General Framework for Evaluating LWC Finalists in Terms of Resistance to Side-Channel Attacks

Jens-Peter Kaps, Kris Gaj, Abubakr Abdulgadir, and Kamyar Mohajerani

Fifth NIST Lightweight Cryptography Workshop 2022

https://cryptography.gmu.edu

GEORGE MASON UNIVERSITY

Acknowledgments

- This work is partially supported by the Department of Commerce (NIST) Grant no. 70NANB18H219

Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Authors

Jens-Peter Kaps

GMU

Kris Gaj

GMU

Kamyar Mohajerani

GMU

Bakry Abdulgadir

PQSecure LLC

Motivation for General Framework

- No single group is likely to have resources and expertise to develop and evaluate SCA-protected implementations of all 10 LWC finalists.
- Self-evaluation by developers may be insufficient and/or error-prone.
- Collective responsibility of the cryptographic engineering community to contribute to the evaluation process and make it as transparent and fair as possible.
- Contributions by multiple groups will make:
  - each group's workload more manageable
  - coverage of implementation platforms more complete
  - results more credible

Benefits for the Cryptographic Community

- Choosing the right algorithm can save the community countless man-hours
- Revealing and highlighting implementation and evaluation methods that rarely get fully disclosed and published
  - Most implementations open-source
  - Most evaluations transparent and reproducible
- Progress in automated generation of protected implementations
- The developed protected implementations can become benchmarks for new attacks and leakage assessment methods

Benefits for Contributing Labs

- Recognition by the cryptographic community that may translate to new collaboration, funding, and publication opportunities
- Topics for Master’s Theses or parts of Ph.D. theses
- Student participants may be rewarded with attractive job offers
- Source of excellent topics for individualized projects in various academic courses
- Projects' high visibility may help to attract investors and collaborators for possible commercialization

General Approach

1. Call for Side-Channel Security Validation Labs
2. Call for Protected Hardware Implementations, targeting low-cost modern FPGAs
3. Call for Protected Software Implementations, targeting low-cost modern embedded processors

- Draft versions announced on lwc-forum in mid-December 2021
- Final versions published in mid-January 2022
- Deadlines in mid-March 2022

Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Side-Channel Security Evaluation Labs

- We called for groups capable and willing to serve as side-channel security evaluation labs to identify their capabilities and contribute to the evaluation process
- Submitters were expected to have access to the equipment used for side-channel leakage assessment and/or attacks, experience, and human resources necessary to perform security analysis
- Suggested devices used for evaluating hardware implementations:
  - Artix-7 and Spartan-7 from Xilinx
  - Cyclone 10 LP from Intel, and
  - ECP5 from Lattice Semiconductor
- Suggested embedded processors used for evaluating software implementations:
  - ARM Cortex-M4F
  - RISC-V (e.g., RV32IMAC)
  - Microchip 8-bit AVR
  - TI MSP430
- A particular lab could specialize in evaluating only hardware implementations, only software implementations, or both

Suggested Deliverables

1. Equipment and Software Used
2. Supported Leakage Assessment Methods
3. Supported Attacks
4. Ability to generate and publish raw measurements to be analyzed by other groups
5. Support for side-channel analysis as service, with the feedback provided to designers of protected implementations during the development process
6. Short description of the personnel and its qualifications
7. Intended period of the lab operation
8. Contact information

Leakage Assessment Methods Supported By At Least One Lab

- Welch's t-test a.k.a. TVLA (Test Vector Leakage Assessment)
- Pearson's χ^2-test
- NICV: Normalized Inter-Class Variance for Detection of Side-Channel Leakage
- DL-LA: Deep Learning Leakage Assessment
- Tests specified in ISO/IEC 17825:2016 Testing Methods for the Mitigation of Non-invasive Attack Classes Against Cryptographic Modules
- SILVER – simulation-based probing security leakage-detection tools

Attacks Supported By At Least One Lab

- Power-Based
  - Simple Power Analysis (SPA)
  - Differential Power Analysis (DPA)
  - Correlation Power Analysis (CPA)
  - Template Attacks (TA)
  - Mutual Information Analysis (MIA)
  - Linear Regression Attack (LRA)
- Electromagnetic Radiation-Based
  - Simple Electromagnetic Analysis (SEMA)
  - Differential Electromagnetic Analysis (DEMA)
  - Correlation Electromagnetic Analysis (CEMA)
- Fault-Based
  - Simple Fault Analysis (SFA)
  - Differential Fault Analysis (DFA)
  - Fault Sensitivity Attack (FSA)
  - Differential Fault Intensity Analysis (DFIA)
  - Fault Behavior Analysis (FBA)

Side-Channel Security Evaluation Labs for HW

Team	Evaluation Platform	Target FPGA Family	Target Boards	Leakage Assessment Methods	Attacks
IAIK, Graz University of Technology, Austria	NewAE ChipWhisperer	Artix-7	NewAE CW305	t-test	
Telecom Paris, France		Spartan-6, Virtex-5, Virtex-II Pro, Stratix II	SASEBO-W, -GII,-G, -R, -B	NICV, t-test, chi-squared test, DL-LA	SPA, DPA, CPA, MIA, TA, LRA, etc.; SEMA, DEMA, CEMA, etc.; SFA, DFA, FSA, DFIA, FBA, etc.
Cryptology and Computer Security Laboratory, Shanghai Jiao Tong University, China	Riscure Inspector, NewAE ChipWhisperer, SAKURA	Kintex-7, Spartan-6,	SAKURA-G, SAKURA-X	t-test, chi-squared test, DL-LA	CPA, TA, MIA, DL-based methods

Side-Channel Security Evaluation Labs for HW

Team	Evaluation Platform	Target FPGA Family	Target Boards	Leakage Assessment Methods	Attacks
Hardware Security and Cryptographic Processor Lab,Tsinghua University, China	SAKURA	Kintex-7, Spartan-6	SAKURA-G, SAKURA-X	NICV, t-test, chi-squared test	SPA, DPA, CPA, MIA, TA, LRA, etc.
CESCA Lab, Radboud University, Netherlands	Riscure Inspector, NewAE ChipWhisperer	Artix-7, Spartan-6	NewAE CW305, SAKURA-G	t-test, chi-squared test, DL-LA	SPA, DPA, CPA, TA; DEMA; DFA, FI attacks
Secure-IC, France	Secure-IC Analyzr, SAKURA	Spartan-6	SAKURA-G	ISO/IEC 17825:2016	
CERG, George Mason University, USA	FOBOS3	Artix-7	NewAE CW305	t-test	
Ruhr-University Bochum, Germany	SILVER and other simulation-based probing security leakage-detection tools				

Side-Channel Security Evaluation Labs for SW

Team	Evaluation Platform	Target Processor	Leakage Assessment Methods	Attacks
IAIK, Graz University of Technology, Austria	NewAE ChipWhisperer	ARM Cortex-M4F	t-test	
Telecom Paris, France	NewAE ChipWhisperer	ARM Cortex-M0, ARM Cortex-M4F, ATxmega128D4	NICV, t-test, chi-squared test, DL-LA	SPA, DPA, CPA, MIA, TA, LRA, etc.; SEMA, DEMA, CEMA, etc.; SFA, DFA, FSA, DFIA, FBA, etc.
Laboratory for Safe and Secure Systems, OTH Regensburg, Germany		ATmega328P, ARM Cortex-M3, RISC-V GD32VF103CBT6, ARM Cortex-M7, Tensilica Xtensa LX6 (2 out of 5)	t-test	
Cryptology and Computer Security Laboratory, Shanghai Jiao Tong University, China	Riscure Inspector, NewAE ChipWhisperer	ARM Cortex-M4F, ATxmega128D4, ATmega128A	t-test, chi-squared test, DL-LA	CPA, TA, MIA, DL-based methods

Side-Channel Security Evaluation Labs for SW

Team	Evaluation Platform	Target Processor	Leakage Assessment Methods	Attacks
Hardware Security and Cryptographic Processor Lab,Tsinghua University, China		ARM Cortex-M4F, ARM Cortex-M3	NICV, t-test, chi-squared test	SPA, DPA, CPA, MIA, TA, LRA, etc.
CESCA Lab, Radboud University, Netherlands	Riscure Inspector, NewAE ChipWhisperer	ARM Cortex-M4F, ATxmega128D4	t-test, chi-squared test, DL-LA	SPA, DPA, CPA, TA; DEMA; DFA, FI attacks

Overview

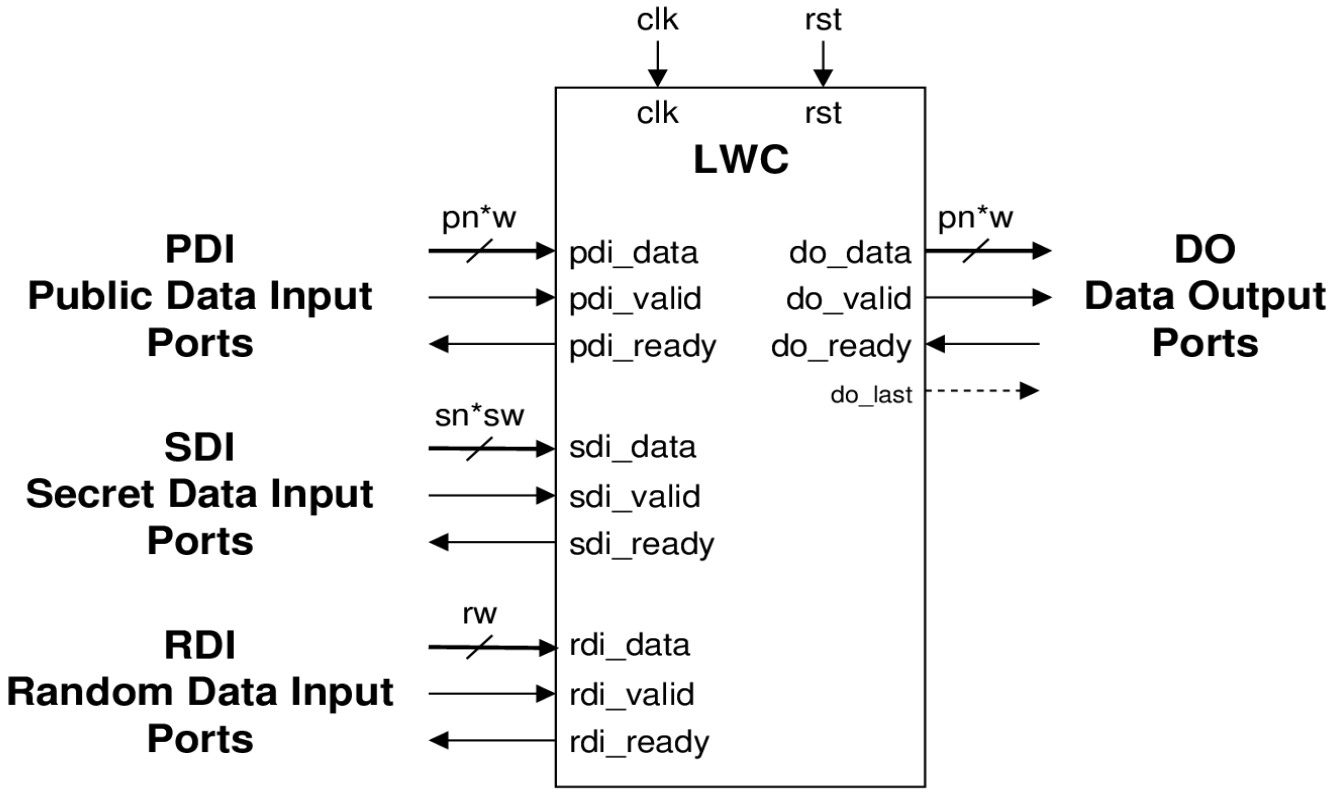
- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Protected Hardware Implementations

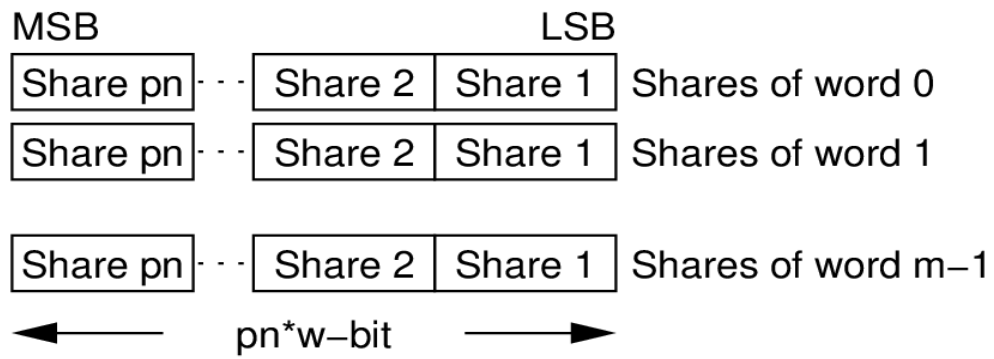
- Submitted designs should demonstrate strong resistance against side-channel attacks when implemented on low-cost modern FPGAs
- A potential for porting the designs to ASIC (Application-Specific Integrated Circuit) technology
- All submitted implementations investigated by one or more Side-Channel Security Evaluation Labs

Protected HW Implementations Submission Requirements

- Compliant with the Extended LWC Hardware API, v1.1 (January 2022) or later
- Interface



- Pre-shared Data



Features Supporting Leakage Assessment Methods

Feature	Proposed Approach
Division of inputs into shares	outside of LWC
Combining shares into outputs	outside of LWC
Passing leakage detection test dependent on	side-channel countermeasures
Random Data Input ports	yes
Overhead of DRBG in terms of area, power, energy	excluded
Sharing DRBG with other units	easy
Changing the source of random bits	easy

Protected Hardware Implementations – Variants

- Variants = Different versions of the design that correspond to
  - different algorithms of the same family
  - different sizes of keys, nonces, tags, etc.
  - different parameters of the interface, such as w and sw
  - different hardware architectures (e.g., basic iterative, unrolled, folded, pipelined, etc.),
  - different protection methods against side-channel attacks,
  - different orders of protection against side-channel attacks

Protected HW Implementations Suggested Deliverables

- For each variant:
  - LICENSE.txt - licensing and copyright information
  - <variant_name>.toml - information characterizing a particular variant, encoded using TOML (Tom’s Obvious Minimal Language)
  - src_rtl - synthesizable source files
  - src_tb - testbenches developed or modified by a given submitter
  - KAT - known-answer tests
  - cref - reference software implementation
  - docs - additional documentation

Protected HW Implementations Documentation

- Protection Methods
  1. Manual design:
     - DOM: Domain-Oriented Masking
     - TI: Threshold Implementation
     - Mode-level robustness against physical attacks
  2. Semi-automated design:
     - HPC2: Hardware Private Circuits 2 (design supported w/ AGEMA)
- Results of the Preliminary Security Evaluation, including at least
  (a) Attack/leakage assessment type
  (b) Number of traces used
  (c) Experimental setup
  (d) Attack/leakage assessment characteristics
  (e) Attack-specific characteristics
  (f) Documentation of results

Protected HW Implementations Submitted To Date

LWC Candidates	Team	No. of variants	Protection Method	Protection Order	Availability	License
ISAP	IAIK, Graz University of Technology, Austria	7	mode-level DPA resistance	N/A	GitHub	GPL-3.0
Ascon, Elephant, GIFT-COFB, ISAP, PHOTON-Beetle, Romulus, SPARKLE, TinyJAMBU, Xoodyak	Ruhr-University Bochum, Germany	Ascon, Xoodyak: 6 Others: 3	HPC2	1, 2, 3	GitHub	GPL-3.0
Elephant, TinyJAMBU, Xoodyak,	CERG, George Mason University, USA	1	DOM	1	TinyJAMBU: GitHub; Elephant, Xoodyak: Per request	GPL-3.0
Ascon	IAIK, Graz University of Technology, Austria	1	DOM	1, 2	Per request	GPL-3.0
Xoodyak	Hardware Security and Cryptographic Processor Lab, Tsinghua University, Beijing, China	2	DOM, TI	1	GitHub	GPL-3.0

Missing Protected Hardware Implementations

- Missing semi-automatically generated implementations:
  - Grain128-AEAD
- Missing manually-designed hardware implementations:
  - GIFT-COFB
  - Grain128-AEAD
  - Photon-Beetle
  - Romulus
  - Sparkle

Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Protected Software Implementations

- Submitted designs should demonstrate strong resistance against side-channel attacks when executed on low-cost modern embedded processors
- The code can contain assembly language instructions specific to a given Instruction Set Architecture (ISA)
- All submitted implementations investigated by one or more Side-Channel Security Evaluation Labs

Protected SW Implementations Submission Requirements

- Compliant with the NIST API defined in Submission Requirements and Evaluation Criteria for the Lightweight Cryptography Standardization Process, published in August 2018
- nsec set to NULL
- C99 standard C suitable for compilation, linkage, and assembly using standard tooling (e.g., GCC) for the target architecture(s)
- Architecture specific optimizations (e.g., assembly language) permitted
- No dependance on any external headers or libraries, including cryptographic libraries (e.g., OpenSSL), outside of the C99 standard
- Exception:
  - randombytes.h header from SUPERCOP

Features Supporting Leakage Assessment Methods

- At least one variant of the protected implementation should support Welch's t-test
- Goal: no spurious correlation from sharing and un-sharing operations
- Method: division of the protected implementation into three functions:
  - `generate_shares_encrypt()`, `crypto_aead_encrypt_shared()`, and `combine_shares_encrypt()`, for encryption, and
  - `generate_shares_decrypt()`, `crypto_aead_decrypt_shared()`, and `combine_shares_decrypt()`, for decryption
- Only `crypto_aead_encrypt_shared()` and `crypto_aead_decrypt_shared()` used for leakage assessment

Protected SW Implementations Submitted To Date

LWC Candidates	Team	No. of variants	Protection Method	Protection Order	Availability	License
ISAP	ISAP Team	5	mode-level DPA resistance	N/A	GitHub	CCO-1.0
Ascon	Ascon Team	6	Masking, share rotation, mode-level security	2	GitHub	CCO-1.0
GIFT-COFB	Alexandre Adomnicai	1	Boolean masking	1	GitHub	CCO-1.0
Romulus	Alexandre Adomnicai	3	Boolean masking	1	GitHub	CCO-1.0
Xoodyak	HW Security and Crypto-graphic Processor Lab, Tsinghua University, Beijing, China	1	ISW Scheme	1	GitHub	CCO-1.0

CCO-1.0: Creative Commons version 1

Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Matching Implementations with Evaluation Labs
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Matching Implementations with Evaluation Labs

- Our Team has already proposed assignments aimed at
  - maximum coverage of all implementations submitted to date using the maximum number of
    - leakage assessment methods
    - attacks
    - evaluation platforms
  - similar evaluations allowed but minimized
- No explicit commitments from all labs yet
- After receiving commitments GMU will maintain a public table: labs / implementations
- Implementers can requests the labs to keep the code, but not the results, confidential

Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Matching Implementations with Evaluation Labs
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Benchmarking Protected Implementations

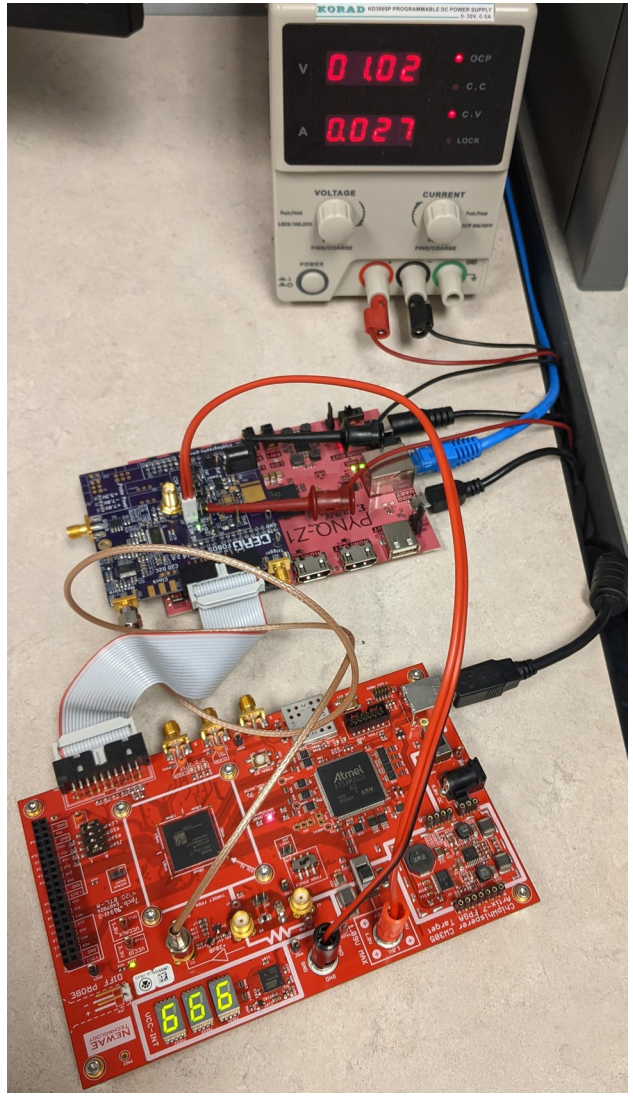
- FPGA Benchmarking by the GMU Team
  - Artix-7 and Spartan-7 from Xilinx
  - Cyclone 10 LP from Intel, and
  - ECP5 from Lattice Semiconductor
- Volunteers to support ASIC and Software Benchmarking very welcome!

Benchmarking of Protected Hardware Implementations

- Ranking reported only for the implementations with the same or very similar protection method and order
- Metrics
  - Resource utilization
  - Number of LUTs (LEs for Cyclone 10LP) and flip-flops, assuming no use of embedded memories (such as BRAMs), DSP units, and embedded multipliers
  - Throughput for multiple sizes of inputs
    - 16 B, 64 B, 1536 B, long
- Power
- Energy per bit
- Maximum number of fresh random bits per clock cycle
- Total number of random bits per each byte of AD and plaintext

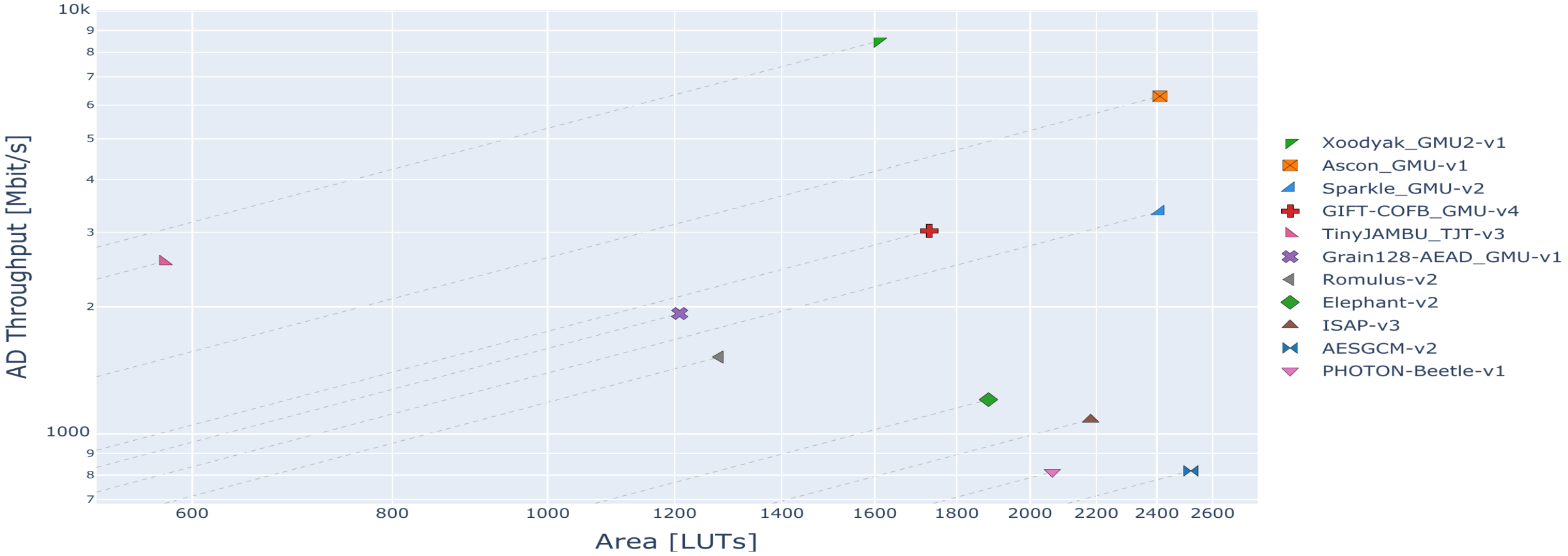
Evaluation using FOBOS 3

- Flexible Open-source workBench fOr Side-channel analysis (FOBOS)
- FOBOS Software:
  - Runs in JupyterLab, controlled via web browser and Jupyter notebooks.
  - Welch's T-test, Power Benchmarking
- Control:
  - PYNQ Z1 – Zynq 7 SoC with dual core ARM processor
  - FOBOS Shield:
    - OpenADC, 105 MS/s, 40 MHz bandwidth, used for side-channel leakage evaluation.
    - DUT Power Supply (not populated due to chip shortage)
    - Voltage measurement (V_{Core} of FPGA)
    - Current Shunt Monitors to measure current consumption of DUT.
- Device under Test (DUT):
  - NewAE CW305 with Xilinx Artix-7
  - Optional: Digilent Nexys-3 with Xilinx Spartan-6
  - Other targets under development



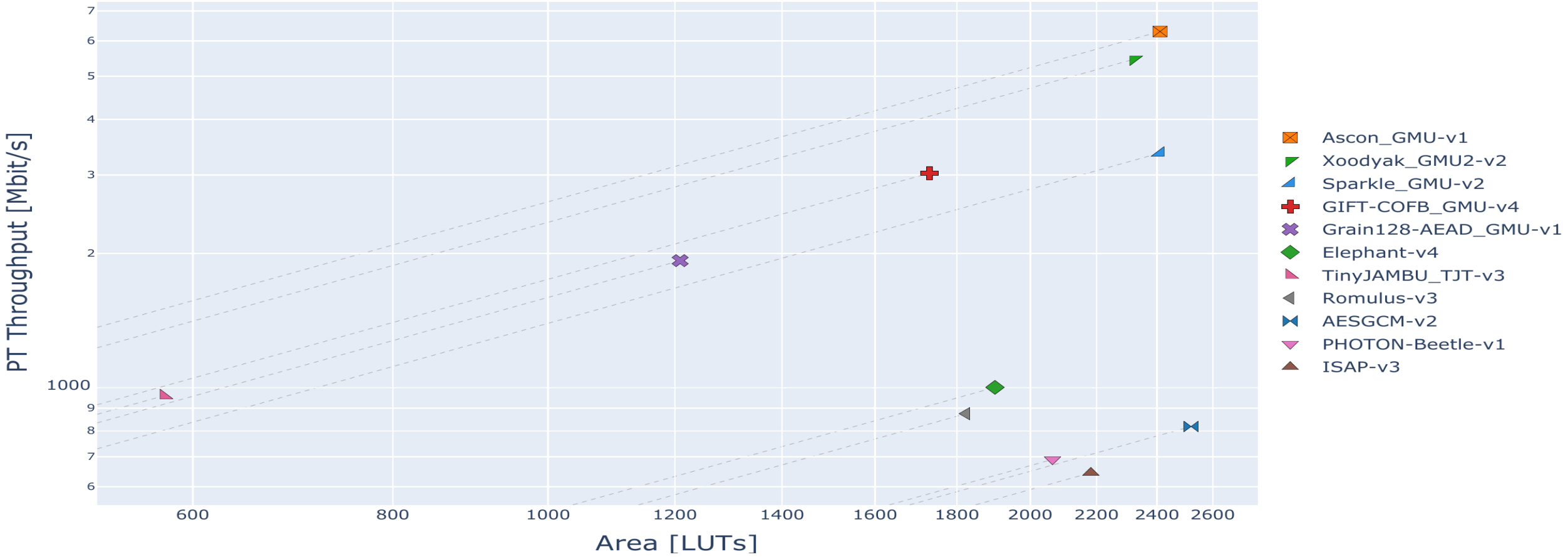
AD Throughput vs Area

- Xilinx(AMD) xc7a12tcsg325-3 Artix-7 FPGA
- Unprotected Designs



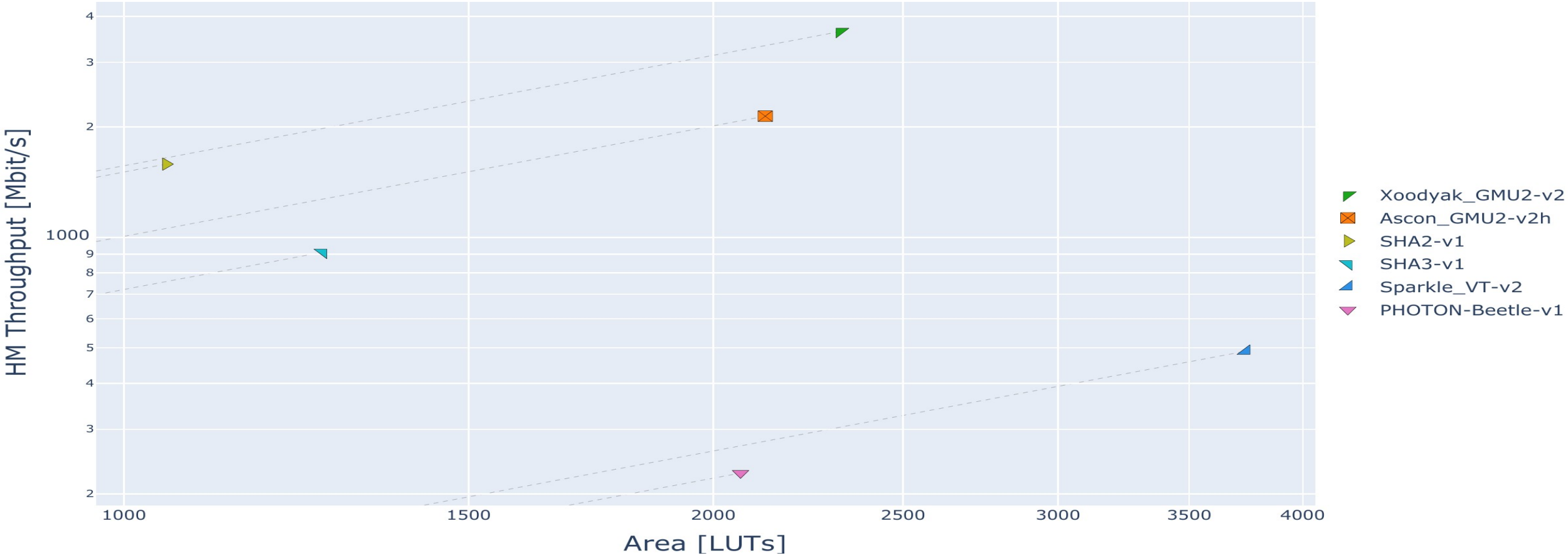
Encryption Throughput vs Area

- Xilinx(AMD) xc7a12tcsg325-3 Artix-7 FPGA
- Unprotected Designs



Hash Throughput vs Area

- Xilinx(AMD) xc7a12tcsg325-3 Artix-7 FPGA
- Unprotected Designs



Overview

- Introduction
- Side-Channel Security Evaluation Labs
- Protected Hardware Implementations
- Protected Software Implementations
- Matching Implementations with Evaluation Labs
- Benchmarking Protected Implementations
- Proposed Revised Timeline

Proposed Revised Timeline

- Lab Commitments (subject to future revisions): May 20, 2022
- Lab Reports: July 20, 2022
- GMU FPGA Benchmarking Report: August 20, 2022

- Details at https://cryptography.gmu.edu/athena/index.php?id=LWC
- New Protected Implementations Welcome Throughout the Entire Process