A perspective on standardization of advanced cryptography at NIST

Luís Brandão

Cryptographic Technology Group
National Institute of Standards and Technology
(Gaithersburg, Maryland, USA)

Presentation at ACS'19
Advanced **C**ryptography **S**tandardization Workshop
August 18, 20**19** @ Santa Barbara, California, USA

Thank you to Lily Chen, Rene Peralta and Andy Regenscheid for provided feedback

Outline

1. NIST introduction
2. Threshold cryptography
3. Some considerations
4. Privacy-enhancing cryptography
5. Concluding remarks

Outline

1. NIST introduction

2. Threshold cryptography

3. Some considerations

4. Privacy-enhancing cryptography

5. Concluding remarks

NIST basics

National Institute of Standards and Technology (NIST)

(National Bureau of Standards 1901–1988) $\rightarrow$ NIST 1988–present

- **Non-regulatory** federal agency (within the U.S. Department of Commerce)
- **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.

Aerial photo of Gaithersburg campus (source: Google Maps, August 2019)

NIST basics

National Institute of Standards and Technology (NIST)

(National Bureau of Standards 1901–1988) $\rightarrow$ NIST 1988–present

- **Non-regulatory** federal agency (within the U.S. Department of Commerce)
- **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.

Wide spectrum of competences

- $\sim 6\text{–}7 \times 10^3$ workers (employees + associates)
- Two campuses (Maryland and Colorado)
- Five laboratories and two centers
- Labs $\rightarrow$ Divisions $\rightarrow$ Groups $\rightarrow$ Projects
- Standards, research and applications

Aerial photo of Gaithersburg campus (source: Google Maps, August 2019)

Labs, divisions, groups

Information Technology Laboratory (ITL)

advancing measurement science, standards, and technology through research and development in information technology, mathematics, and statistics.

Labs, divisions, groups

Information Technology Laboratory (ITL)

advancing measurement science, standards, and technology through research and development in information technology, mathematics, and statistics.

→ **Computer Security Division (CSD):** Cryptographic Technology; Secure Systems and Applications; Security Components and Mechanisms; Security Engineering and Risk Management; Security Testing, Validation and Measurement

Labs, divisions, groups

Information Technology Laboratory (ITL)

advancing measurement science, standards, and technology through research and development in information technology, mathematics, and statistics.

→ **Computer Security Division (CSD):** Cryptographic Technology; Secure Systems and Applications; Security Components and Mechanisms; Security Engineering and Risk Management; Security Testing, Validation and Measurement

→ **Cryptographic Technology Group (CTG):** research, develop, engineer, and produce guidelines, recommendations and best practices for cryptographic algorithms, methods, and protocols.

Labs, divisions, groups

Information Technology Laboratory (ITL)

advancing measurement science, standards, and technology through research and development in information technology, mathematics, and statistics.

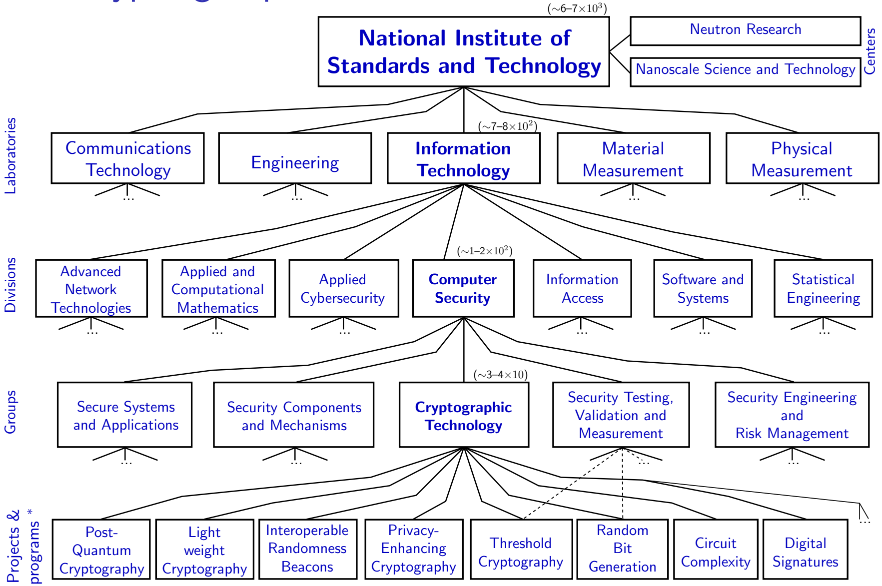
→ **Computer Security Division (CSD):** Cryptographic Technology; Secure Systems and Applications; Security Components and Mechanisms; Security Engineering and Risk Management; Security Testing, Validation and Measurement

→ **Cryptographic Technology Group (CTG):** research, develop, engineer, and produce guidelines, recommendations and best practices for cryptographic algorithms, methods, and protocols.

- Documents: "standards" (**FIPS**); recommendations, guidelines, reference material (**SP 800**); research reports and background information (**NISTIR**)
- International interaction: government, industry, academia, standardization

FIPS = Federal Information Processing Standards; SP 800 = Special Publications in Computer Security; NISTIR = NIST Internal or Interagency Report.

The “Crypto group” at NIST



* (Some projects/programs involve several groups, divisions or labs) (in parenthesis: approximate range # workers, inc. associates and fed. employees)

Some standardized crypto primitives

Traditional focus on "basic" primitives:

Some standardized crypto primitives

Traditional focus on “basic” primitives:

- Block ciphers
- Cipher modes of operation
- Hash functions
- Signatures
- Pair-wise key agreement
- DRBGs

Some standardized crypto primitives

Traditional focus on “basic” primitives:

- Block ciphers: DES (1977), EES (1994), TDEA (1999), AES (2001)
- Cipher modes of operation (1980–): CBC, CT, CCM, GCM ...
- Hash functions (SHS): SHA-1 (1994), SHA-2 (2001), SHA-3 (2015)
- Signatures (DSS): DSA (1997), ECDSA (1998), RSA (2000), EdDSA (2019)
- Pair-wise key agreement, e.g., based on DH (2006) and RSA (2009)
- DRBGs (2006): CTR_, Hash_, HMAC_, Dual_EC_ (withdrawn in 2015 due to concerns of potential subversion)

(Not an exhaustive list; years indicated for perspective; some documentation has subsequent updates)
(Further details in “NIST Cryptographic Standards and Guidelines Development Program Briefing Book”)

Legend:

- AES = Advanced Encryption Standard
- CBC = Cipher block chaining (mode)
- CT = Counter (mode)
- CCM = Counter with Cipher-block chaining
- DES = Data Encryption Standard
- DH = Diffie–Hellman
- DSA = Digital Signature Algorithm
- DSS = Digital Signature Standard
- DRBG = Deterministic Random Bit Generator
- ECDSA = Elliptic curve DSA
- EdDSA = Edwards curve DSA
- EES = Escrowed Encryption Standard
- GCM = Galois counter mode
- RSA = Rivest–Shamir–Adleman
- SHA = Secure Hash Algorithm
- SHS = Secure Hash Standard
- TDEA = Triple Data Encryption Algorithm

Some standardized crypto primitives

Traditional focus on "basic" primitives:

- Block ciphers: DES (1977), EES (1994), TDEA (1999), AES (2001)
- Cipher modes of operation (1980–): CBC, CT, CCM, GCM ...
- Hash functions (SHS): SHA-1 (1994), SHA-2 (2001), SHA-3 (2015)
- Signatures (DSS): DSA (1997), ECDSA (1998), RSA (2000), EdDSA (2019)
- Pair-wise key agreement, e.g., based on DH (2006) and RSA (2009)
- DRBGs (2006): CTR_, Hash_, HMAC_, Dual_EC_ (withdrawn in 2015 due to concerns of potential subversion)

(Not an exhaustive list; years indicated for perspective; some documentation has subsequent updates)
(Further details in "NIST Cryptographic Standards and Guidelines Development Program Briefing Book")

Legend:

- AES = Advanced Encryption Standard
- CBC = Cipher block chaining (mode)
- CT = Counter (mode)
- CCM = Counter with Cipher-block chaining
- DES = Data Encryption Standard
- DH = Diffie–Hellman
- DSA = Digital Signature Algorithm
- DSS = Digital Signature Standard
- DRBG = Deterministic Random Bit Generator
- ECDSA = Elliptic curve DSA
- EdDSA = Edwards curve DSA
- EES = Escrowed Encryption Standard
- GCM = Galois counter mode
- RSA = Rivest–Shamir–Adleman
- SHA = Secure Hash Algorithm
- SHS = Secure Hash Standard
- TDEA = Triple Data Encryption Algorithm

Some standardized crypto primitives

Traditional focus on "basic" primitives:

- Block ciphers: DES (1977), EES (1994), TDEA (1999), AES (2001)
- Cipher modes of operation (1980–): CBC, CT, CCM, GCM ...
- Hash functions (SHS): SHA-1 (1994), SHA-2 (2001), SHA-3 (2015)
- Signatures (DSS): DSA (1997), ECDSA (1998), RSA (2000), EdDSA (2019)
- Pair-wise key agreement, e.g., based on DH (2006) and RSA (2009)
- DRBGs (2006): CTR_, Hash_, HMAC_, Dual_EC_ (withdrawn in 2015 due to concerns of potential subversion)

(Not an exhaustive list; years indicated for perspective; some documentation has subsequent updates)
(Further details in "NIST Cryptographic Standards and Guidelines Development Program Briefing Book")

Legend:
- AES = Advanced Encryption Standard
- CBC = Cipher block chaining (mode)
- CT = Counter (mode)
- CCM = Counter with Cipher-block chaining
- DES = Data Encryption Standard
- DH = Diffie–Hellman
- DSA = Digital Signature Algorithm
- DSS = Digital Signature Standard
- DRBG = Deterministic Random Bit Generator
- ECDSA = Elliptic curve DSA
- EdDSA = Edwards curve DSA
- EES = Escrowed Encryption Standard
- GCM = Galois counter mode
- RSA = Rivest–Shamir–Adleman
- SHA = Secure Hash Algorithm
- SHS = Secure Hash Standard
- TDEA = Triple Data Encryption Algorithm

Some standardized crypto primitives

Traditional focus on "basic" primitives:

- Block ciphers: DES (1977), EES (1994), TDEA (1999), AES (2001)
- Cipher modes of operation (1980–): CBC, CT, CCM, GCM ...
- Hash functions (SHS): SHA-1 (1994), SHA-2 (2001), SHA-3 (2015)
- Signatures (DSS): DSA (1997), ECDSA (1998), RSA (2000), EdDSA (2019)
- Pair-wise key agreement, e.g., based on DH (2006) and RSA (2009)
- DRBGs (2006): CTR_, Hash_, HMAC_, Dual_EC_ (withdrawn in 2015 due to concerns of potential subversion)

(Not an exhaustive list; years indicated for perspective; some documentation has subsequent updates)
(Further details in "NIST Cryptographic Standards and Guidelines Development Program Briefing Book")

Legend:

- AES = Advanced Encryption Standard
- CBC = Cipher block chaining (mode)
- CT = Counter (mode)
- CCM = Counter with Cipher-block chaining
- DES = Data Encryption Standard
- DH = Diffie–Hellman
- DSA = Digital Signature Algorithm
- DSS = Digital Signature Standard
- DRBG = Deterministic Random Bit Generator
- ECDSA = Elliptic curve DSA
- EdDSA = Edwards curve DSA
- EES = Escrowed Encryption Standard
- GCM = Galois counter mode
- RSA = Rivest–Shamir–Adleman
- SHA = Secure Hash Algorithm
- SHS = Secure Hash Standard
- TDEA = Triple Data Encryption Algorithm

Some standardized crypto primitives

Traditional focus on "basic" primitives:

- Block ciphers: DES (1977), EES (1994), TDEA (1999), AES (2001)
- Cipher modes of operation (1980–): CBC, CT, CCM, GCM ...
- Hash functions (SHS): SHA-1 (1994), SHA-2 (2001), SHA-3 (2015)
- Signatures (DSS): DSA (1997), ECDSA (1998), RSA (2000), EdDSA (2019)
- Pair-wise key agreement, e.g., based on DH (2006) and RSA (2009)
- DRBGs (2006): CTR_, Hash_, HMAC_, Dual_EC_ (withdrawn in 2015 due to concerns of potential subversion)

(Not an exhaustive list; years indicated for perspective; some documentation has subsequent updates)
(Further details in "NIST Cryptographic Standards and Guidelines Development Program Briefing Book")

Some of these NIST-standards were specified with reference to standards by other bodies, and with further requirements.

Legend:

- AES = Advanced Encryption Standard
- CBC = Cipher block chaining (mode)
- CT = Counter (mode)
- CCM = Counter with Cipher-block chaining
- DES = Data Encryption Standard
- DH = Diffie–Hellman
- DSA = Digital Signature Algorithm
- DSS = Digital Signature Standard
- DRBG = Deterministic Random Bit Generator
- ECDSA = Elliptic curve DSA
- EdDSA = Edwards curve DSA
- EES = Escrowed Encryption Standard
- GCM = Galois counter mode
- RSA = Rivest–Shamir–Adleman
- SHA = Secure Hash Algorithm
- SHS = Secure Hash Standard
- TDEA = Triple Data Encryption Algorithm

Some standardized crypto primitives

Traditional focus on "basic" primitives:

- Block ciphers: DES (1977), EES (1994), TDEA (1999), AES (2001)
- Cipher modes of operation (1980–): CBC, CT, CCM, GCM ...
- Hash functions (SHS): SHA-1 (1994), SHA-2 (2001), SHA-3 (2015)
- Signatures (DSS): DSA (1997), ECDSA (1998), RSA (2000), EdDSA (2019)
- Pair-wise key agreement, e.g., based on DH (2006) and RSA (2009)
- DRBGs (2006): CTR_, Hash_, HMAC_, Dual_EC_ (withdrawn in 2015 due to concerns of potential subversion)

(Not an exhaustive list; years indicated for perspective; some documentation has subsequent updates)
(Further details in "NIST Cryptographic Standards and Guidelines Development Program Briefing Book")

Some of these NIST-standards were specified with reference to standards by other bodies, and with further requirements.

Several methods:

- Internal or interagency developed techniques
- Adoption of external standards
- Open call, competition, "competition-like"

Legend:

- AES = Advanced Encryption Standard
- CBC = Cipher block chaining (mode)
- CT = Counter (mode)
- CCM = Counter with Cipher-block chaining
- DES = Data Encryption Standard
- DH = Diffie–Hellman
- DSA = Digital Signature Algorithm
- DSS = Digital Signature Standard
- DRBG = Deterministic Random Bit Generator
- ECDSA = Elliptic curve DSA
- EdDSA = Edwards curve DSA
- EES = Escrowed Encryption Standard
- GCM = Galois counter mode
- RSA = Rivest–Shamir–Adleman
- SHA = Secure Hash Algorithm
- SHS = Secure Hash Standard
- TDEA = Triple Data Encryption Algorithm

Other processes (examples)

Other processes (examples)

Ongoing evaluations:

- Post-quantum cryptography: signatures, public-key encryption, key encapsulation
- Lightweight cryptography: ciphers, authenticated encryption, hash functions

Other processes (examples)

Ongoing evaluations:

- Post-quantum cryptography: signatures, public-key encryption, key encapsulation
- Lightweight cryptography: ciphers, authenticated encryption, hash functions

The crypto group has other ongoing projects: https://www.nist.gov/itl/csd/cryptographic-technology

Other processes (examples)

Ongoing evaluations:

- Post-quantum cryptography: signatures, public-key encryption, key encapsulation
- Lightweight cryptography: ciphers, authenticated encryption, hash functions

The crypto group has other ongoing projects: https://www.nist.gov/itl/csd/cryptographic-technology

Previous considerations:

- Paring-based Cryptography: workshop (2008), study and call for feedback on use cases (2011), report (2012–2015) (forming NIST's position on standardization/recommendation: more research is needed).

Other processes (examples)

Ongoing evaluations:

- Post-quantum cryptography: signatures, public-key encryption, key encapsulation
- Lightweight cryptography: ciphers, authenticated encryption, hash functions

The crypto group has other ongoing projects: https://www.nist.gov/itl/csd/cryptographic-technology

Previous considerations:

- Paring-based Cryptography: workshop (2008), study and call for feedback on use cases (2011), report (2012–2015) (forming NIST's position on standardization/recommendation: more research is needed).

Development process:

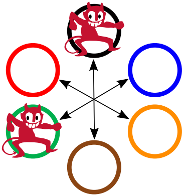
- NISTIR 7977: NIST Cryptographic Standards and Guidelines Development Process (2016). Formalizes several **principles** to follow:
  - transparency
  - openness
  - balance
  - integrity
  - technical merit
  - usability
  - global acceptability
  - continuous improvement
  - innovation and intellectual property

  (and overarching considerations)

Outline

NIST project: Threshold Cryptography

Goal: standardize threshold schemes for cryptographic primitives (signing, public-key decryption, key generation, enciphering, ...)

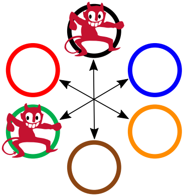


NIST project: Threshold Cryptography

Goal: standardize threshold schemes for cryptographic primitives (signing, public-key decryption, key generation, enciphering, ...)

Distributed computation, with some properties:

- Each component operates only on a share of the key
- Tolerance to compromise of f-out-of-n components
- Enhanced resistance against side-channel attacks

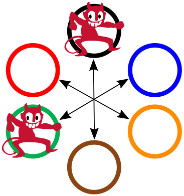


NIST project: Threshold Cryptography

Goal: standardize threshold schemes for cryptographic primitives
(signing, public-key decryption, key generation, enciphering, ...)

Distributed computation, with some properties:

- Each component operates only on a share of the key
- Tolerance to compromise of f-out-of-n components
- Enhanced resistance against side-channel attacks



Some questions:

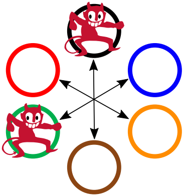
- Potentially many standards ... which ones, what kind?
- Which process(es): direct proposal, open call, adopt other standards, ...?
- Different areas of expertise ... how to involve stakeholders?

NIST project: Threshold Cryptography

Goal: standardize threshold schemes for cryptographic primitives
(signing, public-key decryption, key generation, enciphering, ...)

Distributed computation, with some properties:

- Each component operates only on a share of the key
- Tolerance to compromise of f-out-of-n components
- Enhanced resistance against side-channel attacks



Some questions:

- Potentially many standards ... which ones, what kind?
- Which process(es): direct proposal, open call, adopt other standards, ...?
- Different areas of expertise ... how to involve stakeholders?

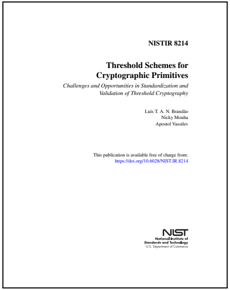
Next slides: steps so far; some challenges; roadmap.

NIST Threshold Cryptography timeline

NIST Threshold Cryptography timeline

(March 2019) **NISTIR 8214**:
NIST report on threshold schemes

(Draft July 2018 → comments → March 2019)


NISTIR 8214

Threshold Schemes for Cryptographic Primitives

Challenges and Opportunities in Standardization and Validation of Threshold Cryptography

Luís T. A. N. Brandão
Nicky Mouha
Apostol Vassilev

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8214

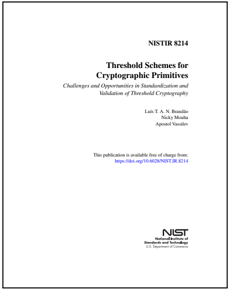
NIST
National Institute of Standards and Technology
U.S. Department of Commerce

https://doi.org/10.6028/NIST.IR.8214

NIST Threshold Cryptography timeline

(March 2019) **NISTIR 8214**:
NIST report on threshold schemes

(Draft July 2018 → comments → March 2019)


NISTIR 8214

Threshold Schemes for
Cryptographic Primitives

Challenges and Opportunities in Standardization and Validation of Threshold Cryptography

Luís T. A. N. Brandão
Nicky Mouha
Apostol Vassilev

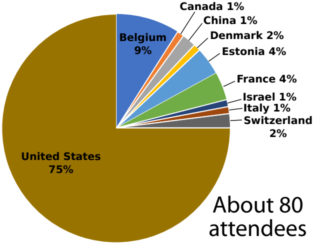
This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8214

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

https://doi.org/10.6028/NIST.IR.8214

(March 2019) **NTCW 2019**:
NIST Threshold Cryptography workshop

(2 panels, 2 keynotes, 5 papers, 8 presentations,
4 NIST talks, 2 feedback moments)

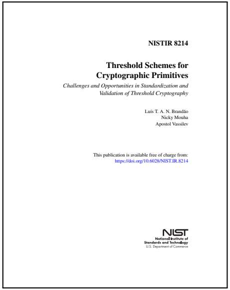


https://csrc.nist.gov/Events/2019/NTCW19

NIST Threshold Cryptography timeline

(March 2019) **NISTIR 8214**:
NIST report on threshold schemes

(Draft July 2018 → comments → March 2019)


NISTIR 8214

Threshold Schemes for Cryptographic Primitives

Challenges and Opportunities in Standardization and Validation of Threshold Cryptography

Luís T. A. N. Brandão
Nicky Mouha
Apostol Vassilev

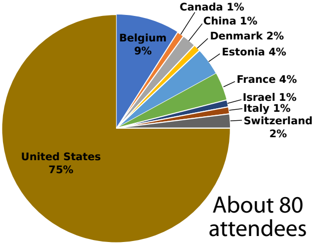
This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8214

NIST
National Institute of Standards and Technology
U.S. Department of Commerce

https://doi.org/10.6028/NIST.IR.8214

(March 2019) **NTCW 2019**:
NIST Threshold Cryptography workshop

(2 panels, 2 keynotes, 5 papers, 8 presentations, 4 NIST talks, 2 feedback moments)



https://csrc.nist.gov/Events/2019/NTCW19

- (Soon) **Draft roadmap towards standardization**
- (After) Get feedback on items for standardization and criteria for calls

Why taking initial steps?

Why bothering to start with the NISTIR and the NTCW !?

Why taking initial steps?

Why bothering to start with the NISTIR and the NTCW !?

- They are steps in *driving an open and transparent process*
- And the reflection and learning process is useful ...

Why taking initial steps?

Why bothering to start with the NISTIR and the NTCW !?

- They are steps in *driving an open and transparent process*
- And the reflection and learning process is useful ...

Some notes:

- need to characterize threshold schemes (multiple dimensions);

Kinds of threshold  Communication interfaces  Executing platform  Setup and maintenance 

Why taking initial steps?

Why bothering to start with the NISTIR and the NTCW !?

- They are steps in *driving an open and transparent process*
- And the reflection and learning process is useful ...

Some notes:

- need to characterize threshold schemes (multiple dimensions);

  Kinds of threshold | Communication interfaces | Executing platform | Setup and maintenance
- dilemmas about granularity;
- separation of single-device vs. multi-party;
- usefulness of explaining rationale and envisioning applications;
- stakeholders' willingness to contribute;
- encouragement to move forward.

Why taking initial steps?

Why bothering to start with the NISTIR and the NTCW !?

- They are steps in *driving an open and transparent process*
- And the reflection and learning process is useful ...

Some notes:

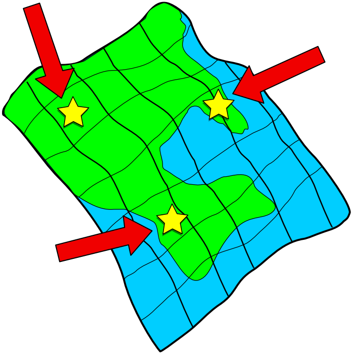
- need to characterize threshold schemes (multiple dimensions);

  Kinds of threshold | Communication interfaces | Executing platform | Setup and maintenance
- dilemmas about granularity;
- separation of single-device vs. multi-party;
- usefulness of explaining rationale and envisioning applications;
- stakeholders' willingness to contribute;
- encouragement to move forward.

These elements are helpful for the next step ... designing a roadmap

Preliminary roadmap (ongoing)

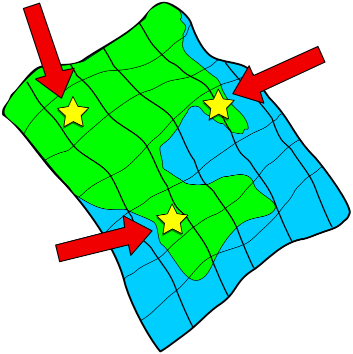
We are writing a draft “roadmap”:



1. getting a map
2. deciding where to go
3. thinking how to get there

Preliminary roadmap (ongoing)

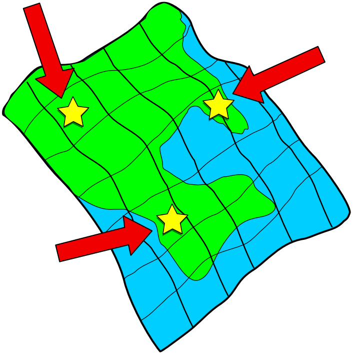
We are writing a draft "roadmap":



1. getting a map
   (**mapping layers**)
2. deciding where to go
   (weighing factors)
3. thinking how to get there
   (collaboration)

Preliminary roadmap (ongoing)

We are writing a draft “roadmap”:



1. getting a map
   (**mapping layers**)
2. deciding where to go
   (weighing factors)
3. thinking how to get there
   (collaboration)

Disclaimer: the structure in the next slides is subject to change.

Mapping layers

An abstract layered decomposition of the threshold standardization space

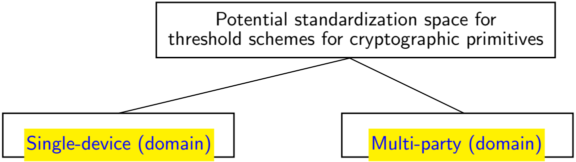
Four layers

Potential standardization space for threshold schemes for cryptographic primitives

Mapping layers

An abstract layered decomposition of the threshold standardization space

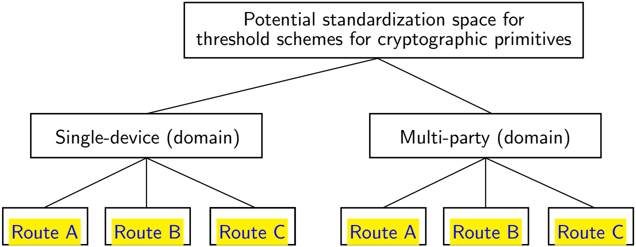
Four layers: domains



Mapping layers

An abstract layered decomposition of the threshold standardization space

Four layers: domains, routes

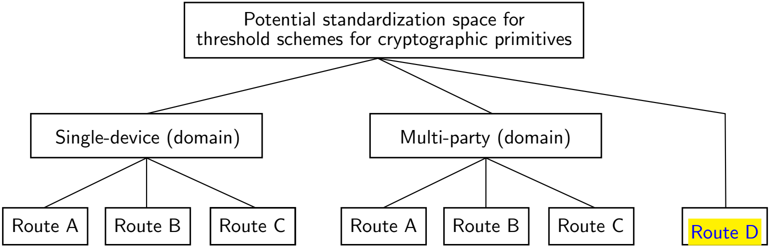


- Route A: simple thresholdization
- Route B: compositional / complex designs
- Route C: new / non-standardized primitives

Mapping layers

An abstract layered decomposition of the threshold standardization space

Four layers: domains, routes

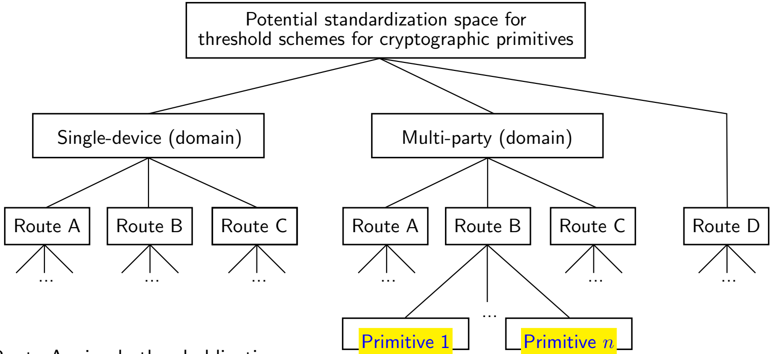


- Route A: simple thresholdization
- Route B: compositional / complex designs
- Route C: new / non-standardized primitives
- Route D: gadgets / building blocks

Mapping layers

An abstract layered decomposition of the threshold standardization space

Four layers: domains, routes, primitives

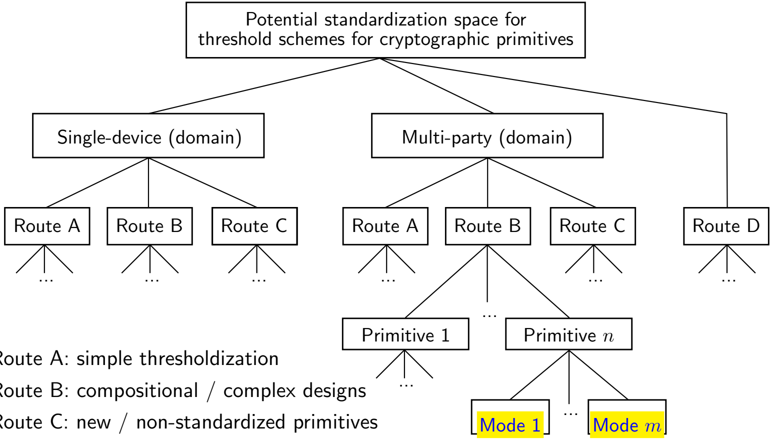


- Route A: simple thresholdization
- Route B: compositional / complex designs
- Route C: new / non-standardized primitives
- Route D: gadgets / building blocks

Mapping layers

An abstract layered decomposition of the threshold standardization space

Four layers: domains, routes, primitives, modes



- Route A: simple thresholdization
- Route B: compositional / complex designs
- Route C: new / non-standardized primitives
- Route D: gadgets / building blocks

Some conceived examples

Primitives across routes:

- **A:**
- **B:**
- **C:**
- **D:**

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen; AES (single-device) threshold circuit design against leakage.
- **B:**
- **C:**
- **D:**

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen; AES (single-device) threshold circuit design against leakage.
- **B:** ECDSA signature; RSA key-gen; AES enciphering; AES (single-device) threshold circuit against combined attacks.
- **C:**
- **D:**

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen;
  AES (single-device) threshold circuit design against leakage.
- **B:** ECDSA signature; RSA key-gen; AES enciphering;
  AES (single-device) threshold circuit against combined attacks.
- **C:** post-quantum signing & decryption; lightweight-crypto threshold; ...
- **D:**

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen;
  AES (single-device) threshold circuit design against leakage.
- **B:** ECDSA signature; RSA key-gen; AES enciphering;
  AES (single-device) threshold circuit against combined attacks.
- **C:** post-quantum signing & decryption; lightweight-crypto threshold; ...
- **D:** secret sharing; distributed RNG; consensus; oblivious transfer; ...

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen;
  AES (single-device) threshold circuit design against leakage.
- **B:** ECDSA signature; RSA key-gen; AES enciphering;
  AES (single-device) threshold circuit against combined attacks.
- **C:** post-quantum signing & decryption; lightweight-crypto threshold; ...
- **D:** secret sharing; distributed RNG; consensus; oblivious transfer; ...

(Some items can be in more than one route; not all need to be a standardization goal)

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen;
  AES (single-device) threshold circuit design against leakage.
- **B:** ECDSA signature; RSA key-gen; AES enciphering;
  AES (single-device) threshold circuit against combined attacks.
- **C:** post-quantum signing & decryption; lightweight-crypto threshold; ...
- **D:** secret sharing; distributed RNG; consensus; oblivious transfer; ...

(Some items can be in more than one route; not all need to be a standardization goal)

Examples of modes:

- **Interchangeable:**
- **Secret-shared IO:**
- **Auditable:**

Some conceived examples

Primitives across routes:

- **A:** RSA decryption & signature; Schnorr signature; ECC key-gen;
  AES (single-device) threshold circuit design against leakage.
- **B:** ECDSA signature; RSA key-gen; AES enciphering;
  AES (single-device) threshold circuit against combined attacks.
- **C:** post-quantum signing & decryption; lightweight-crypto threshold; ...
- **D:** secret sharing; distributed RNG; consensus; oblivious transfer; ...

(Some items can be in more than one route; not all need to be a standardization goal)

Examples of modes:

- **Interchangeable:** interface indistinguishable from conventional primitive (e.g., threshold signature with secret-shared key)
- **Secret-shared IO:** operation on secret-shared plaintext **i**nput (client sends shares of input separately to each component); similar for **o**utput
- **Auditable:** client can learn/prove that computation was thresholdized (e.g., multi-signature with independent keys)

Development process

Generic possible sequence of phases:

1. Roadmap → 2. Calls with criteria → 3. Evaluation → 4. Issue standards

(Each phase to include public feedback. Some Threshold Cryptography workshops along the way?)

Development process

Generic possible sequence of phases:

1. Roadmap → **2. Calls with criteria** → **3. Evaluation** → **4. Issue standards**
(Each phase to include public feedback. Some Threshold Cryptography workshops along the way?)

Different standardization *items* can have **different**:

- **calls for contributions:** feedback on reference protocols; new protocols; reference implementations showing feasibility; research results, ...
- **timelines** (e.g., compare different routes)
- **final formats:** addendum vs. standalone standard, reference other standards, implementation/validation guidelines, reference definitions,

Development process

Generic possible sequence of phases:

1. Roadmap → **2. Calls with criteria** → **3. Evaluation** → **4. Issue standards**
(Each phase to include public feedback. Some Threshold Cryptography workshops along the way?)

Different standardization *items* can have **different**:

- **calls for contributions:** feedback on reference protocols; new protocols; reference implementations showing feasibility; research results, ...
- **timelines** (e.g., compare different routes)
- **final formats:** addendum vs. standalone standard, reference other standards, implementation/validation guidelines, reference definitions,

Public feedback can be useful for specifying:

- standardization *items* (down to the threshold mode / functionality);
- weighing factors: application motivation, validation suitability, features;
- features: rejuvenation, dynamic thresholds; robustness; composability; ...

Outline

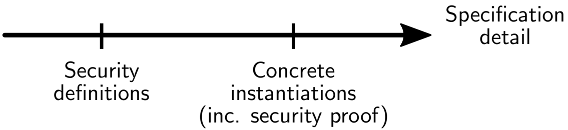
A perspective on the granularity dilemma

Do we need to compromise between:

A perspective on the granularity dilemma

Do we need to compromise between:

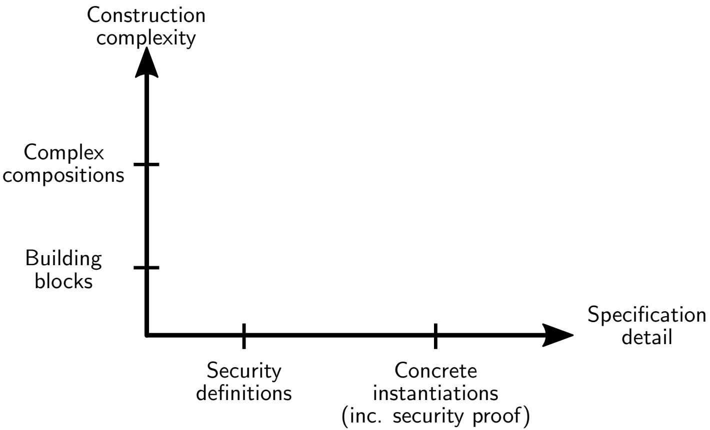
- ideal functionalities vs. concrete protocols of threshold schemes?



A perspective on the granularity dilemma

Do we need to compromise between:

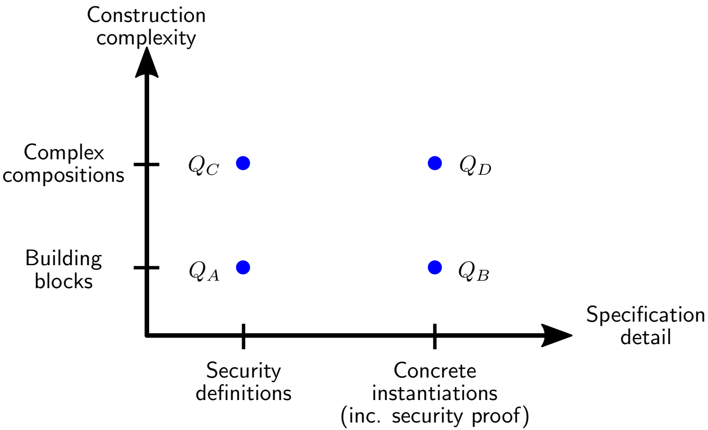
- ideal functionalities vs. concrete protocols of threshold schemes?
- building blocks vs. complex constructions?



A perspective on the granularity dilemma

Do we need to compromise between:

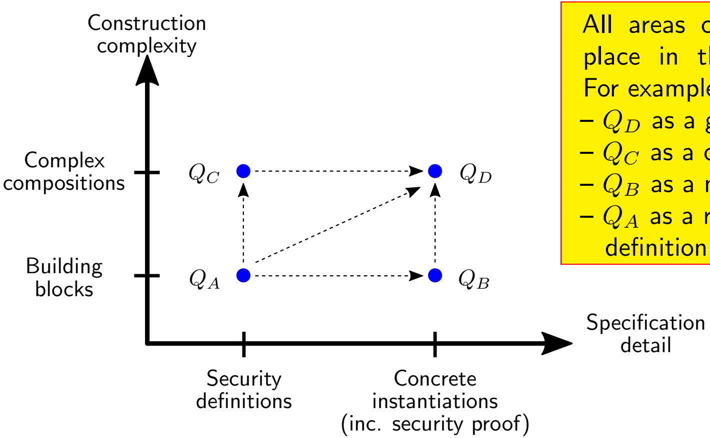
- ideal functionalities vs. concrete protocols of threshold schemes?
- building blocks vs. complex constructions?



A perspective on the granularity dilemma

Do we need to compromise between:

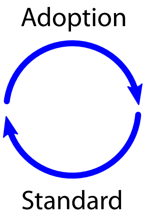
- ideal functionalities vs. concrete protocols of threshold schemes?
- building blocks vs. complex constructions?



All areas can have a place in the process. For example:
- Q_D as a goal;
- Q_C as a criterion;
- Q_B as a module;
- Q_A as a reference definition.

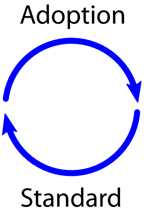
Standardization vs. adoption

What makes a standard *good*? A well-done specification ... and the context.



Standardization vs. adoption

What makes a standard *good*? A well-done specification ... and the context.

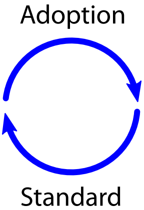


A <u>good</u> standard can be a reference for:

- *best practices* and *minimum defaults*;
- *interoperability*;
- validation and certification;
- what to innovate upon.

Standardization vs. adoption

What makes a standard *good*? A well-done specification ... and the context.



A <u>good</u> standard can be a reference for:

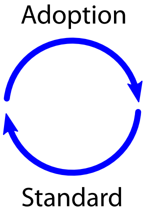
- *best practices* and *minimum defaults*;
- *interoperability*;
- validation and certification;
- what to innovate upon.

If/when compliance is required, a standard can be bad if:

- the technique is obsolete / outdated;
- cannot verify/validate implementations prone to error;
- cannot be corrected / withdrawn / replaced (when it should)

Standardization vs. adoption

What makes a standard *good*? A well-done specification ... and the context.



A good standard can be a reference for:

- *best practices* and *minimum defaults*;
- *interoperability*;
- validation and certification;
- what to innovate upon.

If/when compliance is required, a standard can be bad if:

- the technique is obsolete / outdated;
- cannot verify/validate implementations prone to error;
- cannot be corrected / withdrawn / replaced (when it should)

Note: Adoption goals can vary with the standardization initiative

Intellectual property claims

The topic of intellectual property is relevant:

Intellectual property claims

The topic of intellectual property is relevant:

- Can ask for disclosure of patents: *call* for disclosure, conditions for submitting
- Can promote/require "FRAND" license: **f**air, **r**easonable, **a**nd **n**on-**d**iscriminatory
- Cannot force third party to disclose or enable FRAND terms ... but can choose to specify guidance based on expectation of FRAND terms.

Intellectual property claims

The topic of intellectual property is relevant:

- Can ask for disclosure of patents: *call* for disclosure, conditions for submitting
- Can promote/require "FRAND" license: **f**air, **r**easonable, **a**nd **n**on-**d**iscriminatory
- Cannot force third party to disclose or enable FRAND terms ... but can choose to specify guidance based on expectation of FRAND terms.

Excerpt from NIST-ITL patent policy: "*assurance [...] that [...] party does not hold [...] any essential patent claim(s); or that a license [...] will be made available: under reasonable terms and conditions that are demonstrably free of any unfair discrimination;*" [possibly without compensation]

Excerpt from NISTIR 7977: "*NIST has noted a strong preference among its users for solutions that are unencumbered by royalty-bearing patented technologies. NIST has observed that widespread adoption of cryptographic solutions that it has developed has been facilitated by royalty-free licensing terms.*" [...]

"*NIST will explicitly recognize and respect the value of IP and the need to protect IP if it is incorporated into standards or guidelines.*"

Traceability / transparency

Make evident all the progress/changes in documentation.

Traceability / transparency

Make evident all the progress/changes in documentation. Example: a *diff* version shows the public comments and corresponding answers/changes.

rity model is not enough to assess the effects of ~~and on~~ placing a threshold scheme ~~placed~~ in an adversarial environment. One also needs to characterize implementation aspects whose variation may affect security. These include the types of threshold, the communication interfaces, the target computing platforms, and the setup and maintenance requirements. R14: N9

For example, system models and attack types can differ substantially across different platforms and communication mediums. It should thus be considered how the components inter-communicate, and how they can be assumed separate and independent vs. mutually interfering. In a single device setting, this may involve interaction between different components within a single chip or a single computer. In a contrasting setting, multiple nodes (e.g., R15: A6, E19

#	Ref	Old location	E: Comments by Christian Cachin; Hugo Krawczyk; Tal Rabin; Jason Resch; Chrysoula Stathakopoulou (IBM)	Related	Reply Notes	Rev
38	E19	Sec. 1	Communications Channels Among other things, we were very pleased to see the level of attention that this document dedicates to considerations of the communications channel and how that can impact the security properties of the resulting threshold system. We consider it crucial enough to merit some discussion in the introduction (pages 1-3) as the attack models and realistic schemes differ enormously whether communication and the N components are (a) modules on the same chip; (b) compo-	A6, E19, E20, E21	– **NOTE:** Attack models and security properties can vary substantially with the type of implementation platform and communication between nodes – **CHANGED:** In Sec. 1 (Introduction), mentioned this example after enumerating characterizing features.	R15

https://csrc.nist.gov/CSRC/media/Publications/nistir/8214/final/documents/nistir-8214-diff-comments-received.pdf

Outline

Privacy-Enhancing Cryptography

The NIST PEC project (revived in 2019):

- Interested in privacy promotable by cryptography
- Zero-knowledge proofs (ZKPs), secure multiparty computation (SMPC), ...
- Wants to keep up to date with, and support, external initiatives
- An important goal: develop useful **reference material**

Privacy-Enhancing Cryptography

The NIST PEC project (revived in 2019):

- Interested in privacy promotable by cryptography
- Zero-knowledge proofs (ZKPs), secure multiparty computation (SMPC), ...
- Wants to keep up to date with, and support, external initiatives
- An important goal: develop useful **reference material**

Reference material:

- **Assess** the state of things (feasibility, cost, ...) in a particular area
- **Motivate** application / proofs-of-concept in use-cases
- **Frame** the further development of standards, and future discussions
- **Enable** interoperability for companies that want to do things now

ZKPs / ZKProof

ZKProof (https://www.zkproof.org):

- effort towards the standardization of zero-knowledge proofs (ZKPs)
- open initiative of industry and academia
- produces open documentation — fits the “reference materials” approach

ZKPs / ZKProof

ZKProof (https://www.zkproof.org):

- effort towards the standardization of zero-knowledge proofs (ZKPs)
- open initiative of industry and academia
- produces open documentation — fits the "reference materials" approach

ZKProof asked NIST for feedback → **The PEC team engaged:**

- ported docs (proceedings 1st workshop) to LaTeX and sent comments
- proposed an editorial process to develop a community reference
- participated in the 2nd workshop
- now producing contributions

ZKPs / ZKProof

ZKProof (https://www.zkproof.org):

- effort towards the standardization of zero-knowledge proofs (ZKPs)
- open initiative of industry and academia
- produces open documentation — fits the "reference materials" approach

ZKProof asked NIST for feedback → **The PEC team engaged:**

- ported docs (proceedings 1st workshop) to LaTeX and sent comments
- proposed an editorial process to develop a community reference
- participated in the 2nd workshop
- now producing contributions

In summary:

- This is one activity supporting advanced cryptography standardization
- We also want to promote SMPC (...)

Outline

"Advanced cryptography"?

“Advanced cryptography”?

What makes it advanced (regarding standardization)?

- Protocols (with distributed systems) instead of single-side primitives?
- Many paradigms/options to choose from?
- Complex techniques not previously standardized?
- Uncertainty of adoption or what approach to take?

"Advanced cryptography"?

What makes it advanced (regarding standardization)?

- Protocols (with distributed systems) instead of single-side primitives?
- Many paradigms/options to choose from?
- Complex techniques not previously standardized?
- Uncertainty of adoption or what approach to take?

Two notes:

- What is advanced today may be basic tomorrow!?
- Perhaps we need (more) "advanced standardization" processes?

“Advanced cryptography”?

What makes it advanced (regarding standardization)?

- Protocols (with distributed systems) instead of single-side primitives?
- Many paradigms/options to choose from?
- Complex techniques not previously standardized?
- Uncertainty of adoption or what approach to take?

Two notes:

- What is advanced today may be basic tomorrow!?
- Perhaps we need (more) “advanced standardization” processes?

Call for collaboration. We want to collaborate with open and transparent processes towards standardization of advanced cryptography. Let us know.

Concluding remarks

1. NIST is interested in the development of "advanced cryptography" (secure implementations, technology adoption, interoperability)
2. The standardization **development process** matters
3. Not everything should be standardized by NIST ... but some things should
4. The set of final standards can be of several types
5. Standardization considerations go beyond technical security
6. Humans are in the equation ... collaboration inter-stakeholders is essential
7. NIST is currently active in threshold crypto and PEC

The test of time

Which of today's developing standards will remain, 70 years from now, as building blocks of advanced crypto?

The test of time

Which of today's developing standards will remain, 70 years from now, as building blocks of advanced crypto?

Photo in 1948 *

Photo in 2018: https://www.nist.gov/sites/default/files/documents/2018/06/15/nist_gaithersburg_master_plan_may_7_2018.pdf

The NIST Stone Test Wall: "Constructed [in 1948] to study the performance of stone subjected to weathering. It contains 2352 individual samples of stone, of which 2032 are domestic stone from 47 states, and 320 are stones from 16 foreign countries."

* https://www.nist.gov/el/materials-and-structural-systems-division-73100/nist-stone-wall

Thank you for your attention

A perspective on standardization of advanced cryptography at NIST

Presentation at ACS'19
Advanced **C**ryptography **S**tandardization Workshop
August 18, 20**19** @ Santa Barbara, California, USA

Feedback is very much appreciated

luis.brandao@nist.gov

List of Frames