Post-Quantum Cryptography: Detours, delays, and disasters

Daniel J. Bernstein & Tanja Lange

University of Illinois at Chicago; Ruhr University Bochum; Academia Sinica
&
Eindhoven University of Technology; Academia Sinica

20 August 2022

Cryptographic tools used in TLS (`https`)

TLS relies critically on public-key cryptography for two reasons:

- Making sure the attacker can't pretend to be the server.
  This uses signatures: e.g., RSA-4096.
- Sending data as scrambled "ciphertexts" that the attacker can't understand.
  This uses encryption: e.g., NIST P-256.

Cryptographic tools used in TLS (`https`)

TLS relies critically on public-key cryptography for two reasons:

- Making sure the attacker can't pretend to be the server.
  This uses signatures: e.g., RSA-4096.
- Sending data as scrambled "ciphertexts" that the attacker can't understand.
  This uses encryption: e.g., NIST P-256.

For speed, TLS combines public-key cryptography with symmetric cryptography:

- Use public-key encryption to exchange a key,
  and public-key signatures so the attacker can't substitute a different key.
- Use symmetric encryption with that key to protect confidentiality of user data.
- Use symmetric authentication with that key to protect integrity of user data.

Similar comments for SSH and other popular cryptographic protocols.

Cryptographic tools used in TLS (`https`)

TLS relies critically on public-key cryptography for two reasons:

- Making sure the attacker can't pretend to be the server.
  This uses signatures: e.g., RSA-4096, **which will be broken by quantum computers.**
- Sending data as scrambled "ciphertexts" that the attacker can't understand.
  This uses encryption: e.g., NIST P-256, **which will be broken by quantum computers.**

For speed, TLS combines public-key cryptography with symmetric cryptography:

- Use public-key encryption to exchange a key,
  and public-key signatures so the attacker can't substitute a different key.
- Use symmetric encryption with that key to protect confidentiality of user data.
- Use symmetric authentication with that key to protect integrity of user data.

Similar comments for SSH and other popular cryptographic protocols.

Cryptographic tools used in TLS (`https`)

TLS relies critically on public-key cryptography for two reasons:

- Making sure the attacker can't pretend to be the server.
  This uses signatures: e.g., RSA-4096, **which will be broken by quantum computers.**
- Sending data as scrambled "ciphertexts" that the attacker can't understand.
  This uses encryption: e.g., NIST P-256, **which will be broken by quantum computers.**

For speed, TLS combines public-key cryptography with symmetric cryptography:

- Use public-key encryption to exchange a key,
  and public-key signatures so the attacker can't substitute a different key.
- Use symmetric encryption with that key to protect confidentiality of user data.
- Use symmetric authentication with that key to protect integrity of user data.

Similar comments for SSH and other popular cryptographic protocols.

Post-quantum cryptography: cryptography under the assumption that the attacker has a quantum computer.

Urgency of post-quantum recommendations

Screenshot from 8 May 2016

- All currently used public-key systems on the Internet are broken by quantum computers.
- Today's encrypted communication can be (and is being!) stored by attackers and can be decrypted later with quantum computer – think of medical records, legal proceedings, and state secrets.
- 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 is an EU project in H2020, running 2015 – 2018.
- PQCRYPTO is designing a portfolio of high-security post-quantum public-key systems, and will improve the speed of these systems, adapting to the different performance challenges of mobile devices, the cloud, and the Internet.

Standardize now? Standardize later?

Screenshot from 8 May 2016

- 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.
- Recommend very conservative systems now; users who care will accept performance issues and gladly update to faster/smaller options later.
- But: standardization takes lots of time, so start standardization processes now.

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

Issued in 2015 by the PQCRYPTO project.

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-, ...

So everyone lived happily ever after

So everyone lived happily ever after?

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.
A patent holder then contacts Google and asks for money—oops!

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.
A patent holder then contacts Google and asks for money—oops!

2016.11 Chrome removes `newhope1024` option.

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.
A patent holder then contacts Google and asks for money—oops!

2016.11 Chrome removes `newhope1024` option.

2016.12 US National Institute of Standards and Technology (NIST)
calls for submissions by 2017.11 of post-quantum systems for standardization.

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.
A patent holder then contacts Google and asks for money—oops!

2016.11 Chrome removes `newhope1024` option.

2016.12 US National Institute of Standards and Technology (NIST)
calls for submissions by 2017.11 of post-quantum systems for standardization.
NIST *prohibits* submissions of ECC+PQ hybrids:
"The algorithms shall not incorporate major components that are believed
to be insecure against quantum computers. (For example, hybrid schemes
that include encryption or signatures based on factoring or discrete logs
will not be considered for standardization by NIST in this context.)"

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.
A patent holder then contacts Google and asks for money—oops!

2016.11 Chrome removes `newhope1024` option.

2016.12 US National Institute of Standards and Technology (NIST)
calls for submissions by 2017.11 of post-quantum systems for standardization.
NIST *prohibits* submissions of ECC+PQ hybrids:
"The algorithms shall not incorporate major components that are believed
to be insecure against quantum computers. (For example, hybrid schemes
that include encryption or signatures based on factoring or discrete logs
will not be considered for standardization by NIST in this context.)"
NIST sets up evaluation criteria emphasizing *pre-quantum* security.

Critical post-quantum decisions in 2016

2016.07 Chrome adds `newhope1024` post-quantum option.
Used for an experiment with Google servers.
This encryption layer is *added* to X25519 encryption (ECC):
if `newhope1024` is completely broken, still have pre-quantum security.
`newhope1024` is a new cryptosystem: main components from 2010, 2014, 2015.
A patent holder then contacts Google and asks for money—oops!

2016.11 Chrome removes `newhope1024` option.

2016.12 US National Institute of Standards and Technology (NIST)
calls for submissions by 2017.11 of post-quantum systems for standardization.
NIST *prohibits* submissions of ECC+PQ hybrids:
"The algorithms shall not incorporate major components that are believed
to be insecure against quantum computers. (For example, hybrid schemes
that include encryption or signatures based on factoring or discrete logs
will not be considered for standardization by NIST in this context.)"
NIST sets up evaluation criteria emphasizing *pre-quantum* security.
Further distractions: "flexibility", security "categories", ...

Pressure to wait for NIST

Industry incentives to wait: NIST is promising to collect information about patents and select strong patent-free post-quantum standards.

- Strong: "The security provided by a cryptographic scheme is the most important factor in the evaluation."
- Patent-free: "NIST believes it is critical that this process leads to cryptographic standards that can be freely implemented in security technologies and products."

Pressure to wait for NIST

Industry incentives to wait: NIST is promising to collect information about patents and select strong patent-free post-quantum standards.

- Strong: "The security provided by a cryptographic scheme is the most important factor in the evaluation."
- Patent-free: "NIST believes it is critical that this process leads to cryptographic standards that can be freely implemented in security technologies and products."

Other standardization organizations decide that they'll wait for NIST.

IRTF CFRG, 2017.03: "the current CFRG approach is to define RFCs for a few relatively mature post-quantum primitives, such as hash-based signatures, but to wait for the results of the NIST process for everything else."

ISO internal discussions: ISO will wait for NIST.

Pressure to wait for NIST

Industry incentives to wait: NIST is promising to collect information about patents and select strong patent-free post-quantum standards.

- Strong: "The security provided by a cryptographic scheme is the most important factor in the evaluation."
- Patent-free: "NIST believes it is critical that this process leads to cryptographic standards that can be freely implemented in security technologies and products."

Other standardization organizations decide that they'll wait for NIST.

IRTF CFRG, 2017.03: "the current CFRG approach is to define RFCs for a few relatively mature post-quantum primitives, such as hash-based signatures, but to wait for the results of the NIST process for everything else."

ISO internal discussions: ISO will wait for NIST.

Exception: China runs its own competition!

Pressure to wait for NIST

Industry incentives to wait: NIST is promising to collect information about patents and select strong patent-free post-quantum standards.

- Strong: "The security provided by a cryptographic scheme is the most important factor in the evaluation."
- Patent-free: "NIST believes it is critical that this process leads to cryptographic standards that can be freely implemented in security technologies and products."

Other standardization organizations decide that they'll wait for NIST.

IRTF CFRG, 2017.03: "the current CFRG approach is to define RFCs for a few relatively mature post-quantum primitives, such as hash-based signatures, but to wait for the results of the NIST process for everything else."

ISO internal discussions: ISO will wait for NIST.

Exception: China runs its own competition! But nobody cares what China does.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

2017.12.23 Hülsing–Bernstein–Panny–Lange: scripts breaking RaCoSS.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

2017.12.23 Hülsing–Bernstein–Panny–Lange: scripts breaking RaCoSS.

2017.12.25 Panny: script breaking RVB; RVB withdrawn.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

2017.12.23 Hülsing–Bernstein–Panny–Lange: scripts breaking RaCoSS.

2017.12.25 Panny: script breaking RVB; RVB withdrawn.

2017.12.25 Bernstein–Lange: script breaking HK17.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

2017.12.23 Hülsing–Bernstein–Panny–Lange: scripts breaking RaCoSS.

2017.12.25 Panny: script breaking RVB; RVB withdrawn.

2017.12.25 Bernstein–Lange: script breaking HK17.

2017.12.26 Gaborit: attack reducing McNie security level.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

2017.12.23 Hülsing–Bernstein–Panny–Lange: scripts breaking RaCoSS.

2017.12.25 Panny: script breaking RVB; RVB withdrawn.

2017.12.25 Bernstein–Lange: script breaking HK17.

2017.12.26 Gaborit: attack reducing McNie security level.

2017.12.29 Gaborit: attack reducing Lepton security level.

Attack timeline: month 0 of NIST post-quantum competition

2017.11.30 Deadline for submissions to NIST; some submissions are announced online.

2017.12.18 Bernstein–Groot Bruinderink–Panny–Lange: script breaking CCA for HILA5.

2017.12.21 NIST posts 69 "complete and proper" submissions from 260 people; very few conservative submissions; heavy emphasis on performance.

2017.12.21 Panny: script breaking Guess Again.

2017.12.23 Hülsing–Bernstein–Panny–Lange: scripts breaking RaCoSS.

2017.12.25 Panny: script breaking RVB; RVB withdrawn.

2017.12.25 Bernstein–Lange: script breaking HK17.

2017.12.26 Gaborit: attack reducing McNie security level.

2017.12.29 Gaborit: attack reducing Lepton security level.

2017.12.29 Beullens: attack reducing DME security level.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

2018.01.05 Lequesne–Sendrier–Tillich: attack breaking Edon-K; script posted 2018.02.20; Edon-K withdrawn.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

2018.01.05 Lequesne–Sendrier–Tillich: attack breaking Edon-K; script posted 2018.02.20; Edon-K withdrawn.

2018.01.05 Beullens: script breaking DME.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

2018.01.05 Lequesne–Sendrier–Tillich: attack breaking Edon-K; script posted 2018.02.20; Edon-K withdrawn.

2018.01.05 Beullens: script breaking DME.

2018.01.05 Li–Liu–Pan–Xie, independently Bootle–Tibouchi–Xagawa: attack breaking Compact LWE; script from 2nd team.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

2018.01.05 Lequesne–Sendrier–Tillich: attack breaking Edon-K; script posted 2018.02.20; Edon-K withdrawn.

2018.01.05 Beullens: script breaking DME.

2018.01.05 Li–Liu–Pan–Xie, independently Bootle–Tibouchi–Xagawa: attack breaking Compact LWE; script from 2nd team.

2018.01.11 Castryck–Vercauteren: attack breaking Giophantus.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

2018.01.05 Lequesne–Sendrier–Tillich: attack breaking Edon-K; script posted 2018.02.20; Edon-K withdrawn.

2018.01.05 Beullens: script breaking DME.

2018.01.05 Li–Liu–Pan–Xie, independently Bootle–Tibouchi–Xagawa: attack breaking Compact LWE; script from 2nd team.

2018.01.11 Castryck–Vercauteren: attack breaking Giophantus.

2018.01.22 Blackburn: attack reducing WalnutDSA security level.

Attack timeline: month 1

2018.01.01 Bernstein, building on Bernstein–Lange, Wang–Malluhi, Li–Liu–Pan–Xie: faster script breaking HK17; HK17 withdrawn.

2018.01.02 Steinfeld, independently Albrecht–Postlethwaite–Virdia: script breaking CFPKM.

2018.01.02 Alperin-Sheriff–Perlner: attack breaking pqsigRM.

2018.01.04 Yang–Bernstein–Lange: script breaking SRTPI; SRTPI withdrawn.

2018.01.05 Lequesne–Sendrier–Tillich: attack breaking Edon-K; script posted 2018.02.20; Edon-K withdrawn.

2018.01.05 Beullens: script breaking DME.

2018.01.05 Li–Liu–Pan–Xie, independently Bootle–Tibouchi–Xagawa: attack breaking Compact LWE; script from 2nd team.

2018.01.11 Castryck–Vercauteren: attack breaking Giophantus.

2018.01.22 Blackburn: attack reducing WalnutDSA security level.

2018.01.23 Beullens: another attack reducing WalnutDSA security level.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

2018.05.09 Kotov–Menshov–Ushakov: another attack breaking WalnutDSA.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

2018.05.09 Kotov–Menshov–Ushakov: another attack breaking WalnutDSA.

2018.05.16 Barelli–Couvreur: attack reducing DAGS security level.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

2018.05.09 Kotov–Menshov–Ushakov: another attack breaking WalnutDSA.

2018.05.16 Barelli–Couvreur: attack reducing DAGS security level.

2018.05.30 Couvreur–Lequesne–Tillich: attack breaking "short" parameters for RLCE.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

2018.05.09 Kotov–Menshov–Ushakov: another attack breaking WalnutDSA.

2018.05.16 Barelli–Couvreur: attack reducing DAGS security level.

2018.05.30 Couvreur–Lequesne–Tillich: attack breaking "short" parameters for RLCE.

2018.06.11 Beullens–Castryck–Vercauteren: script breaking Giophantus.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

2018.05.09 Kotov–Menshov–Ushakov: another attack breaking WalnutDSA.

2018.05.16 Barelli–Couvreur: attack reducing DAGS security level.

2018.05.30 Couvreur–Lequesne–Tillich: attack breaking "short" parameters for RLCE.

2018.06.11 Beullens–Castryck–Vercauteren: script breaking Giophantus.

etc.

Attack timeline: round 1 continues

2018.02.01 Beullens: attack breaking WalnutDSA.

2018.02.07 Fabsic–Hromada–Zajac: attack breaking CCA for LEDA.

2018.03.27 Yu–Ducas: attack reducing DRS security level.

2018.04.03 Debris-Alazard–Tillich: attack breaking RankSign; RankSign withdrawn.

2018.04.04 Beullens–Blackburn: script breaking WalnutDSA.

2018.05.09 Kotov–Menshov–Ushakov: another attack breaking WalnutDSA.

2018.05.16 Barelli–Couvreur: attack reducing DAGS security level.

2018.05.30 Couvreur–Lequesne–Tillich: attack breaking "short" parameters for RLCE.

2018.06.11 Beullens–Castryck–Vercauteren: script breaking Giophantus.

etc.

2019.01.30 NIST announces selection of 26 second-round candidates;
keeps 0/13 broken submissions, 3/9 submissions with disproven security claims,
28/47 remaining submissions, biased towards faster submissions; −5 merges.

Some widely used software starts adding post-quantum options

. . . and attacks continue

Incentives for industry are changing: urgency of protecting users is more obvious; NIST has already collected and published patent statements.

2019.04 Lyubashevsky–Schwabe: script breaking some qTESLA parameters.

2019.04 OpenSSH 8.0 adds `sntrup761` option.
Used if client and server configure it.

Some widely used software starts adding post-quantum options

. . . and attacks continue

Incentives for industry are changing: urgency of protecting users is more obvious;
NIST has already collected and published patent statements.

2019.04 Lyubashevsky–Schwabe: script breaking some qTESLA parameters.

2019.04 OpenSSH 8.0 adds `sntrup761` option.
Used if client and server configure it.

2019.07 Google and Cloudflare run a new post-quantum experiment.
Option 1: CECPQ2, encrypting with `ntruhrss701` and ECC.
Option 2: CECPQ2b, encrypting with `sikep434` and ECC.

Some widely used software starts adding post-quantum options

. . . and attacks continue

Incentives for industry are changing: urgency of protecting users is more obvious;
NIST has already collected and published patent statements.

2019.04 Lyubashevsky–Schwabe: script breaking some qTESLA parameters.

2019.04 OpenSSH 8.0 adds `sntrup761` option.
Used if client and server configure it.

2019.07 Google and Cloudflare run a new post-quantum experiment.
Option 1: CECPQ2, encrypting with `ntruhrss701` and ECC.
Option 2: CECPQ2b, encrypting with `sikep434` and ECC.

2019.08 Zaverucha–Kales: attack reducing security level of "provably secure" MQDSS.

Some widely used software starts adding post-quantum options

. . . and attacks continue

Incentives for industry are changing: urgency of protecting users is more obvious;
NIST has already collected and published patent statements.

2019.04 Lyubashevsky–Schwabe: script breaking some qTESLA parameters.

2019.04 OpenSSH 8.0 adds `sntrup761` option.
Used if client and server configure it.

2019.07 Google and Cloudflare run a new post-quantum experiment.
Option 1: CECPQ2, encrypting with `ntruhrss701` and ECC.
Option 2: CECPQ2b, encrypting with `sikep434` and ECC.

2019.08 Zaverucha–Kales: attack reducing security level of "provably secure" MQDSS.

2019.10 Google claims "quantum supremacy".

Some widely used software starts adding post-quantum options

. . . and attacks continue

Incentives for industry are changing: urgency of protecting users is more obvious;
NIST has already collected and published patent statements.

2019.04 Lyubashevsky–Schwabe: script breaking some qTESLA parameters.

2019.04 OpenSSH 8.0 adds `sntrup761` option.
Used if client and server configure it.

2019.07 Google and Cloudflare run a new post-quantum experiment.
Option 1: CECPQ2, encrypting with `ntruhrss701` and ECC.
Option 2: CECPQ2b, encrypting with `sikep434` and ECC.

2019.08 Zaverucha–Kales: attack reducing security level of "provably secure" MQDSS.

2019.10 Google claims "quantum supremacy".

More attacks disprove security claims for further submissions.

Attacks keep getting better

2020.07 NIST announces selection of 15 third-round candidates;
keeps 0/2 broken submissions, 0/5 submissions with disproven security claims,
16/19 remaining submissions; −1 merge.

NIST demonstrates how confident it is in its selections: "If NIST's confidence in better performing signature algorithms is shaken by new analysis , SPHINCS+ could provide an immediately available algorithm for standardization at the end of the third round."

Attacks keep getting better

2020.07 NIST announces selection of 15 third-round candidates;
keeps 0/2 broken submissions, 0/5 submissions with disproven security claims,
16/19 remaining submissions; −1 merge.

NIST demonstrates how confident it is in its selections: "If NIST's confidence in better performing signature algorithms is shaken by new analysis , SPHINCS+ could provide an immediately available algorithm for standardization at the end of the third round."

2020.09 NIST: "The key point here . . . is not that 'attacks have improved, and may continue improving against Rainbow.' . . . the theoretical analysis has been tightened in a way that **matches** the empirical observations of the generic algorithm."

Attacks keep getting better

2020.07 NIST announces selection of 15 third-round candidates;
keeps 0/2 broken submissions, 0/5 submissions with disproven security claims,
16/19 remaining submissions; −1 merge.

NIST demonstrates how confident it is in its selections: "If NIST's confidence in better performing signature algorithms is shaken by new analysis , SPHINCS+ could provide an immediately available algorithm for standardization at the end of the third round."

2020.09 NIST: "The key point here . . . is not that 'attacks have improved, and may continue improving against Rainbow.' . . . the theoretical analysis has been tightened in a way that **matches** the empirical observations of the generic algorithm."

2020.10 Beullens: attack reducing security level of Rainbow.

Attacks keep getting better

2020.07 NIST announces selection of 15 third-round candidates;
keeps 0/2 broken submissions, 0/5 submissions with disproven security claims,
16/19 remaining submissions; −1 merge.

NIST demonstrates how confident it is in its selections: "If NIST's confidence in better performing signature algorithms is shaken by new analysis , SPHINCS+ could provide an immediately available algorithm for standardization at the end of the third round."

2020.09 NIST: "The key point here . . . is not that 'attacks have improved, and may continue improving against Rainbow.' . . . the theoretical analysis has been tightened in a way that **matches** the empirical observations of the generic algorithm."

2020.10 Beullens: attack reducing security level of Rainbow.

2020.11 Tao–Petzoldt–Ding: attack reducing security level of GeMSS.

Attacks keep getting better

2020.07 NIST announces selection of 15 third-round candidates;
keeps 0/2 broken submissions, 0/5 submissions with disproven security claims,
16/19 remaining submissions; −1 merge.

NIST demonstrates how confident it is in its selections: "If NIST's confidence in better performing signature algorithms is shaken by new analysis , SPHINCS+ could provide an immediately available algorithm for standardization at the end of the third round."

2020.09 NIST: "The key point here . . . is not that 'attacks have improved, and may continue improving against Rainbow.' . . . the theoretical analysis has been tightened in a way that **matches** the empirical observations of the generic algorithm."

2020.10 Beullens: attack reducing security level of Rainbow.

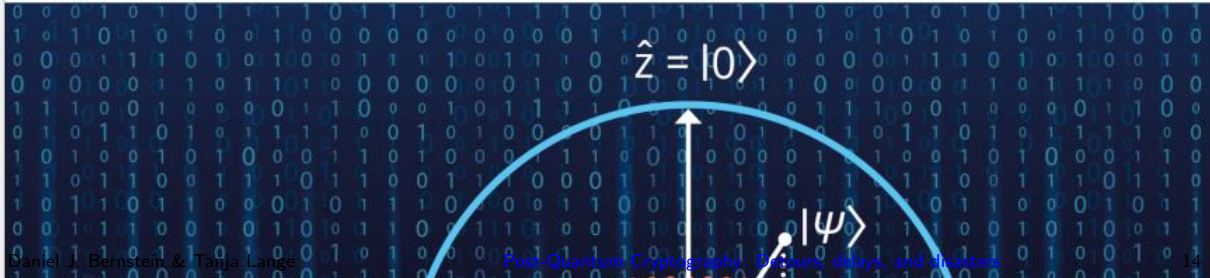
2020.11 Tao–Petzoldt–Ding: attack reducing security level of GeMSS.

2021.02 Beullens: script breaking smallest Rainbow parameter set.

Post-Quantum Cryptography: Current state and quantum mitigation

Ward Beullens, Jan-Pieter D'Anvers, Andreas Hülsing,
Tanja Lange, Lorenz Panny, Cyprien de Saint Guilhem, Nigel P. Smart.
Evangelos Rekleitis, Angeliki Aktypi, Athanasios-Vasileios Grammatopoulos.

EUROPEAN UNION AGE
FOR CYBERSECU



ENISA report: Current state and quantum mitigation

Chapters

1. Introduction
2. Families of Post-Quantum Algorithms
3. Security Notions and Generic Transforms
4. NIST Round 3 Finalists
5. Alternate Candidates
6. Quantum Mitigation
   - 6.1 Hybrid schemes
   - 6.2 Protective measures for pre-quantum cryptography

Report available from ENISA's website.

US government vs. deployment of post-quantum cryptography

2021.05 OpenBSD adds `sntrup761` option for IPsec. Used if client and server configure it.

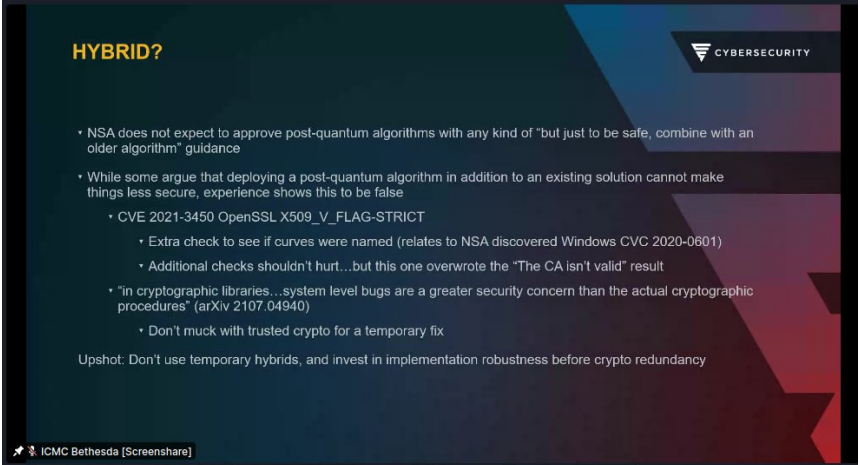
US government vs. deployment of post-quantum cryptography

2021.05 OpenBSD adds `sntrup761` option for IPsec. Used if client and server configure it.

2021.07 Matthew Scholl, Chief of the Computer Security Division in NIST's Information Technology Laboratory, on videotape: "Don't let folks start to buy and implement unstandard, unknown, potentially unsecured implementations before we as a general community have agreed upon standardization."

2021.08 NSA says: "The intention is to update CNSA to remove quantum-vulnerable algorithms and replace them with a subset of the quantum-resistant algorithms selected by NIST ... NSA is waiting for the NIST process to be completed and for standards to be published. ... NSS customers are reminded that NSA does not recommend and policy does not allow implementing or using unapproved, non-standard or experimental cryptographic algorithms. The field of quantum-resistant cryptography is no exception."

2021.09 DHS says: Do not use "post-quantum cryptographic industry products until standardization, implementation, and testing of replacement products with approved algorithms are completed by NIST."


HYBRID?
CYBERSECURITY
• NSA does not expect to approve post-quantum algorithms with any kind of "but just to be safe, combine with an older algorithm" guidance
• While some argue that deploying a post-quantum algorithm in addition to an existing solution cannot make things less secure, experience shows this to be false
• CVE 2021-3450 OpenSSL X509_V_FLAG-STRICT
• Extra check to see if curves were named (relates to NSA discovered Windows CVC 2020-0601)
• Additional checks shouldn't hurt…but this one overwrote the "The CA isn't valid" result
• "in cryptographic libraries…system level bugs are a greater security concern than the actual cryptographic procedures" (arXiv 2107.04940)
• Don't muck with trusted crypto for a temporary fix
Upshot: Don't use temporary hybrids, and invest in implementation robustness before crypto redundancy
ICMC Bethesda [Screenshare]

US ANSI X9 on post-quantum hybrids

2021.10 "As we transition from classical cryptography to post-quantum cryptography (PQC), there is a need to understand the proper ways to use both methods simultaneously. PQC methods will not be able to be used as a direct replacement in all cases. And the confidence and broad acceptance of PQC methods will not be as great as classical cryptography. **Simultaneous use of both classical cryptography and PQC methods for both security and acceptance** is required during a transition and may be required long term as well. There are improper and insecure ways of implementing a hybrid of classical and PQC methods. Specifying the proper methods of using both are required." (emphasis added)

French ANSSI on post-quantum hybrids

2021.12 "Although this new post-quantum toolbox may seem handy for developers, the maturity level of the post-quantum algorithms presented to the NIST process should not be overestimated. Many aspects lack cryptanalytical hindsight or are still research topics, e.g. analysis of the difficulty of the underlying problem in the classical and quantum computation models, dimensioning, integration of schemes in protocols and more importantly the design of secure implementations. This situation will probably last some time after the publication of NIST standards. **Acknowledging the immaturity of PQC is important: ANSSI will not endorse any direct drop-in replacement of currently used algorithms in the short/medium term. However, this immaturity should not serve as an argument for postponing the first deployments.**" (emphasis added)

NIST delays announcement for another half year

2021.12 NIST: "NIST will be selecting the first post-quantum standards for KEMs (and digital signatures) around the end of December or sometime in early January."

2022.02 NIST: "We hope to be able to announce the results and report not later than the end of March."

NIST delays announcement for another half year

. . . and some deployment continues in the meantime

2021.12 NIST: "NIST will be selecting the first post-quantum standards for KEMs (and digital signatures) around the end of December or sometime in early January."

2022.02 NIST: "We hope to be able to announce the results and report not later than the end of March."

2022.02 OpenSSH 8.9 enables `sntrup761` on server by default.
Used if client configures it.

NIST delays announcement for another half year

. . . and some deployment continues in the meantime

2021.12 NIST: "NIST will be selecting the first post-quantum standards for KEMs (and digital signatures) around the end of December or sometime in early January."

2022.02 NIST: "We hope to be able to announce the results and report not later than the end of March."

2022.02 OpenSSH 8.9 enables `sntrup761` on server by default.
Used if client configures it.

2022.03 NIST: "We ask for a little bit more patience since we are not ready to make the announcement today. We still expect to make it very soon."

NIST delays announcement for another half year

. . . and some deployment continues in the meantime

2021.12 NIST: "NIST will be selecting the first post-quantum standards for KEMs (and digital signatures) around the end of December or sometime in early January."

2022.02 NIST: "We hope to be able to announce the results and report not later than the end of March."

2022.02 OpenSSH 8.9 enables `sntrup761` on server by default.
Used if client configures it.

2022.03 NIST: "We ask for a little bit more patience since we are not ready to make the announcement today. We still expect to make it very soon."

2022.04 OpenSSH 9.0 enables `sntrup761` on client by default.

NIST's 5 July 2022 (aka 127 March 2022) announcement

The winners:

- Kyber, a public-key encryption system based on structured lattices
- Dilithium, a public-key signature scheme based on structured lattices
- Falcon, a public-key signature scheme based on structured lattices
- SPHINCS+, a public-key signature scheme based on

NIST's 5 July 2022 (aka 127 March 2022) announcement

The winners:

- Kyber, a public-key encryption system based on structured lattices
- Dilithium, a public-key signature scheme based on structured lattices
- Falcon, a public-key signature scheme based on structured lattices
- SPHINCS+, a public-key signature scheme based on hash functions

NIST's 5 July 2022 (aka 127 March 2022) announcement

The winners:

- Kyber, a public-key encryption system based on structured lattices
- Dilithium, a public-key signature scheme based on structured lattices
- Falcon, a public-key signature scheme based on structured lattices
- SPHINCS+, a public-key signature scheme based on hash functions

Schemes advancing to round 4, so maybe more winners later:

- BIKE, a public-key encryption system based on codes
- Classic McEliece, a public-key encryption system based on codes
- HQC, a public-key encryption system based on codes
- SIKE, a public-key encryption system based on isogenies

https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022

Patent problems addressed?

2022.07 NIST report:

- "Issues relating to patents were a factor in NIST's decision during the third round as NIST became aware of various third-party patents."
- "NIST negotiated with several third parties to enter into various agreements to overcome potential adoption challenges posed by third-party patents."

Patent problems addressed? Not yet

2022.07 NIST report:

- "Issues relating to patents were a factor in NIST's decision during the third round as NIST became aware of various third-party patents."
- "NIST negotiated with several third parties to enter into various agreements to overcome potential adoption challenges posed by third-party patents."
- "If the agreements are not executed by the end of 2022, NIST may consider selecting NTRU instead of KYBER."

Patent problems addressed? Not yet

2022.07 NIST report:

- "Issues relating to patents were a factor in NIST's decision during the third round as NIST became aware of various third-party patents."
- "NIST negotiated with several third parties to enter into various agreements to overcome potential adoption challenges posed by third-party patents."
- "If the agreements are not executed by the end of 2022, NIST may consider selecting NTRU instead of KYBER."

What this tells patent holders: "Feel free to ask NIST for more money."

Patent problems addressed? Not yet

2022.07 NIST report:

- "Issues relating to patents were a factor in NIST's decision during the third round as NIST became aware of various third-party patents."
- "NIST negotiated with several third parties to enter into various agreements to overcome potential adoption challenges posed by third-party patents."
- "If the agreements are not executed by the end of 2022, NIST may consider selecting NTRU instead of KYBER."

What this tells patent holders: "Feel free to ask NIST for more money."

What this tells industry: "Maybe Kyber isn't patent-free. Keep waiting."

2022.07.15 Fluhrer (Cisco): "Hence, until we get the text of the licenses [Cisco] cannot use Kyber. If continues to be true, we will need to seek an alternative solution."

Backups in case lattices are broken?

NIST repeatedly says it's important to have a backup plan:

- 2022.02 Nature article "The race to save the Internet from quantum hackers":
  "In the next few months, the institute will select two algorithms for each application.
  It will then begin to draft standards for one, while keeping the other as a reserve
  in case the first choice ends up being broken by an unexpected attack,
  quantum or otherwise." (emphasis added)
- 2022.06 NIST: "Because this is a new research field,
  we don't want to put all our eggs in one basket and only have lattice algorithms,
  and then an attack comes along and we don't have anything else." (emphasis added)

Backups in case lattices are broken? No

NIST repeatedly says it's important to have a backup plan:

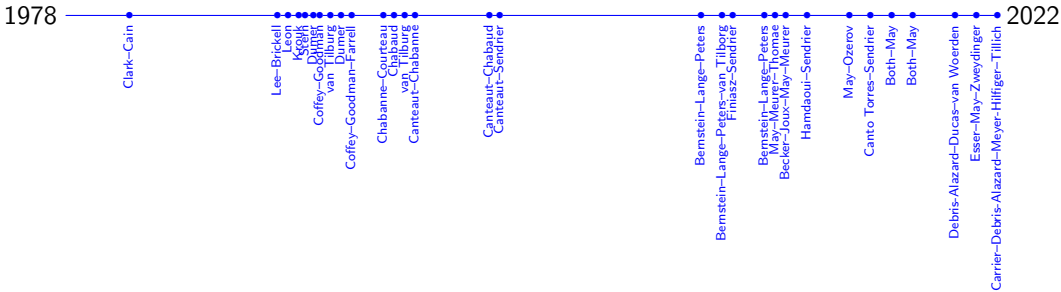
- 2022.02 Nature article "The race to save the Internet from quantum hackers":
  "In the next few months, the institute will select two algorithms for each application.
  It will then begin to draft standards for one, while keeping the other as a reserve
  in case the first choice ends up being broken by an unexpected attack,
  quantum or otherwise." (emphasis added)
- 2022.06 NIST: "Because this is a new research field,
  we don't want to put all our eggs in one basket and only have lattice algorithms,
  and then an attack comes along and we don't have anything else." (emphasis added)

2022.07 NIST:

- Selects 3 signature systems and just 1 encryption system.
  The only encryption option is a new lattice-based cryptosystem,
  with no fallback in case something goes wrong.
- Maybe more selections after round 4, but that won't finish until 2024.

Stable security?

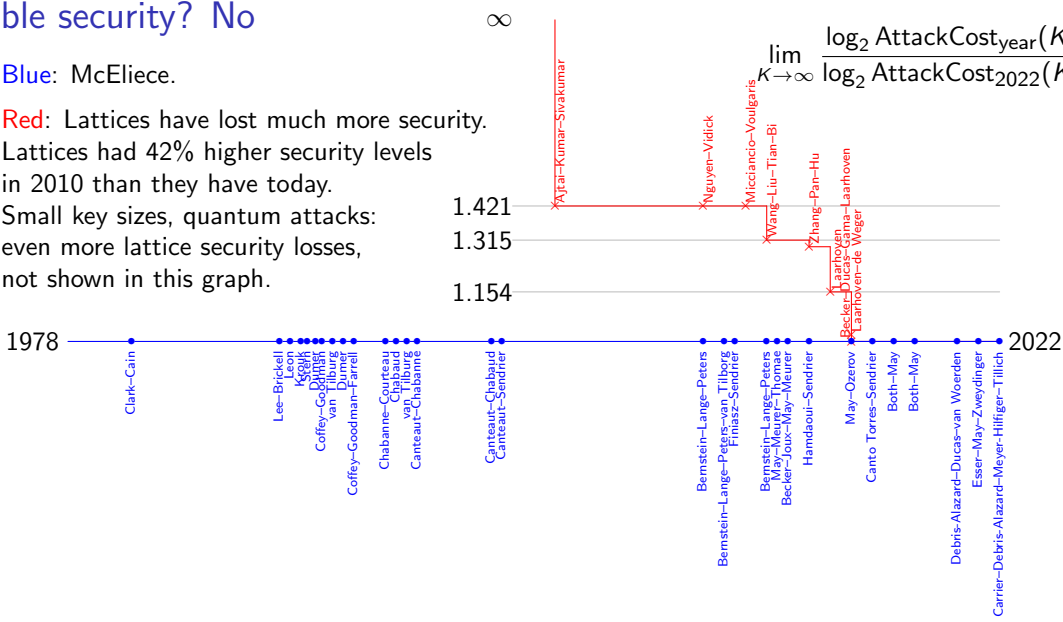
$$\lim_{K\to\infty} \frac{\log_2 \mathsf{AttackCost}_{\mathsf{year}}(K)}{\log_2 \mathsf{AttackCost}_{2022}(K)}$$



Stable security? No

Blue: McEliece.

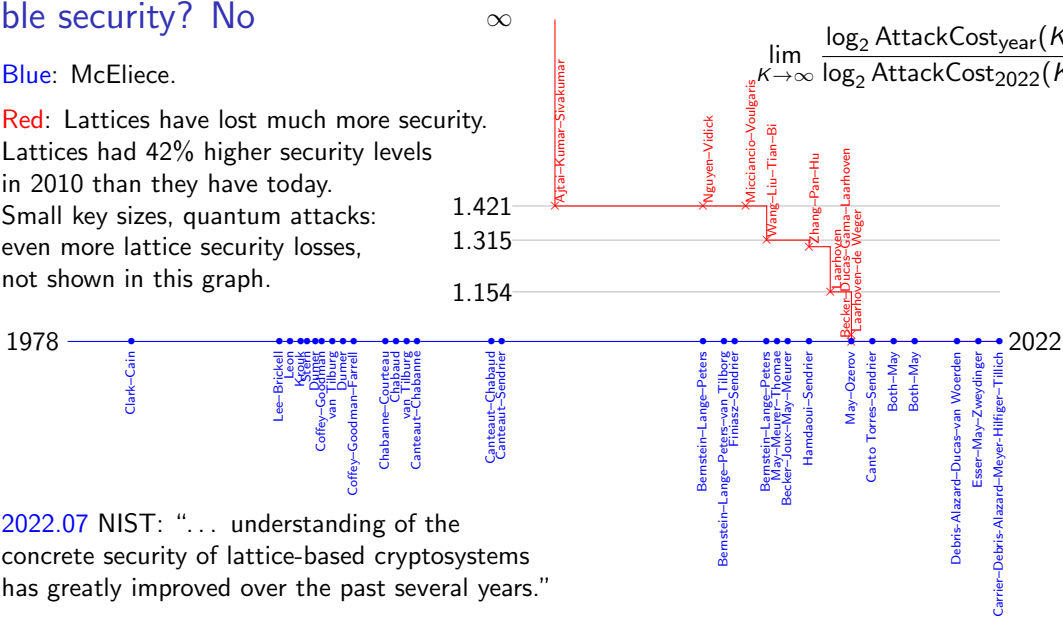
Red: Lattices have lost much more security.
Lattices had 42% higher security levels
in 2010 than they have today.
Small key sizes, quantum attacks:
even more lattice security losses,
not shown in this graph.



Stable security? No

Blue: McEliece.

Red: Lattices have lost much more security. Lattices had 42% higher security levels in 2010 than they have today. Small key sizes, quantum attacks: even more lattice security losses, not shown in this graph.



2022.07 NIST: "... understanding of the concrete security of lattice-based cryptosystems has greatly improved over the past several years."

Security is job #1?

2016.12 NIST call for submissions:

- "Many researchers have begun to investigate post-quantum cryptography . . .
  The goal of this research is to develop cryptographic algorithms that
  would be secure against both quantum and classical computers."
- "The security provided by a cryptographic scheme is
  the most important factor in the evaluation."
- "Assuming good overall security and performance, schemes with greater flexibility" are
  "preferable". Example 4.C.1.e of "flexibility": "The scheme can be incorporated into
  existing protocols and applications, requiring as few changes as possible."

Deployability *on the Internet* is an important feature of post-quantum cryptography.
The exact extent of *protocol* modifications is much less important.

Security is job #1? No

2016.12 NIST call for submissions:

- "Many researchers have begun to investigate post-quantum cryptography ...
  The goal of this research is to develop cryptographic algorithms that
  would be secure against both quantum and classical computers."
- "The security provided by a cryptographic scheme is
  the most important factor in the evaluation."
- "Assuming good overall security and performance, schemes with greater flexibility" are "preferable". Example 4.C.1.e of "flexibility": "The scheme can be incorporated into existing protocols and applications, requiring as few changes as possible."

Deployability *on the Internet* is an important feature of post-quantum cryptography.
The exact extent of *protocol* modifications is much less important.

2022.07 NIST report, change of mission:

- "This field is often referred to as post-quantum cryptography ...
  The goal is to develop schemes that can be deployed in existing communication networks **and protocols without significant modifications**." (emphasis added)

"It's easy! First you shoot the arrow, then you just take your paint brush and ..."

www.brotherjuniper.com

http://brotherjuniper.com/2016/05/its-easy-first-you-shoot-the-arrow/

Another one bites the dust

2022.07.30 Castryck–Decru: "An efficient key recovery attack on SIDH (preliminary version)". Script breaking *all* proposed SIKE parameters.

2022.08.02 Pope: Sage script reimplementing Castryck–Decru attack with various speedups. Several others quickly joined the ~~beating~~ optimization efforts.

2022.08.08 Maino–Martindale: "An attack on SIDH with arbitrary starting curve." (Independent of Castryck–Decru.)

2022.08.11 Robert: "Breaking SIDH in polynomial time."

2022.08.12 Oudompheng, Wesolowski: Papers describing speedups.

https://github.com/jack4818/Castryck-Decru-SageMath

Breaking SIDH on a Laptop

~ Running Time	`SIKEp64`	`$IKEp217`	`SIKEp434`	`SIKEp503`	`SIKEp610`	`SIKEp751`
Paper Implementation (Magma)	-	6 minutes	62 minutes	2h19m	8h15m	20h37m
Our implementation (SageMath)	5 seconds	2 minutes	10 minutes	15 minutes	25 minutes	1-2 hours
Direct Computation (Oudompheng)	2 seconds	9 seconds	22 seconds	2 minutes	15 minutes	1 hour

Note: Especially for the higher NIST levels, a lot of time is spent getting the first digits, and so performance time varies based on whether or not the first few values are `0` (fastest) or `2` (slowest).

Understanding of the concrete security of SIKE has greatly improved over the past days.

It's 2022 and PQC is still not widely deployed.

That's the real disaster!

What can you do now? Deploy hybrids!

Combine one (or more) pre-quantum schemes with one (or more) post-quantum schemes.

Public-key signatures:
All individual signatures must be valid for the hybrid signature to be valid.

Public-key encryption:
Use multiple systems to jointly generate key for use in symmetric cryptography.

Examples of options to "encrypt the encryption":

- Wrap PQC as payload inside pre-quantum (benefit for length fields).
- Wrap pre-quantum inside PQC (limit the attack surface – quantum attacker cannot even break pre-quantum scheme).

Choice of systems:

- Different recommendations for rollout in different risk scenarios:
  - Use most efficient systems with ECC or RSA, to ease usage and gain familiarity.
    Matches Google and Cloudflare experiments.
  - Use most conservative systems with ECC or RSA, to ensure that data really remains secure.
    If you actually have some data you need to protect.
- Some PQ libraries exist, quality is getting better.

Further information

- https://pqcrypto.org our overview page.
- PQCrypto 2016, PQCrypto 2017, PQCrypto 2018, PQCrypto 2019, PQCrypto 2020, PQCrypto 2021 with many slides and videos online.
- Coming soon PQCrypto 2022 online.
- https://pqcrypto.eu.org: PQCRYPTO EU Project.
  - PQCRYPTO recommendations.
  - Free software libraries (libpqcrypto, pqm4, pqhw).
  - Many reports, scientific articles, (overview) talks.
- YouTube channel Tanja Lange: Post-quantum cryptography.
- https://2017.pqcrypto.org/school: PQCRYPTO summer school with 21 lectures on video, slides, and exercises.
- https://2017.pqcrypto.org/exec and https://pqcschool.org/index.html: Executive school (less math, more perspective).
- Quantum Threat Timeline from Global Risk Institute, 2019; 2021 update.
- Status of quantum computer development (by German BSI).
- NIST PQC competition.