

CASA

CYBER SECURITY IN THE AGE
OF LARGE-SCALE ADVERSARIES

HAWKEYE – Recovering Symmetric Cryptography From Hardware Circuits

CRYPTO 2024, August 19

Gregor Leander, Christof Paar, Julian Speith and Lukas Stennes

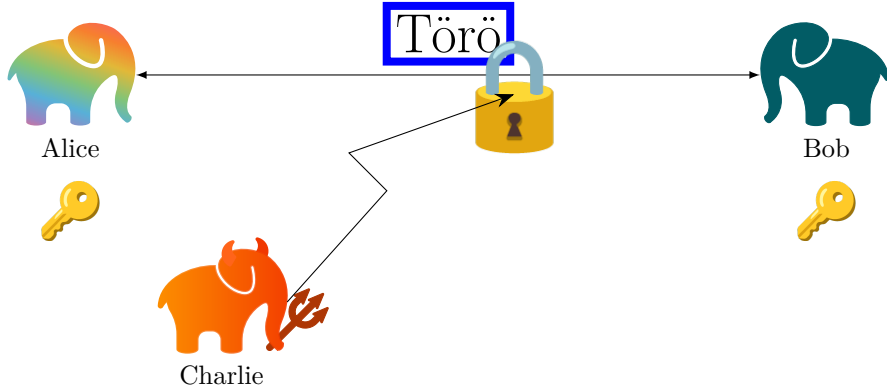
RUHR
UNIVERSITÄT
BOCHUM

RUB

MAX PLANCK INSTITUTE
FOR SECURITY AND PRIVACY



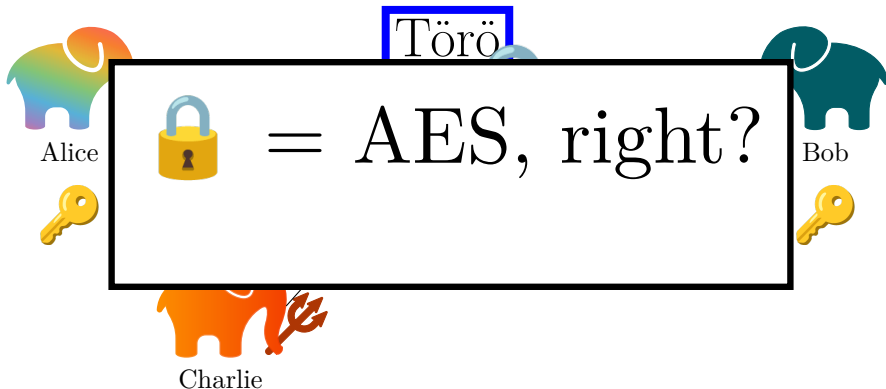
Symmetric Cryptography (Theory)



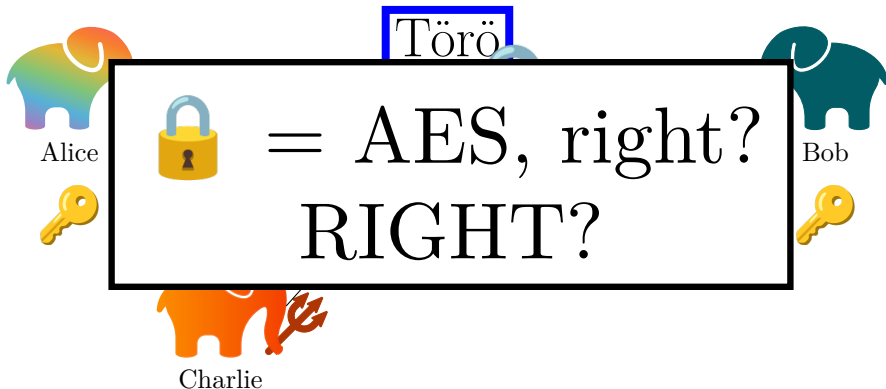
Symmetric Cryptography (Theory)



Symmetric Cryptography (Theory)



Törö

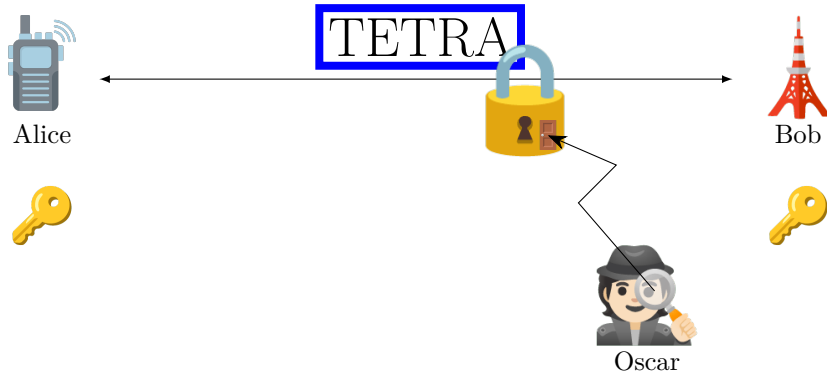


Alice = AES, right?
RIGHT?

Bob

Charlie

Symmetric Cryptography (Practice)



All cops are broadcasting: TETRA under scrutiny

Carlo Meijer
Midnight Blue
c.meijer@midnightblue.nl

Wouter Bokslag
Midnight Blue
w.bokslag@midnightblue.nl

Jos Wetzels
Midnight Blue
j.wetzels@midnightblue.nl

USENIX 2023

There is bad crypto.

There is bad crypto. Let's find it.

How to Find (Bad) Crypto

- ▶ Documents
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents **easy**
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents **easy**
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software ***Where's Crypto?*, Meijer, Moonsamy, Wetzels at USENIX 2021**
 - ▶ Hardware

How to Find (Bad) Crypto

- ▶ Documents **easy**
 - ▶ Academic papers, standards, patents . . .
- ▶ Reverse Engineering
 - ▶ Software ***Where's Crypto?, Meijer, Moonsamy, Wetzels at USENIX 2021***
 - ▶ Hardware **???**

How to Find (Bad) Crypto

- ▶ Documents **easy**
 - ▶ Academic papers, etc.
- ▶ Reverse Engineering
 - ▶ Software **Where**
 - ▶ Hardware



presented at **USENIX 2021**



Background

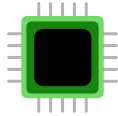
Hardware Reverse Engineering

ASICs

FPGAs

Hardware Reverse Engineering

ASICs

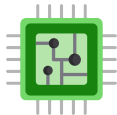


Decapsulation

FPGAs

Hardware Reverse Engineering

ASICs

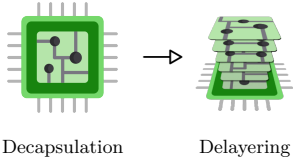


Decapsulation

FPGAs

Hardware Reverse Engineering

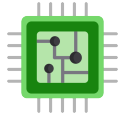
ASICs



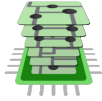
FPGAs

Hardware Reverse Engineering

ASICs



Decapsulation



Delayering

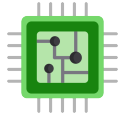


Imaging

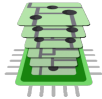
FPGAs

Hardware Reverse Engineering

ASICs



Decapsulation



Delayering



Imaging

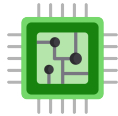


Image Processing

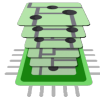
FPGAs

Hardware Reverse Engineering

ASICs



Decapsulation



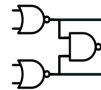
Delayering



Imaging



Image Processing

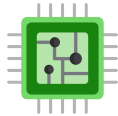


Netlist

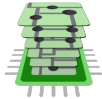
FPGAs

Hardware Reverse Engineering

ASICs



Decapsulation



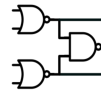
Delayering



Imaging



Image Processing



Netlist

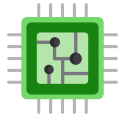


Netlist Analysis

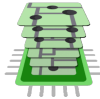
FPGAs

Hardware Reverse Engineering

ASICs



Decapsulation



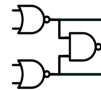
Delayering



Imaging



Image Processing



Netlist



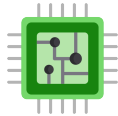
Netlist Analysis

FPGAs

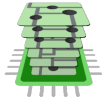


Hardware Reverse Engineering

ASICs



Decapsulation



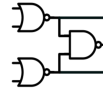
Delayering



Imaging



Image Processing



Netlist



Netlist Analysis

FPGAs

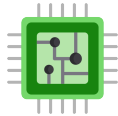


Bitstream Extraction



Hardware Reverse Engineering

ASICs



Decapsulation



Delayering



Imaging



Image Processing



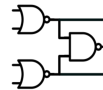
FPGAs



Bitstream Extraction



Bitstream Conversion



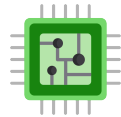
Netlist



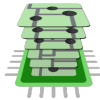
Netlist Analysis

Hardware Reverse Engineering

ASICs



Decapsulation



Delayering



Imaging



Image Processing



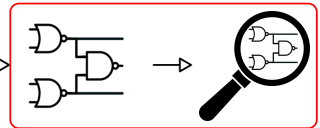
FPGAs



Bitstream Extraction



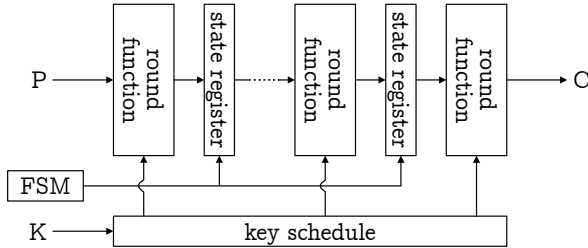
Bitstream Conversion



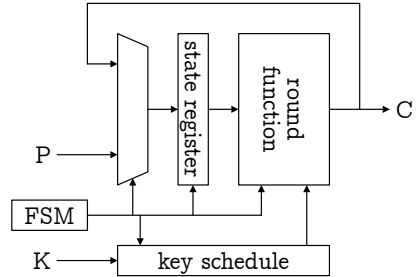
Netlist

Netlist Analysis

Symmetric Cryptography in Hardware

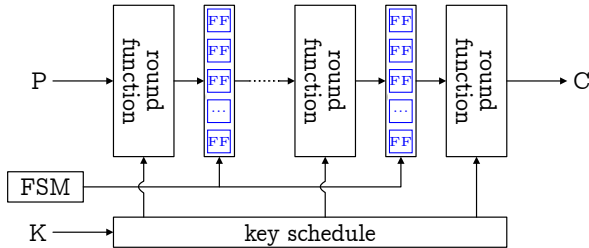


Pipelined

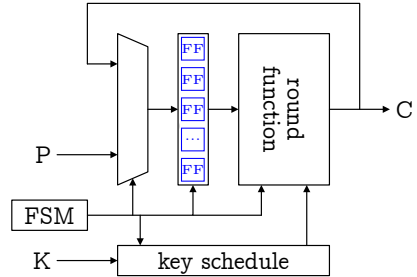


Round-Based

Symmetric Cryptography in Hardware

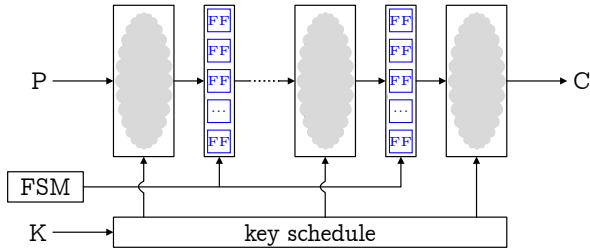


Pipelined

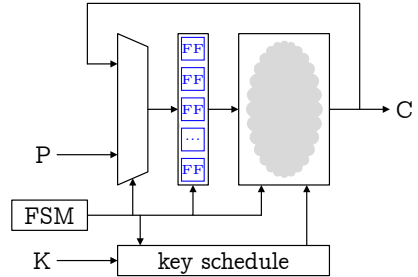


Round-Based

Symmetric Cryptography in Hardware

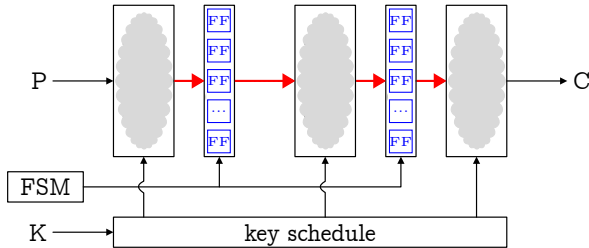


Pipelined

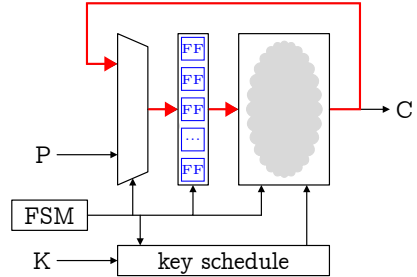


Round-Based

Symmetric Cryptography in Hardware



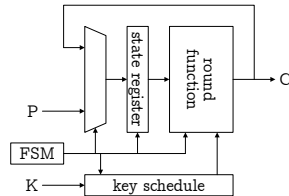
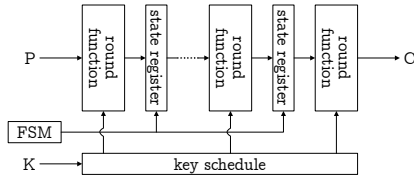
Pipelined



Round-Based

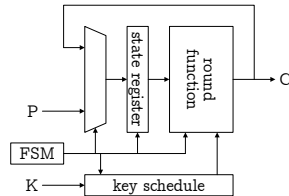
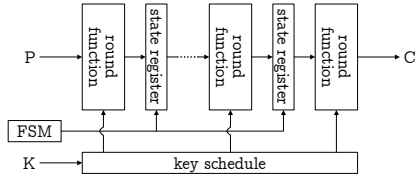
What Is Special About Symmetric Cryptography?

- ▶ Flip-flops in state register influence only state register and ciphertext output
 - ▶ State register flip-flops and ciphertext flip-flops are distinguishable
- ▶ Avalanche effect: Bits in first state register influences all bits of later state registers
- ▶ Round function only depends on plaintext, round keys, and finite state machine control signals



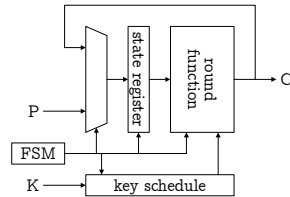
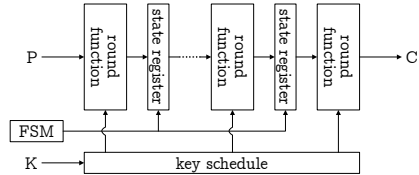
What Is Special About Symmetric Cryptography?

- ▶ Flip-flops in state register influence only state register and ciphertext output
 - ▶ State register flip-flops and ciphertext flip-flops are distinguishable
- ▶ Avalanche effect: Bits in first state register influences all bits of later state registers
- ▶ Round function only depends on plaintext, round keys, and finite state machine control signals



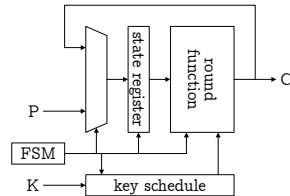
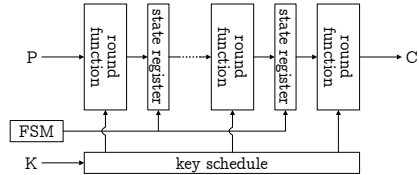
What Is Special About Symmetric Cryptography?

- ▶ Flip-flops in state register influence only state register and ciphertext output
 - ▶ State register flip-flops and ciphertext flip-flops are distinguishable
- ▶ Avalanche effect: Bits in first state register influences all bits of later state registers
- ▶ Round function only depends on plaintext, round keys, and finite state machine control signals



What Is Special About Symmetric Cryptography?

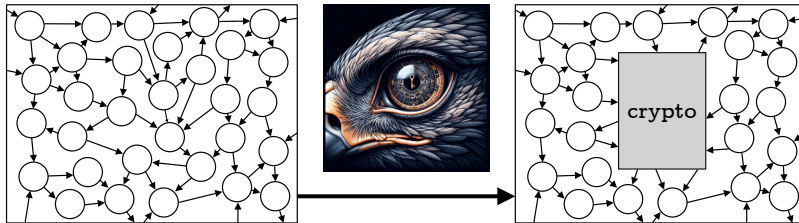
- ▶ Flip-flops in state register influence only state register and ciphertext output
 - ▶ State register flip-flops and ciphertext flip-flops are distinguishable
- ▶ Avalanche effect: Bits in first state register influences all bits of later state registers
- ▶ Round function only depends on plaintext, round keys, and finite state machine control signals



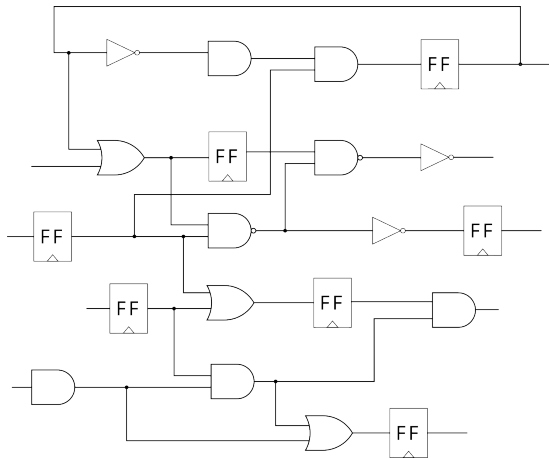


Techniques

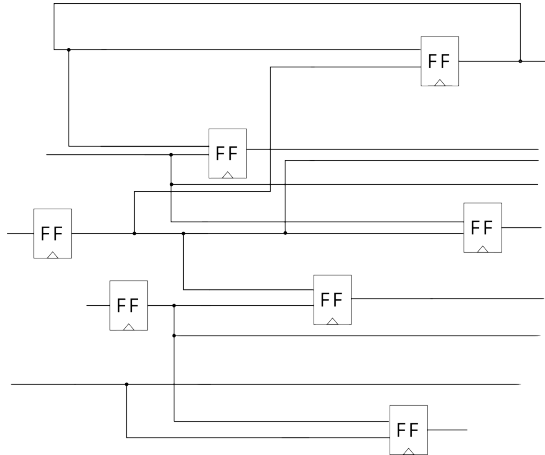
The Goal



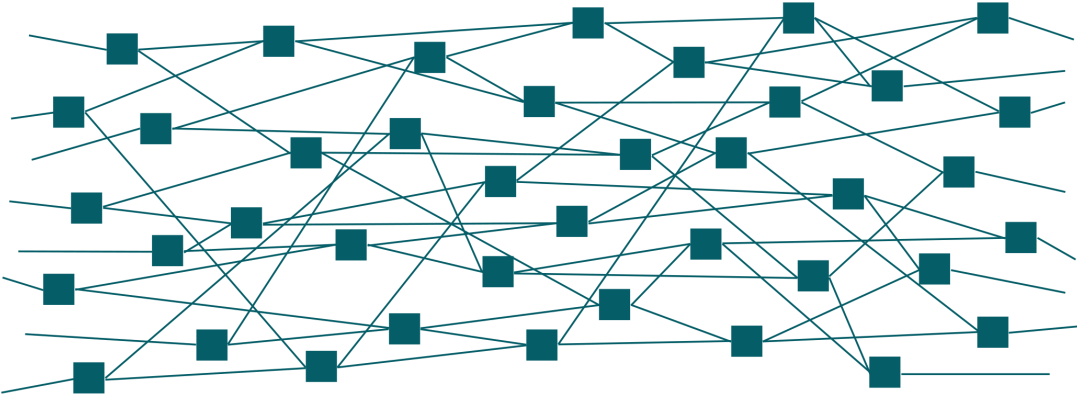
Candidate Search – Preprocessing



Candidate Search – Preprocessing



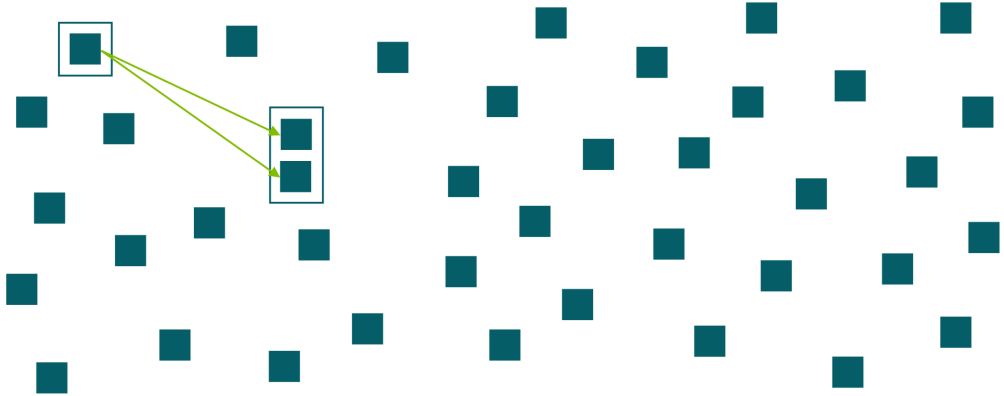
Candidate Search



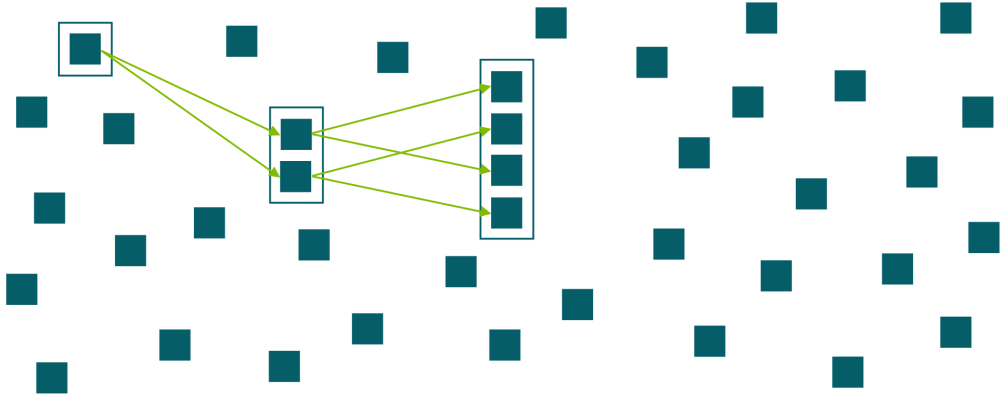
Candidate Search



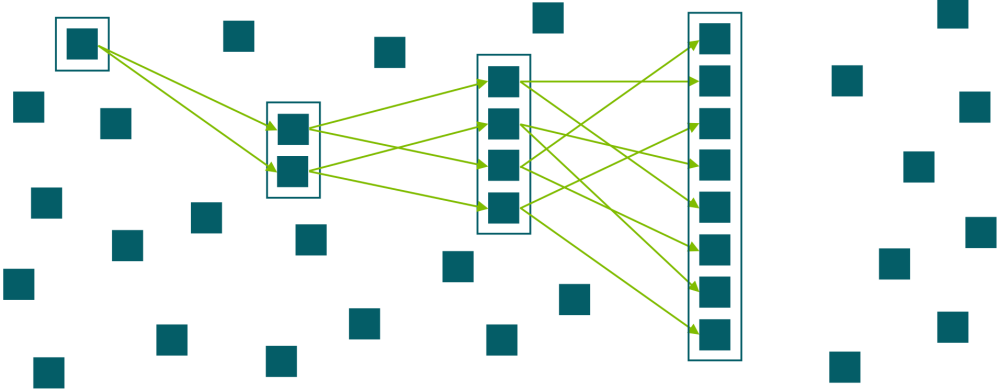
Candidate Search



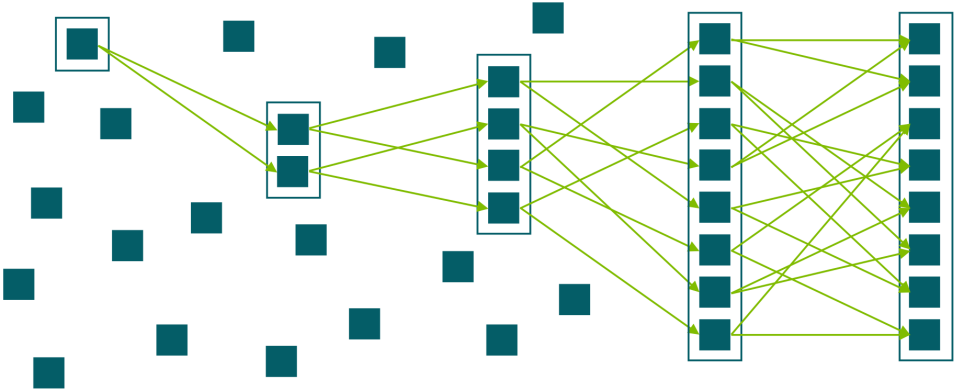
Candidate Search



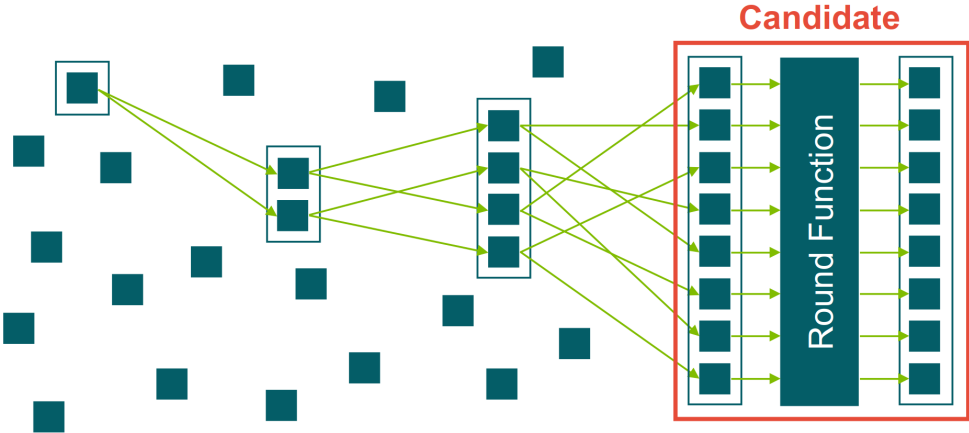
Candidate Search



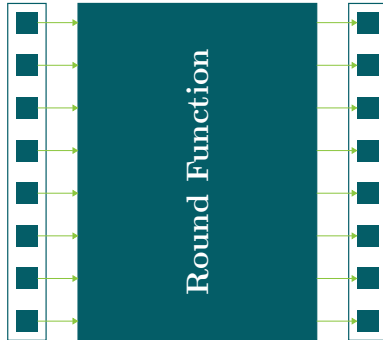
Candidate Search



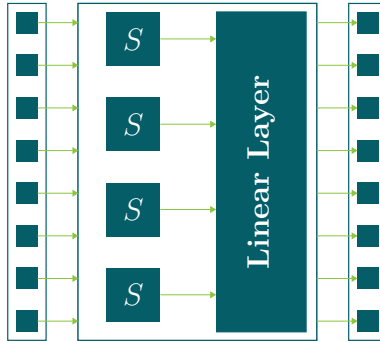
Candidate Search



Round Function Analysis



Round Function Analysis





Evaluation

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

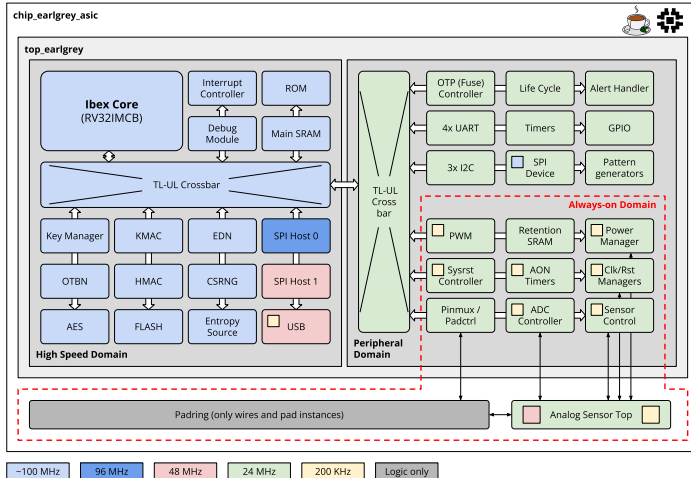
Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

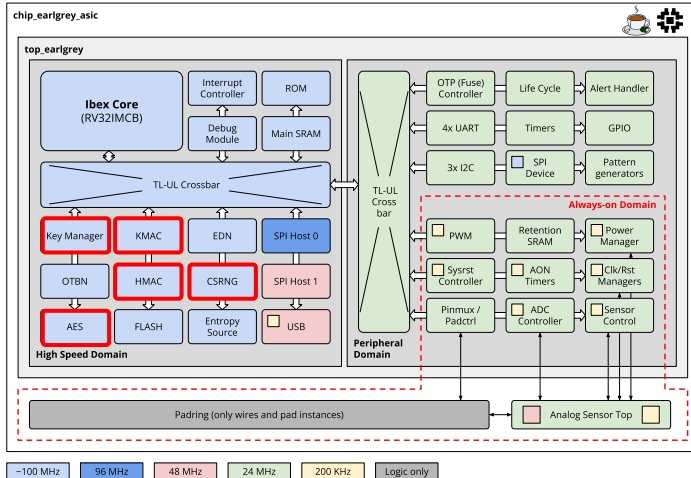
Evaluation

- ▶ Our techniques are based on *heuristics*
- ▶ Imperative to *evaluate* the techniques
- ▶ Hardware reverse engineering (ASIC/FPGA → netlist) not an option
- ▶ We synthesized open source netlists (hardware design → netlists)
 - ▶ OpenTitan: *industry-grade* chip
 - ▶ Cryptographic Accelerators in a small system-on-chips
 - ▶ Isolated (non-)cryptographic benchmarks
- ▶ Confident that our techniques generalise also to *unknown* ciphers
- ▶ Implementation available as artifact

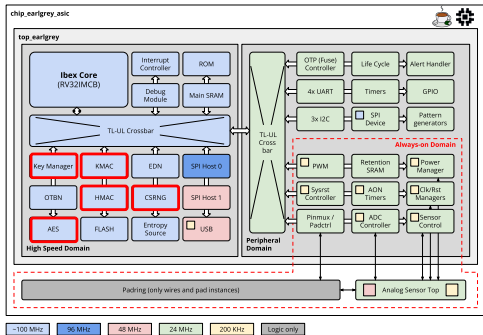
Evaluation: OpenTitan



Evaluation: OpenTitan



Evaluation: OpenTitan

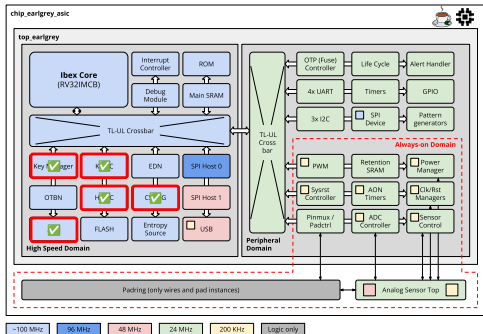


Contains 424.341 gates

After 44s on Apple M2:

| #FFs | Description |
|------|-----------------------|
| 640 | partial Keccak state |
| 128 | AES state |
| 256 | AES key state |
| 256 | SHA-2 state |
| 256 | Xoshiro256++ state |
| 192 | PRESENT state and key |
| 64 | PRINCE output |
| ... | ... |

Evaluation: OpenTitan



Contains 424.341 gates

After 44s on Apple M2:

| #FFs | Description |
|------|-----------------------|
| 640 | partial Keccak state |
| 128 | AES state |
| 256 | AES key state |
| 256 | SHA-2 state |
| 256 | Xoshiro256++ state |
| 192 | PRESENT state and key |
| 64 | PRINCE output |
| ... | ... |

Evaluation: Cryptographic Accelerators and Isolated Benchmarks

- ▶ We evaluate HAWKEYE on a variety of symmetric ciphers and *noise*
- ▶ Run time is in the seconds (roughly linear in size of netlist)
- ▶ HAWKEYE finds almost all ciphers
- ▶ Only very few false positives
 - ▶ Mostly recurring, e.g., counters
 - ▶ Could be filtered out

Evaluation: Cryptographic Accelerators and Isolated Benchmarks

- ▶ We evaluate HAWKEYE on a variety of symmetric ciphers and *noise*
- ▶ Run time is in the seconds (roughly linear in size of netlist)
- ▶ HAWKEYE finds almost all ciphers
- ▶ Only very few false positives
 - ▶ Mostly recurring, e.g., counters
 - ▶ Could be filtered out

Evaluation: Cryptographic Accelerators and Isolated Benchmarks

- ▶ We evaluate HAWKEYE on a variety of symmetric ciphers and *noise*
- ▶ Run time is in the seconds (roughly linear in size of netlist)
- ▶ HAWKEYE finds almost all ciphers
- ▶ Only very few false positives
 - ▶ Mostly recurring, e.g., counters
 - ▶ Could be filtered out

Evaluation: Cryptographic Accelerators and Isolated Benchmarks

- ▶ We evaluate HAWKEYE on a variety of symmetric ciphers and *noise*
- ▶ Run time is in the seconds (roughly linear in size of netlist)
- ▶ HAWKEYE finds almost all ciphers
- ▶ Only very few false positives
 - ▶ Mostly recurring, e.g., counters
 - ▶ Could be filtered out

Evaluation: Cryptographic Accelerators and Isolated Benchmarks

- ▶ We evaluate HAWKEYE on a variety of symmetric ciphers and *noise*
- ▶ Run time is in the seconds (roughly linear in size of netlist)
- ▶ HAWKEYE finds almost all ciphers
- ▶ Only very few false positives
 - ▶ Mostly recurring, e.g., counters
 - ▶ Could be filtered out

Evaluation: Cryptographic Accelerators and Isolated Benchmarks

- ▶ We evaluate HAWKEYE on a variety of symmetric ciphers and *noise*
- ▶ Run time is in the seconds (roughly linear in size of netlist)
- ▶ HAWKEYE finds almost all ciphers
- ▶ Only very few false positives
 - ▶ Mostly recurring, e.g., counters
 - ▶ Could be filtered out

Future Work

- ▶ Symmetric cryptography based on shift registers
- ▶ Side-channel protected implementation
- ▶ Actually finding unknown cryptography
- ▶ **You have a real-world device to look at? Please reach out to us!**

Future Work

- ▶ Symmetric cryptography based on shift registers
- ▶ Side-channel protected implementation
- ▶ Actually finding unknown cryptography
- ▶ **You have a real-world device to look at? Please reach out to us!**

Future Work

- ▶ Symmetric cryptography based on shift registers
- ▶ Side-channel protected implementation
- ▶ Actually finding unknown cryptography
- ▶ You have a real-world device to look at? Please reach out to us!

Future Work

- ▶ Symmetric cryptography based on shift registers
- ▶ Side-channel protected implementation
- ▶ Actually finding unknown cryptography
- ▶ **You have a real-world device to look at? Please reach out to us!**

Future Work

- ▶ Symmetric cryptography based on shift registers
- ▶ Side-channel protected implementation
- ▶ Actually finding unknown cryptography
- ▶ **You have a real-world device to look at? Please reach out to us!**



Thank You!