Post-quantum cryptography

Daniel J. Bernstein & Tanja Lange

University of Illinois at Chicago & Ruhr University Bochum & Technische Universiteit Eindhoven

10 June 2019

Cryptography

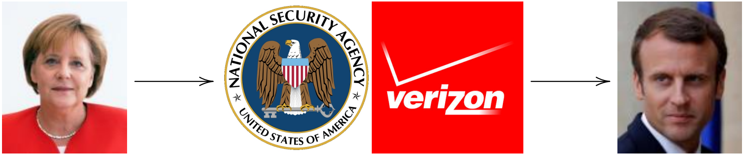
- Motivation #1: Communication channels are spying on our data.
- Motivation #2: Communication channels are modifying our data.

Cryptography

- Motivation #1: Communication channels are spying on our data.
- Motivation #2: Communication channels are modifying our data.

Cryptography

- Motivation #1: Communication channels are spying on our data.
- Motivation #2: Communication channels are modifying our data.



Sender
“Alice”

Untrustworthy network
“Eve”

Receiver
“Bob”

- Literal meaning of cryptography: “secret writing”.
- Security goal #1: **Confidentiality** despite Eve’s espionage.
- Security goal #2: **Integrity**, i.e., recognizing Eve’s sabotage.

Cryptographic applications in daily life

- Mobile phones connecting to cell towers.
- Credit cards, EC-cards, access codes for banks.
- Electronic passports; electronic ID cards.
- Internet commerce, online tax declarations, webmail.
- Facebook, Gmail, WhatsApp, iMessage on iPhone.
- Any webpage with `https`.
- Encrypted file system on iPhone: see Apple vs. FBI.

Cryptographic applications in daily life

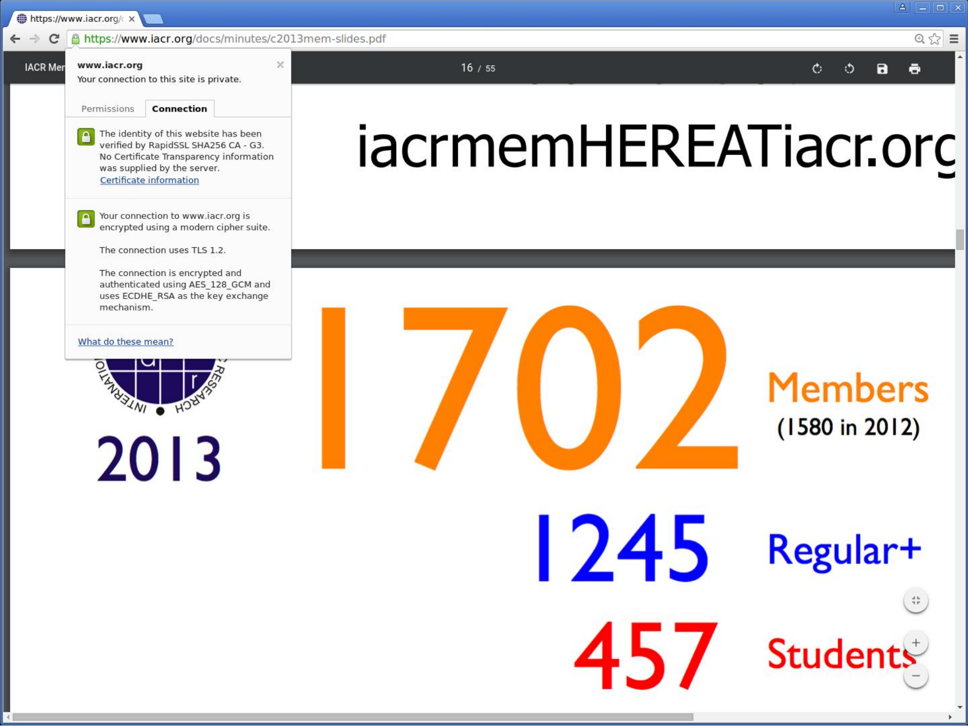
- Mobile phones connecting to cell towers.
- Credit cards, EC-cards, access codes for banks.
- Electronic passports; electronic ID cards.
- Internet commerce, online tax declarations, webmail.
- Facebook, Gmail, WhatsApp, iMessage on iPhone.
- Any webpage with `https`.
- Encrypted file system on iPhone: see Apple vs. FBI.
- PGP encrypted email, Signal, Tor, Tails, Qubes OS.
- VPN to company network.

Cryptographic applications in daily life

- Mobile phones connecting to cell towers.
- Credit cards, EC-cards, access codes for banks.
- Electronic passports; electronic ID cards.
- Internet commerce, online tax declarations, webmail.
- Facebook, Gmail, WhatsApp, iMessage on iPhone.
- Any webpage with `https`.
- Encrypted file system on iPhone: see Apple vs. FBI.
- PGP encrypted email, Signal, Tor, Tails, Qubes OS.
- VPN to company network.

Snowden in Reddit AmA

> *Arguing that you don't care about the right to privacy because you have nothing to hide is no different than saying you don't care about free speech because you have nothing to say.*



www.iacr.org

Your connection to this site is private.

The identity of this website has been verified by RapidSSL SHA256 CA - G3. No Certificate Transparency information was supplied by the server.

Certificate information

Your connection to www.iacr.org is encrypted using a modern cipher suite.

The connection uses TLS 1.2.

The connection is encrypted and authenticated using AES_128_GCM and uses ECDHE_RSA as the key exchange mechanism.

What do these mean?

iacrmemHEREATiacr.org

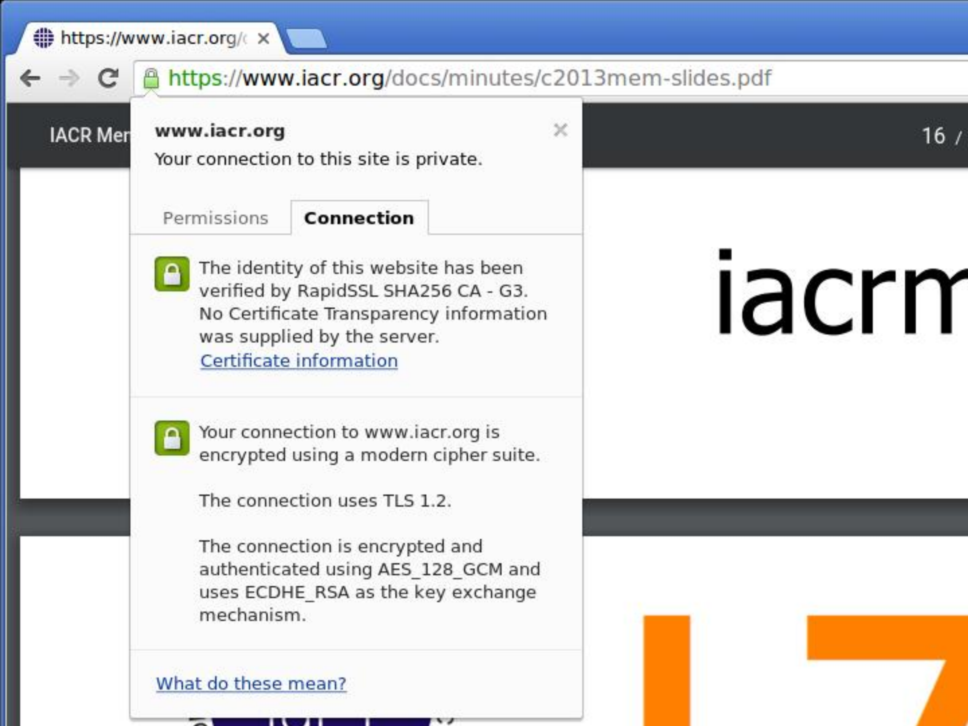
2013

1702 Members

(1580 in 2012)

1245 Regular+

457 Students



www.iacr.org

Your connection to this site is private.

The identity of this website has been verified by RapidSSL SHA256 CA - G3. No Certificate Transparency information was supplied by the server.

Certificate information

Your connection to www.iacr.org is encrypted using a modern cipher suite.

The connection uses TLS 1.2.

The connection is encrypted and authenticated using AES_128_GCM and uses ECDHE_RSA as the key exchange mechanism.

What do these mean?

Cryptographic tools

Many factors influence the security and privacy of data:

- Secure storage, physical security; access control.
- Protection against alteration of data
  $\Rightarrow$ public-key signatures, message-authentication codes.
- Protection of sensitive content against reading
  $\Rightarrow$ encryption.

Many more security goals studied in cryptography

- Protecting against denial of service.
- Stopping traffic analysis.
- Securely tallying votes.
- Searching in and computing on encrypted data.
- ...

Cryptanalysis

- Cryptanalysis is the study of security of cryptosystems.
- Breaking a system can mean that the hardness assumption was not hard or that it just was not as hard as previously assumed.
- Public cryptanalysis is ultimately constructive – ensure that secure systems get used, not insecure ones.
- Weakened crypto ultimately backfires – attacks in 2018 because of crypto wars in the 90s.
- Good arsenal of general approaches to cryptanalysis. There are some automated tools.
- This area is constantly under development; researchers revisit systems continuously.

Security assumptions

- Hardness assumptions at the basis of all public-key and essentially all symmetric-key systems result from (failed) attempts at breaking systems. Security proofs are built only on top of those assumptions.
- A solid symmetric system is required to be as strong as exhaustive key search.
- For public-key systems the best attacks are faster than exhaustive key search. Parameters are chosen to ensure that the best attack is infeasible.

Key-size recommendations

			Future System Use	
	Parameter	Legacy	Near Term	Long Term
Symmetric Key Size	k	80	128	256
Hash Function Output Size	m	160	256	512
MAC Output Size*	m	80	128	256
RSA Problem	$\ell(n) \geq$	1024	3072	15360
Finite Field DLP	$\ell(p^n) \geq$	1024	3072	15360
	$\ell(p), \ell(q) \geq$	160	256	512
ECDLP	$\ell(q) \geq$	160	256	512
Pairing	$\ell(p^{k \cdot n}) \geq$	1024	6144	15360
	$\ell(p), \ell(q) \geq$	160	256	512

- Source: ECRYPT-CSA "Algorithms, Key Size and Protocols Report" (2018).
- These recommendations take into account attacks known today.
- Use extrapolations to larger problem sizes.
- Attacker power typically limited to 2^{128} operations (less for legacy).
- More to come on long-term security . . .

Summary: current state of the art

- Currently used crypto (check the lock icon in your browser) starts with RSA, Diffie-Hellman (DH) in finite fields, or elliptic-curve Diffie-Hellman (ECDH).
- Older standards are RSA or elliptic curves from NIST (or Brainpool), e.g. NIST P256 or ECDSA.
- Internet currently moving over to Curve25519 (Bernstein) and Ed25519 (Bernstein, Duif, Lange, Schwabe, and Yang).
- For symmetric crypto TLS (the protocol behind https) uses AES or ChaCha20 and some MAC, e.g. AES-GCM or ChaCha20-Poly1305. High-end devices have support for AES-GCM, smaller ones do better with ChaCha20-Poly1305.
- Security is getting better. Some obstacles: bugs; untrustworthy hardware;

Summary: current state of the art

- Currently used crypto (check the lock icon in your browser) starts with RSA, Diffie-Hellman (DH) in finite fields, or elliptic-curve Diffie-Hellman (ECDH).
- Older standards are RSA or elliptic curves from NIST (or Brainpool), e.g. NIST P256 or ECDSA.
- Internet currently moving over to Curve25519 (Bernstein) and Ed25519 (Bernstein, Duif, Lange, Schwabe, and Yang).
- For symmetric crypto TLS (the protocol behind https) uses AES or ChaCha20 and some MAC, e.g. AES-GCM or ChaCha20-Poly1305. High-end devices have support for AES-GCM, smaller ones do better with ChaCha20-Poly1305.
- Security is getting better. Some obstacles: bugs; untrustworthy hardware; let alone anti-security measures such as laws restricting encryption in Australia, China, Iran, Russia, UK.

Algorithms for Quantum Computation: Discrete Logarithms and Factoring

Peter W. Shor
AT&T Bell Labs
Room 2D-149
600 Mountain Ave.
Murray Hill, NJ 07974, USA

Abstract

A computer is generally considered to be a universal computational device; i.e., it is believed able to simulate any physical computational device with a cost in computation time of at most a polynomial factor. It is not clear whether this is still true when quantum mechanics is taken into consideration. Several researchers, starting with David Deutsch, have developed models for quantum mechanical computers and have investigated their computational properties. This paper gives Las Vegas algorithms for finding discrete logarithms and factoring integers on a quantum computer that take a number of steps which is polynomial in the input size, e.g., the number of digits of the integer to be factored. These two problems are generally considered hard on a classical computer and have been used as the basis of several proposed cryptosystems. (We thus give the first examples of quantum cryptanalysis.)

[1, 2]. Although he did not ask whether quantum mechanics conferred extra power to computation, he did show that a Turing machine could be simulated by the reversible unitary evolution of a quantum process, which is a necessary prerequisite for quantum computation. Deutsch [9, 10] was the first to give an explicit model of quantum computation. He defined both quantum Turing machines and quantum circuits and investigated some of their properties.

The next part of this paper discusses how quantum computation relates to classical complexity classes. We will thus first give a brief intuitive discussion of complexity classes for those readers who do not have this background. There are generally two resources which limit the ability of computers to solve large problems: time and space (i.e., memory). The field of analysis of algorithms considers the asymptotic demands that algorithms make for these resources as a function of the problem size. Theoretical computer scientists generally classify algorithms as efficient when the number of steps of the algorithms grows as a polynomial in the size of the input. The class of prob-

Universal quantum computers are coming, and are scary

- Massive research effort. Tons of progress summarized in, e.g., https://en.wikipedia.org/wiki/Timeline_of_quantum_computing.

Universal quantum computers are coming, and are scary

- Massive research effort. Tons of progress summarized in, e.g., https://en.wikipedia.org/wiki/Timeline_of_quantum_computing.
- Mark Ketchen, IBM Research, 2012, on quantum computing: "We're actually doing things that are making us think like, 'hey this isn't 50 years off, this is maybe just 10 years off, or 15 years off.' It's within reach."
- Fast-forward to 2022, or 2027. Universal quantum computers exist.

Universal quantum computers are coming, and are scary

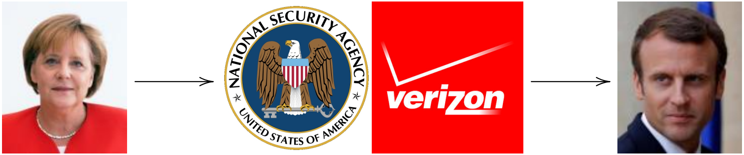
- Massive research effort. Tons of progress summarized in, e.g., https://en.wikipedia.org/wiki/Timeline_of_quantum_computing.
- Mark Ketchen, IBM Research, 2012, on quantum computing: "We're actually doing things that are making us think like, 'hey this isn't 50 years off, this is maybe just 10 years off, or 15 years off.' It's within reach."
- Fast-forward to 2022, or 2027. Universal quantum computers exist.
- Shor's algorithm solves in polynomial time:
  - Integer factorization. RSA is dead.
  - The discrete-logarithm problem in finite fields. DSA is dead.
  - The discrete-logarithm problem on elliptic curves. ECDSA is dead.
- This breaks all current public-key cryptography on the Internet!

Universal quantum computers are coming, and are scary

- Massive research effort. Tons of progress summarized in, e.g., https://en.wikipedia.org/wiki/Timeline_of_quantum_computing.
- Mark Ketchen, IBM Research, 2012, on quantum computing: "We're actually doing things that are making us think like, 'hey this isn't 50 years off, this is maybe just 10 years off, or 15 years off.' It's within reach."
- Fast-forward to 2022, or 2027. Universal quantum computers exist.
- Shor's algorithm solves in polynomial time:
  - Integer factorization. RSA is dead.
  - The discrete-logarithm problem in finite fields. DSA is dead.
  - The discrete-logarithm problem on elliptic curves. ECDSA is dead.
- This breaks all current public-key cryptography on the Internet!
- Also, Grover's algorithm speeds up brute-force searches.
- Example: Only 2^{64} quantum operations to break AES-128;
  2^{128} quantum operations to break AES-256.

Cryptography

- Motivation #1: Communication channels are spying on our data.
- Motivation #2: Communication channels are modifying our data.



Sender
"Alice"

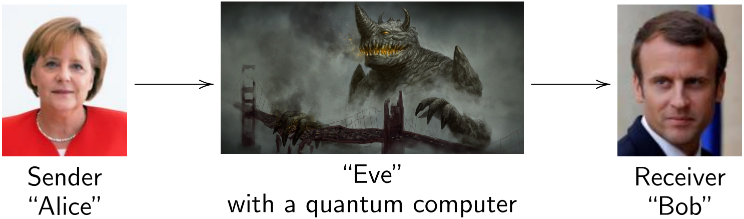
Untrustworthy network
"Eve"

Receiver
"Bob"

- Literal meaning of cryptography: "secret writing".
- Security goal #1: **Confidentiality** despite Eve's espionage.
- Security goal #2: **Integrity**, i.e., recognizing Eve's sabotage.

Post-quantum cryptography

- Motivation #1: Communication channels are spying on our data.
- Motivation #2: Communication channels are modifying our data.



Sender
"Alice"

"Eve"
with a quantum computer

Receiver
"Bob"

- Literal meaning of cryptography: "secret writing".
- Security goal #1: **Confidentiality** despite Eve's espionage.
- Security goal #2: **Integrity**, i.e., recognizing Eve's sabotage.
- Post-quantum cryptography adds to the model that Eve has a quantum computer.

Post-quantum cryptography:
Cryptography designed
under the assumption that
the **attacker** (not the user!)
has a large quantum computer.

History of post-quantum cryptography

- 2003 Daniel J. Bernstein introduces term Post-quantum cryptography.
- PQCrypto 2006: International Workshop on Post-Quantum Cryptography.

History of post-quantum cryptography

- 2003 Daniel J. Bernstein introduces term Post-quantum cryptography.
- PQCrypto 2006: International Workshop on Post-Quantum Cryptography.
- PQCrypto 2008, PQCrypto 2010, PQCrypto 2011, PQCrypto 2013.
- 2014 EU publishes H2020 call including post-quantum crypto as topic.
- ETSI working group on “Quantum-safe” crypto.
- PQCrypto 2014.
- April 2015 NIST hosts first workshop on post-quantum cryptography
- August 2015 NSA wakes up

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying "Don't use post-quantum crypto, the NSA wants you to use it!".

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying "Don't use post-quantum crypto, the NSA wants you to use it!". Or "NSA says NIST P-384 is post-quantum secure".

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying "Don't use post-quantum crypto, the NSA wants you to use it!". Or "NSA says NIST P-384 is post-quantum secure". Or "NSA has abandoned ECC."

NSA announcements

August 11, 2015

> *IAD recognizes that there will be a move, in the not distant future, to a quantum resistant algorithm suite.*

August 19, 2015

> *IAD will initiate a transition to quantum resistant algorithms in the not too distant future.*

NSA comes late to the party and botches its grand entrance.

Worse, now we get people saying "Don't use post-quantum crypto, the NSA wants you to use it!". Or "NSA says NIST P-384 is post-quantum secure". Or "NSA has abandoned ECC." Or "The NSA can break lattices and wants you to use them."

Post-quantum becoming mainstream

- PQCrypto 2016: 22–26 Feb in Fukuoka, Japan, $>$ 200 people

- 2016: Every agency posts something (NCSC UK, NCSC NL, NSA).
- 2016: After public input, NIST calls for submissions to "Post-Quantum Cryptography Standardization Project". Solicits submissions on signatures and encryption (deadline Nov 2017).

PQCrypto 2018
The Ninth International Conference on Post-Quantum Cryptography
Fort Lauderdale, Florida, April 9-11, 2018

National Academy of Sciences (US)

4 December 2018: Report on quantum computing

Don't panic. "Key Finding 1: Given the current state of quantum computing and recent rates of progress, it is highly unexpected that a quantum computer that can compromise RSA 2048 or comparable discrete logarithm-based public key cryptosystems will be built within the next decade."

National Academy of Sciences (US)

4 December 2018: Report on quantum computing

Don't panic. "Key Finding 1: Given the current state of quantum computing and recent rates of progress, it is highly unexpected that a quantum computer that can compromise RSA 2048 or comparable discrete logarithm-based public key cryptosystems will be built within the next decade."

Panic. "Key Finding 10: Even if a quantum computer that can decrypt current cryptographic ciphers is more than a decade off, the hazard of such a machine is high enough—and the time frame for transitioning to a new security protocol is sufficiently long and uncertain—that prioritization of the development, standardization, and deployment of post-quantum cryptography is critical for minimizing the chance of a potential security and privacy disaster."

Confidence-inspiring crypto takes time to build

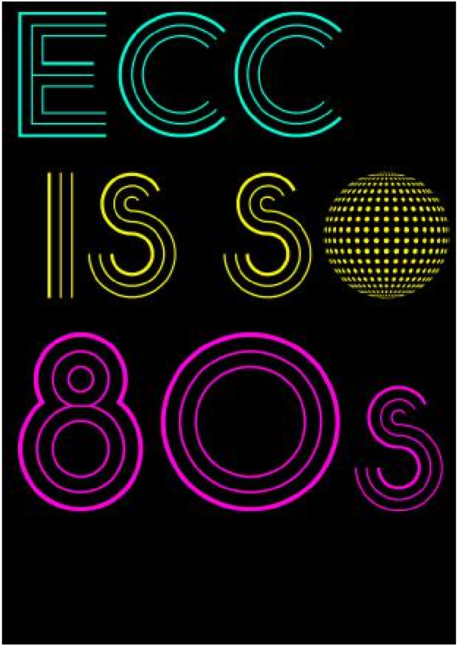
- Many stages of research from cryptographic design to deployment:
  - Explore space of cryptosystems.
  - Study algorithms for the attackers.
  - Focus on secure cryptosystems.

Confidence-inspiring crypto takes time to build

- Many stages of research from cryptographic design to deployment:
    - Explore space of cryptosystems.
    - Study algorithms for the attackers.
    - Focus on secure cryptosystems.
    - Study algorithms for the users.
    - Study implementations on real hardware.
    - Study side-channel attacks, fault attacks, etc.
    - Focus on secure, reliable implementations.
    - Focus on implementations meeting performance requirements.
    - Integrate securely into real-world applications.

Confidence-inspiring crypto takes time to build

- Many stages of research from cryptographic design to deployment:
  - Explore space of cryptosystems.
  - Study algorithms for the attackers.
  - Focus on secure cryptosystems.
  - Study algorithms for the users.
  - Study implementations on real hardware.
  - Study side-channel attacks, fault attacks, etc.
  - Focus on secure, reliable implementations.
  - Focus on implementations meeting performance requirements.
  - Integrate securely into real-world applications.
- Example: ECC introduced **1985**; big advantages over RSA.
  Robust ECC started to take over the Internet in **2015**.
- Can't wait for quantum computers before finding a solution!


ECC
IS SO
80s

Even higher urgency for long-term confidentiality

- Today's encrypted communication is being stored by attackers and will be decrypted years later with quantum computers. Danger for human-rights workers, medical records, journalists, security research, legal proceedings, state secrets, . . .

- Signature schemes can be replaced once a quantum computer is built – but there will not be a public announcement

Even higher urgency for long-term confidentiality

- Today's encrypted communication is being stored by attackers and will be decrypted years later with quantum computers. Danger for human-rights workers, medical records, journalists, security research, legal proceedings, state secrets, …

TOP SECRET//COMINT//REL TO USA, AUS, CAN, GBR, NZL
Where is X-KEYSCORE?
KEYSC

- Signature schemes can be replaced once a quantum computer is built – but there will not be a public announcement … and an important function of signatures is to protect operating system upgrades.
- Protect your upgrades *now* with post-quantum signatures.

Standardize now? Standardize later?

- Standardize now!
  - Rolling out crypto takes long time.
  - Standards are important for adoption (?)
  - Need to be up & running when quantum computers come.

Standardize now? Standardize later?

- Standardize now!
  - Rolling out crypto takes long time.
  - Standards are important for adoption (?)
  - Need to be up & running when quantum computers come.
- Standardize later!
  - Current options are not satisfactory.
  - Once rolled out, it's hard to change systems.
  - Please wait for the research results, will be much better!

Standardize now? Standardize later?

- Standardize now!
  - Rolling out crypto takes long time.
  - Standards are important for adoption (?)
  - Need to be up & running when quantum computers come.
- Standardize later!
  - Current options are not satisfactory.
  - Once rolled out, it's hard to change systems.
  - Please wait for the research results, will be much better!
- But what about users who rely on long-term secrecy of today's communication?
- Recommend now, standardize later. General roll out later.
- Recommend very conservative systems now; users who care will accept performance issues and gladly update to faster/smaller options later.
- But: Find out now where you rely on crypto; make an inventory.
- Important to raise awareness.

Urgency of post-quantum recommendations

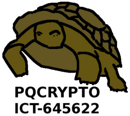
- If users want or need post-quantum systems **now**, what can they do?

Urgency of post-quantum recommendations

- If users want or need post-quantum systems **now**, what can they do?
- Post-quantum secure cryptosystems exist (to the best of our knowledge) but are under-researched – we can recommend secure systems now, but they are big and slow

Urgency of post-quantum recommendations

- If users want or need post-quantum systems **now**, what can they do?
- Post-quantum secure cryptosystems exist (to the best of our knowledge) but are under-researched – we can recommend secure systems now, but they are big and slow hence the logo of the PQCRYPTO project.



Urgency of post-quantum recommendations

- If users want or need post-quantum systems **now**, what can they do?
- Post-quantum secure cryptosystems exist (to the best of our knowledge) but are under-researched – we can recommend secure systems now, but they are big and slow hence the logo of the PQCRYPTO project.

PQCRYPTO
ICT-645622

- PQCRYPTO was an EU project in H2020, running 2015 – 2018.
- PQCRYPTO designed a portfolio of high-security post-quantum public-key systems, and improved the speed of these systems, adapting to the different performance challenges of mobile devices, the cloud, and the Internet.

Initial recommendations of long-term secure post-quantum systems

Daniel Augot, Lejla Batina, Daniel J. Bernstein, Joppe Bos,
Johannes Buchmann, Wouter Castryck, Orr Dunkelman,
Tim Güneysu, Shay Gueron, Andreas Hülsing,
Tanja Lange, Mohamed Saied Emam Mohamed,
Christian Rechberger, Peter Schwabe, Nicolas Sendrier,
Frederik Vercauteren, Bo-Yin Yang

Initial recommendations

- **Symmetric encryption** Thoroughly analyzed, 256-bit keys:
  - AES-256
  - Salsa20 with a 256-bit key

  Evaluating: Serpent-256, . . .
- **Symmetric authentication** Information-theoretic MACs:
  - GCM using a 96-bit nonce and a 128-bit authenticator
  - Poly1305
- **Public-key encryption** McEliece with binary Goppa codes:
  - length $n = 6960$, dimension $k = 5413$, $t = 119$ errors

  Evaluating: QC-MDPC, Stehlé-Steinfeld NTRU, . . .
- **Public-key signatures** Hash-based (minimal assumptions):
  - XMSS with any of the parameters specified in CFRG draft
  - SPHINCS-256

  Evaluating: HFEv-, . . .

Systems expected to survive

- Code-based encryption and signatures.
- Hash-based signatures.
- Isogeny-based encryption.
- Lattice-based encryption and signatures.
- Multivariate-quadratic encryption and signatures.
- Symmetric encryption and authentication.

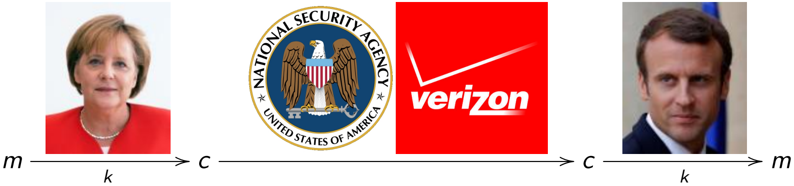
This list is based on the best known attacks (as always).

These are categories of mathematical problems;
individual systems may be insecure if the problem is not used correctly.

Short summaries

- Code-based encryption: short ciphertexts and large public keys. More in a moment.
- Hash-based signatures: very solid security and small public keys. Require only a secure hash function (hard to find second preimages). More in a moment.
- Isogeny-based encryption: new kid on the block, promising short keys and ciphertexts and non-interactive key exchange. Systems rely on hardness of finding isogenies between elliptic curves over finite fields.
- Lattice-based encryption and signatures: possibility for balanced sizes. Security relies on finding short vectors in some (typically special) lattice.
- Multivariate-quadratic signatures: short signatures and large public keys. Systems rely on hardness of solving systems of multi-variate equations over finite fields.

Post-quantum secret-key authenticated encryption



- Very easy solutions if secret key k is long uniform random string:
  - "One-time pad" for encryption.
  - "Wegman–Carter MAC" for authentication.
- AES-256: Standardized method to expand 256-bit k into string indistinguishable from long k.
- AES introduced in 1998 by Daemen and Rijmen. Security analyzed in papers by dozens of cryptanalysts.
- No credible threat from quantum algorithms. Grover costs 2^{128}.
- Some recent results assume attacker has quantum access to computation, then some systems are weaker . . . but I'd know if my laptop had turned into a quantum computer.

Post-quantum secret-key authenticated encryption

$$m \xrightarrow[k]{} c \xrightarrow{\qquad\qquad\qquad} c \xrightarrow[k]{} m$$

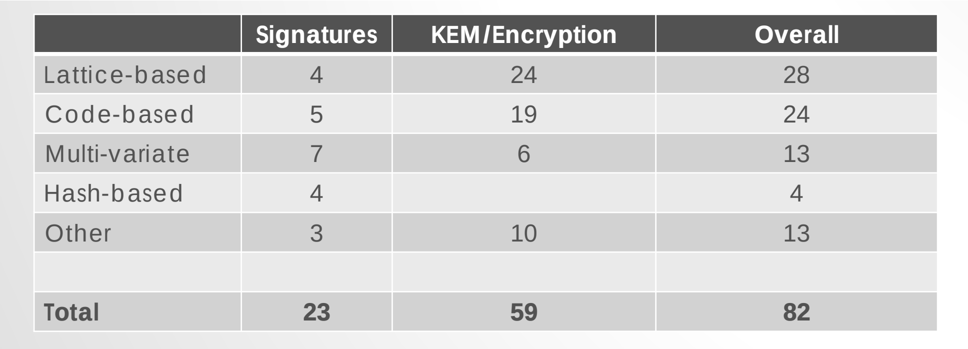
- Very easy solutions if secret key k is long uniform random string:
  - “One-time pad” for encryption.
  - “Wegman–Carter MAC” for authentication.
- AES-256: Standardized method to expand 256-bit k into string indistinguishable from long k.
- AES introduced in 1998 by Daemen and Rijmen. Security analyzed in papers by dozens of cryptanalysts.
- No credible threat from quantum algorithms. Grover costs 2^{128}.
- Some recent results assume attacker has quantum access to computation, then some systems are weaker . . . but I’d know if my laptop had turned into a quantum computer.

NIST Post-Quantum Competition

December 2016, after public feedback: NIST calls for submissions of post-quantum cryptosystems to standardize.

30 November 2017: NIST receives 82 submissions.

Overview from Dustin Moody's (NIST) talk at Asiacrypt 2017:



	Signatures	KEM/Encryption	Overall
Lattice-based	4	24	28
Code-based	5	19	24
Multi-variate	7	6	13
Hash-based	4		4
Other	3	10	13
Total	**23**	**59**	**82**

1.5 years ago in the NIST competition . . .

21 December 2017: NIST posts 69 submissions from 260 people.

BIG QUAKE. BIKE. CFPKM. Classic McEliece. Compact LWE. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. DAGS. Ding Key Exchange. DME. DRS. DualModeMS. Edon-K. EMBLEM and R.EMBLEM. FALCON. FrodoKEM. GeMSS. Giophantus. Gravity-SPHINCS. Guess Again. Gui. HILA5. HiMQ-3. HK17. HQC. KINDI. LAC. LAKE. LEDAkem. LEDApkc. Lepton. LIMA. Lizard. LOCKER. LOTUS. LUOV. McNie. Mersenne-756839. MQDSS. NewHope. NTRUEncrypt. pqNTRUSign. NTRU-HRSS-KEM. NTRU Prime. NTS-KEM. Odd Manhattan. OKCN/AKCN/CNKE. Ouroboros-R. Picnic. pqRSA encryption. pqRSA signature. pqsigRM. QC-MDPC KEM. qTESLA. RaCoSS. Rainbow. Ramstake. RankSign. RLCE-KEM. Round2. RQC. RVB. SABER. SIKE. SPHINCS+. SRTPI. Three Bears. Titanium. WalnutDSA.

1.5 years ago . . . there were already attacks

By end of 2017: 8 out of 69 submissions attacked.

BIG QUAKE. BIKE. CFPKM. Classic McEliece. Compact LWE. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. DAGS. Ding Key Exchange. DME. DRS. DualModeMS. Edon-K. EMBLEM and R.EMBLEM. FALCON. FrodoKEM. GeMSS. Giophantus. Gravity-SPHINCS. Guess Again. Gui. HILA5. HiMQ-3. HK17. HQC. KINDI. LAC. LAKE. LEDAkem. LEDApkc. Lepton. LIMA. Lizard. LOCKER. LOTUS. LUOV. McNie. Mersenne-756839. MQDSS. NewHope. NTRUEncrypt. pqNTRUSign. NTRU-HRSS-KEM. NTRU Prime. NTS-KEM. Odd Manhattan. OKCN/AKCN/CNKE. Ouroboros-R. Picnic. pqRSA encryption. pqRSA signature. pqsigRM. QC-MDPC KEM. qTESLA. RaCoSS. Rainbow. Ramstake. RankSign. RLCE-KEM. Round2. RQC. RVB. SABER. SIKE. SPHINCS+. SRTPI. Three Bears. Titanium. WalnutDSA.

Some less security than claimed; some really broken; some attack scripts.

Do cryptographers have any idea what they're doing?

By end of 2018: 22 out of 69 submissions attacked.

BIG QUAKE. BIKE. CFPKM. Classic McEliece. Compact LWE. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. DAGS. Ding Key Exchange. DME. DRS. DualModeMS. Edon-K. EMBLEM and R.EMBLEM. FALCON. FrodoKEM. GeMSS. Giophantus. Gravity-SPHINCS. Guess Again. Gui. HILA5. HiMQ-3. HK17. HQC. KINDI. LAC. LAKE. LEDAkem. LEDApkc. Lepton. LIMA. Lizard. LOCKER. LOTUS. LUOV. McNie. Mersenne-756839. MQDSS. NewHope. NTRUEncrypt. pqNTRUSign. NTRU-HRSS-KEM. NTRU Prime. NTS-KEM. Odd Manhattan. OKCN/AKCN/CNKE. Ouroboros-R. Picnic. pqRSA encryption. pqRSA signature. pqsigRM. QC-MDPC KEM. qTESLA. RaCoSS. Rainbow. Ramstake. RankSign. RLCE-KEM. Round2. RQC. RVB. SABER. SIKE. SPHINCS+. SRTPI. Three Bears. Titanium. WalnutDSA.

Some less security than claimed; some really broken; some attack scripts.

Some attempts to explain the situation

"What's safe is lattice-based cryptography." — Are you sure about that?

Some attempts to explain the situation

“What’s safe is lattice-based cryptography.” — Are you sure about that?

Lattice-based submissions: Compact LWE. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. Ding Key Exchange. DRS. EMBLEM and R.EMBLEM. FALCON. FrodoKEM. HILA5. KINDI. LAC. LIMA. Lizard. LOTUS. NewHope. NTRUEncrypt. NTRU-HRSS-KEM. NTRU Prime. Odd Manhattan. OKCN/AKCN/CNKE. pqNTRUSign. qTESLA. Round2. SABER. Titanium.

Some attempts to explain the situation

“What’s safe is lattice-based cryptography.” — Are you sure about that?

Lattice-based submissions: Compact LWE. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. Ding Key Exchange. DRS. EMBLEM and R.EMBLEM. FALCON. FrodoKEM. HILA5. KINDI. LAC. LIMA. Lizard. LOTUS. NewHope. NTRUEncrypt. NTRU-HRSS-KEM. NTRU Prime. Odd Manhattan. OKCN/AKCN/CNKE. pqNTRUSign. qTESLA. Round2. SABER. Titanium.

Many recent papers improving lattice attacks.
e.g. D’Anvers–Vercauteren–Verbauwhede papers in November+December: “On the impact of decryption failures on the security of LWE/LWR based schemes”; “The impact of error dependencies on Ring/Mod-LWE/LWR based schemes”.

Some attempts to explain the situation

“What’s safe is using the portfolio from the European PQCRYPTO project.” — Are you sure about that?

Some attempts to explain the situation

“What’s safe is using the portfolio from the European PQCRYPTO project.” — Are you sure about that?

The portfolio: BIG QUAKE. BIKE. Classic McEliece. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. DAGS. FrodoKEM. Gui. KINDI. LUOV. MQDSS. NewHope. NTRU-HRSS-KEM. NTRU Prime. Picnic. qTESLA. Rainbow. Ramstake. SABER. SPHINCS+.

Some attempts to explain the situation

“What’s safe is using the portfolio from the European PQCRYPTO project.” — Are you sure about that?

The portfolio: BIG QUAKE. BIKE. Classic McEliece. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. DAGS. FrodoKEM. Gui. KINDI. LUOV. MQDSS. NewHope. NTRU-HRSS-KEM. NTRU Prime. Picnic. qTESLA. Rainbow. Ramstake. SABER. SPHINCS+.

69 submissions = **denial-of-service attack against security evaluation**. Maybe cryptanalysts focused on submissions from outside the project.

Do cryptographers have any idea what they're doing?

By end of 2018: 22 out of 69 submissions attacked.

BIG QUAKE. BIKE. CFPKM. Classic McEliece. Compact LWE. CRYSTALS-DILITHIUM. CRYSTALS-KYBER. DAGS. Ding Key Exchange. DME. DRS. DualModeMS. Edon-K. EMBLEM and R.EMBLEM. FALCON. FrodoKEM. GeMSS. Giophantus. Gravity-SPHINCS. Guess Again. Gui. HILA5. HiMQ-3. HK17. HQC. KINDI. LAC. LAKE. LEDAkem. LEDApkc. Lepton. LIMA. Lizard. LOCKER. LOTUS. LUOV. McNie. Mersenne-756839. MQDSS. NewHope. NTRUEncrypt. pqNTRUSign. NTRU-HRSS-KEM. NTRU Prime. NTS-KEM. Odd Manhattan. OKCN/AKCN/CNKE. Ouroboros-R. Picnic. pqRSA encryption. pqRSA signature. pqsigRM. QC-MDPC KEM. qTESLA. RaCoSS. Rainbow. Ramstake. RankSign. RLCE-KEM. Round2. RQC. RVB. SABER. SIKE. SPHINCS+. SRTPI. Three Bears. Titanium. WalnutDSA.

Some less security than claimed; some really broken; some attack scripts.

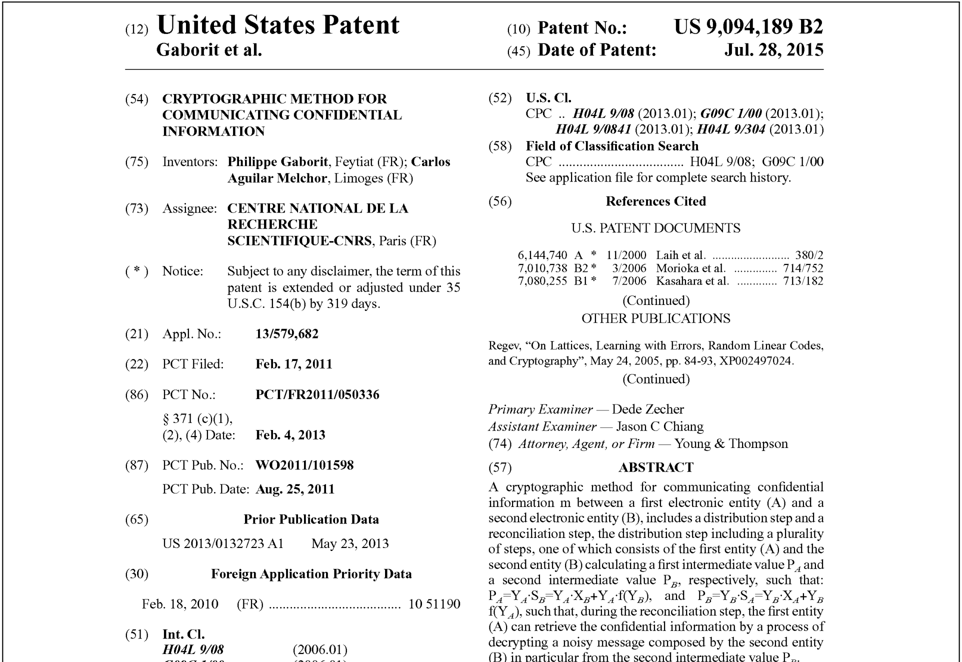
NIST round two

30 January 2019: 26 candidates retained for second round.

. BIKE. . Classic McEliece. .
CRYSTALS-DILITHIUM. CRYSTALS-KYBER. .
. . . . .
. FALCON. FrodoKEM. GeMSS. .
. . . HILA5. . . HQC.
. LAC. LAKE. LEDAkem. LEDApkc. . . .
LOCKER. . LUOV. . . MQDSS.
NewHope. NTRUEncrypt. . NTRU-HRSS-KEM. NTRU
Prime. NTS-KEM. . .
Ouroboros-R. Picnic. . . .
. qTESLA. . Rainbow. . .
. Round2. RQC. . SABER. SIKE. SPHINCS+. .
Three Bears. . .

Some less security than claimed; some really broken; some attack scripts.
Merges: HILA5 & Round2; LAKE, LOCKER, & Ouroboros-R;
LEDAkem & LEDApkc; NTRUEncrypt & NTRU-HRSS-KEM.

Warning: Some submissions are covered by patents



(12) **United States Patent**
Gaborit et al.

(10) **Patent No.:** **US 9,094,189 B2**
(45) **Date of Patent:** **Jul. 28, 2015**

(54) **CRYPTOGRAPHIC METHOD FOR COMMUNICATING CONFIDENTIAL INFORMATION**

(75) Inventors: **Philippe Gaborit**, Feytiat (FR); **Carlos Aguilar Melchor**, Limoges (FR)

(73) Assignee: **CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE-CNRS**, Paris (FR)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 319 days.

(21) Appl. No.: **13/579,682**

(22) PCT Filed: **Feb. 17, 2011**

(86) PCT No.: **PCT/FR2011/050336**

§ 371 (c)(1),
(2), (4) Date: **Feb. 4, 2013**

(87) PCT Pub. No.: **WO2011/101598**

PCT Pub. Date: **Aug. 25, 2011**

(65) **Prior Publication Data**

US 2013/0132723 A1 May 23, 2013

(30) **Foreign Application Priority Data**

Feb. 18, 2010 (FR) 10 51190

(51) **Int. Cl.**

H04L 9/08	(2006.01)
G09C 1/00	(2006.01)

(52) **U.S. Cl.**
CPC .. ***H04L 9/08*** (2013.01); ***G09C 1/00*** (2013.01); ***H04L 9/0841*** (2013.01); ***H04L 9/304*** (2013.01)

(58) **Field of Classification Search**
CPC H04L 9/08; G09C 1/00
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,144,740	A *	11/2000	Laih et al.	....	380/2
7,010,738	B2 *	3/2006	Morioka et al.	....	714/752
7,080,255	B1 *	7/2006	Kasahara et al.	....	713/182

(Continued)

OTHER PUBLICATIONS

Regev, "On Lattices, Learning with Errors, Random Linear Codes, and Cryptography", May 24, 2005, pp. 84-93, XP002497024.

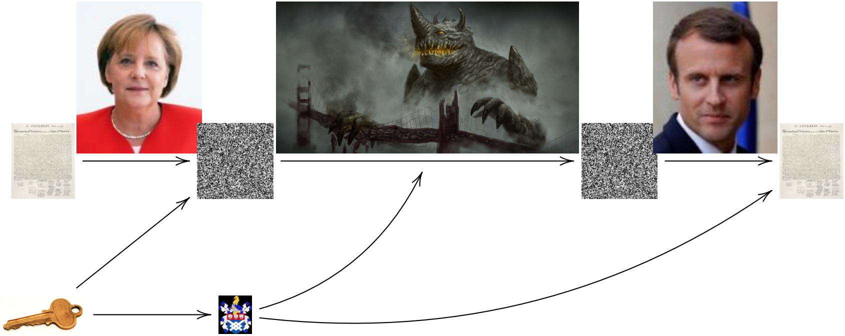
(Continued)

Primary Examiner — Dede Zecher
Assistant Examiner — Jason C Chiang
(74) *Attorney, Agent, or Firm* — Young & Thompson

(57) **ABSTRACT**

A cryptographic method for communicating confidential information m between a first electronic entity (A) and a second electronic entity (B), includes a distribution step and a reconciliation step, the distribution step including a plurality of steps, one of which consists of the first entity (A) and the second entity (B) calculating a first intermediate value P_A and a second intermediate value P_B, respectively, such that: $P_A=Y_A \cdot S_B=Y_A \cdot X_B+Y_A \cdot f(Y_B)$, and $P_B=Y_B \cdot S_A=Y_B \cdot X_A+Y_B$ $f(Y_A)$, such that, during the reconciliation step, the first entity (A) can retrieve the confidential information by a process of decrypting a noisy message composed by the second entity (B) in particular from the second intermediate value P_B.

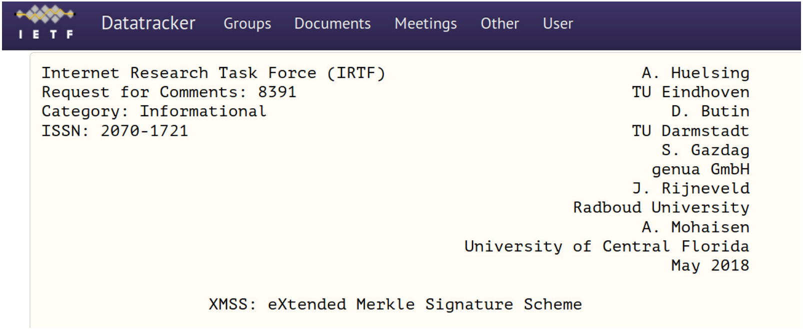
Post-quantum public-key signatures: hash-based



- Secret key , public key .
- Only one prerequisite: a good hash function, e.g. SHA3-512, . . . Hash functions map long strings to fixed-length strings.

  Signature schemes use hash functions in handling .
- Old idea: 1979 Lamport one-time signatures.
- 1979 Merkle extends to more signatures.

Pros and cons



```
Internet Research Task Force (IRTF)                          A. Huelsing
Request for Comments: 8391                                  TU Eindhoven
Category: Informational                                         D. Butin
ISSN: 2070-1721                                            TU Darmstadt
                                                               S. Gazdag
                                                              genua GmbH
                                                            J. Rijneveld
                                                      Radboud University
                                                             A. Mohaisen
                                          University of Central Florida
                                                                May 2018


                 XMSS: eXtended Merkle Signature Scheme
```

Pros:

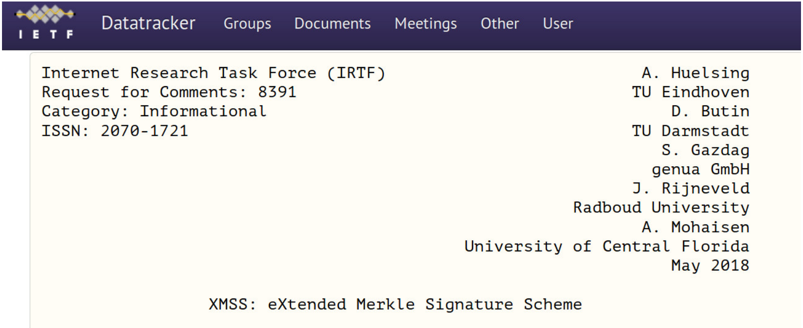
- Security well understood
- Only need secure hash function
- Small public key
- Fast

Cons:

- Biggish signature
- Stateful

  Adam Langley "for most environments it's a huge foot-cannon."

Pros and cons



```
Internet Research Task Force (IRTF)                       A. Huelsing
Request for Comments: 8391                               TU Eindhoven
Category: Informational                                      D. Butin
ISSN: 2070-1721                                          TU Darmstadt
                                                            S. Gazdag
                                                           genua GmbH
                                                         J. Rijneveld
                                                   Radboud University
                                                          A. Mohaisen
                                      University of Central Florida
                                                             May 2018

               XMSS: eXtended Merkle Signature Scheme
```

Pros:

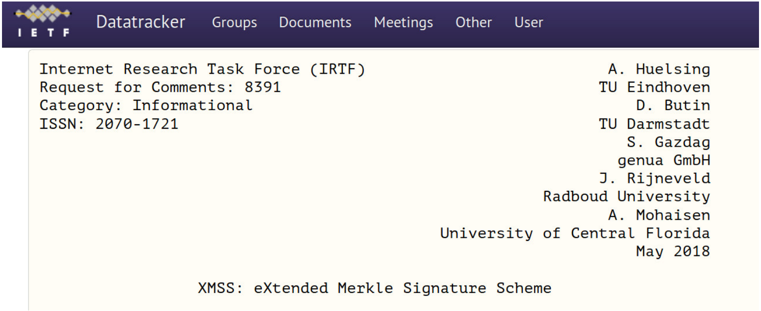
- Security well understood
- Only need secure hash function
- Small public key
- Fast
- We can count: OS update, code signing, . . . do keep state.

Cons:

- Biggish signature
- Stateful
  Adam Langley "for most environments it's a huge foot-cannon."

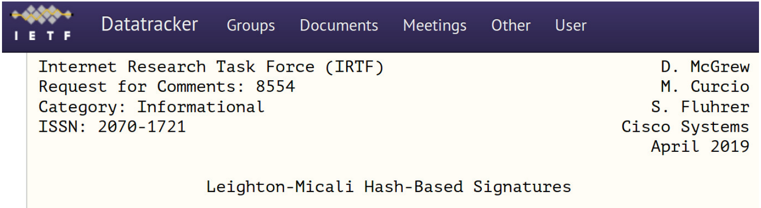
Standardization progress

- CFRG has published 2 RFCs: RFC 8391 and RFC 8554



```
Internet Research Task Force (IRTF)                          A. Huelsing
Request for Comments: 8391                                 TU Eindhoven
Category: Informational                                        D. Butin
ISSN: 2070-1721                                            TU Darmstadt
                                                              S. Gazdag
                                                             genua GmbH
                                                           J. Rijneveld
                                                     Radboud University
                                                            A. Mohaisen
                                         University of Central Florida
                                                               May 2018

                XMSS: eXtended Merkle Signature Scheme
```



```
Internet Research Task Force (IRTF)                            D. McGrew
Request for Comments: 8554                                     M. Curcio
Category: Informational                                       S. Fluhrer
ISSN: 2070-1721                                            Cisco Systems
                                                              April 2019

                  Leighton-Micali Hash-Based Signatures
```

Standardization progress

- CFRG has published 2 RFCs: RFC 8391 and RFC 8554
- NIST has gone through two rounds of requests for public input, most are positive and recommend standardizing XMSS and LMS. Only concern is about statefulness in general.

NIST

Information Technology Laboratory

COMPUTER SECURITY RESOURCE CENTER

PROJECTS

Stateful Hash-Based Signatures

Standardization progress

- CFRG has published 2 RFCs: RFC 8391 and RFC 8554
- NIST has gone through two rounds of requests for public input, most are positive and recommend standardizing XMSS and LMS. Only concern is about statefulness in general.

NIST

Information Technology Laboratory

COMPUTER SECURITY RESOURCE CENTER

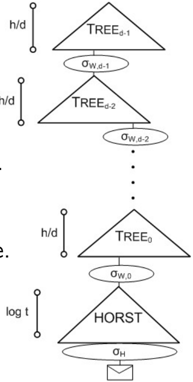
PROJECTS

Stateful Hash-Based Signatures

- ISO SC27 JTC1 WG2 has started a study period on stateful hash-based signatures.

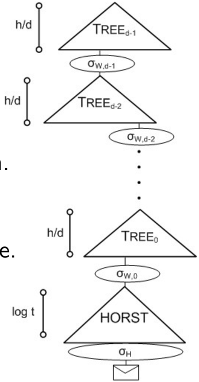
Stateless hash-based signatures

- Idea from 1987 Goldreich:
  - Signer builds huge tree of certificate authorities.
  - Signature includes certificate chain.
  - Each CA is a hash of master secret and tree position. This is deterministic, so don't need to store results.
  - **Random** bottom-level CA signs message. Many bottom-level CAs, so one-time signature is safe.

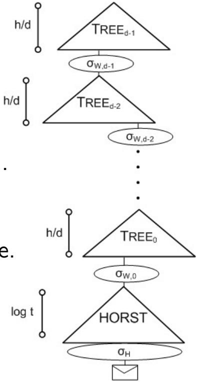


Stateless hash-based signatures

- Idea from 1987 Goldreich:
  - Signer builds huge tree of certificate authorities.
  - Signature includes certificate chain.
  - Each CA is a hash of master secret and tree position. This is deterministic, so don't need to store results.
  - **Random** bottom-level CA signs message. Many bottom-level CAs, so one-time signature is safe.
- 0.6 MB: Goldreich's signature with good 1-time signature scheme.
- 1.2 MB: average Debian package size.
- 1.8 MB: average web page in Alexa Top 1000000.



Stateless hash-based signatures



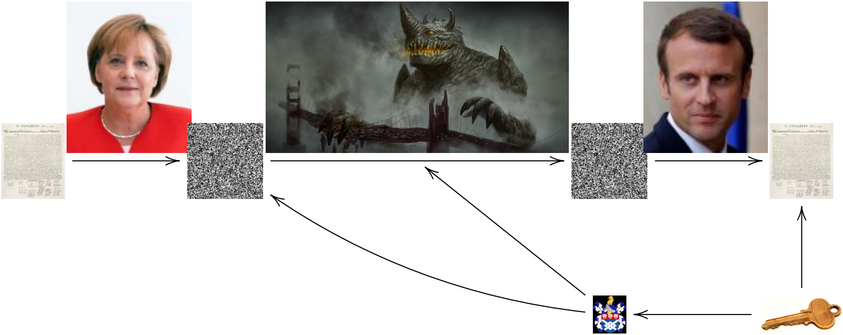
- Idea from 1987 Goldreich:
  - Signer builds huge tree of certificate authorities.
  - Signature includes certificate chain.
  - Each CA is a hash of master secret and tree position. This is deterministic, so don't need to store results.
  - **Random** bottom-level CA signs message. Many bottom-level CAs, so one-time signature is safe.
- 0.6 MB: Goldreich's signature with good 1-time signature scheme.
- 1.2 MB: average Debian package size.
- 1.8 MB: average web page in Alexa Top 1000000.
- 0.041 MB: SPHINCS signature, new optimization of Goldreich. Modular, guaranteed as strong as its components (hash, PRNG). Well-known components chosen for 2^{128} post-quantum security. `sphincs.cr.yp.to`

NIST submission SPHINCS+

- Same as SPHINCS in terms of high level scheme design, but better few-time signatures.
- New protection against multi-target attacks.
- New few-time signature scheme FORS instead of HORST (different way of combining Merkle trees).
- Smaller signatures – 30kB instead of 41kB – or more signatures.
- Smaller public keys.
- Three versions (different hash functions)
  - SPHINCS+-SHA3 (using SHAKE256),
  - SPHINCS+-SHA2 (using SHA-256),
  - SPHINCS+-Haraka (using the Haraka short-input hash function).

See `https://sphincs.org/` for more details.

Post-quantum public-key encryption: code-based



- Alice uses Bob's public key to encrypt.
- Bob uses his secret key to decrypt.
- Code-based crypto proposed by McEliece in 1978 using Goppa codes.
- Almost as old as RSA, but much stronger security history.
- Many further improvements, e.g. Niederreiter system for smaller keys.

One-wayness (OW-CPA)

Fundamental security question:
Given random parity-check matrix H and syndrome s,
can attacker efficiently find low-weight e with $s = He$?

One-wayness (OW-CPA)

Fundamental security question:
Given random parity-check matrix H and syndrome s,
can attacker efficiently find low-weight e with $s = He$?

1962 Prange: simple attack idea
guiding sizes in 1978 McEliece.

One-wayness (OW-CPA)

Fundamental security question:
Given random parity-check matrix H and syndrome s,
can attacker efficiently find low-weight e with $s = He$?

1962 Prange: simple attack idea
guiding sizes in 1978 McEliece.

The McEliece system (with later key-size optimizations)
uses $(c_0 + o(1))\lambda^2(\lg \lambda)^2$-bit keys as $\lambda \to \infty$
to achieve 2^λ security against Prange's attack.

Here $c_0 \approx 0.7418860694$.

40 years and more than 30 analysis papers later

1962 Prange; 1981 Clark–Cain, crediting Omura; 1988 Lee–Brickell; 1988 Leon; 1989 Krouk; 1989 Stern; 1989 Dumer; 1990 Coffey–Goodman; 1990 van Tilburg; 1991 Dumer; 1991 Coffey–Goodman–Farrell; 1993 Chabanne–Courteau; 1993 Chabaud; 1994 van Tilburg; 1994 Canteaut–Chabanne; 1998 Canteaut–Chabaud; 1998 Canteaut–Sendrier; 2008 Bernstein–Lange–Peters; 2009 Bernstein–Lange–Peters–van Tilborg; 2009 Bernstein (**post-quantum**); 2009 Finiasz–Sendrier; 2010 Bernstein–Lange–Peters; 2011 May–Meurer–Thomae; 2012 Becker–Joux–May–Meurer; 2013 Hamdaoui–Sendrier; 2015 May–Ozerov; 2016 Canto Torres–Sendrier; 2017 Kachigar–Tillich (**post-quantum**); 2017 Both–May; 2018 Both–May; 2018 Kirshanova (**post-quantum**).

40 years and more than 30 analysis papers later

1962 Prange; 1981 Clark–Cain, crediting Omura; 1988 Lee–Brickell; 1988 Leon; 1989 Krouk; 1989 Stern; 1989 Dumer; 1990 Coffey–Goodman; 1990 van Tilburg; 1991 Dumer; 1991 Coffey–Goodman–Farrell; 1993 Chabanne–Courteau; 1993 Chabaud; 1994 van Tilburg; 1994 Canteaut–Chabanne; 1998 Canteaut–Chabaud; 1998 Canteaut–Sendrier; 2008 Bernstein–Lange–Peters; 2009 Bernstein–Lange–Peters–van Tilborg; 2009 Bernstein (**post-quantum**); 2009 Finiasz–Sendrier; 2010 Bernstein–Lange–Peters; 2011 May–Meurer–Thomae; 2012 Becker–Joux–May–Meurer; 2013 Hamdaoui–Sendrier; 2015 May–Ozerov; 2016 Canto Torres–Sendrier; 2017 Kachigar–Tillich (**post-quantum**); 2017 Both–May; 2018 Both–May; 2018 Kirshanova (**post-quantum**).

The McEliece system uses $(c_0 + o(1))\lambda^2(\lg \lambda)^2$-bit keys as $\lambda \to \infty$ to achieve 2^λ security against all attacks known today.
Same $c_0 \approx 0.7418860694$.

40 years and more than 30 analysis papers later

1962 Prange; 1981 Clark–Cain, crediting Omura; 1988 Lee–Brickell; 1988 Leon; 1989 Krouk; 1989 Stern; 1989 Dumer; 1990 Coffey–Goodman; 1990 van Tilburg; 1991 Dumer; 1991 Coffey–Goodman–Farrell; 1993 Chabanne–Courteau; 1993 Chabaud; 1994 van Tilburg; 1994 Canteaut–Chabanne; 1998 Canteaut–Chabaud; 1998 Canteaut–Sendrier; 2008 Bernstein–Lange–Peters; 2009 Bernstein–Lange–Peters–van Tilborg; 2009 Bernstein (**post-quantum**); 2009 Finiasz–Sendrier; 2010 Bernstein–Lange–Peters; 2011 May–Meurer–Thomae; 2012 Becker–Joux–May–Meurer; 2013 Hamdaoui–Sendrier; 2015 May–Ozerov; 2016 Canto Torres–Sendrier; 2017 Kachigar–Tillich (**post-quantum**); 2017 Both–May; 2018 Both–May; 2018 Kirshanova (**post-quantum**).

The McEliece system uses $(c_0 + o(1))\lambda^2(\lg\lambda)^2$-bit keys as $\lambda \to \infty$ to achieve 2^λ security against all attacks known today.
Same $c_0 \approx 0.7418860694$.

Replacing λ with 2λ stops all known *quantum* attacks.

NIST submission Classic McEliece

- Security asymptotics unchanged by 40 years of cryptanalysis.
- Short ciphertexts.
- Efficient and straightforward conversion of OW-CPA PKE into IND-CCA2 KEM.
- Constant-time software implementations.
- FPGA implementation of full cryptosystem.
- Open-source (public domain) implementations.
- No patents.

Metric	**mceliece6960119**	**mceliece8192128**
Public-key size	1047319 bytes	1357824 bytes
Secret-key size	13908 bytes	14080 bytes
Ciphertext size	226 bytes	240 bytes
Key-generation time	839556968 cycles	1198956300 cycles
Encapsulation time	174276 cycles	185368 cycles
Decapsulation time	321580 cycles	342640 cycles

See `https://classic.mceliece.org` for more details.

NIST submission NTRU Prime

- Lattice-based encryption – smaller public keys.
- Less structure for the attacker to use:
  - Computation is done modulo prime instead of modulo power of 2.
  - Rings change from using polynomial $x^n - 1$ or $x^n + 1$ to $x^p - x - 1$, p prime.
  - No (nontrivial) subrings or fields.
- No decryption failures.

Metric	**sntrup4596761**	**ntrulpr4591761**
Public-key size	1218 bytes	1047 bytes
Secret-key size	1600 bytes	1238 bytes
Ciphertext size	1047 bytes	1175 bytes
Key-generation time	940852 cycles	44948 cycles
Encapsulation time	44788 cycles	81144 cycles
Decapsulation time	93676 cycles	113708 cycles

See `https://ntruprime.cr.yp.to/` for more details.

Links and upcoming events

- NIST PQC competition https://csrc.nist.gov/Projects/Post-Quantum-Cryptography
- 1 & 2 July 2019: Executive summer school on PQC in Eindhoven https://pqcschool.org/index.html.
- PQCRYPTO EU project https://pqcrypto.eu.org:
  - Expert recommendations.
  - Free software libraries (libpqcrypto, pqm4, pqhw).
  - Lots of reports, scientific papers, (overview) presentations.
- PQCRYPTO summer school 2017 with 21 lectures on video + slides + exercises. https://2017.pqcrypto.org/school:
- Executive school 2017 (12 lectures), less math, more overview. https://2017.pqcrypto.org/exec
- PQCrypto 2019 conference.
- PQCrypto 2018 conference.
- PQCrypto 2017 conference.
- PQCrypto 2016 with slides and videos from lectures + school.
- https://pqcrypto.org: Our survey site.
  - Many pointers: e.g., PQCrypto conference series.
  - Bibliography for 4 major PQC systems.