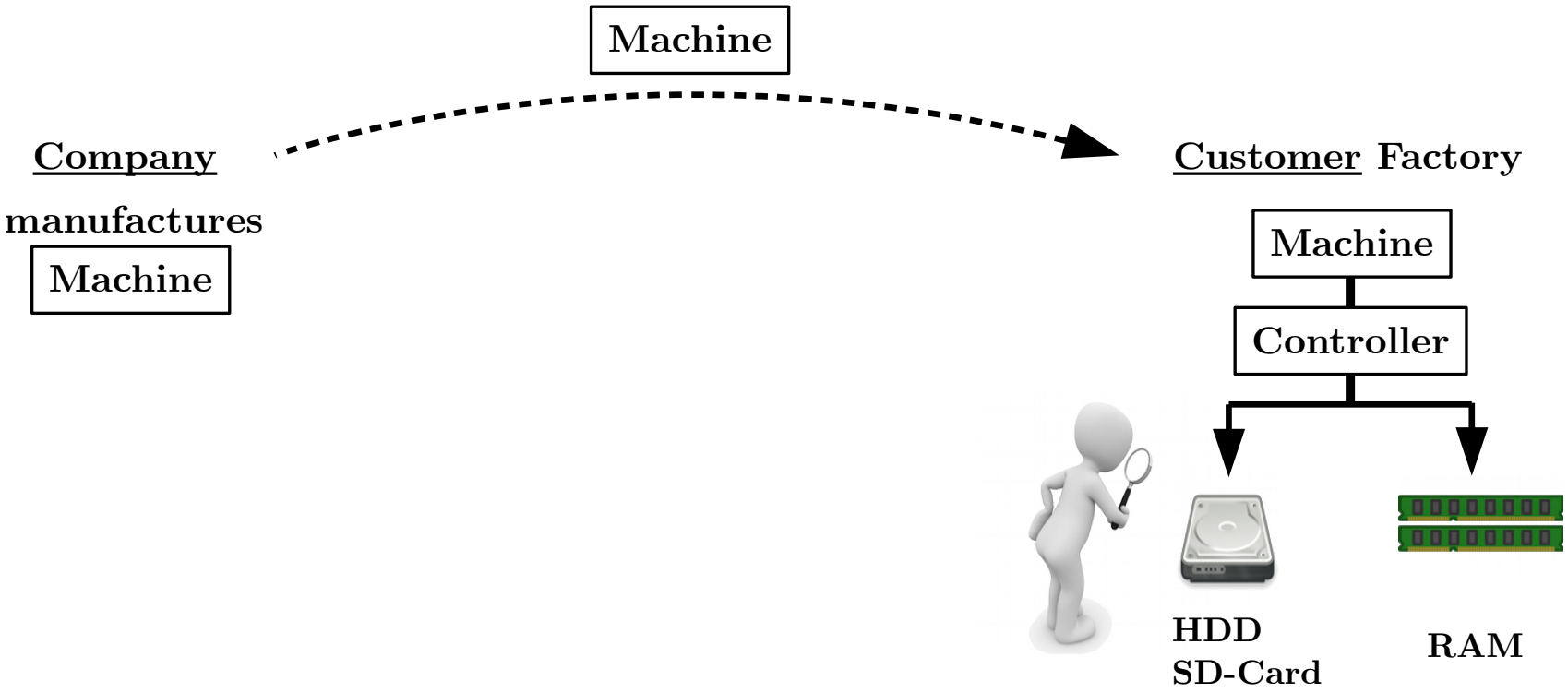
Content

- Memory encryption to protect from physical access
- Differential power analysis (DPA) practically feasible
- MEAS: Memory Encryption and Authentication secure against DPA
  - Combines re-keying and authentication trees
    - Limit inputs processed under one key by two
  - Higher-order protection: masking of plaintexts
  - No need for DPA-protected implementations

Content

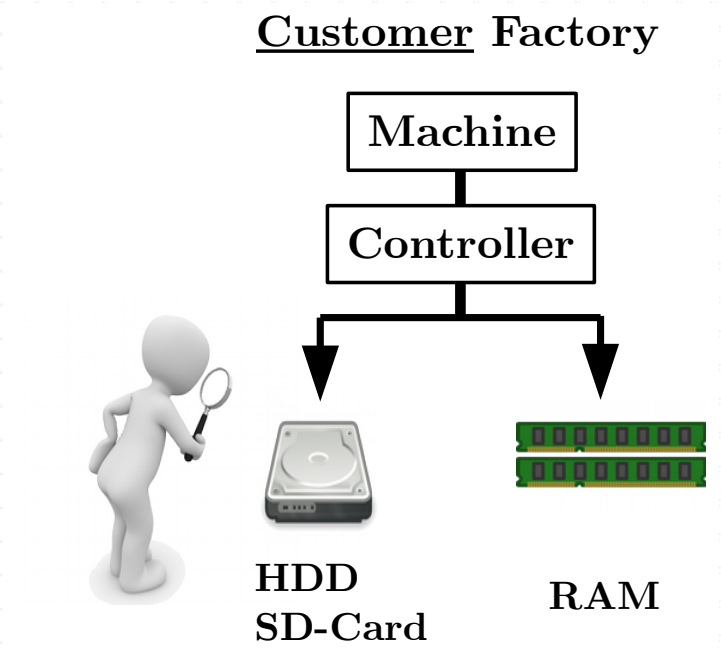
- Memory encryption to protect from physical access
- Differential power analysis (DPA) practically feasible
- MEAS: Memory Encryption and Authentication secure against DPA
  - Combines re-keying and authentication trees
    - Limit inputs processed under one key by two
  - Higher-order protection: masking of plaintexts
  - No need for DPA-protected implementations
  - Suitable for both RAM and NVM
  - Memory overhead: 7.3% for 512-byte disk sectors

Motivation


Machine
Company
manufactures
Machine
Customer Factory
Machine
Controller
HDD
SD-Card
RAM

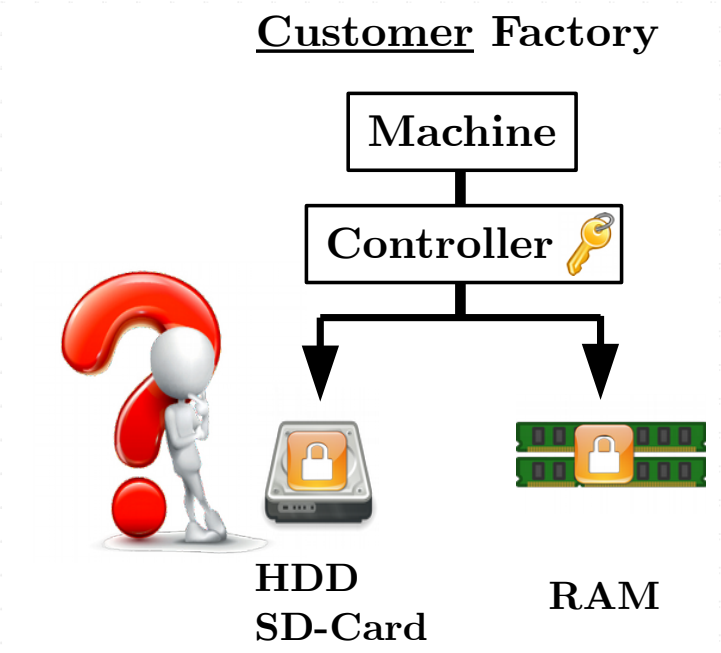
Motivation

- Memory contains high-value assets
- Customer interested in:
  - Intellectual property
    - Source code
    - Control parameters
  - Billing data
    - Pay per use



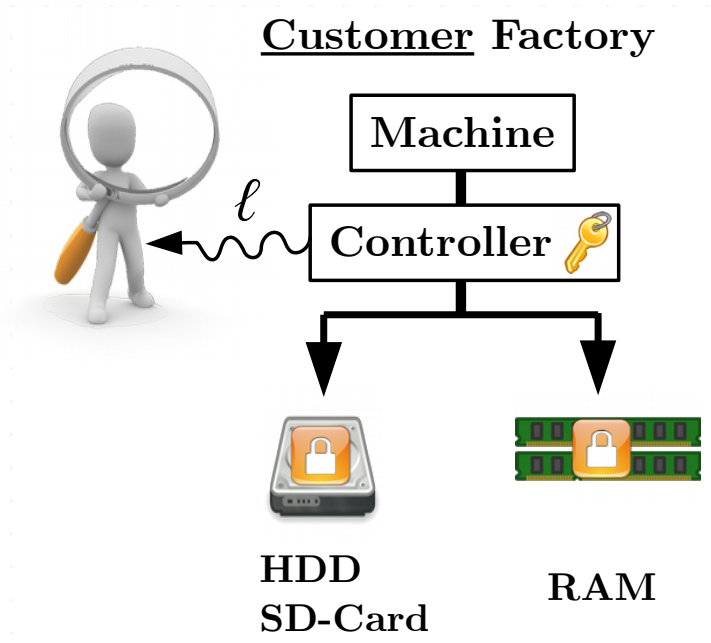
Motivation

- Countermeasure:
  - Encrypt memory
  - Store key securely



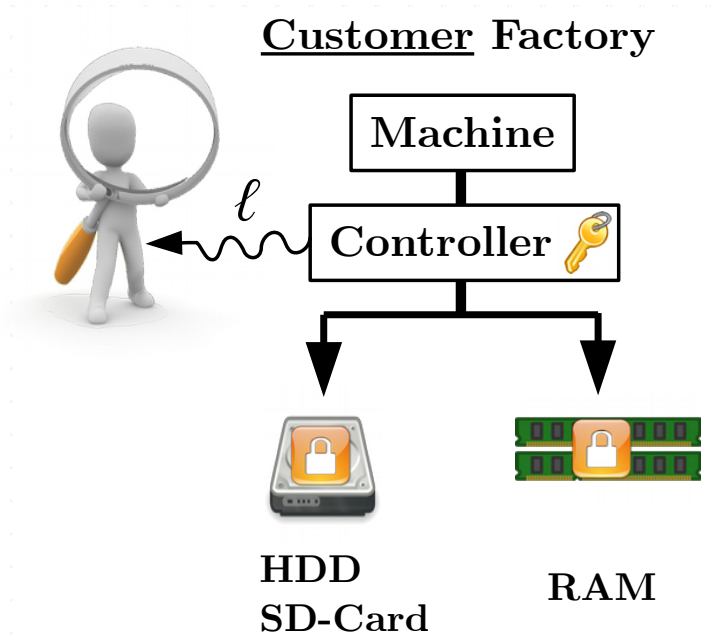
Motivation

- Countermeasure:
  - Encrypt memory
  - Store key securely
- Side-channel leakage:
  - Running device emits information on processed data
  - Power, EM, timing,...



Motivation

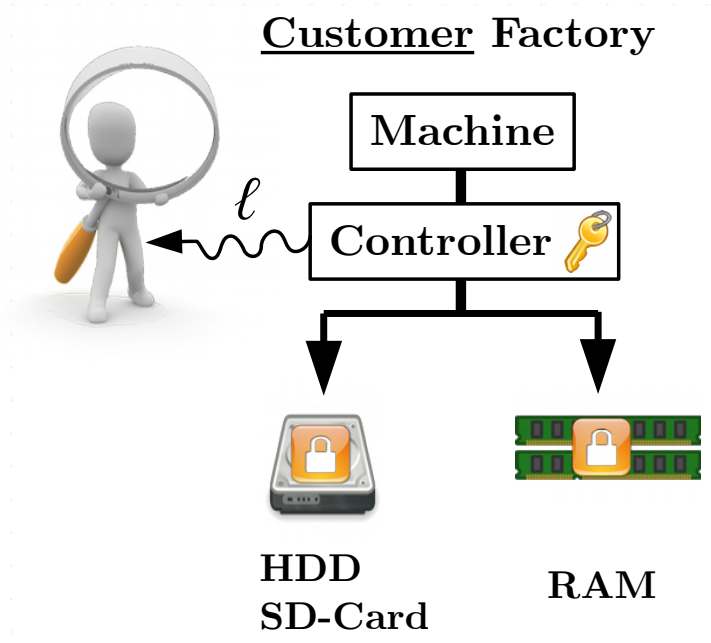
- Countermeasure:
  - Encrypt memory
  - Store key securely
- Side-channel leakage:
  - Running device emits information on processed data
  - Power, EM, timing,...



$\Rightarrow$ Side-channel attacks possible

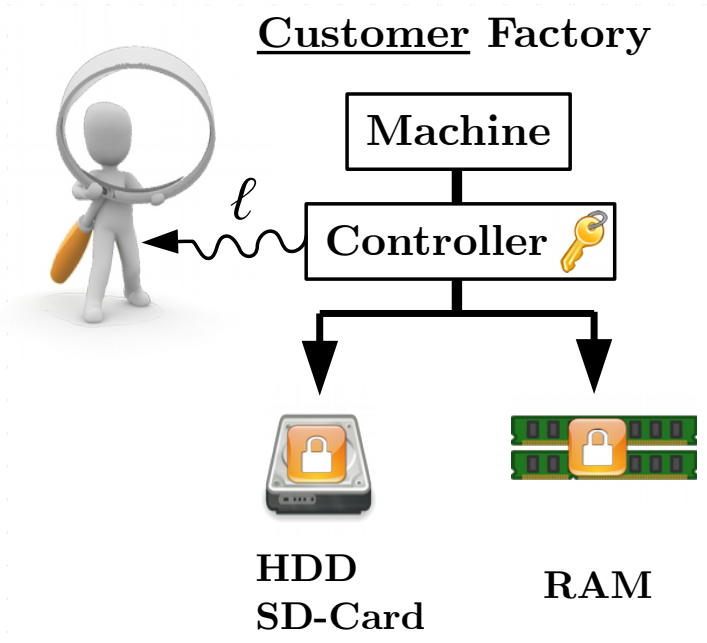
Attack Setting

- Computing device running in hostile environment
- Memory contains sensitive information



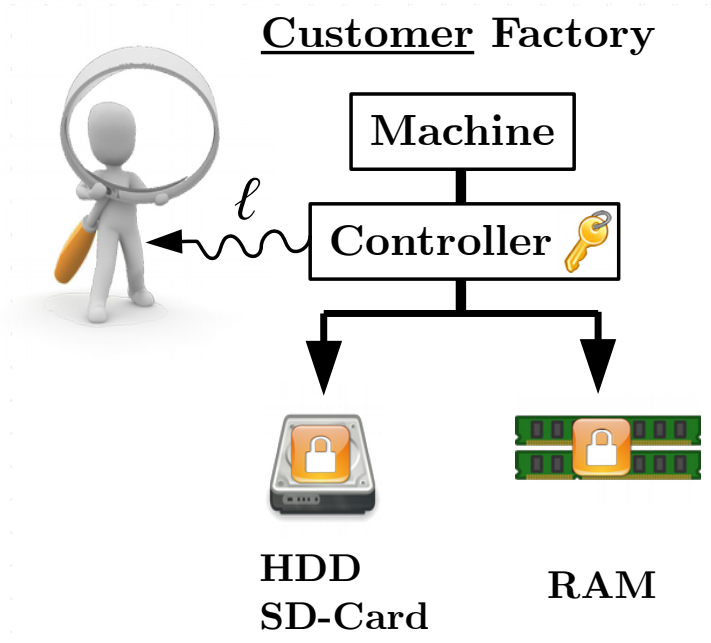
Attack Setting

- Computing device running in hostile environment
- Memory contains sensitive information
- Attacker can access device and (encrypted) memory
- Side-channel information available to the attacker



Attack Setting

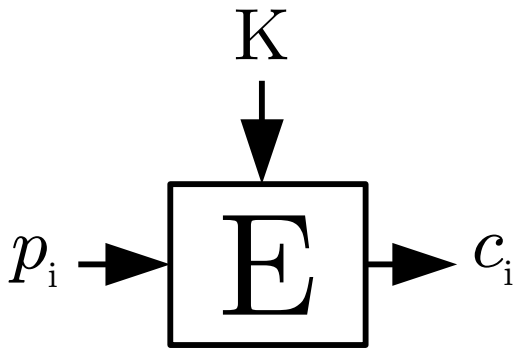
- Computing device running in hostile environment
- Memory contains sensitive information
- Attacker can access device and (encrypted) memory
- Side-channel information available to the attacker



$\Rightarrow$ Simple memory encryption is insufficient

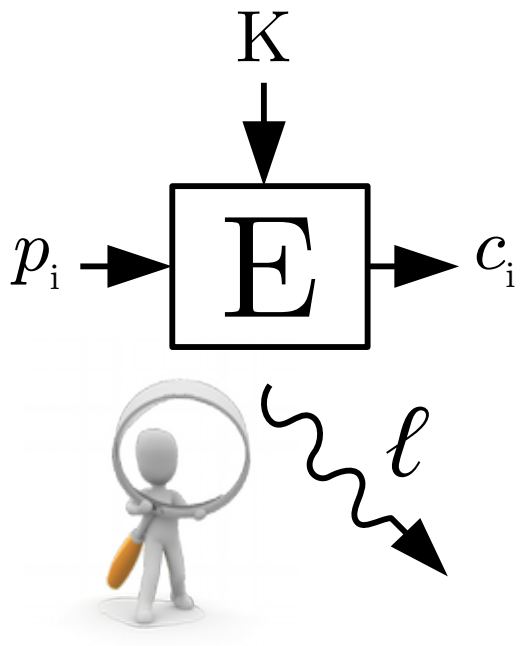
Differential Power Analysis

- Key K used for multiple p_i, c_i
  - n encryptions: $E_K(p_i)$



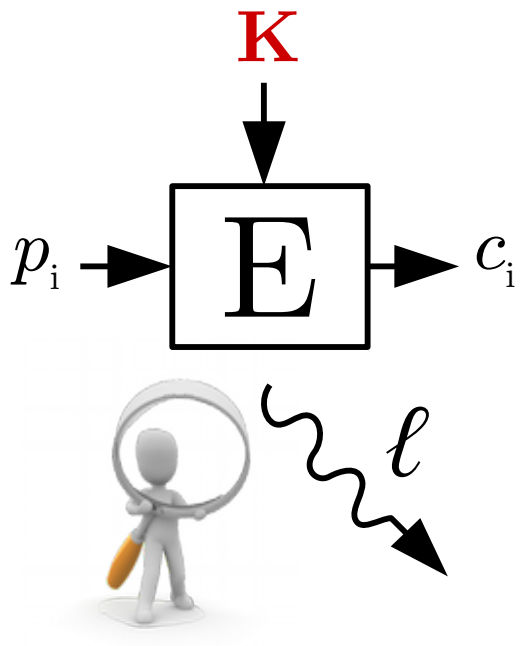
Differential Power Analysis

- Key K used for multiple p_i, c_i
  - n encryptions: $E_K(p_i)$
- Observe power consumption



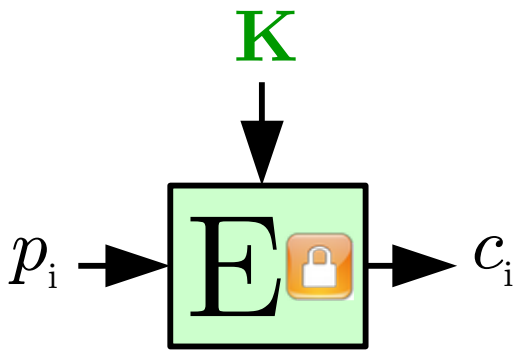
Differential Power Analysis

- Key K used for multiple p_i, c_i
  - n encryptions: $E_K(p_i)$
- Observe power consumption
- Power model for $E_K(p_i) \; \forall \; K$
  - Divide-and-conquer approach
- Statistical analysis reveals K
  - E.g. correlation



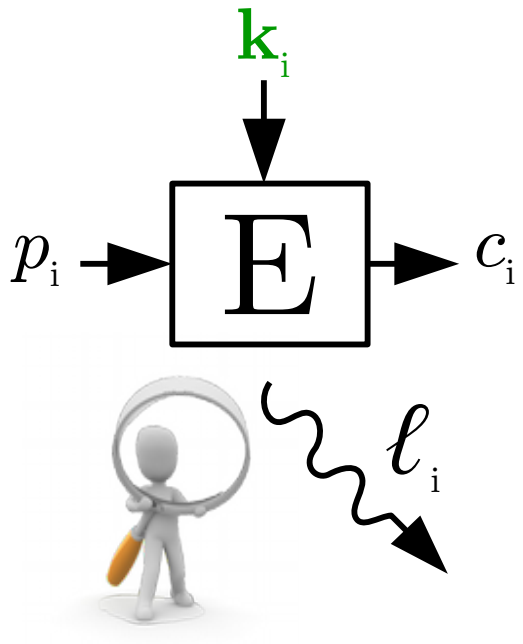
DPA Countermeasures

- Protect implementation (masking)



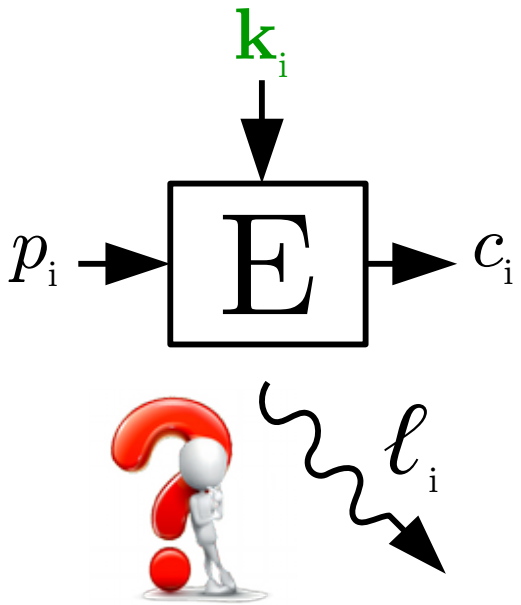
DPA Countermeasures

- Protect implementation (masking)
- Change key frequently (re-keying)
  - Reduce input data complexity
  - Protects the key and thus plaintext



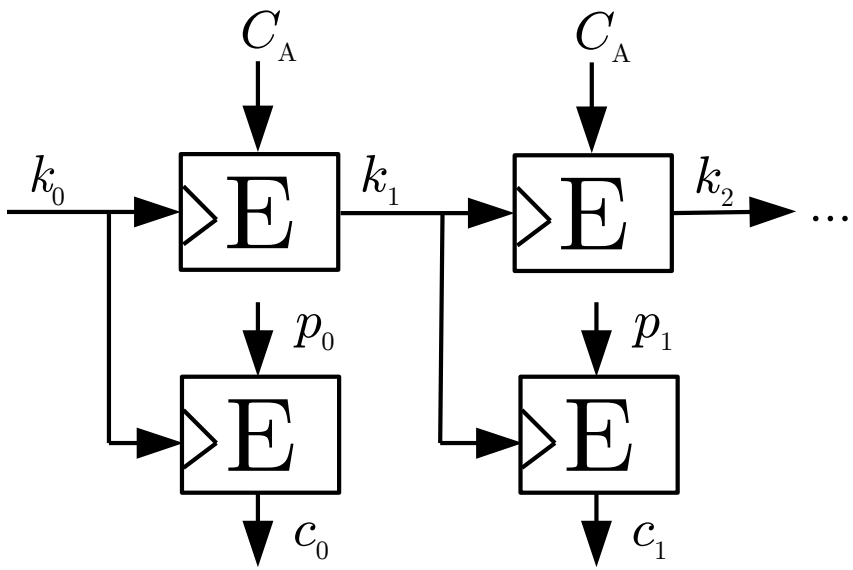
DPA Countermeasures

- Protect implementation (masking)
- Change key frequently (re-keying)
  - Reduce input data complexity
  - Protects the key and thus plaintext
  - Leakage-resilient encryption
  - Application to memory encryption?



Re-Keying for Memory Encryption

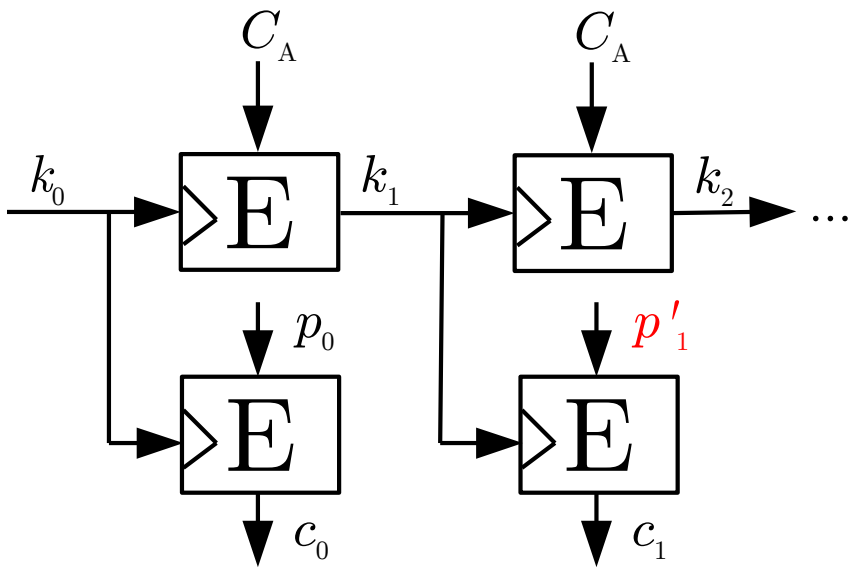
- Leakage-resilient schemes for arbitrary length



Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Re-Keying for Memory Encryption

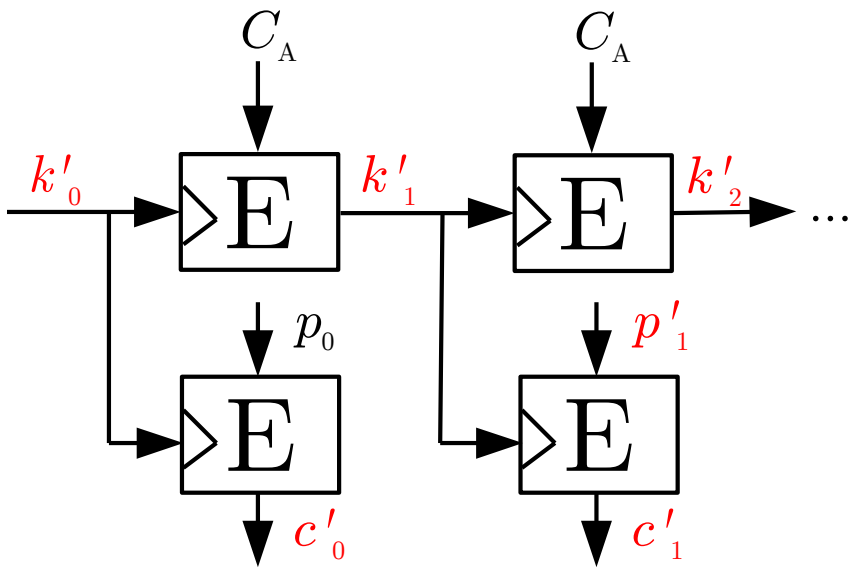
- Leakage-resilient schemes for arbitrary length
- Memory: read-modify-write operation



Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Re-Keying for Memory Encryption

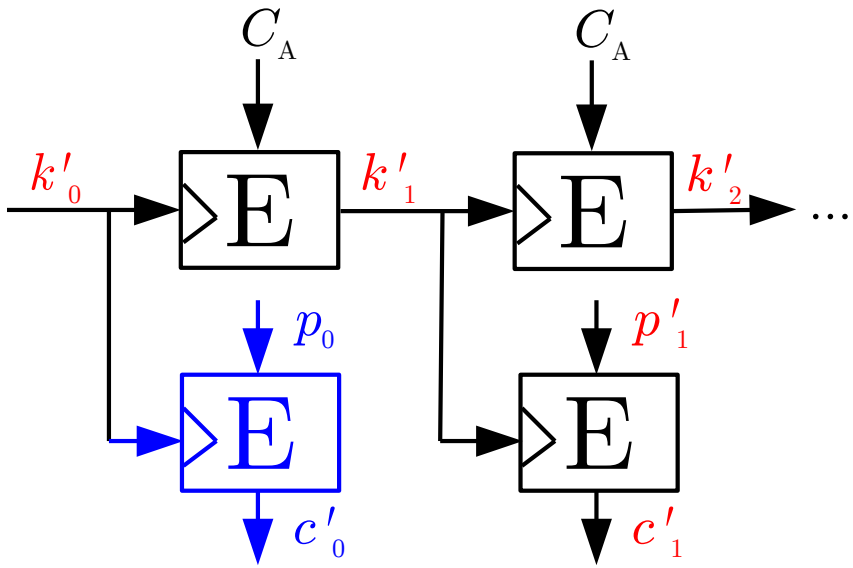
- Leakage-resilient schemes for arbitrary length
- Memory: read-modify-write operation



Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

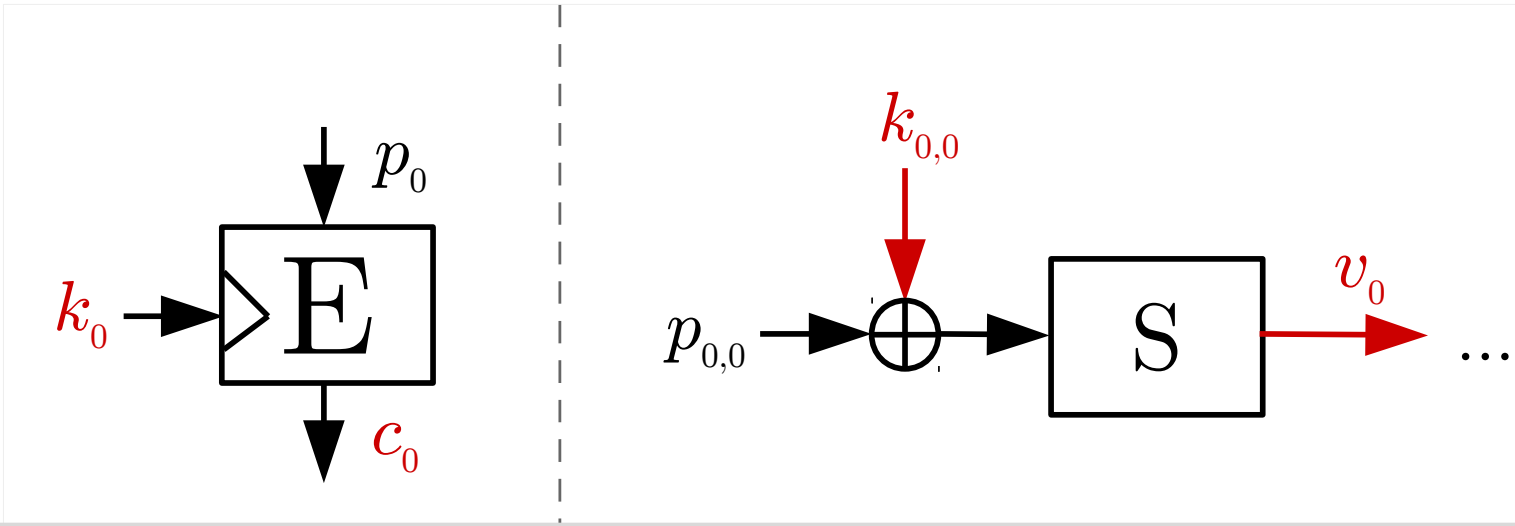
Re-Keying for Memory Encryption

- Leakage-resilient schemes for arbitrary length
- Memory: read-modify-write operation
- 2nd-order DPA to learn constant plaintexts



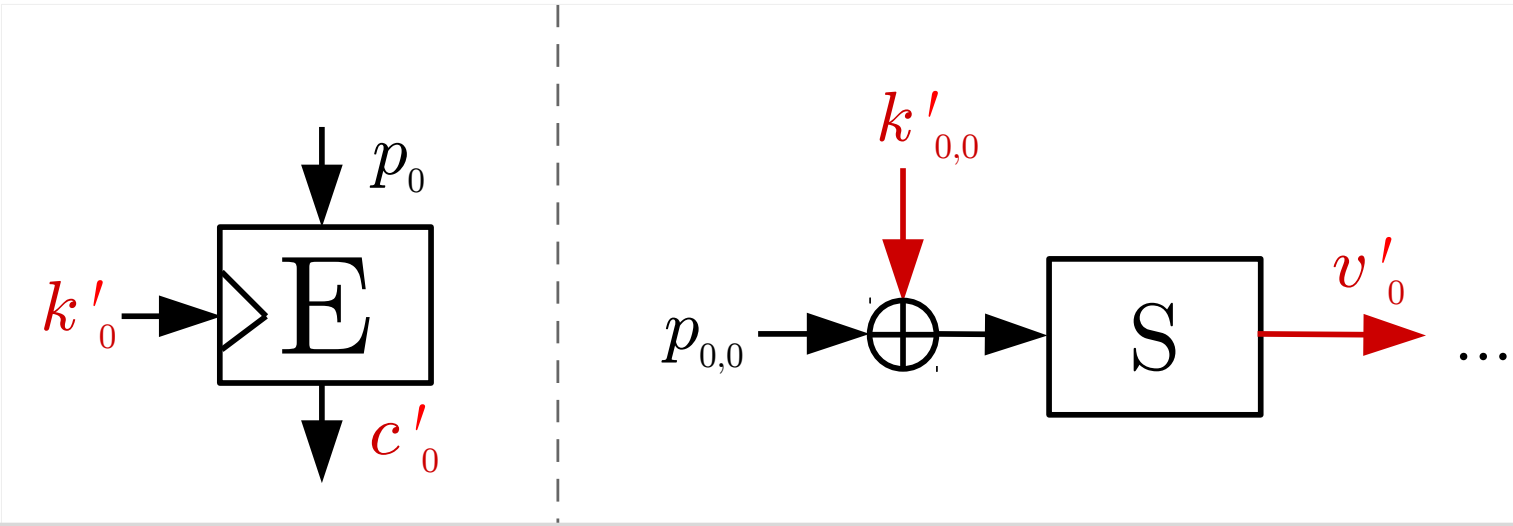
Re-Keying for Memory Encryption (2)

- Encryption of constant p_0 with different keys k_0, k_0', k_0''



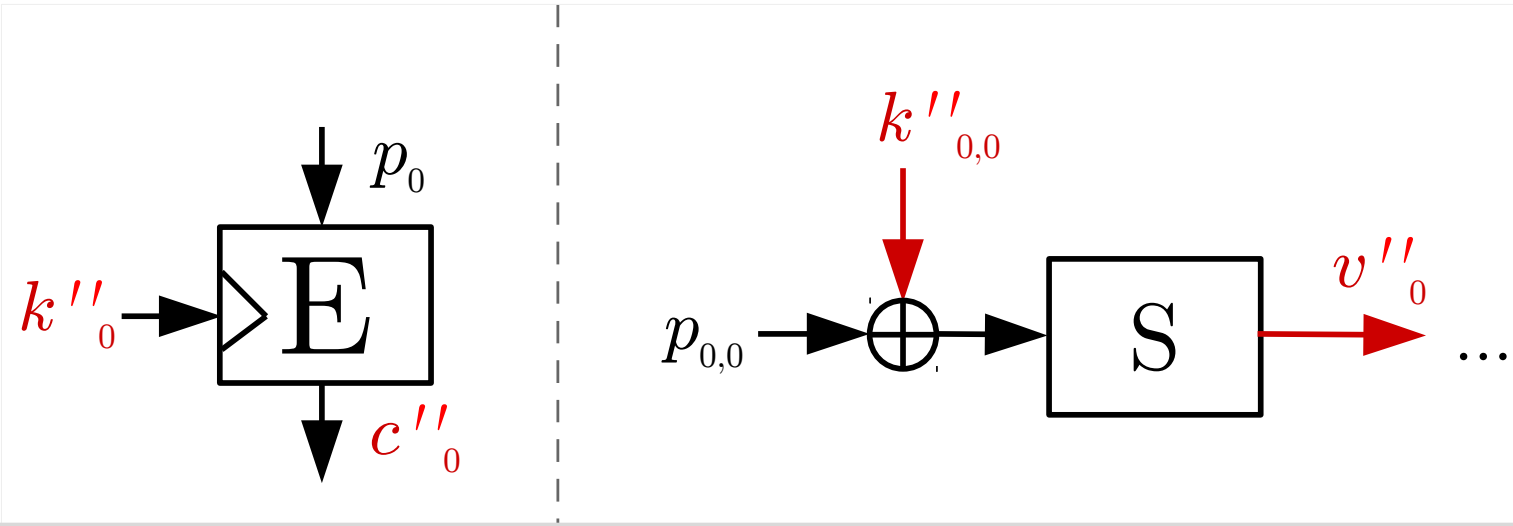
Re-Keying for Memory Encryption (2)

- Encryption of constant p_0 with different keys k_0, k_0', k_0''



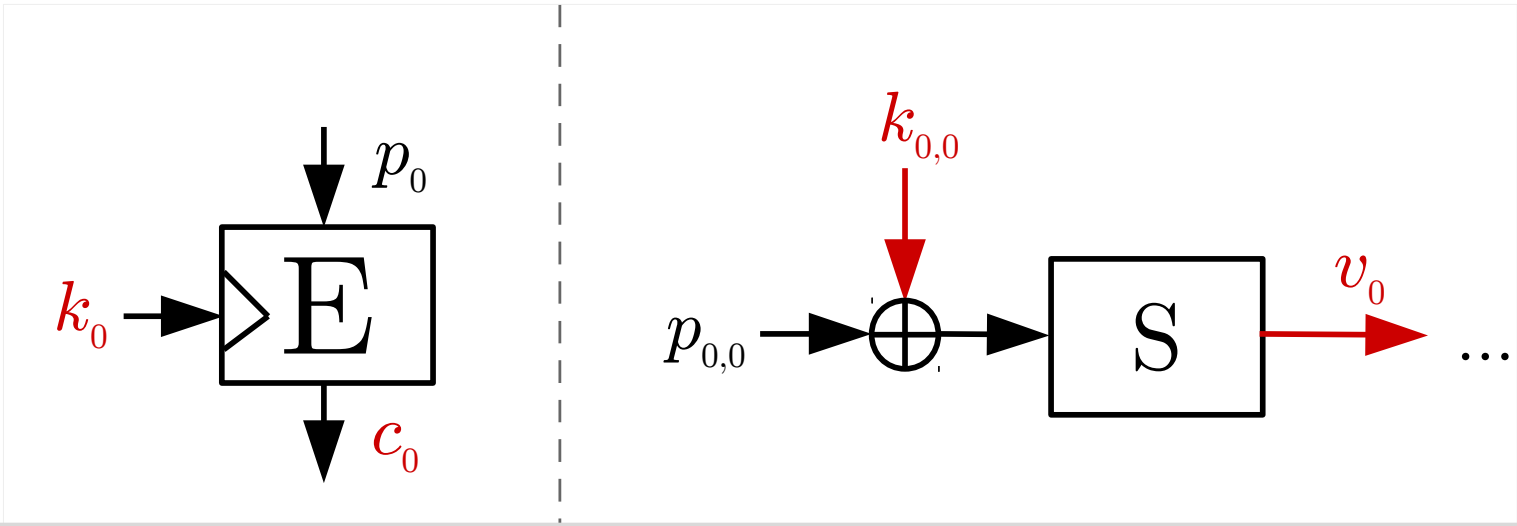
Re-Keying for Memory Encryption (2)

- Encryption of constant p_0 with different keys k_0, k_0', k_0''



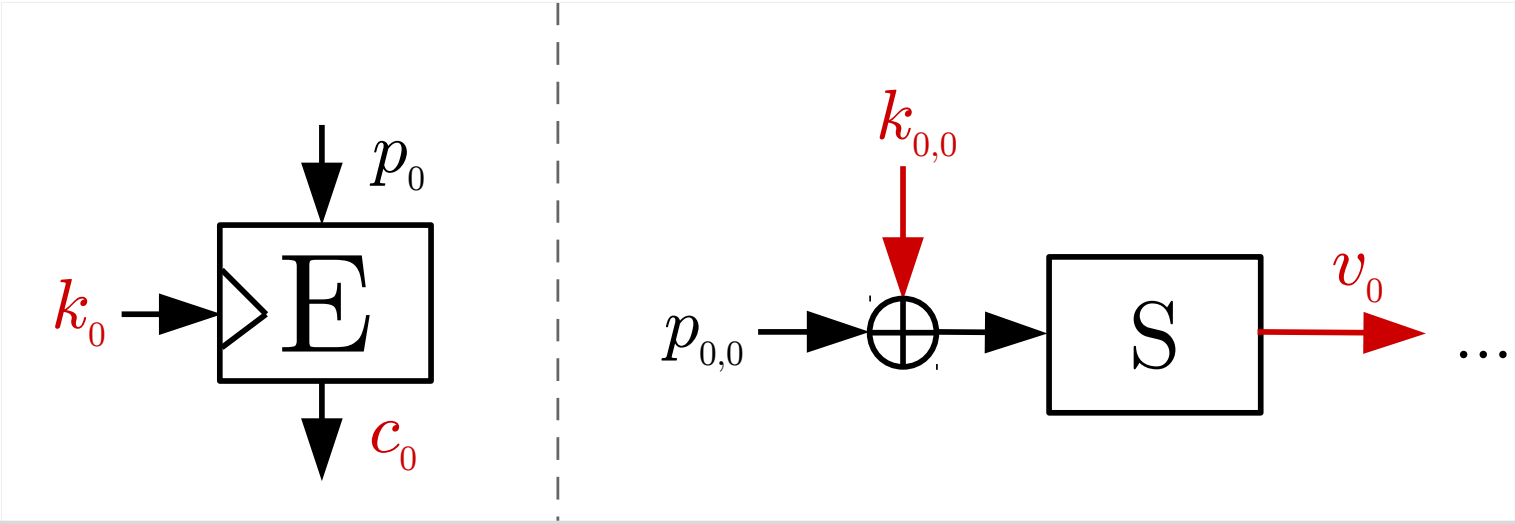
Re-Keying for Memory Encryption (2)

- Encryption of constant p_0 with different keys k_0, k_0', k_0''
- Profiling phase: templates for $k_{0,0}$ and v_0



Re-Keying for Memory Encryption (2)

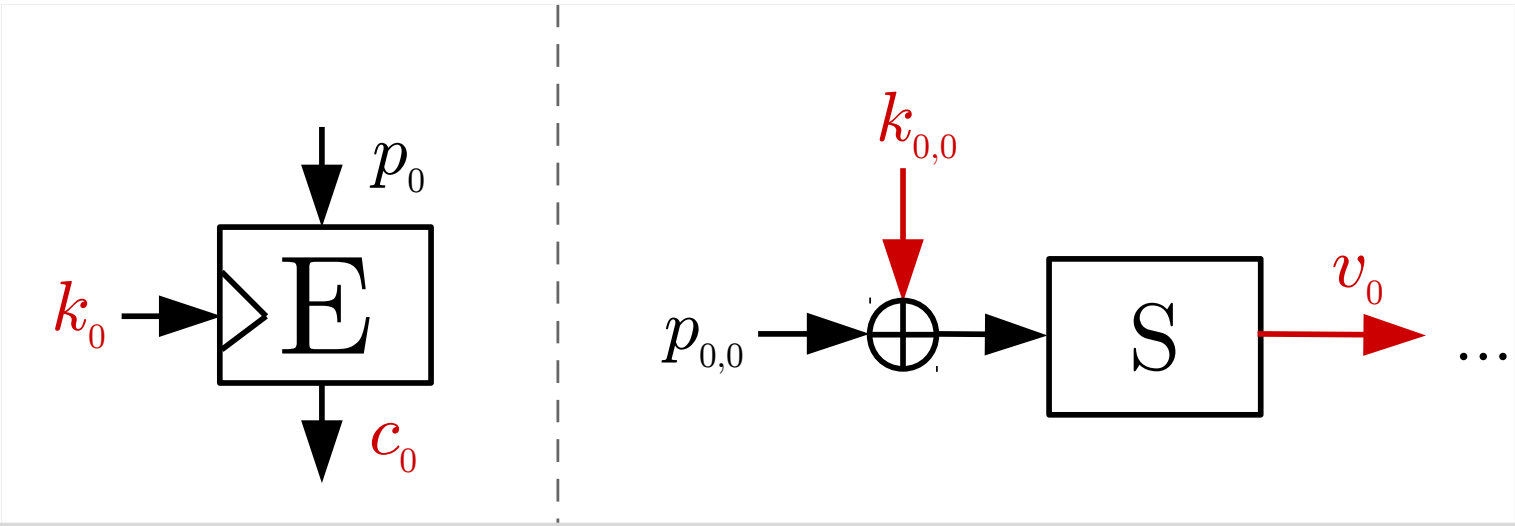
- Encryption of constant p_0 with different keys k_0, k_0', k_0''
- Profiling phase: templates for $k_{0,0}$ and v_0
- Attack phase:
  - Probabilities for $k_{0,0}$ and v_0



Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Re-Keying for Memory Encryption (2)

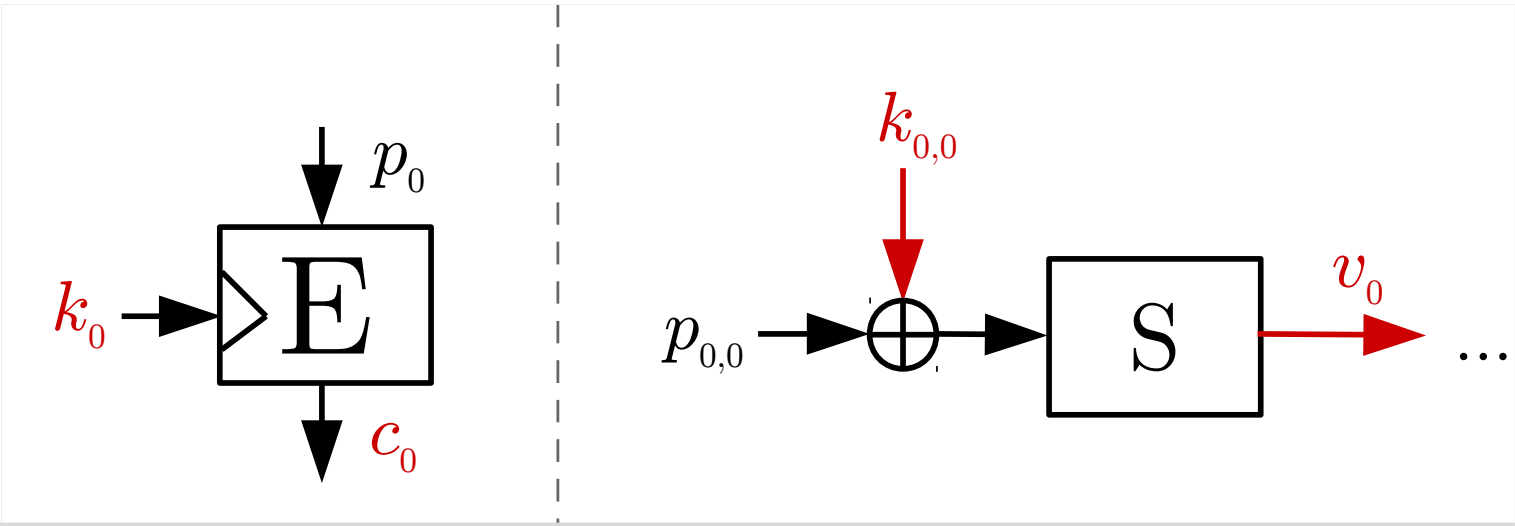
- Encryption of constant p_0 with different keys k_0, k_0', k_0''
- Profiling phase: templates for $k_{0,0}$ and v_0
- Attack phase:
  - Probabilities for $k_{0,0}$ and v_0
  - Joint probability of $k_{0,0}$ and $v_0 \rightarrow p_{0,0}$



Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Re-Keying for Memory Encryption (2)

- Encryption of constant p_0 with different keys k_0, k_0', k_0''
- Profiling phase: templates for $k_{0,0}$ and v_0
- Attack phase:
  - Probabilities for $k_{0,0}$ and v_0
  - Joint probability of $k_{0,0}$ and $v_0 \rightarrow p_{0,0}$
  - Many different keys to get unique $p_{0,0}$



Re-Keying for Memory Encryption (3)

- 2nd-order template attack
  - Very powerful attack
  - Hard to perform in practice

Re-Keying for Memory Encryption (3)

- 2nd-order template attack
  - Very powerful attack
  - Hard to perform in practice
- DPA cannot be prevented completely
  - 1st-order DPA security

Re-Keying for Memory Encryption (3)

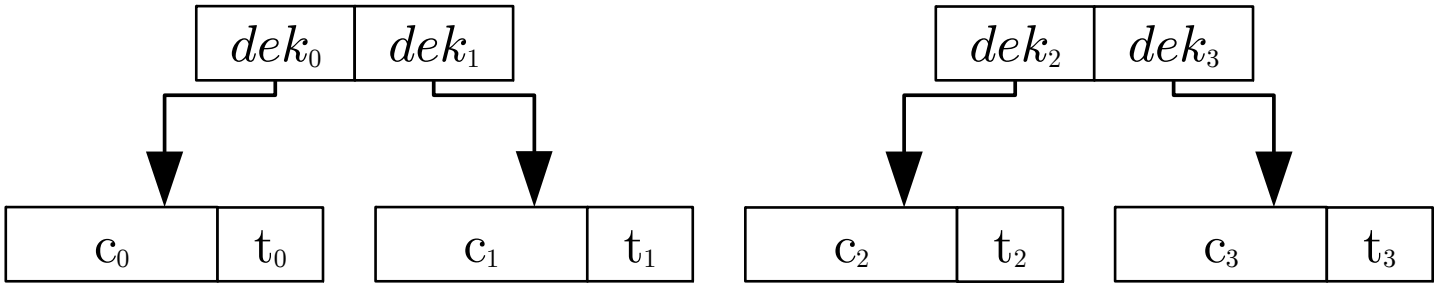
- 2nd-order template attack
  - Very powerful attack
  - Hard to perform in practice
- DPA cannot be prevented completely
  - 1st-order DPA security
- Streaming mode vs. random access
  - Split memory in blocks with different keys
  - Secure key storage on trusted chip
  - Minimal storage: tree approach
    - C.f., Merkle tree

MEAS

- Memory split into m blocks $p_0, p_1, \ldots, p_{m-1}$

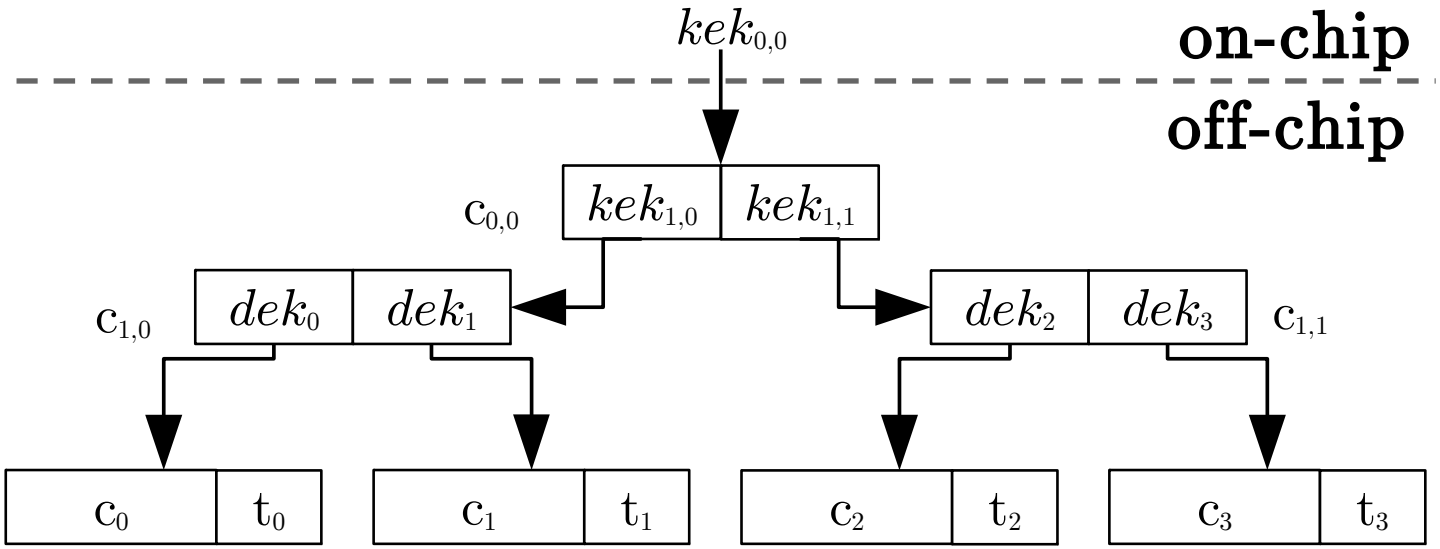
MEAS

- Memory split into m blocks p_0,p_1,...,p_{m-1}
- Apply AE scheme: $(c_i, t_i) = AE(dek_i; p_i)$



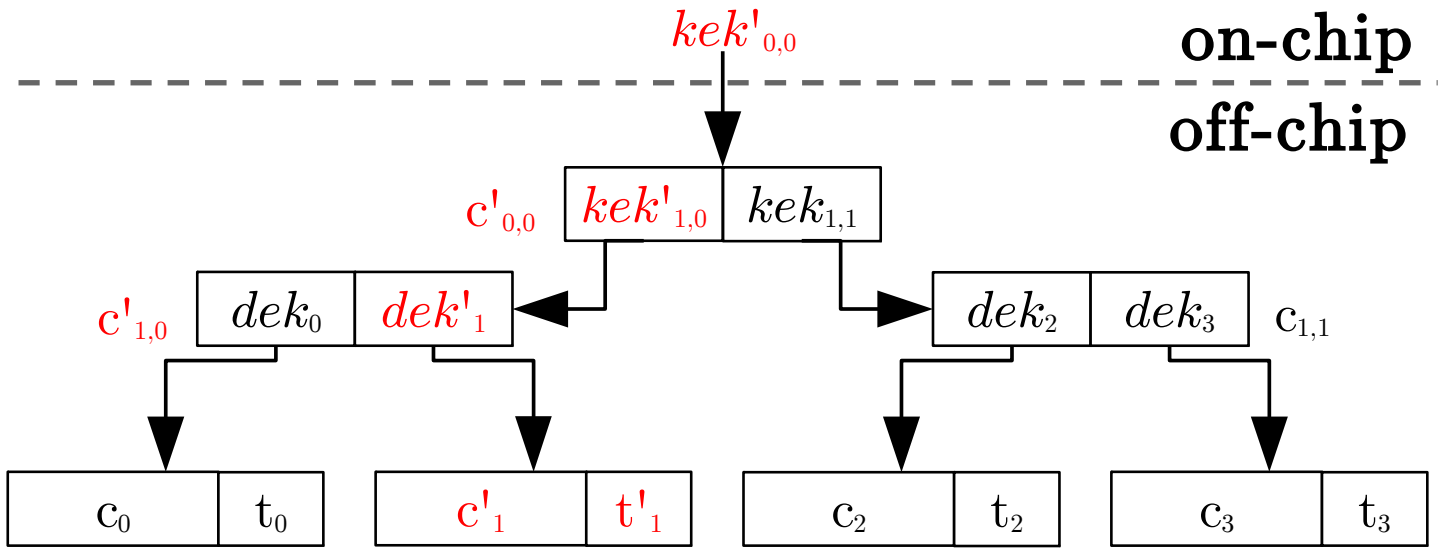
MEAS

- Memory split into m blocks $p_0, p_1, \ldots, p_{m-1}$
- Apply AE scheme: $(c_i, t_i) = AE(dek_i; p_i)$
- Recursive key encryption using ENC



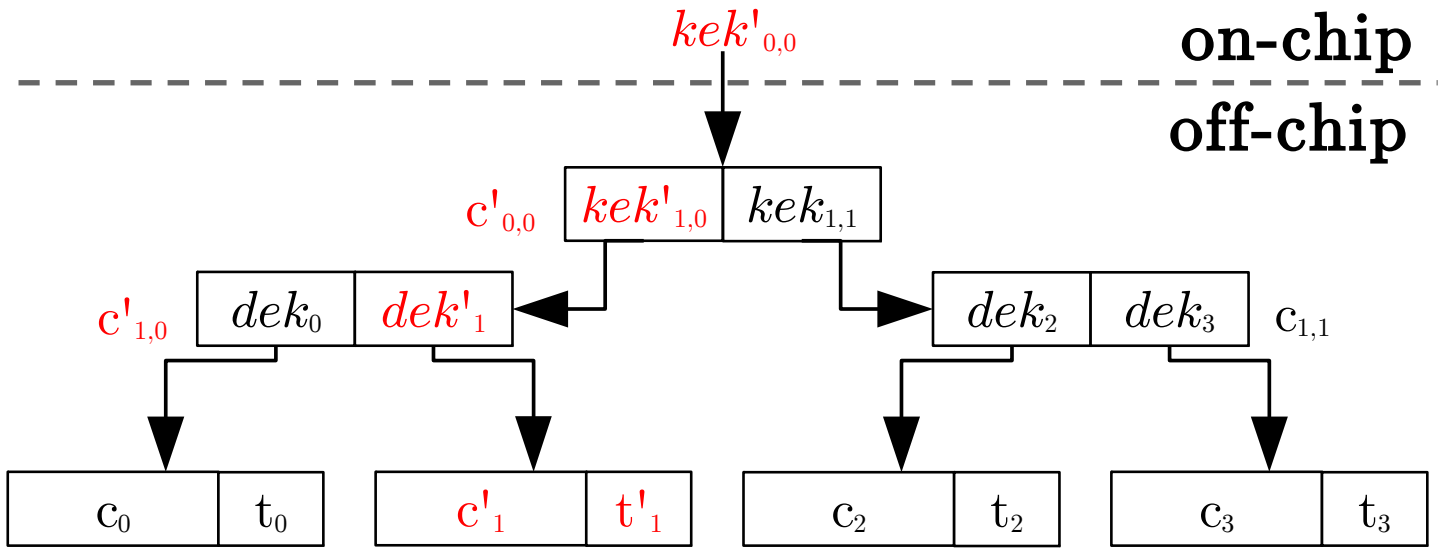
MEAS

- Re-keying: choose fresh, random keys on write



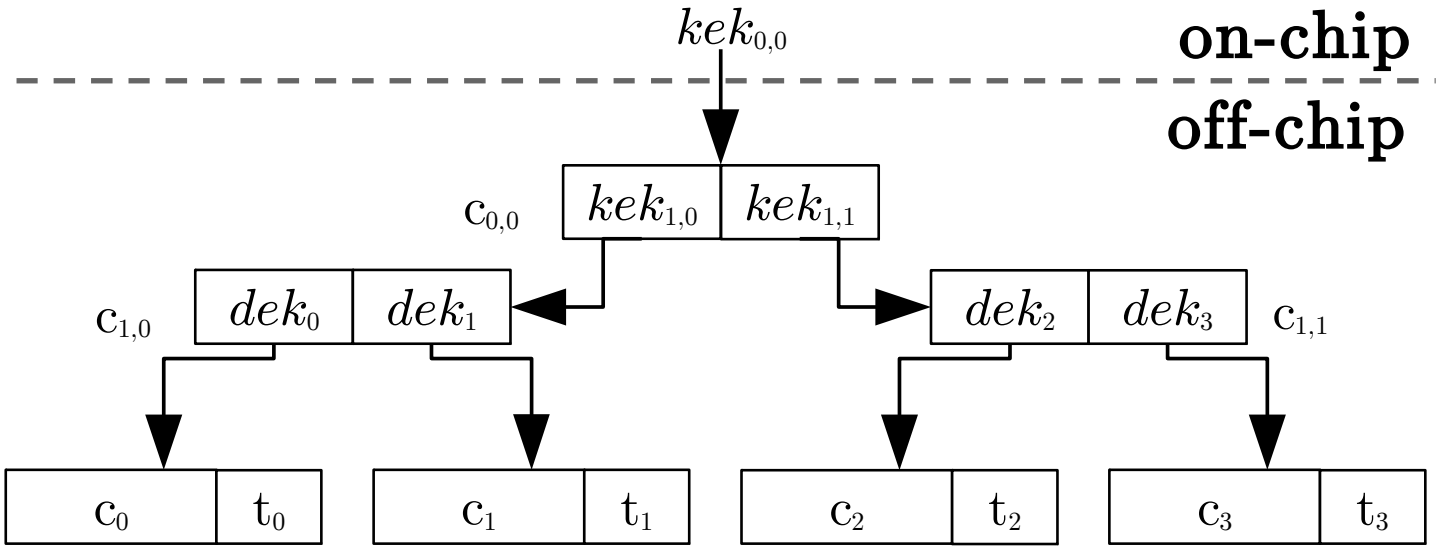
MEAS

- Re-keying: choose fresh, random keys on write
- Authenticity failure: reset or re-keying



MEAS

- Re-keying: choose fresh, random keys on write
- Authenticity failure: reset or re-keying
- Data complexity for DPA limited to 2



Higher-order DPA Security

- Tree-based scheme: 1st-order DPA security
  - Constant data encrypted with different keys

Higher-order DPA Security

- Tree-based scheme: 1st-order DPA security
  - Constant data encrypted with different keys
- Randomization of plaintext to increase attack order

Higher-order DPA Security

- Tree-based scheme: 1st-order DPA security
  - Constant data encrypted with different keys
- Randomization of plaintext to increase attack order
- Masking with dth-order security:
  - Tree node with b blocks $p_0, \ldots, p_{b-1}$

Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Higher-order DPA Security

- Tree-based scheme: 1st-order DPA security
  - Constant data encrypted with different keys
- Randomization of plaintext to increase attack order
- Masking with dth-order security:
  - Tree node with b blocks $p_0, \ldots, p_{b-1}$
  - Generate $d-1$ random masks $m_0, \ldots, m_{d-2}$
  - $r_i = p_i \oplus m_0 \oplus \ldots \oplus m_{d-2}$

Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Higher-order DPA Security

- Tree-based scheme: 1st-order DPA security
  - Constant data encrypted with different keys
- Randomization of plaintext to increase attack order
- Masking with dth-order security:
  - Tree node with b blocks $p_0, \ldots, p_{b-1}$
  - Generate $d-1$ random masks $m_0, \ldots, m_{d-2}$
  - $r_i = p_i \oplus m_0 \oplus \ldots \oplus m_{d-2}$
  - $c = ENC(dek; m_0||\ldots||m_{d-2}||r_0||\ldots||r_b)$
  - Masks updated when key changes

Thomas Unterluggauer, Mario Werner, and Stefan Mangard, Graz University of Technology

Higher-order DPA Security

- Tree-based scheme: 1st-order DPA security
  - Constant data encrypted with different keys
- Randomization of plaintext to increase attack order
- Masking with dth-order security:
  - Tree node with b blocks $p_0, \ldots, p_{b-1}$
  - Generate $d-1$ random masks $m_0, \ldots, m_{d-2}$
  - $r_i = p_i \oplus m_0 \oplus \ldots \oplus m_{d-2}$
  - $c = ENC(dek; m_0||\ldots||m_{d-2}||r_0||\ldots||r_b)$
  - Masks updated when key changes
- Various trade-offs possible

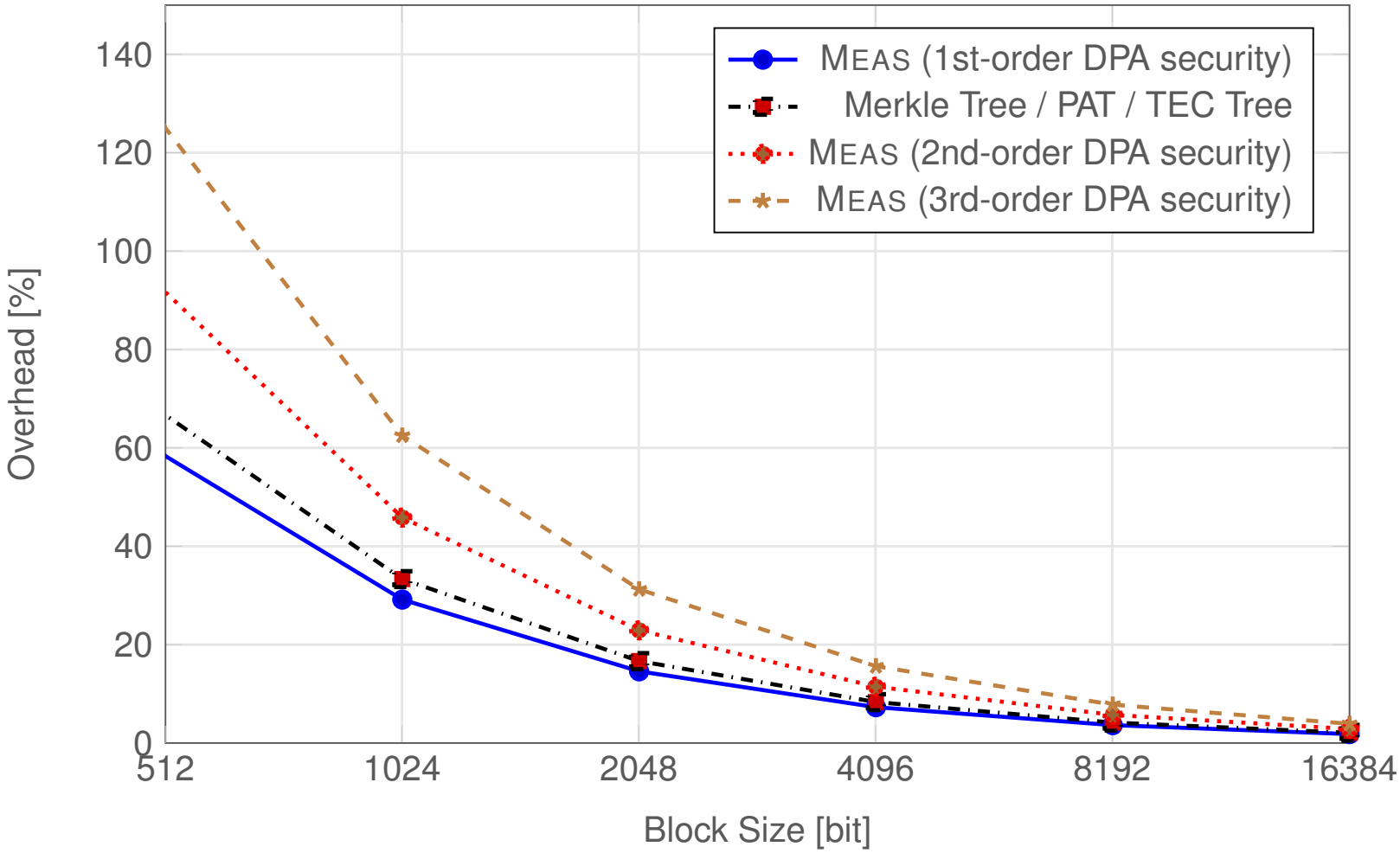
Comparison

	Auth.	Conf.	DPA Security	
MEAS	✓	✓	✓	
PAT	✓			
TEC Tree	✓	✓		
Merkle Tree	✓		✓	
XTS / XEX		✓		

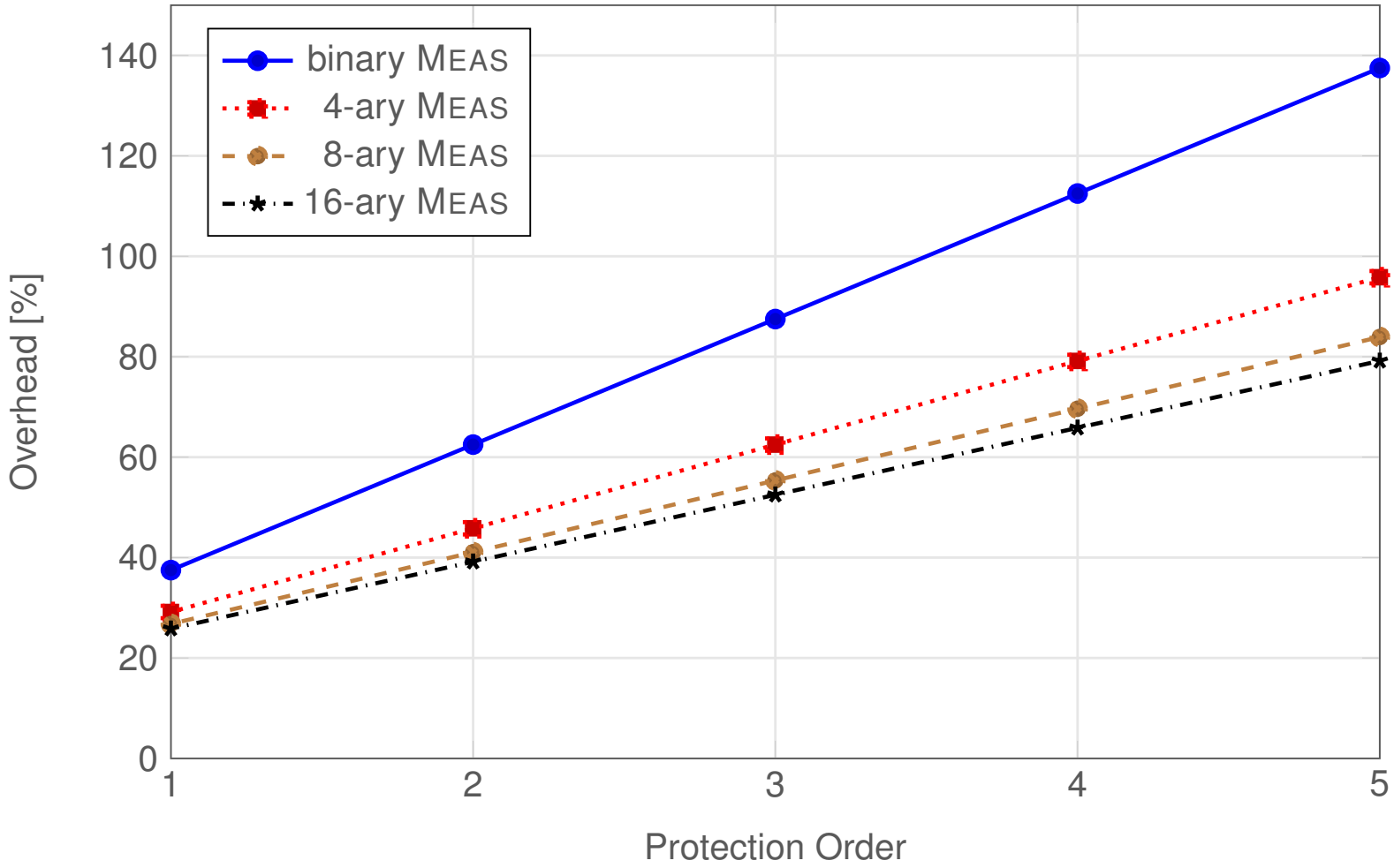
Comparison

	Auth.	Conf.	DPA Security	Parallelizable Read	Parallelizable Write
MEAS	✓	✓	✓		
PAT	✓			✓	✓
TEC Tree	✓	✓		✓	✓
Merkle Tree	✓		✓	✓	
XTS / XEX		✓		✓	✓

Memory Overhead



Memory Overhead



Conclusion

- Memory requires protection from physical access
- Power analysis feasible for most physical attackers
- Frequent re-keying as one DPA countermeasure

Conclusion

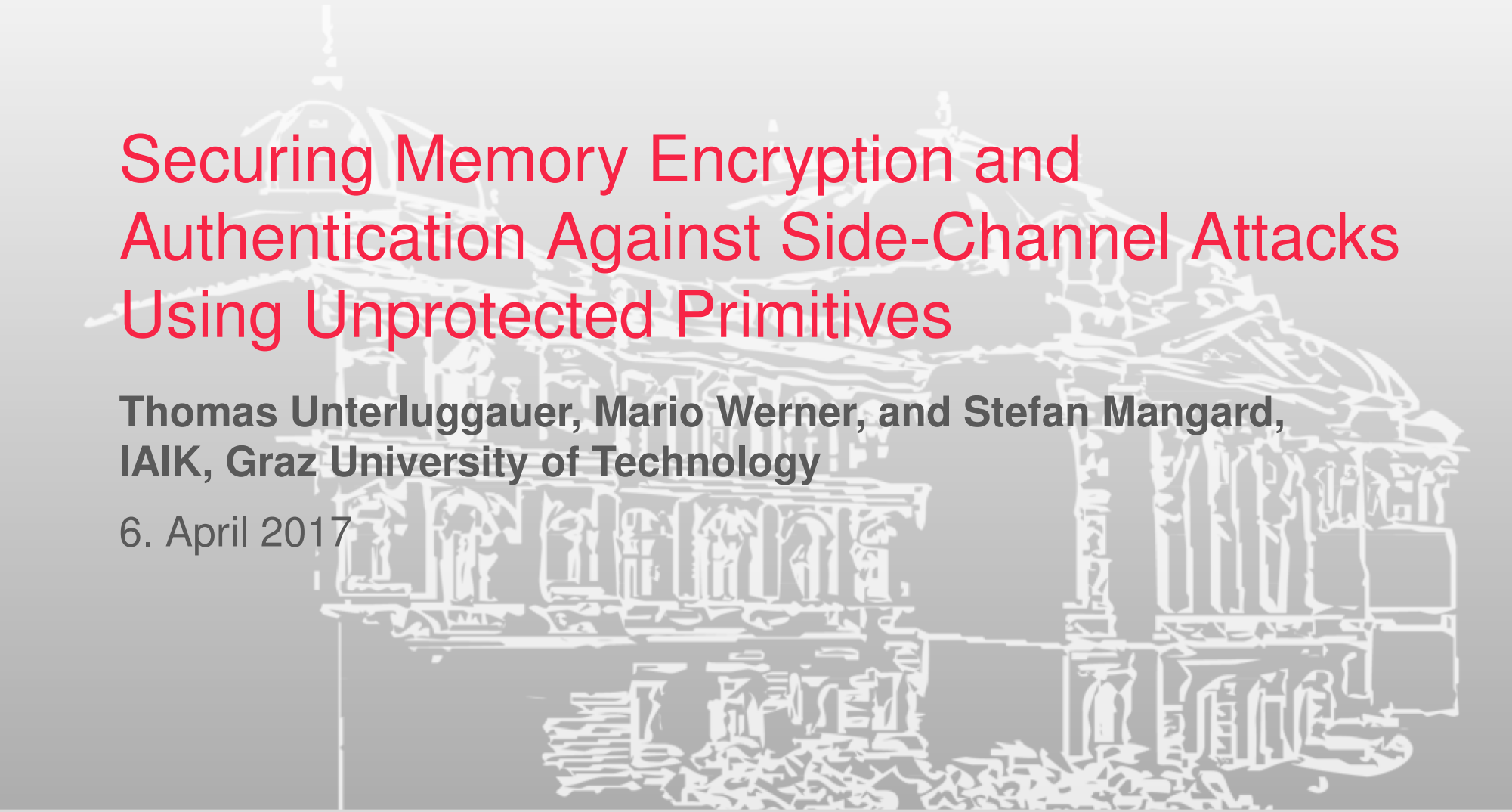
- Memory requires protection from physical access
- Power analysis feasible for most physical attackers
- Frequent re-keying as one DPA countermeasure
- MEAS: memory encryption and authentication with DPA protection
  - No DPA-protected implementation required

Conclusion

- Memory requires protection from physical access
- Power analysis feasible for most physical attackers
- Frequent re-keying as one DPA countermeasure
- MEAS: memory encryption and authentication with DPA protection
  - No DPA-protected implementation required
  - Combines re-keying and authentication trees
  - Masking of plaintexts for higher-order protection

Conclusion

- Memory requires protection from physical access
- Power analysis feasible for most physical attackers
- Frequent re-keying as one DPA countermeasure
- MEAS: memory encryption and authentication with DPA protection
  - No DPA-protected implementation required
  - Combines re-keying and authentication trees
  - Masking of plaintexts for higher-order protection
  - Memory overhead as existing authentication trees



Securing Memory Encryption and Authentication Against Side-Channel Attacks Using Unprotected Primitives

Thomas Unterluggauer, Mario Werner, and Stefan Mangard,
IAIK, Graz University of Technology

6. April 2017