Digital signatures: the postquantum challenge

2022.6.23

Mehdi Tibouchi
NTT Social Informatics Laboratories

Digital signatures

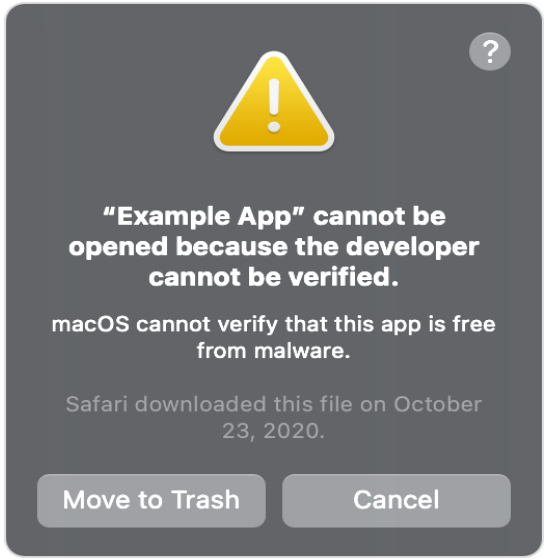
Security definition

- **Accepted security definition [GMR84]**

  existential unforgeability
  under chosen message attacks

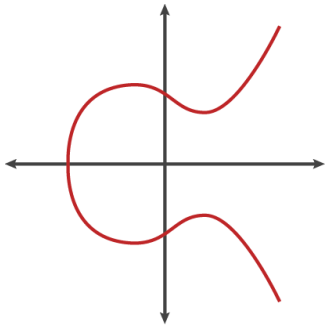
- **hard to construct valid signatures without the signing key**
- **on any message of the adversary's choosing**
- **even after seeing signatures on arbitrarily many other chosen messages**

Digital signatures are everywhere



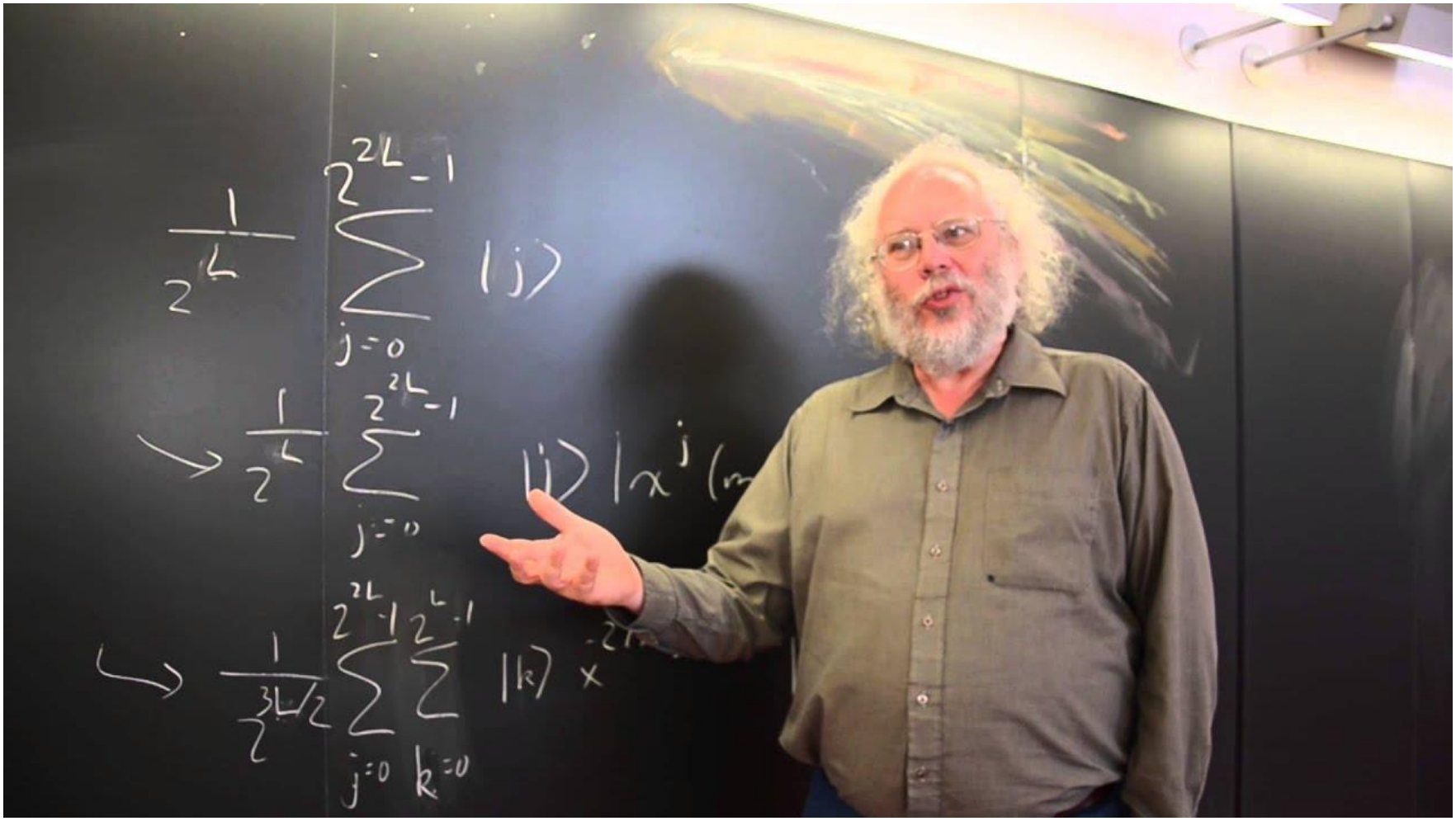
Currently deployed schemes

- Two families of signature schemes popular nowadays
- RSA® vs.



- RSA signatures (RSA-FDH, OAEP, PKCS…): security related to integer factorization
- Elliptic curve-based signatures (ECDSA, EC-Schnorr, EdDSA…): security related to the discrete logarithm problem

The [Shor94] problem



Circa 2015

Innovative R&D by NTT

ars technica

MAIN MENU · MY STORIES: 25 · FORUMS · SUBSCRIBE · JOBS

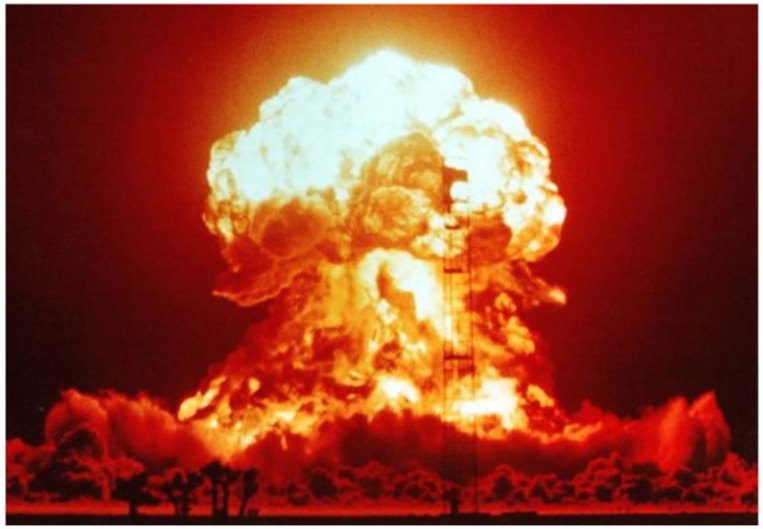
RISK ASSESSMENT / SECURITY & HACKTIVISM

NSA advisory sparks concern of secret advance ushering in cryptoapocalypse

Once elliptic curve crypto was viewed as a savior. Now its future looks doomed.

by Dan Goodin - Oct 23, 2015 7:15am JST

Share · Tweet · Email · 112



SECURELIST

THREATS · CATEGORIES · TAGS · ENCYCLOPEDIA

The cryptopocalypse is nigh

Finally, we cannot overemphasize the importance of cryptographic standards in maintaining the functional value of the internet as an information-sharing and transactional tool of unparalleled promise. These cryptographic standards rely on the expectation that the computational power required to break their encrypted output is simply above and beyond our combined means as a species. But what happens when we take a paradigmatic leap in computational capabilities as promised by future breakthroughs in quantum computing? Though quantum capabilities will not be initially available to the common cybercriminal, it signals a breakdown in the reliability of current crypto-standards and a need to design and implement 'post-quantum cryptography'. Given the poor rate of adoption or proper implementation of high-quality cryptography as it is, we do not foresee a smooth transition to counterbalance cryptographic failures at scale.

EurekAlert! The Global Source for Science News | AAAS

HOME · NEWS · MULTIMEDIA · MEETINGS · PORTALS · ABOUT

PUBLIC RELEASE: 24-SEP-2015

UMD awarded $1 million from NIST for next-generation cryptography

UNIVERSITY OF MARYLAND

NTT

Postquantum signatures

- **Pre-2015:**
  - a number of hardness assumptions not known to be broken by quantum computers
  - signature schemes based on some of them, often fairly theoretical
  - limited security analysis
  - very few concrete implementations or even parameters
  - almost no discussion of implementation security (side-channel protection, etc.)

- **2016: NIST launches standardization process**
  - let's solve those problems!

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

Why the push to move quickly

- Cryptographically relevant quantum computers might be decades away in the future
- But signatures may need to stay secure for a long time (e.g. V2V, OTA updates of IoT…)
- Powerful adversaries might covertly get them earlier than the public
- Updating standards is slow and cumbersome

NIST PQ process timeline

- **Timeline**
  - 2016/12: initial call for proposals
  - 2017/11: deadline for submissions
  - 2017/12: round 1 candidate list (69, incl. 20 signatures)
  - 2019/1: round 2 candidate list (26, incl. 9 signatures)
  - 2020/6: round 3 candidate list (7+8, incl. 3+3 signatures)
  - 2022 (any time now): end of round 3, possible round 4, extra call for proposals for signatures
  - 2024: final standard published?

What the process clarified so far

- **Identified a number of schemes we are fairly confident are secure**
  - and also eliminated insecure or dubious approaches
- **Candidates come with concrete parameters and mostly deployment-ready implementations**
  - new methodologies to evaluate security for various assumptions
  - usually constant-time code, decently optimized
  - also efforts on embedded devices (Cortex-M4)
- **Spurred lots of research activity**
  - new schemes, new attacks, new proof techniques
  - new results that came after the deadline

Postquantum assumptions (I)

- **Code-based cryptography**
  - linear codes with “hidden” structure, hard to decode without knowing the structure
  - no NIST signature candidate (too early)

- **Isogeny-based cryptography**
  - hardness of finding isogenies between elliptic curves
  - no NIST signature candidate (too early)

- **Multivariate cryptography**
  - related to the hardness of solving multivariate polynomial systems
  - works well for signatures, but need to consider optimization carefully (LUOV, Rainbow both broken)

Postquantum assumptions (II)

Innovative R&D by NTT

- **Hash-based cryptography**
  - hash functions and other “symmetric-key” primitives are typically postquantum secure
  - turns out that signatures can be constructed from them!
  - very safe approach, but not so efficient
- **Lattice-based cryptography**
  - hardness of finding short vectors in Euclidean lattices
  - “structured” lattices give rise to fairly compact and efficient schemes (2 out of 3 NIST finalists)
  - long-term security debated
- **Misc.**
  - e.g. Picnic: signatures from MPC-in-the-head

NTT

How to build PQ signatures

1. Solve the problem theoretically
2. Make the solution at least somewhat practical
3. Iron out parameters / security analysis
4. Implement, address implementation security
5. Improve and optimize
6. Goto 3

Example I: Hash-based

1. Lamport one-time signature (from any OWF)
2. Extend to multiple-time signatures and compress verification key using Merkle trees
3. Eliminate the state (SPHINCS)
4. Iron out parameters and optimize (SPHICS+)
5. More precise security analysis (e.g. QROM security, Q2 security…)

Example II: FSwA signatures

1. **Starting point: Schnorr signatures**
   1. Signature on m is (g^r, cx – r mod q), c=H(r,m)
   2. Second element is random for uniformly random r

2. **[Lyu09], [Lyu12]: similar approach for lattices using aborts**

3. **GLP, BLISS: instantiate the framework efficiently**

4. **Dilithium: address implementation issues**

5. **Further work: QROM security, masking countermeasures, etc.**

Example III: lattice h&s signatures

1. Insecure attempts: GGH, NTRUSign
2. GPV'08: first provable scheme (large and cumbersome)
3. Peikert'10, MP'12: better trapdoors, simpler sampling
4. DLP'14: first efficient implementation (still slow quadratic signing)
5. FALCON: efficient (quasilinear), compact, better security analysis
6. Further work: side-channel analysis, improved variants (Mitaka, compression)

Round 3 candidates vs. ECC/RSA

Scheme	vk size	sig. size	sig. speed	verif. speed
Ed25519	32	64	45k	160k
RSA-2048	256	256	3.3M	48k
FALCON-512	897	659	350k	71k
Dilithium-2	1312	2420	259k	118k
Rainbow-I	157,800	66	50k	24k
GeMSS-I	417,408	48	1510M	161k
SPHINCS+	64	17,088	57M	3.3M

Conclusion

- **Designing good cryptographic schemes is hard (esp. signatures)**
- **But transition to PQ schemes urgent**
- **Lots of progress under NIST process**
- **Many problems remain**
  - are the underlying problem really secure?
  - nothing as good as ECC; can we do better?
  - implementation issues abound
- **NIST round 4 + extra call for proposals ahead: plenty more work to do**