Tackling advanced cryptography ... toward standards?

Presented* at SSR 2023 & STAP'23 (joint session)

April 22, 2023 | Lyon (France)

SSR 2023: Security Standardisation Research Conference

STAP'23: Symmetric Techniques for Advanced Protocols

* Luís Brandão: At NIST as a Foreign Guest Researcher (non-employee), Contractor from Strativia.

Expressed opinions are from the speaker and should not be construed as official NIST views. Joint work with René Peralta.

Outline

1. On a few used words
2. NIST Intro
3. NIST PEC and Threshold Crypto
4. The Threshold Call
5. Interaction and Feedback

(Slides will be made publicly available)

NIST = National Institute of Standards and Technology. PEC = Privacy-Enhancing Crytpography.

Outline

1. On a few used words
2. NIST Intro
3. NIST PEC and Threshold Crypto
4. The Threshold Call
5. Interaction and Feedback

NIST = National Institute of Standards and Technology. PEC = Privacy-Enhancing Crytpography.

Parsing some words in the title

"*Tackling* ***advanced*** *cryptography ... toward* ***standards?***"

- "**Standards**":

Parsing some words in the title

"*Tackling **advanced** cryptography ... toward **standards?***"

- "**Standards**":
  - **Standard** ... as a specification that can or should (or should not) be followed
  - **Standard**ization ... as a process (including prior to the "standard")
  - **Standard**ization bodies/communities, **standard**ization-related workshops

Parsing some words in the title

"*Tackling **advanced** cryptography ... toward **standards?***"

- "**Standards**":
  - **Standard** ... as a specification that can or should (or should not) be followed
  - **Standard**ization ... as a process (including prior to the "standard")
  - **Standard**ization bodies/communities, **standard**ization-related workshops
- "**Advanced**": contextual, relative to something (next slide)

Parsing some words in the title

"*Tackling **advanced** cryptography ... toward **standards?***"

- "**Standards**":
  - **Standard** ... as a specification that can or should (or should not) be followed
  - **Standard**ization ... as a process (including prior to the "standard")
  - **Standard**ization bodies/communities, **standard**ization-related workshops
- "**Advanced**": contextual, relative to something (next slide)
- "**?**": ? ? ? many questions

Parsing some words in the title

"*Tackling **advanced** cryptography ... toward **standards?***"

- "**Standards**":
  - **Standard** ... as a specification that can or should (or should not) be followed
  - **Standard**ization ... as a process (including prior to the "standard")
  - **Standard**ization bodies/communities, **standard**ization-related workshops
- "**Advanced**": contextual, relative to something (next slide)
- "**?**": ? ? ? many questions
- **Others:** "tackling", "cryptography", "...", "toward"

“Advanced” cryptography

Tradition: standards for building blocks for “traditional” data security.

	Traditional	
Data status	*At rest* or *In transit*	
Operation being secured	Storage or Communication	
Example **crypto** primitives	Encryption, Signatures, Hashing	
NIST crypto standards today?	**Yes**	

“Advanced” cryptography

Tradition: standards for building blocks for “traditional” data security.

	Traditional	**Advanced**
Data status	*At rest* or *In transit*	*In use*
Operation being secured	Storage or Communication	**Computation**
Example **crypto** primitives	Encryption, Signatures, Hashing	MPC, HE, ZKP
NIST crypto standards today?	**Yes**	**No**

Legend: HE = homomorphic encryption; MP = multi-party; MPC = (secure) MP computation; ZKP = zero-knowledge proof

"Advanced" cryptography

Tradition: standards for building blocks for "traditional" data security.

	Traditional	**Advanced**
Data status	*At rest* or *In transit*	*In use*
Operation being secured	Storage or Communication	**Computation**
Example **crypto** primitives	Encryption, Signatures, Hashing	MPC, HE, ZKP
NIST crypto standards today?	**Yes**	**No**

Legend: HE = homomorphic encryption; MP = multi-party; MPC = (secure) MP computation; ZKP = zero-knowledge proof

Modernization: *advanced crypto* (enhanced features, composition, distributed systems, ...)

Outline

1. On a few used words
2. NIST Intro
3. NIST PEC and Threshold Crypto
4. The Threshold Call
5. Interaction and Feedback

NIST = National Institute of Standards and Technology. PEC = Privacy-Enhancing Crytpography.

NIST: Laboratories → Divisions → Groups

- **Non-regulatory** federal agency (@ U.S. Dept. Commerce)
- **Mission:** ... innovation ... industrial competitiveness ... measurement science, standards, and technology ... economic security ... quality of life.

NIST name and address plate (source: nist.gov)

NIST: Laboratories → Divisions → Groups

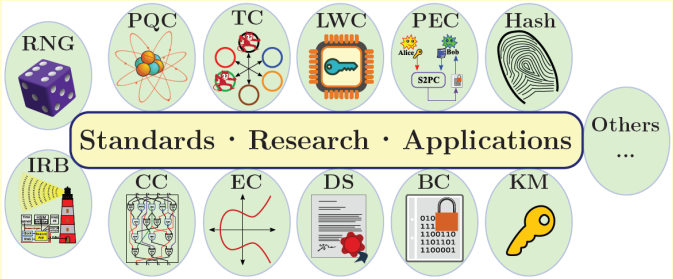
- **Non-regulatory** federal agency (@ U.S. Dept. Commerce)
- **Mission:** ... innovation ... industrial competitiveness ... measurement science, standards, and technology ... economic security ... quality of life.

NIST name and address plate (source: nist.gov)

INFORMATION TECHNOLOGY LABORATORY → **Computer Security Division (CSD):**

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

Activities in the “Crypto” Group



- **Public documentation:** FIPS; Special Publications (SP 800); NIST Reports (IR).
- **International cooperation:** government, industry, academia, standardization bodies.

Legend: **BC** = **B**lock **C**iphers. **CC** = **C**ircuit **C**omplexity. **Crypto** = **Crypto**graphy. **DS** = **D**igital **S**ignatures. **EC** = **E**lliptic **C**urves. **FIPS** = **F**ederal **I**nformation **P**rocessing **S**tandards. **IR** = **I**nternal or **I**nteragency (denoting that the public NIST report was developed internally at NIST or in an interagency collaboration, respectively. **IRB** = **I**nteroperable **R**andomness **B**eacons. **KM** = **K**ey **M**anagement. **LWC** = **L**ightweight **C**rypto. **PEC** = **P**rivacy-**E**nhancing **C**rypto. **PQC** = **P**ost-**Q**uantum **C**rypto. **RNG** = **R**andom-**N**umber **G**eneration. **SP 800** = **S**pecial **P**ublications in Computer Security. **TC** = [Multi-Party] **T**hreshold **C**rypto).

More details at https://www.nist.gov/itl/csd/cryptographic-technology

Some examples of NIST Crypto Projects

- **PQC:** [standardization] "**post-quantum**" signatures and key-encapsulation
- **LWC:** [standardization] "**lightweight**" **A**uth. **E**nc. w/ **A**ssoc. **D**ata, and hashing

Legend: **AEAD** = **A**uth[enticated] **E**nc[ryption] w[ith] **A**ssoc[iated] **D**ata. **CTG** = **C**ryptographic **T**echnology **G**roup. **LWC** = **L**ightweight **C**ryptography. **MPTC** = **M**ulti-**P**arty **T**hreshold **C**ryptography. **NIST** = **N**ational **I**nstitute of **S**tandards and **T**echnology. **PEC** = **P**rivacy-**E**nhancing **C**ryptography. **PQC** = **P**ost-**Q**uantum **C**ryptography.

Some examples of NIST Crypto Projects

- **PQC:** [standardization] "**post-quantum**" signatures and key-encapsulation
- **LWC:** [standardization] "**lightweight**" **A**uth. **E**nc. w/ **A**ssoc. **D**ata, and hashing
- **PEC:** [exploratory] "**privacy-enhancing**" (advanced) features/functionalities
- **MPTC:** [exploratory] "**multi-party threshold**" schemes for crypto primitives
- **...** various others https://www.nist.gov/itl/csd/cryptographic-technology

Legend: **AEAD** = **A**uth[enticated] **Enc**[ryption] w[ith] **Assoc**[iated] **D**ata. **CTG** = **C**ryptographic **T**echnology **G**roup. **LWC** = **L**ightweight **C**ryptography. **MPTC** = **M**ulti-**P**arty **T**hreshold **C**ryptography. **NIST** = **N**ational **I**nstitute of **S**tandards and **T**echnology. **PEC** = **P**rivacy-**E**nhancing **C**ryptography. **PQC** = **P**ost-**Q**uantum **C**ryptography.

Some examples of NIST Crypto Projects

- **PQC:** [standardization] "**post-quantum**" signatures and key-encapsulation
- **LWC:** [standardization] "**lightweight**" **A**uth. **E**nc. w/ **A**ssoc. **D**ata, and hashing
- **PEC:** [exploratory] "**privacy-enhancing**" (advanced) features/functionalities
- **MPTC:** [exploratory] "**multi-party threshold**" schemes for crypto primitives
- **...** various others https://www.nist.gov/itl/csd/cryptographic-technology

The "Threshold Call" (from MPTC+PEC): to gather **reference material** for public analysis ... aiming for **recommendations** (in a 1st phase), including about PEC.

Legend: **AEAD** = **A**uth[enticated] **E**nc[ryption] w[ith] **A**ssoc[iated] **D**ata. **CTG** = **C**ryptographic **T**echnology **G**roup. **LWC** = **L**ightweight **C**ryptography. **MPTC** = **M**ulti-**P**arty **T**hreshold **C**ryptography. **NIST** = **N**ational **I**nstitute of **S**tandards and **T**echnology. **PEC** = **P**rivacy-**E**nhancing **C**ryptography. **PQC** = **P**ost-**Q**uantum **C**ryptography.

Some NIST Crypto "Standardization" Updates

- **Post-Quantum (PQC):** [Aim] Draft Standards of selected schemes (Summer 2023).
  – Public call (2022) for more PQ-signatures (submit by June 1st).
- **Lightweight (LWC):** Feb 2023, selected ASCON (**A**uth. **E**nc. w/ **A**ssoc. **D**ata; hash).
  – Workshop on June 21–22 (submit by May 1st). [Aim] Draft Standard (late 2023).
- **Threshold Call (MPTC/PEC):** Call Draft (Jan. 25th); public comments (April 10th).
  – [Aim] Call finalized in 2023 2nd half; submissions deadline within 2024 1st half.

Legend: AEAD = **A**uth[enticated] **Enc**[ryption] w[ith] **Assoc**[iated] **D**ata. **EdDSA** = **Ed**wards-Curve **D**igital **S**ignature **A**lgorithm. **Feb** = **Feb**ruary. **FIPS** = **F**ederal **I**nformation **P**rocessing **S**tandards. **Jan** = **Jan**uary. **Oct** = **Oct**ober. **RBG** = **R**andom **B**it **G**eneration. **SP** = **S**pecial **P**ublication (800 series) [in Computer Security]

Some NIST Crypto "Standardization" Updates

- **Post-Quantum (PQC):** [Aim] Draft Standards of selected schemes (Summer 2023).
  – Public call (2022) for more PQ-signatures (submit by June 1st).
- **Lightweight (LWC):** Feb 2023, selected ASCON (**A**uth. **E**nc. w/ **A**ssoc. **D**ata; hash).
  – Workshop on June 21–22 (submit by May 1st). [Aim] Draft Standard (late 2023).
- **Threshold Call (MPTC/PEC):** Call Draft (Jan. 25th); public comments (April 10th).
  – [Aim] Call finalized in 2023 2nd half; submissions deadline within 2024 1st half.
- **Crypto Publication Review:** Revising Standards (FIPS & SP) older than 5 years.
- **FIPS 186-5 (signatures, including EdDSA):** Standard (final) published Feb. 7th.
- **RBG workshop** (May 30th); **Cipher Modes workshop** (Oct. 3rd; submit by July 1st).

Legend: **AEAD** = **A**uth[enticated] **Enc**[ryption] w[ith] **Assoc**[iated] **D**ata. **EdDSA** = **Ed**wards-Curve **D**igital **S**ignature **A**lgorithm. **Feb** = **Feb**ruary. **FIPS** = **F**ederal **I**nformation **P**rocessing **S**tandards. **Jan** = **Jan**uary. **Oct** = **Oct**ober. **RBG** = **R**andom **B**it **G**eneration. **SP** = **S**pecial **P**ublication (800 series) [in Computer Security]

Some NIST Crypto "Standardization" Updates

- **Post-Quantum (PQC):** [Aim] Draft Standards of selected schemes (Summer 2023).
  – Public call (2022) for more PQ-signatures (submit by June 1st).
- **Lightweight (LWC):** Feb 2023, selected ASCON (**A**uth. **E**nc. w/ **A**ssoc. **D**ata; hash).
  – Workshop on June 21–22 (submit by May 1st). [Aim] Draft Standard (late 2023).
- **Threshold Call (MPTC/PEC):** Call Draft (Jan. 25th); public comments (April 10th).
  – [Aim] Call finalized in 2023 2nd half; submissions deadline within 2024 1st half.
- **Crypto Publication Review:** Revising Standards (FIPS & SP) older than 5 years.
- **FIPS 186-5 (signatures, including EdDSA):** Standard (final) published Feb. 7th.
- **RBG workshop** (May 30th); **Cipher Modes workshop** (Oct. 3rd; submit by July 1st).

Legend: **AEAD** = **A**uth[enticated] **Enc**[ryption] w[ith] **Assoc**[iated] **D**ata. **EdDSA** = **Ed**wards-Curve **D**igital **S**ignature **A**lgorithm. **Feb** = **Feb**ruary. **FIPS** = **F**ederal **I**nformation **P**rocessing **S**tandards. **Jan** = **Jan**uary. **Oct** = **Oct**ober. **RBG** = **R**andom **B**it **G**eneration. **SP** = **S**pecial **P**ublication (800 series) [in Computer Security]

Outline

1. On a few used words
2. NIST Intro
3. NIST PEC and Threshold Crypto
4. The Threshold Call
5. Interaction and Feedback

NIST = National Institute of Standards and Technology. PEC = Privacy-Enhancing Crytpography.

Privacy-Enhancing Cryptography (PEC): NIST Project

Cryptography (that can be) used to **enhance privacy**.

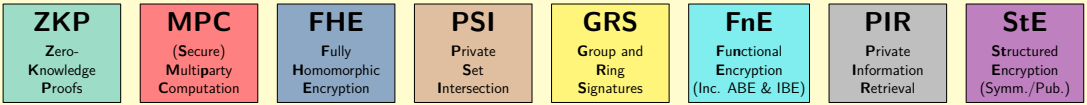
(emphasis on non-standardized tools)

Privacy-Enhancing Cryptography (PEC): NIST Project

Cryptography (that can be) used to **enhance privacy**.
(emphasis on non-standardized tools)

Goals:

1. Accompany the progress of **emerging *PEC tools***.



Legend: ABE: attribute-based encryption. IBE: identity-based encryption. PEC: privacy-enhancing cryptography. Symm./pub.: symmetric-key or public-key based.

Privacy-Enhancing Cryptography (PEC): NIST Project

Cryptography (that can be) used to **enhance privacy**.
(emphasis on non-standardized tools)

Goals:

1. Accompany the progress of **emerging *PEC tools***.
2. Promote development of PEC **reference material**.

PEC tools

STPPA (series of talks)

PEC use-case suite

Threshold schemes

ZKProof collaboration

Encounter metrics

Email list (PEC Forum)

https://csrc.nist.gov/projects/pec

ZKP	**MPC**	**FHE**	**PSI**	**GRS**	**FnE**	**PIR**	**StE**
Zero- **K**nowledge **P**roofs	(**S**ecure) **M**ultiparty **C**omputation	**F**ully **H**omomorphic **E**ncryption	**P**rivate **S**et **I**ntersection	**G**roup and **R**ing **S**ignatures	**Fun**ctional **E**ncryption (Inc. ABE & IBE)	**P**rivate **I**nformation **R**etrieval	**St**ructured **E**ncryption (Symm./Pub.)

Legend: ABE: attribute-based encryption. IBE: identity-based encryption. PEC: privacy-enhancing cryptography. Symm./pub.: symmetric-key or public-key based.

Privacy-Enhancing Cryptography (PEC): NIST Project

Cryptography (that can be) used to **enhance privacy**.
(emphasis on non-standardized tools)

Goals:

1. Accompany the progress of **emerging *PEC tools***.
2. Promote development of PEC **reference material**.
3. **Exploratory work** to assess potential for recommendations, standardization; ...

PEC tools
STPPA (series of talks)
PEC use-case suite
Threshold schemes
ZKProof collaboration
Encounter metrics
Email list (PEC Forum)

https://csrc.nist.gov/projects/pec

ZKP	**MPC**	**FHE**	**PSI**	**GRS**	**FnE**	**PIR**	**StE**
Zero- **K**nowledge **P**roofs	(**S**ecure) **M**ultiparty **C**omputation	**F**ully **H**omomorphic **E**ncryption	**P**rivate **S**et **I**ntersection	**G**roup and **R**ing **S**ignatures	**Fun**ctional **E**ncryption (Inc. ABE & IBE)	**P**rivate **I**nformation **R**etrieval	**St**ructured **E**ncryption (Symm./Pub.)

Legend: ABE: attribute-based encryption. IBE: identity-based encryption. PEC: privacy-enhancing cryptography. Symm./pub.: symmetric-key or public-key based.

Multi-Party Threshold Cryptography: NIST project

Cryptographic primitives:


Signing

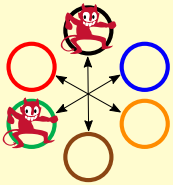
Encryption


KeyGen


Hashing

etc.

Threshold schemes (for cryptographic primitives):



https://csrc.nist.gov/projects/threshold-cryptography

Multi-Party Threshold Cryptography: NIST project

Cryptographic primitives:


Signing

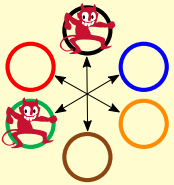
Encryption


KeyGen


Hashing

etc.

Threshold schemes (for cryptographic primitives):



1. Split (**secret-share**) the secret/private-key across multiple parties.
2. Use **MPC** to perform needed operation (with split key), e.g., sign.
   (MPC = secure multiparty computation ... or call it "Threshold Cryptography")

https://csrc.nist.gov/projects/threshold-cryptography

Multi-Party Threshold Cryptography: NIST project

Cryptographic primitives:


Signing

Encryption

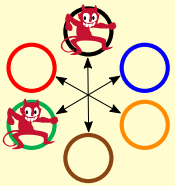

KeyGen


Hashing

etc.

Threshold schemes (for cryptographic primitives):

1. Split (**secret-share**) the secret/private-key across multiple parties.
2. Use **MPC** to perform needed operation (with split key), e.g., sign.
   (MPC = secure multiparty computation ... or call it "Threshold Cryptography")



- **"Threshold" (f):** Operation is secure if number of corrupted parties is $\leq f$.
- **Decentralized** trust about key **(not reconstructed)**: avoids single-point of failure.

https://csrc.nist.gov/projects/threshold-cryptography

Why care about threshold schemes?

Strong feasibility result (theory): can be applied to any cryptographic primitive.

But, in practice, some primitives are ***threshold-friendlier**** than others.

(* i.e., informally, easier in practice to thresholdize, or amenable to "better" threshold schemes)

Why care about threshold schemes?

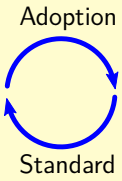
Strong feasibility result (theory): can be applied to any cryptographic primitive.

But, in practice, some primitives are ***threshold-friendlier**** than others.

(* i.e., informally, easier in practice to thresholdize, or amenable to "better" threshold schemes)

- Standards "should" focus on high need and potential for **adoption**
- **Threshold friendliness:** desirable feature $\rightarrow$ improves **adoptability**

  (e.g., determ. vs. prob. threshold EdDSA/Schnorr signatures [NISTIR 8214B ipd])



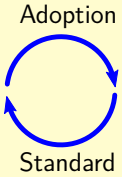
Why care about threshold schemes?

Strong feasibility result (theory): can be applied to any cryptographic primitive.

But, in practice, some primitives are ***threshold-friendlier**** than others.

(* i.e., informally, easier in practice to thresholdize, or amenable to "better" threshold schemes)

- Standards "should" focus on high need and potential for **adoption**
- **Threshold friendliness:** desirable feature → improves **adoptability**
  (e.g., determ. vs. prob. threshold EdDSA/Schnorr signatures [NISTIR 8214B ipd])



How to explore the threshold space?:

- applicable to a **wide scope** of primitives
- bringing **added complexity**

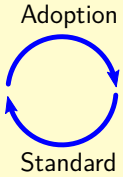
Why care about threshold schemes?

Strong feasibility result (theory): can be applied to any cryptographic primitive.

But, in practice, some primitives are ***threshold-friendlier**** than others.

(* i.e., informally, easier in practice to thresholdize, or amenable to "better" threshold schemes)

- Standards "should" focus on high need and potential for **adoption**
- **Threshold friendliness:** desirable feature → improves **adoptability**
  (e.g., determ. vs. prob. threshold EdDSA/Schnorr signatures [NISTIR 8214B ipd])



How to explore the threshold space?:

- applicable to a **wide scope** of primitives
- bringing **added complexity**

Next section: A public **Call** for reference material ... toward **recommendations**.

Outline

1. On a few used words
2. NIST Intro
3. NIST PEC and Threshold Crypto
4. The Threshold Call
5. Interaction and Feedback

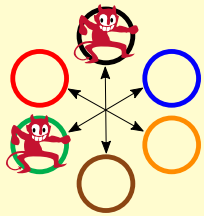
NIST = National Institute of Standards and Technology. PEC = Privacy-Enhancing Crytpography.

The NIST Call for *Multi-Party Threshold Schemes*

NISTIR 8214C ipd (**i**nitial **p**ublic **d**raft)

Email public comments to nistir-8214C-comments@nist.gov, by **2023-April-10**.

Calling for threshold schemes for diverse primitives:

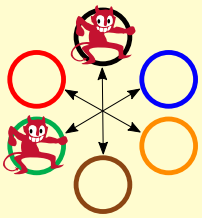


The NIST Call for *Multi-Party Threshold Schemes*

NISTIR 8214C ipd (**i**nitial **p**ublic **d**raft)

Email public comments to nistir-8214C-comments@nist.gov, by **2023-April-10**.

Calling for threshold schemes for diverse primitives:



- **Cat1: Selected NIST-standardized primitives**
  - In EdDSA, ECDSA, RSA, AES, ECC-KE, ...
- **Cat2: Primitives in schemes not standardized by NIST**
  - *Threshold friendly*, and possibly with advanced features (e.g., in FHE, IBE, ZKP)

Legend: AES = **A**dvanced **E**ncryption **S**tandard. EC = **E**lliptic **c**urve. ECC-KE = **EC** **c**ryptography (based) **k**ey-exchange. FHE = **f**ully-**h**omomorphic **e**ncryption. EdDSA = **Ed**wards-Curve **d**igital **s**ignature **a**lgorithm. ECDSA = **EC** **d**igital **s**ignature **a**lgorithm. IBE = **i**dentity-**b**ased **e**ncryption. NIST = **N**ational **I**nstitute of **S**tandards and **T**echnology. RSA = **R**ivest–**S**hamir–**A**dleman. ZKP = **z**ero-**k**nowledge **p**roofs.

Category Cat1 of NIST Call for Multi-Party Threshold Schemes

Too many acronyms, we know. (Legend further below)

Subcategory: Type
C1.1: **Signing**
C1.2: **PKE**
C1.3: **2KA**
C1.4: **Symmetric**
C1.5: **Keygen**

Legend: **2KA**: **pair**-wise **k**ey-agreement. **2KE**: **pair**-wise **k**ey-establisment. **AES**: **A**dvanced **E**ncryption **S**tandard. **CDH**: cofactor **D**iffie–**H**ellman. **ECC**: **E**lliptic-curve **c**ryptography (or, if used as an adjective, **EC**-based). **ECDSA**: **E**lliptic-curve **D**igital **S**ignature **A**lgorithm. **EdDSA**: **Ed**wards-curve **D**igital **S**ignature **A**lgorithm. **E**lliptic-curve based **K**ey-**E**stablishment. **FIPS**: **F**ederal **I**nformation **P**rocessing **S**tandard. **KC**: **K**ey-confirmtion. **KDM**: **K**ey-**d**erivation **m**echanism. **Keygen**: **K**ey-**gen**eration. **MQV**: Menezes-Qu-Vanstone. **PKE**: **p**ublic-**k**ey **e**ncryption. **RSA**: **R**ivest–**S**hamir–**A**dleman (signature and encryption schemes). **RSADSA**: **RSA** digital signature algorithm. **SP 800**: **S**pecial **P**ublication (in Computer Security). **Note:** In the 2nd column, each item within a subcategory is itself called a family of specifications, since it may include diverse primitives or modes/variants.

Category Cat1 of NIST Call for Multi-Party Threshold Schemes

Too many acronyms, we know. (Legend further below)

Subcategory: Type	Families of specifications	NIST references
C1.1: **Signing**	EdDSA sign, ECDSA sign, RSADSA sign	FIPS 186-5 (see also NISTIR 8214B)

Legend: 2KA: **pair**-wise **k**ey-agreement. 2KE: **pair**-wise **k**ey-establisment. **AES**: **A**dvanced **E**ncryption **S**tandard. **CDH**: cofactor **D**iffie–**H**ellman. **ECC**: **E**lliptic-curve **c**ryptography (or, if used as an adjective, **EC**-based). **ECDSA**: **E**lliptic-curve **D**igital **S**ignature **A**lgorithm. **EdDSA**: **Ed**wards-curve **D**igital **S**ignature **A**lgorithm. **E**lliptic-curve based **K**ey-**E**stablishment. **FIPS**: **F**ederal **I**nformation **P**rocessing **S**tandard. **KC**: **K**ey-confirmtion. **KDM**: **K**ey-**d**erivation **m**echanism. **Keygen**: **K**ey-**gen**eration. **MQV**: Menezes-Qu-Vanstone. **PKE**: **p**ublic-**k**ey **e**ncryption. **RSA**: **R**ivest–**S**hamir–**A**dleman (signature and encryption schemes). **RSADSA**: **RSA** digital signature algorithm. **SP 800**: **S**pecial **P**ublication (in Computer Security). **Note:** In the 2nd column, each item within a subcategory is itself called a family of specifications, since it may include diverse primitives or modes/variants.

Category Cat1 of NIST Call for Multi-Party Threshold Schemes

Too many acronyms, we know. (Legend further below)

Subcategory: Type	Families of specifications	NIST references
C1.2: **PKE**	RSA decrypt, RSA encrypt (a secret value)	SP 800-56B Rev2

Legend: **2KA**: **pair**-wise **k**ey-agreement. **2KE**: **pair**-wise **k**ey-establisment. **AES**: **A**dvanced **E**ncryption **S**tandard. **CDH**: cofactor **D**iffie–**H**ellman. **ECC**: **E**lliptic-**c**urve **c**ryptography (or, if used as an adjective, **EC**-based). **ECDSA**: **E**lliptic-**c**urve **D**igital **S**ignature **A**lgorithm. **EdDSA**: **Ed**wards-curve **D**igital **S**ignature **A**lgorithm. **E**lliptic-**c**urve based **K**ey-**E**stablishment. **FIPS**: **F**ederal **I**nformation **P**rocessing **S**tandard. **KC**: **K**ey-**c**onfirmtion. **KDM**: **K**ey-**d**erivation **m**echanism. **Keygen**: **K**ey-**gen**eration. **MQV**: Menezes-Qu-Vanstone. **PKE**: **p**ublic-**k**ey **e**ncryption. **RSA**: **R**ivest–**S**hamir–**A**dleman (signature and encryption schemes). **RSADSA**: **RSA** digital signature algorithm. **SP 800**: **S**pecial **P**ublication (in Computer Security). **Note:** In the 2nd column, each item within a subcategory is itself called a family of specifications, since it may include diverse primitives or modes/variants.

Category Cat1 of NIST Call for Multi-Party Threshold Schemes

Too many acronyms, we know. (Legend further below)

Subcategory: Type	Families of specifications	NIST references
C1.4: **Symmetric**	AES encipher/decipher, KDM/KC (for 2KE)	FIPS 197, SP 800-56C Rev2, ...

Legend: **2KA**: **pair**-wise **k**ey-agreement. **2KE**: **pair**-wise **k**ey-establisment. **AES**: **A**dvanced **E**ncryption **S**tandard. **CDH**: cofactor **D**iffie–**H**ellman. **ECC**: **E**lliptic-curve **c**ryptography (or, if used as an adjective, **EC**-based). **ECDSA**: **E**lliptic-curve **D**igital **S**ignature **A**lgorithm. **EdDSA**: **Ed**wards-curve **D**igital **S**ignature **A**lgorithm. **E**lliptic-curve based **K**ey-**E**stablishment. **FIPS**: **F**ederal **I**nformation **P**rocessing **S**tandard. **KC**: **K**ey-confirmtion. **KDM**: **K**ey-derivation **m**echanism. **Keygen**: **K**ey-**gen**eration. **MQV**: Menezes-Qu-Vanstone. **PKE**: **p**ublic-**k**ey **e**ncryption. **RSA**: **R**ivest–**S**hamir–**A**dleman (signature and encryption schemes). **RSADSA**: **RSA** digital signature algorithm. **SP 800**: **S**pecial **P**ublication (in Computer Security). **Note:** In the 2nd column, each item within a subcategory is itself called a family of specifications, since it may include diverse primitives or modes/variants.

Category Cat1 of NIST Call for Multi-Party Threshold Schemes

Too many acronyms, we know. (Legend further below)

Subcategory: Type	Families of specifications	NIST references
C1.1: **Signing**	EdDSA sign, ECDSA sign, RSADSA sign	FIPS 186-5 (see also NISTIR 8214B)
C1.2: **PKE**	RSA decrypt, RSA encrypt (a secret value)	SP 800-56B Rev2
C1.3: **2KA**	ECC-CDH, ECC-MQV	SP 800-56A Rev3
C1.4: **Symmetric**	AES encipher/decipher, KDM/KC (for 2KE)	FIPS 197, SP 800-56C Rev2, ...
C1.5: **Keygen**	ECC keygen, RSA keygen, bitstring keygen	(corresponding references above)

Legend: **2KA**: **pair**-wise **k**ey-agreement. **2KE**: **pair**-wise **k**ey-establisment. **AES**: **A**dvanced **E**ncryption **S**tandard. **CDH**: cofactor **D**iffie–**H**ellman. **ECC**: **E**lliptic-curve **c**ryptography (or, if used as an adjective, **EC**-based). **ECDSA**: **E**lliptic-curve **D**igital **S**ignature **A**lgorithm. **EdDSA**: **Ed**wards-curve **D**igital **S**ignature **A**lgorithm. **E**lliptic-curve based **K**ey-**E**stablishment. **FIPS**: **F**ederal **I**nformation **P**rocessing **S**tandard. **KC**: **K**ey-confirmtion. **KDM**: **K**ey-**d**erivation **m**echanism. **Keygen**: **K**ey-**gen**eration. **MQV**: Menezes-Qu-Vanstone. **PKE**: public-**k**ey **e**ncryption. **RSA**: **R**ivest–**S**hamir–**A**dleman (signature and encryption schemes). **RSADSA**: **RSA** digital signature algorithm. **SP 800**: **S**pecial **P**ublication (in Computer Security). **Note:** In the 2nd column, each item within a subcategory is itself called a family of specifications, since it may include diverse primitives or modes/variants.

Category Cat2 of the NIST “Threshold” Call

Subcategory: Type
C2.1: **Signing**
C2.2: **PKE**
C2.3: **Key-agreem.**
C2.4: **Symmetric**
C2.5: **Keygen**

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Category Cat2 of the NIST "Threshold" Call

TF = **t**hreshold **f**riendly. QR = **q**uantum **r**esistant.

Subcategory: Type	Example types of schemes	Example primitives
C2.1: **Signing**	TF succinct & verifiably-deterministic signatures	Sign
|	TF-QR signatures	Sign

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Category Cat2 of the NIST “Threshold” Call

Subcategory: Type
C2.6: **Advanced**
C2.7: **ZKPoK**
C2.8: **Gadgets**

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Category Cat2 of the NIST "Threshold" Call

TF = **t**hreshold **f**riendly. QR = **q**uantum **r**esistant.

Subcategory: Type	Example types of schemes	Example primitives
C2.6: **Advanced**	TF-QR fully-homomorphic encryption	Decryption; Keygen
|	TF identity-based and attribute-based encryption	Decryption; Keygens

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Category Cat2 of the NIST “Threshold” Call

Subcategory: Type	Example types of schemes	Example primitives
C2.7: **ZKPoK**	**Z**ero-**k**nowledge **p**roof **o**f **k**nowledge of private key	ZKPoK.Generate

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Category Cat2 of the NIST “Threshold” Call

Subcategory: Type	Example types of schemes	Example primitives
C2.8: **Gadgets**	Garbled circuit (GC)	GC.generate; GC.evaluate

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Category Cat2 of the NIST "Threshold" Call

TF = **t**hreshold **f**riendly. QR = **q**uantum **r**esistant.

Subcategory: Type	Example types of schemes	Example primitives
C2.1: **Signing**	TF succinct & verifiably-deterministic signatures	Sign
|	TF-QR signatures	Sign
C2.2: **PKE**	TF-QR **p**ublic-**k**ey **e**ncryption (PKE)	Decrypt/Encrypt (a secret value)
C2.3: **Key-agreem.**	TF Low-round multi-party key-agreement	Single-party primitives
C2.4: **Symmetric**	TF blockcipher/PRP	Encipher/decipher
|	TF key-derivation / key-confirmation	PRF and hash function
C2.5: **Keygen**	Any of the above	Keygen
C2.6: **Advanced**	TF-QR fully-homomorphic encryption	Decryption; Keygen
|	TF identity-based and attribute-based encryption	Decryption; Keygens
C2.7: **ZKPoK**	**Z**ero-**k**nowledge **p**roof **o**f **k**nowledge of private key	ZKPoK.Generate
C2.8: **Gadgets**	Garbled circuit (GC)	GC.generate; GC.evaluate

Note: While TF-QR is desired for any type of scheme, some examples show just TF to highlight that it is welcome even if not QR

Legend: agreem. = **agreem**ent. **Keygen** = **key-gen**eration. **PKE** = **p**ublic-**k**ey **e**ncryption. **PRF** = **p**seudorandom **f**unction [family]. **PRP** = **p**seudorandom **p**ermutation [family]. **QR** = **q**uantum **r**esistant. **TF** = **t**hreshold-**f**riendly. **ZKPoK** = **z**ero **k**nowledge **p**roof **o**f **k**nowledge.

Welcome/needed interaction with the community

1. **2023: Interactive feedback about the call:**
   a. We got 12 public comments about the ipd (compilation to appear next week)
   b. We expect/welcome subsequent feedback via the MPTC-forum
   c. Feedback will be used to improve the final call

Welcome/needed interaction with the community

1. **2023: Interactive feedback about the call:**
   a. We got 12 public comments about the ipd (compilation to appear next week)
   b. We expect/welcome subsequent feedback via the MPTC-forum
   c. Feedback will be used to improve the final call
2. **2024: Concrete submissions:**
   – Structured specification, open-source implementation, evaluation, ...
3. **2024/2025: Public scrutiny of submitted schemes:**
   – Evaluation comments (can impact subsequent recommendations)

Public comments received in first phase

#	Main topics (informal)
#1	Scope; quantum resistance.
#2	Innovation; models.
#3	Threshold motivation and alternatives; some expired patents.
#4	Mandatory checks; KAT values; implementation complexity.
#5	Fully homomorphic encryption (FHE).
#6	Threshold & oblivious pseudo-random functions (PRF); keygen; robustness; asynchronicity.
#7	Shamir Secret-sharing (safe evaluation points)
#8	Scope; keygen; adaptive security; key-refresh; bounds; broadcast; thresholds; party's state.
#9	Attribute-based encryption (ABE): ciphertext-policy, key-policy, multi-authority.
#10	All-or-nothing transform (AONT) and homomophic encryption.
#11	Implementation dependencies, KAT values in randomized multi-party runs.
#12	Robustness.

Some takeaways about the “Threshold Call”

- **Reference material**
- **Clarification toward recommendations**
- **Synergies**

Suggested reading: NISTIR 8214C ipd

NIST First Call for Multi-Party Threshold Schemes

(Initial Public Draft) [2023-Jan-25]

Some takeaways about the "Threshold Call"

- **Reference material:** The initial process is **not a competition** aiming to select a winner, but the public exposure is deemed useful.
- **Clarification toward recommendations:** The submissions and their analyses will clarify useful system models, security requirements ... and **future processes**.
- **Synergies:** Submissions of schemes in standardization development in other bodies and/or by **community efforts** are also very welcome!

Suggested reading: NISTIR 8214C ipd

NIST First Call for Multi-Party Threshold Schemes

(Initial Public Draft) [2023-Jan-25]

Outline

1. On a few used words
2. NIST Intro
3. NIST PEC and Threshold Crypto
4. The Threshold Call
5. Interaction and Feedback

NIST = National Institute of Standards and Technology. PEC = Privacy-Enhancing Crytpography.

The initial question (in the title):

Tackling advanced cryptography ... toward standards?

Yes, but ...

- it's a process (many processes)
- it *takes a village* (many villages)
- it depends on which "standards"

Thank you for your attention! Questions?

- Questions from the audience?
- (Next slide) Brainstorming questions to the audience

Threshold Call

MPTC-Forum

PEC-Forum

Tackling advanced cryptography ... toward standards?

Presented at the SSR 2023 & STAP'23 (joint session)

April 22, 2023 @ Lyon (France) — luis.brandao@nist.gov

Brainstorming on crypto standardization

1. On the **timing** & **speed** of processes: what is too soon, too late, too slow, and too fast?

Brainstorming on crypto standardization

1. On the **timing** & **speed** of processes: what is too soon, too late, too slow, and too fast?

2. What **value** is there in still pursuing new standards for **quantum-breakable** primitives?

Brainstorming on crypto standardization

1. On the **timing** & **speed** of processes: what is too soon, too late, too slow, and too fast?

2. What **value** is there in still pursuing new standards for **quantum-breakable** primitives?

3. How to handle the standardization tension between **innovation** and **interoperability**?

Brainstorming on crypto standardization

1. On the **timing** & **speed** of processes: what is too soon, too late, too slow, and too fast?
2. What **value** is there in still pursuing new standards for **quantum-breakable** primitives?
3. How to handle the standardization tension between **innovation** and **interoperability**?
4. Which crypto functionalities/features make sense to **prioritize** for standardization?

Brainstorming on crypto standardization

1. On the **timing** & **speed** of processes: what is too soon, too late, too slow, and too fast?
2. What **value** is there in still pursuing new standards for **quantum-breakable** primitives?
3. How to handle the standardization tension between **innovation** and **interoperability**?
4. Which crypto functionalities/features make sense to **prioritize** for standardization?
5. What **synergies** to aim for between academia, industry, gov and standards bodies?