Quantum computers
the future attack that breaks today’s messages

Tanja Lange
Technische Universiteit Eindhoven

24 January 2019
Cybersecurity in de zorg
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 tools

Many factors influence the security and privacy of data:

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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Legacy</th>
<th>Near Term</th>
<th>Long Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetric Key Size</td>
<td>$k$</td>
<td>80</td>
<td>128</td>
</tr>
<tr>
<td>Hash Function Output Size</td>
<td>$m$</td>
<td>160</td>
<td>256</td>
</tr>
<tr>
<td>MAC Output Size*</td>
<td>$m$</td>
<td>80</td>
<td>128</td>
</tr>
<tr>
<td>RSA Problem</td>
<td>$\ell(n) \geq$</td>
<td>1024</td>
<td>3072</td>
</tr>
<tr>
<td>Finite Field DLP</td>
<td>$\ell(p^n) \geq$</td>
<td>1024</td>
<td>3072</td>
</tr>
<tr>
<td></td>
<td>$\ell(p), \ell(q) \geq$</td>
<td>160</td>
<td>256</td>
</tr>
<tr>
<td>ECDLP</td>
<td>$\ell(q) \geq$</td>
<td>160</td>
<td>256</td>
</tr>
<tr>
<td>Pairing</td>
<td>$\ell(p^k \cdot n) \geq$</td>
<td>1024</td>
<td>6144</td>
</tr>
<tr>
<td></td>
<td>$\ell(p), \ell(q) \geq$</td>
<td>160</td>
<td>256</td>
</tr>
</tbody>
</table>

- 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;
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 backdoors or hacking back.
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 power [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...
Effects of large universal quantum computers


▶ 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.
  - The discrete-logarithm problem in finite fields.
  - The discrete-logarithm problem on elliptic curves.

▶ This breaks all current public-key cryptography on the Internet!
Effects of large universal quantum computers


- 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. ECDH 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.

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

For more information, visit: [pqcrypto.eu.org](https://pqcrypto.eu.org)
Post-quantum cryptography

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

Literary 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.
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 totally insecure if the problem is not used correctly.

We have a good understanding of what a quantum computer can do, but new systems need more analysis.
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) 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.”
High 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

https://pqcrypto.eu.org
High 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 ... and an important function of signatures is to protect operating system upgrades.
- Protect your upgrades now with post-quantum signatures.

https://pqcrypto.eu.org
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 PQCRIPTO project.

https://pqcrypto.eu.org
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 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-, . . .
Standardization efforts

- NIST (National Institute for Standards and Technology) asked for submissions to post-quantum project. Ongoing efforts to analyze, implement, select; final results expected in 4-6 years.
- ETSI QSC: several whitepapers.
- ISO: working on whitepaper.
- OASIS: KMIP (key management) standard with PQC.
- ANSI and IEEE have standardized NTRU (not for PQC parameters).

https://pqcrypto.eu.org
Tanja Lange
Quantum computers – the future attack that breaks today’s messages
### Standardization efforts

- NIST (National Institute for Standards and Technology) asked for submissions to post-quantum project. Ongoing efforts to analyze, implement, select; final results expected in 4-6 years.
- ETSI QSC: several whitepapers.
- ISO: working on whitepaper.
- OASIS: KMIP (key management) standard with PQC.
- ANSI and IEEE have standardized NTRU (not for PQC parameters).

*Internet Research Task Force (IRTF)*  
Request for Comments: 8391  
Category: Informational  
ISSN: 2070-1721

---

XMSS: eXTended Merkle Signature Scheme

[https://pqcrypto.eu.org](https://pqcrypto.eu.org)  
Tanja Lange  
Quantum computers – the future attack that breaks today’s messages
Deployment issues & solutions

- Different recommendations for rollout:
  - Use most efficient systems with ECC or RSA, to ease usage and gain familiarity.
  - Use most conservative systems (possibly with ECC), to ensure that data really remains secure.

These recommendations match different risk scenarios.

- Protocol integration and implementation problems:
  - Key sizes or message sizes are larger for post-quantum systems, but IPv6 guarantees only delivery of \( \leq 1280 \)-byte packets.
  - Google experimented with larger keys and noticed delays and dropped connections.
  - Long-term keys require extra care (reaction attacks).

- Some libraries exist, but mostly for experiments, not production quality.
- Google and Cloudflare very recently announced some experiments of including post-quantum systems into TLS.
Links and upcoming events

▶ 1 & 2 July 2019: Executive summer school on post-quantum cryptography in Eindhoven
▶ https://pqcrypto.eu.org: PQCRIPTO EU project.  
  ▶ Expert recommendations.  
  ▶ Free software libraries (libpqcrypto, pqm4, pqhw).  
  ▶ Lots of reports, scientific papers, (overview) presentations.
▶ https://2017.pqcrypto.org/exec: Executive school (12 lectures), less math, more overview. So far slides, soon videos.
▶ PQCrypto 2018 & PQCrypto 2017 conferences.
▶ PQCrypto 2016 with slides and videos from lectures + school.
▶ https://pqcrypto.org: Survey site by D.J. Bernstein and me.
  ▶ Many pointers: e.g., PQCrypto conference series.
  ▶ Bibliography for 4 major PQC systems.
Bonus Slides
NIST submission SPHINCS+

- Signature based on hash functions.
- Requires only a secure hash function, no further assumptions.
- Based on ideas of Lamport (1979) and Merkle (1979).
- Developed starting from SPHINCS with
  - improve multi-signature,
  - smaller keys,
  - Option for shorter signatures (30kB instead of 41kB) if “only” $2^{50}$ messages signed.
- Three versions (using different hash functions)
  - SPHINCS+-SHA3 (with SHAKE256),
  - SPHINCS+-SHA2 (with SHA-256),
  - SPHINCS+-Haraka (with Haraka, a hash function for short inputs).

More info at https://sphincs.org/.
NIST submission Classic McEliece

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

<table>
<thead>
<tr>
<th>Metric</th>
<th>mceliece6960119 (bytes)</th>
<th>mceliece8192128 (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public-key size</td>
<td>1047319</td>
<td>1357824</td>
</tr>
<tr>
<td>Secret-key size</td>
<td>13908</td>
<td>14080</td>
</tr>
<tr>
<td>Ciphertext size</td>
<td>226</td>
<td>240</td>
</tr>
<tr>
<td>Key-generation time</td>
<td>1108833108 cycles</td>
<td>1173074192 cycles</td>
</tr>
<tr>
<td>Encapsulation time</td>
<td>153940 cycles</td>
<td>188520 cycles</td>
</tr>
<tr>
<td>Decapsulation time</td>
<td>318088 cycles</td>
<td>343756 cycles</td>
</tr>
</tbody>
</table>

See [https://classic.mceliece.org](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.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Streamlined NTRU Prime</th>
<th>NTRU LPrime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public-key size</td>
<td>1218 bytes</td>
<td>1047 bytes</td>
</tr>
<tr>
<td>Secret-key size</td>
<td>1600 bytes</td>
<td>1238 bytes</td>
</tr>
<tr>
<td>Ciphertext size</td>
<td>1047 bytes</td>
<td>1175 bytes</td>
</tr>
<tr>
<td>Key-generation time</td>
<td>940852 cycles</td>
<td>44948 cycles</td>
</tr>
<tr>
<td>Encapsulation time</td>
<td>44788 cycles</td>
<td>81144 cycles</td>
</tr>
<tr>
<td>Decapsulation time</td>
<td>93676 cycles</td>
<td>113708 cycles</td>
</tr>
</tbody>
</table>

See [https://ntruprime.cr.yp.to/](https://ntruprime.cr.yp.to/) for more details.