Applications and Standardization of Lightweight Cryptography

Meltem Sönmez Turan

National Institute of Standards and Technology, Gaithersburg, MD, USA

SAC Summer School (S^3), 2018, Calgary, Canada

August 13, 2018

Outline

Session I

- Brief intro to NIST's Crypto Standards
- NIST's Lightweight Crypto Project
- Real World Applications of Lightweight Crypto

Session II

- Standardization of Lightweight Cryptography
- NIST's Call for Submission and Next Steps

NIST’s Cryptography Standards

National Institute of Standards and Technology

- Non-regulatory federal agency within U.S. Department of Commerce.
- Founded in 1901, known as the National Bureau of Standards (NBS) prior to 1988.
- Headquarters in Gaithersburg, Maryland, and laboratories in Boulder, Colorado.
- Employs around 3,000 employees, and 2,700 associates.

National Institute of Standards and Technology

- Non-regulatory federal agency within U.S. Department of Commerce.
- Founded in 1901, known as the National Bureau of Standards (NBS) prior to 1988.
- Headquarters in Gaithersburg, Maryland, and laboratories in Boulder, Colorado.
- Employs around 3,000 employees, and 2,700 associates.

NIST's Mission

to promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology in ways that enhance economic security and improve our quality of life.

NIST Organization Chart

Seven Laboratory Programs

- Center for Nanoscale Science and Technology
- Communications Technology Lab.
- Engineering Lab.
- **Information Technology Lab.**
- Material Measurement Lab.
- NIST Center for Neutron Research
- Physical Measurement Lab.

Information Technology Lab.

- Advanced Network Technologies
- Applied and Computational Mathematics
- Applied Cybersecurity
- **Computer Security**
- Information Access
- Software and Systems
- Statistical Engineering

Computer Security Division

- Cryptographic Technology
- Secure Systems and Applications
- Security Outreach and Integration
- Security Components and Mechanisms
- Security Test, Validation and Measurements

Computer Security Division (CSD)

Conducts research, development and outreach necessary to provide standards and guidelines, mechanisms, tools, metrics and practices to protect nation's information and information systems.

Who does CSD work with?

- Government: Core user community.
- Industry: On adoption of cryptographic algorithms, feedback mechanism on standards.
- Academic Researchers: Development of new algorithms/modes schemes to advance the science of cryptography.
- Standards Developing Organizations: Adoption and development of new standards.

Computer Security Division (CSD)

Conducts research, development and outreach necessary to provide standards and guidelines, mechanisms, tools, metrics and practices to protect nation's information and information systems.

Who does CSD work with?

- Government: Core user community.
- Industry: On adoption of cryptographic algorithms, feedback mechanism on standards.
- Academic Researchers: Development of new algorithms/modes schemes to advance the science of cryptography.
- Standards Developing Organizations: Adoption and development of new standards.

How can you work with NIST?

- NRC postdoctoral researcher, Foreign guest researcher, Internship, NIST ITL grant programs

CSD Publications

- Federal Information Processing Standards (FIPS): Specify approved crypto standards.

CSD Publications

- Federal Information Processing Standards (FIPS): Specify approved crypto standards.
- NIST Special Publications (SPs): Guidelines, technical specifications, recommendations and reference materials, including multiple sub-series:
  - SP 800 - Computer security
  - SP 1800 - Cybersecurity practice guides
  - SP 500 - Information technology (relevant documents)

CSD Publications

- Federal Information Processing Standards (FIPS): Specify approved crypto standards.
- NIST Special Publications (SPs): Guidelines, technical specifications, recommendations and reference materials, including multiple sub-series:
    - SP 800 - Computer security
    - SP 1800 - Cybersecurity practice guides
    - SP 500 - Information technology (relevant documents)
- NIST Internal or Interagency Reports (NISTIR):Reports of research findings, including background information for FIPS and SPs.

CSD Publications

- Federal Information Processing Standards (FIPS): Specify approved crypto standards.
- NIST Special Publications (SPs): Guidelines, technical specifications, recommendations and reference materials, including multiple sub-series:
  - SP 800 - Computer security
  - SP 1800 - Cybersecurity practice guides
  - SP 500 - Information technology (relevant documents)
- NIST Internal or Interagency Reports (NISTIR):Reports of research findings, including background information for FIPS and SPs.
- NIST Information Technology Laboratory (ITL) Bulletins: Monthly overviews of NIST's security and privacy publications, programs and projects

to subscribe for publication announcements:

`https://public.govdelivery.com/accounts/USNIST/subscriber/new?qsp=USNIST_3`

How does CSD develop standards?

- International “competitions”: Engage community through an open competition (e.g., AES, SHA-3, PQC, Lightweight Crypto).
- Adoption of existing standards: Collaboration with accredited standards organizations (e.g., RSA, HMAC).
- Open call for proposals: Ongoing open invitation (e.g., modes of operations).
- Development of new algorithms: if no suitable standard exists (e.g., DRBGs).

Public Review and Outreach

- FIPS, SP and IR documents are published as drafts with a 30 to 90-day comment period to solicit public feedback.
- Public workshops and forums.
- Involvement in Standards Development Organizations.
- Engaging community at industry/academic conferences, meetings and other events.

NIST-approved Crypto Standards (NIST SP 800 131A)(1/7)

Encryption/Decryption Using Block Ciphers

- AES (FIPS 197) with three key sizes 128, 192 and 256 bits, acceptable using approved modes specified in SP 800-38 series.
- Triple DES (SP 800-67)
  - Two-key TDEA Encryption: disallowed, Decryption: legacy use.
  - Three-key TDEA Encryption: disallowed after 2023, Decryption: legacy use.
- SKIPJACK (FIPS 185) Encryption: disallowed, Decryption: legacy use.

NIST-approved Crypto Standards (NIST SP 800 131A)(2/7)

Modes of Operation (SP 800 38 series)

- Confidentiality modes (ECB, CBC, CFB, OFB),

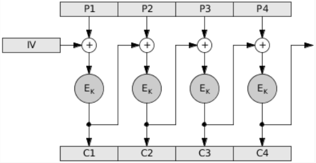


Figure 1: CBC mode

- Confidentiality mode for storage devices (XTS-AES),
- Format preserving encryption modes (FF1, FF3) (e.g., to encrypt 16-digit credit card number, and the ciphertext is another 16-bit number.

NIST-approved Crypto Standards (NIST SP 800 131A)(3/7)

Authenticated Encryption (simultaneously provides confidentiality, integrity, and authenticity of the data.)

- CCM mode for AES, CCM combines the counter mode for confidentiality with the cipher block chaining technique for authentication. (SP 800-38C)
- Galois/Counter Mode (GCM) for AES. GCM combines the counter mode for confidentiality, with an authentication mechanism that is based on a universal hash function. (SP 800-38D)

NIST-approved Crypto Standards (NIST SP 800 131A)(4/7)

Hash Functions (maps input of arbitrary size to a fixed size.)

- SHA-1
  - Disallowed for digital signature generation
  - Legacy use for digital signature verification
  - Acceptable to use in non-digital-signature applications

NIST-approved Crypto Standards (NIST SP 800 131A)(4/7)

Hash Functions (maps input of arbitrary size to a fixed size.)

- SHA-1
  - Disallowed for digital signature generation
  - Legacy use for digital signature verification
  - Acceptable to use in non-digital-signature applications
- SHA-2 family including SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, SHA-512/256 (FIPS 180): acceptable for hash function applications.

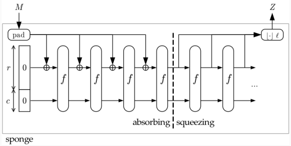
NIST-approved Crypto Standards (NIST SP 800 131A)(4/7)

Hash Functions (maps input of arbitrary size to a fixed size.)

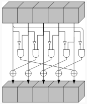
- SHA-1
  - Disallowed for digital signature generation
  - Legacy use for digital signature verification
  - Acceptable to use in non-digital-signature applications
- SHA-2 family including SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, SHA-512/256 (FIPS 180): acceptable for hash function applications.
- SHA-3 family including SHA3-224, SHA3-256, SHA3-382 and SHA3-512(FIPS 202), TupleHash and ParallelHash (SP 800-185): acceptable for hash function applications.
- Two extendable output functions (XOFs): SHAKE128 and SHAKE256: not approved hash algorithms, subject to additional security considerations.

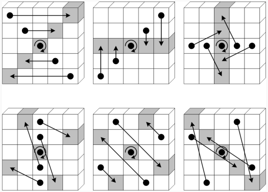
NIST-approved Crypto Standards (NIST SP 800 131A)(4/7)

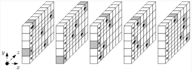
SHA-3 [1] Sponge Construction:

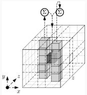


Underlying functions:









[1] https://keccak.team/figures.html

NIST-approved Crypto Standards (NIST SP 800 131A)(5/7)

Message Authentication Codes (to protect the integrity of the message and the authenticity of the source of the message)

- Based on block ciphers and hash functions
  - CMAC (SP 800 38B) based on AES or TDEA
  - GMAC (SP 800 38D) based on AES (GCM with no message to be encrypted).
- Based on hash functions
  - HMAC (FIPS 198-1) with any approved function, two calls to the underlying hash function.
  - KMAC (SP 800-185) based on Keccak

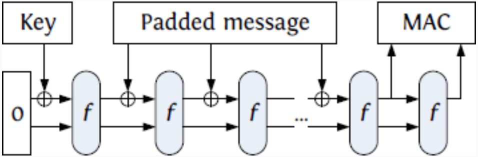


Figure 2: Generating a MAC using permutation *f*

NIST-approved Crypto Standards (NIST SP 800 131A)(6/7)

Random Bit Generation (used extensively in cryptography, to generate keys, nonce, IV, challenge etc.)

- Hash_DRBG and HMAC_DRBG: acceptable (SP 800 90A)
- CTR_DRBG with three-key TDEA: disallowed after 2023
- CTR_DRBG with AES: acceptable
- DUAL_EC_DRBG: disallowed

NIST-approved Crypto Standards (NIST SP 800 131A)(6/7)

Random Bit Generation (used extensively in cryptography, to generate keys, nonce, IV, challenge etc.)

- Hash_DRBG and HMAC_DRBG: acceptable (SP 800 90A)
- CTR_DRBG with three-key TDEA: disallowed after 2023
- CTR_DRBG with AES: acceptable
- DUAL_EC_DRBG: disallowed

Stream Ciphers

- No approved dedicated stream ciphers, except the stream cipher modes of block ciphers (e.g., counter mode, OFB)

NIST-approved Crypto Standards (NIST SP 800 131A)(7/7)

Other standards on

- Digital Signature Generation
- Key Agreement using Diffie-Hellman and MQV
- Key Agreement and Key Transport using RSA
- Key Wrapping
- Key Derivation from Cryptographic Keys

Some of the Ongoing Projects at NIST (1/3)

Post-Quantum Cryptography (Quantum Resistant cryptography)

- Substantial amount of research on quantum computers,
- If large-scale quantum computers are ever built, they will be able to break many of the public-key cryptosystems currently in use.
- Goal: Develop cryptographic systems that are secure against both quantum and classical computers, and can operate with existing communications protocols and networks.
- Scope: Digital signatures, public-key encryption and KEMs
- 64 submissions
  - 26 lattice, 19 code based, 9 multivariate, 10 other
  - 19 signatures and 45 KEM/encryption
- Second NIST PQC workshop in August 2019.

Some of the Ongoing Projects at NIST (2/3)

Threshold Cryptography

- In order to decrypt an encrypted message or to sign a message, several parties (k out of n) cooperate in the decryption or signature protocol.
- Draft NISTIR 8214: To understand the challenges and opportunities offered by threshold cryptography.
  - Public comment period until October 22, 2018.
- Upcoming NIST workshop on Threshold cryptography, March 11-12, 2019.

Some of the Ongoing Projects at NIST (2/3)

Threshold Cryptography

- In order to decrypt an encrypted message or to sign a message, several parties (k out of n) cooperate in the decryption or signature protocol.
- Draft NISTIR 8214: To understand the challenges and opportunities offered by threshold cryptography.
  - Public comment period until October 22, 2018.
- Upcoming NIST workshop on Threshold cryptography, March 11-12, 2019.

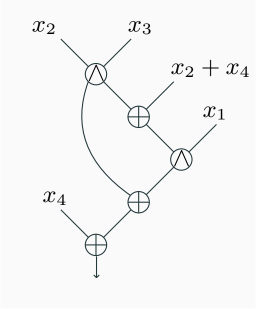
Random Number Generation

- NIST SP 800 90 Series include deterministic random bit generators (SP 800 90A), design and validation of entropy sources (SP 800 90B), and their combination (SP 800 90C)
- NIST Randomness Beacon: A source of public randomness that uses two independent hardware entropy sources. Outputs full-entropy bit-strings in blocks of 512 bits in every 60 seconds.

Some of the Ongoing Projects at NIST (3/3)

Circuit Complexity

- Efficient implementations of Boolean functions and Sboxes
- Smaller gate complexity, smaller circuit depth, smaller number of AND gates
- Circuits for the AES Sbox using 32 AND gates, 83 XOR/XNOR gates and depth 28



NIST's Lightweight Crypto Project

NIST’s Lightweight Crypto Project (1/3)

Motivation

- Shift from general-purpose computers to dedicated resource-constrained (with limited processing and storage capabilities) devices such as RFID tags, sensor networks, IoT devices

NIST's Lightweight Crypto Project (1/3)

Motivation

- Shift from general-purpose computers to dedicated resource-constrained (with limited processing and storage capabilities) devices such as RFID tags, sensor networks, IoT devices
- New applications (e.g., home automation, smart city technologies, digital assistants, healthcare) that collect, store and process private information (e.g., sleep patterns, heart beat, exercise routines, medical information, location)

NIST's Lightweight Crypto Project (1/3)

Motivation

- Shift from general-purpose computers to dedicated resource-constrained (with limited processing and storage capabilities) devices such as RFID tags, sensor networks, IoT devices
- New applications (e.g., home automation, smart city technologies, digital assistants, healthcare) that collect, store and process private information (e.g., sleep patterns, heart beat, exercise routines, medical information, location)
- Lack of crypto standards that are suitable for constrained devices.

NIST’s Lightweight Crypto Project (2/3)

Scope:

- All cryptographic primitives and modes that are needed in constrained environments.
- Initial Focus: Symmetric Cryptography.

NIST’s Lightweight Crypto Project (3/3)

Goal

- Understanding the growing industry need for lightweight cryptography.
- Understanding where the performance of the current NIST-approved cryptography is unacceptable.
- Developing new guidelines, recommendations and standards for the use of lightweight cryptography.

What is Lightweight Cryptography?

- Subfield of cryptography that aims to provide crypto solutions tailored to constrained environments.

What is Lightweight Cryptography?

- Subfield of cryptography that aims to provide crypto solutions tailored to constrained environments.
- May have different security properties than for general use.

What is Lightweight Cryptography?

- Subfield of cryptography that aims to provide crypto solutions tailored to constrained environments.
- May have different security properties than for general use.
- Not weak crypto, must be sufficient for the target application.

What is Lightweight Cryptography?

- Subfield of cryptography that aims to provide crypto solutions tailored to constrained environments.
- May have different security properties than for general use.
- Not weak crypto, must be sufficient for the target application.
- Not intended to replace conventional crypto standards.

New Trends in Lightweight Crypto Design

New design may benefit from the developments of the state-of-art after the design of AES

- Many iterations of simple rounds, simple operations (e.g., 4x4 Sboxes, bit permutations).
- Smaller block/key sizes, smaller security margins by design.
- Simpler key schedules.

Modifications of well-analyzed designs: e.g., DESL, DESXL.

Old interesting algorithms: e.g., RC5, TEA, XTEA.

New dedicated algorithms: e.g., Skinny, Pride, Gimli, Simon, Speck, Simeck, Present, etc.

Weight of Cryptographic Primitives

Weight of an algorithm is a property of its implementation depending on different metrics of the target platform.

Weight of Cryptographic Primitives

Weight of an algorithm is a property of its implementation depending on different metrics of the target platform.

Hardware applications:

- Area: physical area needed for a circuit that implements the primitive.
- Throughput: amount of input processed per time unit.
- Latency: time it takes to obtain the output of the circuit once the input is set.
- Power consumption: amount of power needed to use the circuit.

Weight of Cryptographic Primitives

Weight of an algorithm is a property of its implementation depending on different metrics of the target platform.

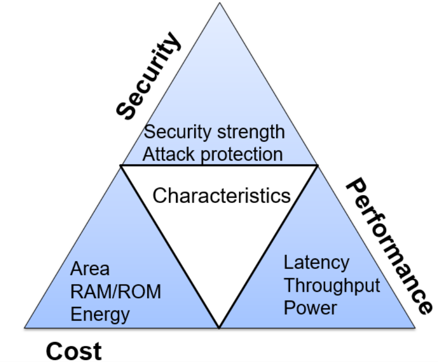
Hardware applications:

- Area: physical area needed for a circuit that implements the primitive.
- Throughput: amount of input processed per time unit.
- Latency: time it takes to obtain the output of the circuit once the input is set.
- Power consumption: amount of power needed to use the circuit.

Software applications:

- RAM: amount of memory used during each evaluation of the algorithm.
- Code size: size of the compiled source code that is executed.
- Throughput: amount of input processed during each clock cycle.

Tradeoff between Security, Performance and Cost



Optimal tradeoff depends on the target technology and application.

Due to the variability of applications/requirements, hard to select a one-size-fits-all algorithm.

Do we need lightweight crypto standards?

NIST Standards in Constrained Environments (1/2)

AES:

- Many optimized implementations of AES in hardware and software
  - Atomic-AES implementation of enc/dec core (Banik et al., 2016)
  - 1947/2090 GEs (8-bit serial implementation) (Mathew et al., 2015)
  - Fast AES-128-CTR on ARM Cortex-M3 with side channel resistance (Schwabe, Stoffelen, 2016)

[2] Ronen et al., IoT Goes Nuclear: Creating a Zigbee Chain Reaction, IEEE Security and Privacy, 2017

NIST Standards in Constrained Environments (1/2)

AES:

- Many optimized implementations of AES in hardware and software
  - Atomic-AES implementation of enc/dec core (Banik et al., 2016)
  - 1947/2090 GEs (8-bit serial implementation) (Mathew et al., 2015)
  - Fast AES-128-CTR on ARM Cortex-M3 with side channel resistance (Schwabe, Stoffelen, 2016)
- Not always feasible to implement, e.g., on RL78 16-bit microcontroller, combined enc/dec implementation is not possible within 512 bytes of ROM and 128 bytes of RAM (Moriai, 2016).

[2] Ronen et al., IoT Goes Nuclear: Creating a Zigbee Chain Reaction, IEEE Security and Privacy, 2017

NIST Standards in Constrained Environments (1/2)

AES:

- Many optimized implementations of AES in hardware and software
  - Atomic-AES implementation of enc/dec core (Banik et al., 2016)
  - 1947/2090 GEs (8-bit serial implementation) (Mathew et al., 2015)
  - Fast AES-128-CTR on ARM Cortex-M3 with side channel resistance (Schwabe, Stoffelen, 2016)
- Not always feasible to implement, e.g., on RL78 16-bit microcontroller, combined enc/dec implementation is not possible within 512 bytes of ROM and 128 bytes of RAM (Moriai, 2016).
- AES is fast on 8-bit microcontrollers, but requires to store the Sbox.

[2] Ronen et al., IoT Goes Nuclear: Creating a Zigbee Chain Reaction, IEEE Security and Privacy, 2017

NIST Standards in Constrained Environments (1/2)

AES:

- Many optimized implementations of AES in hardware and software
  - Atomic-AES implementation of enc/dec core (Banik et al., 2016)
  - 1947/2090 GEs (8-bit serial implementation) (Mathew et al., 2015)
  - Fast AES-128-CTR on ARM Cortex-M3 with side channel resistance (Schwabe, Stoffelen, 2016)
- Not always feasible to implement, e.g., on RL78 16-bit microcontroller, combined enc/dec implementation is not possible within 512 bytes of ROM and 128 bytes of RAM (Moriai, 2016).
- AES is fast on 8-bit microcontrollers, but requires to store the Sbox.
- Devices with hardware acceleration modules may have side channel vulnerabilities
  - e.g., Updates for Phillips light bulbs are authenticated using AES-based MAC with fixed (secret) key, possible to recover key and push malicious updates [2].

[2] Ronen et al., IoT Goes Nuclear: Creating a Zigbee Chain Reaction, IEEE Security and Privacy, 2017

NIST Standards in Constrained Environments (2/2)

SHA-2 and SHA-3 Families

- Large memory requirements, 1600 bits for SHA-3 and 512 bits for SHA-2.
- Lightweight versions of SHA-3 with smaller permutation sizes (200-, 400- and 800-bits) are specified in the FIPS 202, but currently not approved.

NIST Standards in Constrained Environments (2/2)

SHA-2 and SHA-3 Families

- Large memory requirements, 1600 bits for SHA-3 and 512 bits for SHA-2.
- Lightweight versions of SHA-3 with smaller permutation sizes (200-, 400- and 800-bits) are specified in the FIPS 202, but currently not approved.

Do we need lightweight cryptography standards?

- Yes, dedicated algorithms with inherent side channel resistance may provide security and performance advantages over AES.
- Hash functions with smaller internal size, and that can share crypto logic to provide other functionalities are more suitable for constrained devices.

Industry-Specific Standards

- Due to the lack of lightweight primitives, in early 90's many proprietary non-standard algorithms were used.
- Most of them are reverse-engineered or released through leaks, and practically broken.
- Examples: A5/1 and A5/2 in cell phones, DSC in cordless phones, E0 in Bluetooth, Cryptomeria in DVD players, KeeLoq in car locks and Megamos in vehicle immobilizers, PC-1 in Amazon Kindle etc.

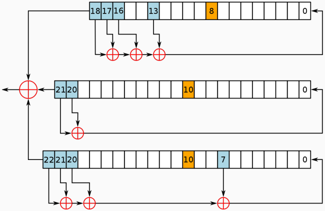


Figure 3: Keystream generation in A5/1

Feedback from Industry

- Two workshops on Lightweight Cryptography at NIST, in July'15 and Oct.'16.
  - Consensus on the need for lightweight crypto standards.
- Published NISTIR 8114 - Report on Lightweight Cryptography (March 2017)
  - Overview of lightweight cryptography: Target devices, performance metrics, lightweight primitives, performance of NIST standards in constrained environments, and other lightweight crypto standards.
  - NIST's lightweight crypto project: Scope, design considerations, profiles, and evaluation process.
  - **Request for feedback:** 22 questions to industry partners to understand their need and target applications.

Request for Feedback

- Application: Target functionality? Typical plaintext, tag sizes?

Request for Feedback

- Application: Target functionality? Typical plaintext, tag sizes?
- Constraints: What limits are imposed on the energy and/or power that is available to the device? Does the device have to respond within a specific time?

Request for Feedback

- Application: Target functionality? Typical plaintext, tag sizes?
- Constraints: What limits are imposed on the energy and/or power that is available to the device? Does the device have to respond within a specific time?
- Cryptographic keys: How are the keys generated? Where are they stored, and for how long? How much data is processed under the same key?

Request for Feedback

- Application: Target functionality? Typical plaintext, tag sizes?
- Constraints: What limits are imposed on the energy and/or power that is available to the device? Does the device have to respond within a specific time?
- Cryptographic keys: How are the keys generated? Where are they stored, and for how long? How much data is processed under the same key?
- Software implementations: Which platforms? Which specific types of processors? Limits on the amount of registers, RAM and ROM? Is it necessary to obfuscate the implementation?

Request for Feedback

- Application: Target functionality? Typical plaintext, tag sizes?
- Constraints: What limits are imposed on the energy and/or power that is available to the device? Does the device have to respond within a specific time?
- Cryptographic keys: How are the keys generated? Where are they stored, and for how long? How much data is processed under the same key?
- Software implementations: Which platforms? Which specific types of processors? Limits on the amount of registers, RAM and ROM? Is it necessary to obfuscate the implementation?
- Hardware implementations: Which types of hardware are considered (FPGA, ASIC, etc.)? Limits on the amount of logic blocks or GEs?

Request for Feedback

- Application: Target functionality? Typical plaintext, tag sizes?
- Constraints: What limits are imposed on the energy and/or power that is available to the device? Does the device have to respond within a specific time?
- Cryptographic keys: How are the keys generated? Where are they stored, and for how long? How much data is processed under the same key?
- Software implementations: Which platforms? Which specific types of processors? Limits on the amount of registers, RAM and ROM? Is it necessary to obfuscate the implementation?
- Hardware implementations: Which types of hardware are considered (FPGA, ASIC, etc.)? Limits on the amount of logic blocks or GEs?
- Side channel resistance: Protection against side-channel or fault attacks required?

Responses to the Questionnaire

- Target applications: Hardware encrypted data storage device, low-cost and low-consumption sensor data transmission, RAIN RFID tags for anti-counterfeiting solutions, IoTs, wearables, low power wireless sensor networks.
- Target functionality: Encryption, AE, hashing, key agreement, sensor/tag authentication.
- Target devices: ARM Cortex-M0 processors, Intel Quark SoC X1021, Atom E3826.
- Side channel resistance: In general, good to have.

Rain RFID for anti-counterfeiting (1/2) [3]

RFID (Radio Frequency Identification) uses EM waves to identify and track objects.

Different types at different frequencies:

- Low Frequency (LF): Short read range of 10 cm, slower read rate, used for access control and livestock tracking.
- High Frequency (HF): Read range between 10 cm and 1 m, used for ticketing and payment systems.
- Ultra-high Frequency (UHF): Read range as long as 12 m, cheaper to manufacture, used for retail inventory, anti-counterfeiting.

RAIN (RAdio frequency IdentificatioN) industry alliance to promote the use of UHF RFID

[3] M. Robshaw and T. Williamson, RAIN RFID and the Internet of Things: Industry Snapshot and Security Needs, NIST Lightweight Cryptography Workshop 2015

Rain RFID for anti-counterfeiting (2/2)

- Counterfeiting can be avoided by authenticating the tags using a challenge-response protocol.
- Most RAIN RFID chips have small amount of user memory (typically $<$ 64 bits, some special chips have $<$2k bits).
- TDEA and AES bring performance compromise to RAIN RFID.
- Algorithms that run on limited hardware and small power supply are desired.

More Applications [4]

Smart Home Appliances

- Many new electrical home appliances with low-end CPUs.
- Software-oriented primitives that consume less CPU time and smaller ROM requirements are desirable.

Automobiles

- In-vehicle, vehicle-to-vehicle and road-to-vehicle communication, driving assistance systems.
- High throughput and low latency are desirable.

Medical Sensors

- Measuring blood pressure, pulse, blood sugar etc.
- Hardware oriented primitives with low power consumption are desirable.

[4] CRYPTREC Cryptographic Technology Guideline, March 2017

Summary - Session I

- Brief intro to NIST's cryptography standards and ongoing projects
- NIST's lightweight cryptography projects, how NIST standards perform in constrained environments
- Lightweight cryptography applications

BREAK

Lightweight Cryptography Standards

Lightweight Cryptography standards/proposals by

- ISO/IEC: International Organization of Standardization (ISO) and the International Electrotechnical Commission (IEC).
- Cryptrec: Cryptography Research and Evaluation Committees set up by the Japanese Government .
- Ecrypt: European Network of Excellence in Cryptology, 4-year European research initiative
- CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness

ISO/IEC

ISO/IEC JTC 1/SC 27 focus on IT Security techniques

- A standardization subcommittee of International Organization of Standardization (ISO) and the International Electrotechnical Commission (IEC)
- Five working groups.
    - Working Group 2 (WG2) is for Cryptography and Security Mechanisms. Lightweight Crypto is one of the projects of WG2. The standards developed under this project are in ISO/IEC 29192 series.

ISO/IEC 29192 Lightweight Cryptography (1/4)

Part 1 - General (2012)

- Defines security, classification and implementation requirements.
- 80-bit security is considered as minimum security strength for lightweight cryptography. At least 112-bit security is recommended for longer periods.

ISO/IEC 29192 Lightweight Cryptography (1/4)

Part 1 - General (2012)

- Defines security, classification and implementation requirements.
- 80-bit security is considered as minimum security strength for lightweight cryptography. At least 112-bit security is recommended for longer periods.

Part 2 - Block ciphers

- PRESENT: block size 64 bits, and key size of 80 or 128 bits (CHES 2007)
- CLEFIA: block size of 128 bits, and key size of 128, 192, 256 bits (FSE 2007)
- Amendment was proposed to include SIMON and SPECK designed by NSA, but they are not approved.

ISO/IEC 29192 Lightweight Cryptography (2/4)

Part 3 - Stream ciphers

- Enocoro: key size of 80 or 128 bits, based on a finite state machine and uses operations defined over the finite field $GF(2^4)$ and $GF(2^8)$.
- Trivium: key size of 80 bits, three nonlinear feedback registers, 288 bits of internal size.

ISO/IEC 29192 Lightweight Cryptography (2/4)

Part 3 - Stream ciphers

- Enocoro: key size of 80 or 128 bits, based on a finite state machine and uses operations defined over the finite field $GF(2^4)$ and $GF(2^8)$.
- Trivium: key size of 80 bits, three nonlinear feedback registers, 288 bits of internal size.

Part 4 - Asymmetric techniques

- Three mechanisms: cryptoGPS, authenticated lightweight key exchange (ALIKE), and ID-based signature IBS
- Amendment 1: includes an Elliptic Curve based authentication scheme ELLI, designed for RFID reader to authenticate RFID tag.

ISO/IEC 29192 Lightweight Cryptography (3/4)

Part 5 - Hash functions

- Photon with permutation sizes 100, 144, 196, 256 and 288 bits and hash output lengths 80, 128, 160, 224 and 256 respectively.
- Spongent with permutation sizes of 88, 136, 176, 240 and 272 bits and hash output lengths 88, 128, 160, 224, and 256 respectively (CHES 2011)
- Lesamnta-LW with permutation size 384 bits and hash output length 256 bits (ICISC 2010)

ISO/IEC 29192 Lightweight Cryptography (3/4)

Part 5 - Hash functions

- Photon with permutation sizes 100, 144, 196, 256 and 288 bits and hash output lengths 80, 128, 160, 224 and 256 respectively.
- Spongent with permutation sizes of 88, 136, 176, 240 and 272 bits and hash output lengths 88, 128, 160, 224, and 256 respectively (CHES 2011)
- Lesamnta-LW with permutation size 384 bits and hash output length 256 bits (ICISC 2010)

Part 6 - MAC

- Under development. Chaskey was submitted in 2014 - a permutation based MAC algorithm that uses ARX design strategy (SAC 2014)

ISO/IEC 29192 Lightweight Cryptography (4/4)

Summary

- Covers a relatively large range w.r.t. key size, block size and hash value.
- Includes symmetric and public key (for authentication for specific applications) primitives
- By revisions/amendments, additional algorithms and mechanisms may be included and the ones already specified may be removed through national experts' proposals.

ISO/IEC 29167 Automatic identification and data capture techniques

- Part I -Security services for RFID air interfaces: Defines various security mechanisms that can be implemented by a tag, and the requirements for crypto suites.
- Part 10-19 includes includes AES-128, Present-80, ECC-DH, Grain-128A, AES OFB, XOR, ECDSA-ECDH, cryptoGPS and RAMON.

Cryptrec (Cryptography Research and Evaluation Committees)

- Project to evaluate and monitor the security of cryptographic techniques used in Japanese e-Government systems.
- Publishes three lists
  - e-Government recommended cipher list: approved in terms of security and implementation aspects as well as current and future market development.
  - Candidate recommended cipher list: approved in terms of security and implementation aspects.
  - Monitored ciphers list: not recommended for use, because of high risk of compromise, allowed to use only for interoperability with legacy systems.

Cryptrec (Cryptography Research and Evaluation Committees)

- Project to evaluate and monitor the security of cryptographic techniques used in Japanese e-Government systems.
- Publishes three lists
  - e-Government recommended cipher list: approved in terms of security and implementation aspects as well as current and future market development.
  - Candidate recommended cipher list: approved in terms of security and implementation aspects.
  - Monitored ciphers list: not recommended for use, because of high risk of compromise, allowed to use only for interoperability with legacy systems.
- In March'17, published a guideline on lightweight cryptography.

Cryptrec

- Target algorithms:
    - Block ciphers: AES, Camellia, CLEFIA, TDES, LED, PRINCE, PRESENT, Piccolo, TWINE, SIMON, SPECK, Midori.
    - Authenticated Encryption: ACORN, AES-GCM, AES-OTR, Ascon, CLOC, SILC, JAMBU, Ketje, Minalpher, AES-OCB.
- Initial suggestions for different applications
    - Electrical home appliances e.g., SPECK, SIMON, Piccolo, and Twine
    - RFID Tags e.g., SIMON, SPECK, Piccolo, PRINCE
    - Medical Sensors e.g., SIMON, SPECK, Piccolo, PRESENT
    - Automobiles e.g., Midori, PRINCE, PRESENT, SIMON

ECRYPT eSTREAM Project

A 4-year network of excellence funded project started in 2004 by European Network of Excellence for Cryptology (ECRYPT)

Goal: to identify new stream ciphers that migth be suitable for widespread adoption and to stimulate work in stream ciphers.

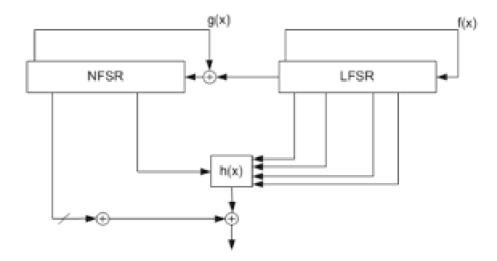
Profiles:

- *Profile I:* for software applications with high throughput requirements with key size of 256 bits.
- *Profile II:* for hardware applications with restricted resources with key size of 80 bits.

ECRYPT eSTREAM - Profile II Finalists (1/3)

Grain

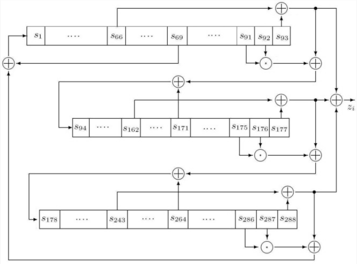
- Widely analyzed, updated version featuring authentication, tweaked twice
- A version that also supports authentication, Grain128A



ECRYPT eSTREAM - Profile II Finalists (2/3)

Trivium

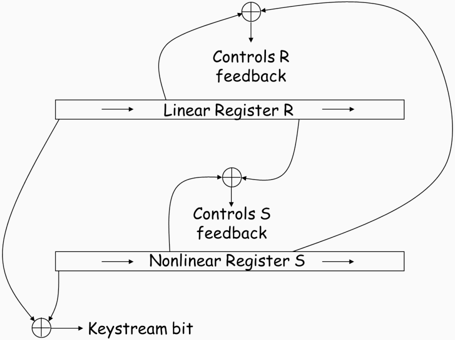
- Widely analyzed, simple and elegant, not tweaked
- Only has an 80-bit version



ECRYPT eSTREAM - Profile II Finalists (3/3)

Mickey

- Lightly analyzed, security depends on the hardness of analysis
- Less implementation flexibility, due to irregular clocking, susceptible to timing and power analysis



CAESAR

- Competition for Authenticated Encryption: Security, Applicability, and Robustness, organized by a committee of 18 cryptographers.
- Aims to select a portfolio of algorithms: with advantages over AES-GCM, and suitable for widespread adoption.
- 57 submissions in March 2014, 7 finalists announced in March’18

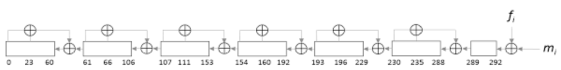
CAESAR

- Competition for Authenticated Encryption: Security, Applicability, and Robustness, organized by a committee of 18 cryptographers.
- Aims to select a portfolio of algorithms: with advantages over AES-GCM, and suitable for widespread adoption.
- 57 submissions in March 2014, 7 finalists announced in March'18
- July'16, three use cases are announced.
  - **Use Case 1:** Lightweight applications (constrained environments): fits into small hardware area, small code for 8-bit CPUs, side channel resistance, hardware performance, especially energy/bit, speed on 8-bit CPUs, optimized for short messages.
  - Use Case 2: High performance applications.
  - Use Case 3: Defense in depth.

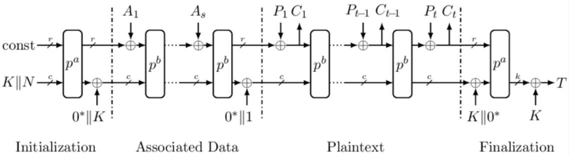
CAESAR Finalists

Finalists announced in March'18.

- **Use Case 1:**
  - ACORN: Stream cipher based, 128-bit key, 128-bit IV, 128-bit tag. Internal state 293 bits, concatenation of 6 LFSRs.



  - Ascon: Based on the duplex sponge mode, 128-bit key, 128-bit IV, 128-bit tag. internal state 320 bits.



- Use Case 2: AEGIS, MORUS and OCB
- Use Case 3: COLM, and Dexoys-II.

NIST's Call for Submission and Next Steps

Profiles

In April 2017, NIST published profiles for LWC Standardization Process as a white paper. NIST portfolio will be recommended in the context of one or more profiles.

Set of engineering requirements that consists of

- Functionality provided by the algorithm (e.g., authenticated encryption).
- Design goals for intended applications (e.g., efficient for short messages).
- Physical characteristics of the environment implementation resides in (hardware or software).
- Performance characteristics (e.g., latency, throughput or power).
- Security characteristics (e.g., security strength, relevant attack models, side channel resistance).

Profile Template

Profile<profile name>	
Functionality	Purpose of cryptographic algorithm (e.g., encryption, authenticated encryption scheme, hashing, message authentication, etc.)
Design goals	List design goals.
Physical characteristics	Name physical characteristic(s), and provide acceptable range(s) (e.g., 64 to 128 bytes of RAM)
Performance characteristics	Name performance characteristic(s), and provide acceptable range(s)
Security characteristics	Minimum security strength, relevant attack models, side channel resistance requirements, etc.

Developing Profiles

Due to the variability of the requirements, use cases, and target devices, having a single profile may not be optimal.

Biryukov and Perrin [5] proposed to split into two areas

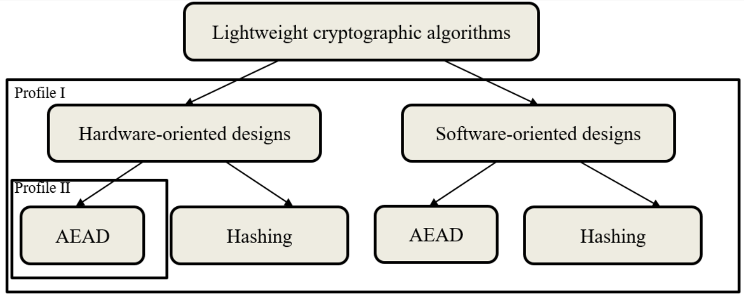
- Ultra-lightweight cryptography: lower security level, for more constrained devices,
- IoT/Ubiquitous cryptography: similar to the requirements of conventional crypto.

[5] A. Biryukov, L. Perrin, *State of the Art in Lightweight Symmetric Cryptography*, IACR Cryptology ePrint Archive, 2017:511, 2017

Draft Profiles for Lightweight Cryptography

NIST published Draft White Paper *Profiles for Lightweight Cryptography Standardization Process* in April 2017.

- Profile I Authenticated Encryption with Associated Data (AEAD) and Hashing for constrained software and hardware environments.
- Profile II AEAD for constrained hardware environments.


Lightweight cryptographic algorithms
Profile I
Hardware-oriented designs
Software-oriented designs
Profile II
AEAD
Hashing
AEAD
Hashing

Profile I - AEAD and Hashing for Constrained Environments (1/4)

Functionality:

- Authenticated encryption with associated data and hashing

Design goals:

- Performs significantly better in constrained environments (hardware and embedded software platforms) compared to current NIST standards.
- Both algorithms should be optimized to be efficient for short messages (e.g., as short as 8 bytes).
- The message length shall be an integer number of bytes.

Physical characteristics

- Compact hardware implementations and embedded software implementations with low RAM and ROM usage should be possible.

Profile I - AEAD and Hashing for Constrained Environments (2/4)

Performance characteristics:

- The performance on ASIC and FPGA should consider various standard cell libraries, the flexibility to support various implementation strategies (low energy, low power, low latency), with significant improvements over current NIST standards.
- The performance on microcontrollers should consider a wide range of 8-bit, 16-bit and 32-bit microcontroller architectures.
- The preprocessing of a key (in terms of computation time and memory footprint) should be efficient.

Profile I - AEAD and Hashing for Constrained Environments (3/4)

Security characteristics for AEAD

- A key length of 128 bits shall be supported.
- Nonce and tag lengths of up to 128 bits shall be supported.
- Plaintext and associated data lengths of up to $2^{50} - 1$ bytes shall be supported.
- At least $2^{50} - 1$ bytes can be processed securely under a single key.
- Cryptanalytic attacks should require at least 2^{112} computations on a classical computer in a single-key setting.
- Lends itself to countermeasures against various side-channel attacks, including timing attacks, simple and differential power analysis (SPA/DPA), and simple and differential electromagnetic analysis (SEMA/DEMA).

Profile I - AEAD and Hashing for Constrained Environments (4/4)

Security characteristics for hashing

- Cryptanalytic attacks should require at least 2^{112} computations on a classical computer.
- Hash outputs of 256 bits must be supported, and longer hash values may be supported as well.
- A maximum message length of $2^{50} - 1$ bytes shall be supported.
- Lends itself to countermeasures against various side-channel attacks, including timing attacks, simple and differential power analysis (SPA/DPA), and simple and differential electromagnetic analysis (SEMA/DEMA).

Profile II- AEAD for Constrained Hardware Environments (1/2)

Functionality:

- Authenticated encryption with associated data

Design goals:

- Performs significantly better compared to current NIST standards.
- The performance for short messages (e.g., as short as 8 bytes) is important.
- The message length shall be an integer number of bytes.

Physical characteristics

- Targeted towards constrained hardware platforms.
- Compact hardware implementations should be possible.

Profile II - AEAD for Constrained Hardware Environments (2/2)

Performance characteristics:

- The performance on ASIC and FPGA should consider a wide range of standard cell libraries and vendors.
- Algorithm should be flexible to support various implementation strategies (low energy, low power, low latency).
- The preprocessing of a key (in terms of computation time and memory footprint) should be efficient.

Security characteristics:

- Same as security characteristics of AEAD in Profile I.

Draft Submission Requirements and Evaluation Criteria

- In May 2018, NIST published the Draft Submission Requirements and Evaluation Criteria.
- After a comment period (approx. 45 days), NIST prepared the final call for submissions.
- Instead of having two calls for each draft profile, we included a single call for AEAD with optional hashing.
- Final call will be publicly available very soon.

Draft Submission Requirements and Evaluation Criteria

- Requirements to be a complete submission
  - Cover sheet, specification, supporting documents, source code, test vectors, and IP statements
- Requirements to be a proper submission
  - AEAD security requirements
  - Hash function security requirements
  - Design requirements
  - Additional AEAD+hashing requirements
  - Implementation requirements

AEAD Security Requirements (1/2)

An authenticated encryption with associated data (AEAD) algorithm is a function with four byte-string inputs and one byte-string output. The four inputs are a variable-length plaintext, variable-length associated data, a fixed-length nonce, and a fixed-length key. The output is a variable-length ciphertext.

- Confidentiality of the plaintexts (under adaptive chosen-plaintext attacks) + Integrity of the ciphertexts (under adaptive forgery attempts)
- Nonce is assumed to be unique under the same key.
- Similarities with CAESAR call for submission

AEAD Security Requirements (2/2)

- Family of algorithms
  - One primary member with key, nonce and tag lengths of at least 128, 96 and 64 bits, respectively.
  - Limits on the input sizes for the primary member **shall not** be smaller than $2^{50} - 1$.
  - Family can include at most 10 members.
- Keys sizes **shall** at least be 128 bits. Attacks **shall** require at least 2^{112} computations. If larger key sizes are supported, it is recommended that at least one member has key size of 256 bits (and resistance to attacks **shall** at least be 2^{224} computations).
- Well-understood, and analyzed. Submissions are expected to have third-party analysis.

Hash Function Requirements

A hash function is a function with one byte-string input and one byte-string output. The input is a variable-length message. The output is a fixed-length hash value.

- Computationally infeasible to find a collision or a (second) preimage. Resistance to length extension attacks.
- Cryptanalytic attacks on the hash function **shall** require at least 2^{112} computations on a classical computer.
- The hash function **shall not** specify hash values that are smaller than 256 bits.
- Family of algorithms
  - One primary member has a hash size of 256 bits.
  - Limits on the input sizes for the primary member **shall not** be smaller than $2^{50} - 1$.
  - Family can include at most 10 members.

Additional Requirements for Submissions with AEAD and Hashing

- Submissions **shall** state which design components the AEAD and hashing algorithms have in common, and explain how these common components lead to a reduced implementation cost.
- Submissions **shall** specify list of pairs of AEAD and hash function family members to be evaluated jointly. This list is permitted to be as short as one recommendation. Primary member of the AEAD family and primary member of the hash function family **shall** be paired together. This list **shall not** be longer than ten recommendations.

Design Requirements

- Submissions **shall** perform significantly better in constrained environments (HW and SW platforms) compared to NIST standards.
- Optimized to be efficient for short messages.
- Implementations should lend themselves to countermeasures against
  - Side Channel Attacks gain info from the implementation of the primitive (e.g., timing, power, electromagnetic field).
  - Fault Attacks alter the normal functioning of a physical electronic device (e.g., by changing the supply voltage), such that it causes an error in the computation that can be leveraged to perform an attack.
- Designs can make tradeoffs between performance metrics, and submitters are allowed to prioritize certain performance requirements over others.

Implementation Requirements

- Reference software implementation in C, to support public understanding.
- An implementation for all variants.
- Code without compiler intrinsics, platform-specific headers, or compiler-specific features.
- Compatible API with eBACS: ECRYPT Benchmarking of Cryptographic Systems.
- Optimized implementations that use the same API, or additional implementations that highlight specific implementation features of the algorithms. There are no restrictions on the API for the additional implementations.
- The correctness of the reference implementation will be verified on the NIST test vector verification platform.

Evaluation Process

- Submissions will be analyzed based on security, performance and side channel resistance.
- Submissions that have significant third-party analysis or leverage components of existing standards will be favored for selection.

Tentative timeline

- Early September 2018, NIST will publish FRN (Federal Register Notice) and the final Call for Submissions.
- December 2018, option for early submission for initial review.
- February 2019, deadline for submissions.
- NIST will publish the complete and proper submissions.
- Initial evaluation will be for approximately 12 months.
- Workshop will be held ten to twelve months after the submission deadline.
- Standardization within two to four years, after the public analysis starts.

Thanks!

More information on Lightweight Cryptography Project available at
`https://www.nist.gov/programs-projects/lightweight-cryptography`

Subscribe to our mailing list: `lwc-forum@list.nist.gov`

Additional comments/questions?
Email the team at `lightweight-crypto@nist.gov`

Further Reading

- A. Biryukov and L. Perrin, *State of the Art in Lightweight Symmetric Cryptography*, ePrint 511/2017.
- CRYPTREC Cryptographic Technology Guideline (Lightweight Cryptography), March 2017.
- NISTIR 8114 - Report on Lightweight Cryptography, March 2017.