Queen's University
Belfast

Towards Quantum Resistant Cryptography Standards

Dr. Markku-Juhani O. Saarinen
Research Fellow, Queen's University Belfast
`m.saarinen@qub.ac.uk`

Invited Talk, CTCrypt '16
Yaroslavl, Russia – 07 June 2016

Background and Terminology

Quantum Computing (QC) uses quantum superpositions, rather than binary digits, to perform computations. This computational model was first considered in early 1980s.

Quantum Algorithms are algorithms for Quantum Computers. They often have different performance asymptotics from classical algorithms.

Shor's Algorithm (1994) can factor integers and compute discrete logarithms efficiently. It has also been extended to the Elliptic Curve Discrete Logarithm Problem. Together, these attacks can be devastating to most of current public key cryptography.

Grover's Search Algorithm (1996) can be used to search for a k-bit secret key with $\sqrt{2^k} = 2^{k/2}$ quantum effort. It effectively doubles the required key sizes for ciphers.

QC has been dismissed by practical cryptographers until recent years. General purpose quantum computers do not currently exist; however progress is being made.

CNSS Advisory Memorandum 02-15 in August 2015

In August 2015 the Committee on National Security Systems (CNSS) and National Security Agency (NSA) suddenly revised their cryptographic recommendations in CNSSAM 02-15.

"Based on analysis of the effect of quantum computing [..] the set of authorized algorithms is [changed] as we anticipate a need to shift to quantum-resistant cryptography in the near future."

The recommendation killed off shorter key lengths (AES-128, SHA-256, RSA-2048, DL-2048, ECC P-256) allowed in "Suite B".

The new (interim) set of algorithms is called "Commercial National Security Algorithm Suite" and is approved up to TOP SECRET:

RSA 3072, DH 3072, ECDH/DSA P-384, SHA-384, AES-256.

Practical Impact of CNSSAM 02-15

- **Significant financial and industry impact** due to the large installed base and scope of National Security Systems – and related, compatible international standards and systems. Example: Most P-256 ECC hardware is now essentially redundant.
- **New risk analysis, QC prognosis.** The change implies changes in NSA's risk outlook. Note that the change affects handling of TOP SECRET information, which should remain resistant to passive cryptanalysis for 50 years or more. In addition to D-Wave and IBM's effort, NSA probably has additional non-public data points in their curve tracking the development of quantum computing.
- **Symmetric cryptography is least affected.** Most effective symmetric attack techniques such as differential and linear cryptanalysis are mostly bound by known plain / ciphertext data complexity – quantum computing does not change this. However key sizes need to be doubled due to Grover's search algorithm.
- **Post-Quantum Standardization.** Standards for public key cryptography need to be significantly revised. We need altogether new post-quantum algorithms.

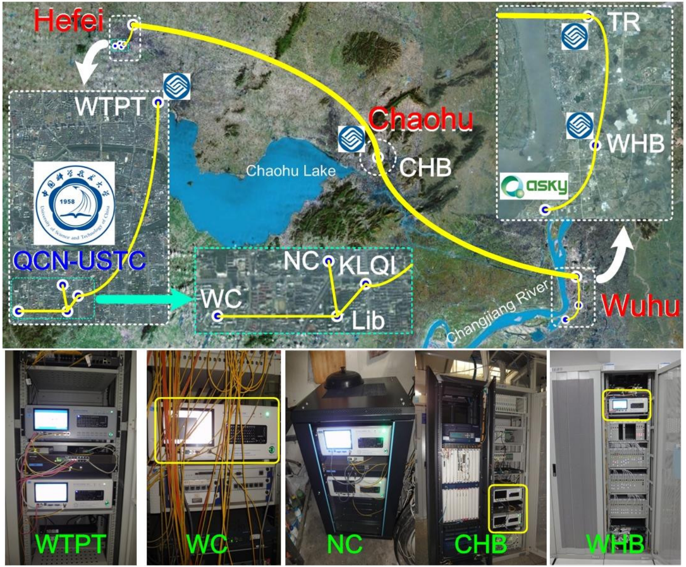
Physicists' Approach: Quantum Key Distribution

QKD is an approach to key distribution whose security is derived from quantum mechanical channels.

Some countries have massive QKD programs (incl. WANs). China is e.g. performing QKD satellite experiments.

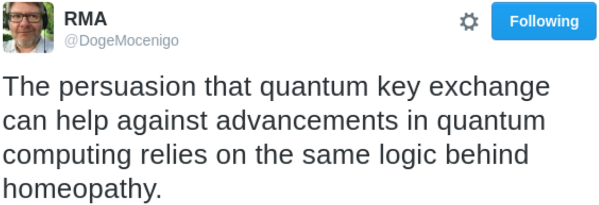
After keys are obtained, bulk encryption and authentication use symmetric algorithms.

Problems: Channel hacking and infrastructure costs.



Post-Quantum Cryptographer’s T-Shirt

Roberto Avanzi was **not** endorsing homeopathy when he wrote:



The persuasion that quantum key exchange can help against advancements in quantum computing relies on the same logic behind homeopathy.

Prof. Daniel J. Bernstein (on EU’s €1-billion “Quantum Manifesto”):

> *“Fundamentally, quantum cryptography [..] aims for security in an oversimplified model of the physical world, takes resources away from more serious security techniques, and ends up damaging security in the real world.”*

CESG White Paper on Quantum Key Distribution

In February 2016 **CESG** (the Information Security Arm of **GCHQ**) equally surprisingly published a white paper on QKD.

QKD is distinct from post-quantum public key cryptography, which is based on classical mathematical problems that are hard to solve even in the presence of quantum computers.

Quote: *"[This paper] makes the case for research into developing post-quantum public key cryptography as a more practical and cost-effective step towards defending real-world communications systems from the threat of a future quantum computer."*

U.K. Government position: Forget QKD, use Post-Quantum.

NIST’s Quantum-Resistant Standardization Plan

In February 2016 NIST announced a forthcoming plan to select and standardize post-quantum public key algorithms for signatures, encryption, and key establishment.

In April 2016 this plan was formally set forth in **NIST Internal Report 8105** “Report on Post-Quantum Cryptography.”

Algorithms must be public with no IPR restrictions. The selection process will be run in a similar fashion to AES and SHA3 competitions. However there may not be a single winner. NIST will choose one or more in each category.

NIST expects standardization to be complete in early 2020s. Relevant work is also ongoing within ETSI and IETF.

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

By end of 2016: *(Soon!)*
Formal Call for Proposals.

Late in 2017:
Deadline for submissions.

Following 3–5 years:
Public analysis phase. NIST will report its findings.

So what's available ?

Lattice-based cryptography. Usually actually based on (Ring or Ideal Lattice variants of) the Shortest Integer Solution (SIS) and Learning With Errors (LWE) problems. Encryption (R-LWE, NTRU?), Signatures ("BLISS"), and Key Exchange ("New Hope").

Code-based cryptography. Based on coding theory. There are still-secure parameters for the 1978 McEliece cryptosystem. Bigger keys. Good mainly for encryption.

Multivariate polynomial cryptography. Based on solving systems of multivariate polynomials. Many schemes have been broken, but signatures still have potential.

Hash-based signatures. These are based on cryptographic hash functions, and their PQ security is well understood. The algorithms have limitations, large signature size. Strong proposals for signatures (XMSS – CFRG Draft, SPHINCS – EuroCrypt '15).

Other. There are additional proposals such as the isogeny problem of supersingular curves (Crypto '16 Isogeny Diffie-Hellman key exchange proposal), braid groups, etc.

Practical Lattice-Based Post-Quantum Algorithms

I’ve spent the last year studying practical Lattice-based signature, encryption, and key exchange algorithms. There is something of a consensus on viable design approaches.

Security estimates are more complex and undefined than algorithms themselves.

These are based on a number of theorems, algorithms, attacks, countermeasures, simplifications, and some plain conjectures. Claims of “provable security” do not rule out the prospect of cryptanalysis.

There are at least half a dozen fundamental attacks on these schemes, each of which must be considered in parameter selection.

More analysis is required for confidence.

Attack	(m, b)	Classical	Quantum	Plausible
My code “Trunc8”		$q = 12289, n = 512, \sigma = 4.859$		
Primal	$(660, 496)$	144	129	102
Dual	$(674, 494)$	144	**129**	102
NTRU Prime [BeChLa+16]		$q = 9829, n = 739, \sigma = 0.7430$		
Primal	$(600, 480)$	140	125	99
Dual	$(618, 478)$	139	**125**	99
NTRU-743 [HoPiSc+15]		$q = 2^{12}, n = 743, \sigma = 0.8164$		
Primal	$(613, 603)$	176	157	125
Dual	$(635, 600)$	175	**157**	124
BCNS [BoCoNa+15]		$q = 2^{32} - 1, n = 1024, \sigma = 3.192$		
Primal	$(1062, 296)$	86	77	61
Dual	$(1055, 296)$	86	**77**	61
NewHope [AlDuPo+16]		$q = 12289, n = 1024, \sigma = 2.828$		
Primal	$(1100, 967)$	282	253	200
Dual	$(1100, 962)$	280	**252**	199
JarJar [AlDuPo+16]		$q = 12289, n = 512, \sigma = 3.464$		
Primal	$(623, 449)$	131	117	93
Dual	$(602, 448)$	130	**117**	92

Implementing Lattice-Based Post-Quantum Algorithms

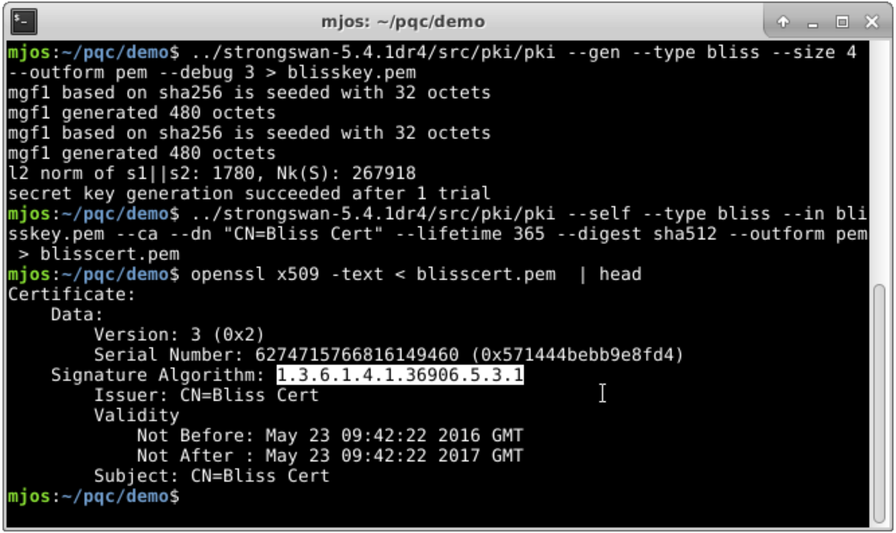
A complete Ring-LWE based suite has emerged, consisting of BLISS Signatures (Crypto '13), "New Hope" KEX (USENIX Security '16), and public key encryption in same the anti-cyclic rings formed of polynomials modulo $f(x) = x^n + 1$ with coefficients in the field $\mathbb{Z}_q$. Here q is a (small) prime such as $q = 3 * 2^{12} + 1 = 12289$.

From implementation perspective, the main **new requirements** for Ring-LWE are:

- **Fast polynomial ring arithmetic**. In rings of dimension $n = \{256, 512, 1024\}$ implementations can utilize Finite Field FFT and vectorization optimizations.
- **Nonuniform random number generation.** Efficient and cryptographic quality "sampling" from discrete Gaussian and Binomial distributions.
- **Encoding and decoding.** Due to their nonuniform nature, keys, signatures, and ciphertexts should be encoded with Huffman or Arithmetic codes.

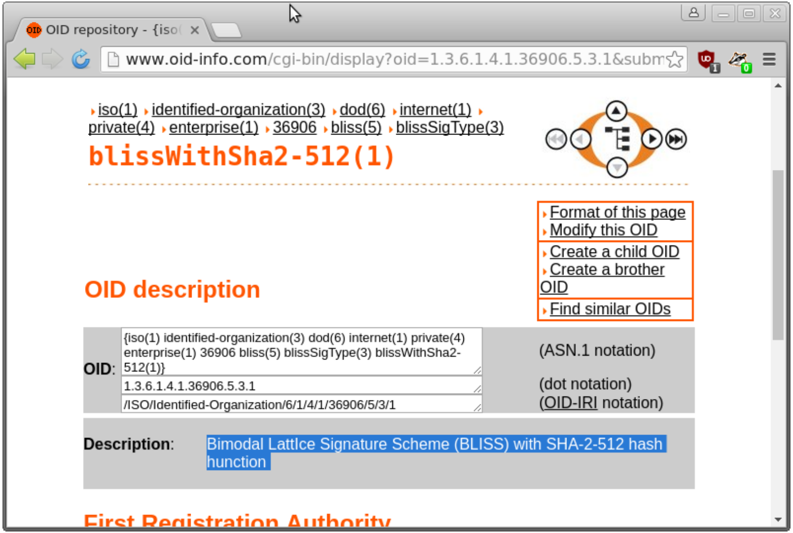
..In the rest of the slides, I will discuss some recent industry developments.

strongSwan OpenSource IPsec-based VPN: strongswan.org



```
mjos:~/pqc/demo$ ../strongswan-5.4.1dr4/src/pki/pki --gen --type bliss --size 4
--outform pem --debug 3 > blisskey.pem
mgf1 based on sha256 is seeded with 32 octets
mgf1 generated 480 octets
mgf1 based on sha256 is seeded with 32 octets
mgf1 generated 480 octets
l2 norm of s1||s2: 1780, Nk(S): 267918
secret key generation succeeded after 1 trial
mjos:~/pqc/demo$ ../strongswan-5.4.1dr4/src/pki/pki --self --type bliss --in bli
sskey.pem --ca --dn "CN=Bliss Cert" --lifetime 365 --digest sha512 --outform pem
 > blisscert.pem
mjos:~/pqc/demo$ openssl x509 -text < blisscert.pem  | head
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 6274715766816149460 (0x571444bebb9e8fd4)
    Signature Algorithm: 1.3.6.1.4.1.36906.5.3.1
        Issuer: CN=Bliss Cert
        Validity
            Not Before: May 23 09:42:22 2016 GMT
            Not After : May 23 09:42:22 2017 GMT
        Subject: CN=Bliss Cert
mjos:~/pqc/demo$
```

BLISS Ring-LWE Certificates can use OID 1.3.6.1.4.1.36906.5.x



OID repository - {iso(
www.oid-info.com/cgi-bin/display?oid=1.3.6.1.4.1.36906.5.3.1&subm

iso(1) identified-organization(3) dod(6) internet(1) private(4) enterprise(1) 36906 bliss(5) blissSigType(3)

blissWithSha2-512(1)

- Format of this page
- Modify this OID
- Create a child OID
- Create a brother OID
- Find similar OIDs

OID description

OID:	{iso(1) identified-organization(3) dod(6) internet(1) private(4) enterprise(1) 36906 bliss(5) blissSigType(3) blissWithSha2-512(1)}	(ASN.1 notation)
	1.3.6.1.4.1.36906.5.3.1	(dot notation)
	/ISO/Identified-Organization/6/1/4/1/36906/5/3/1	(OID-IRI notation)

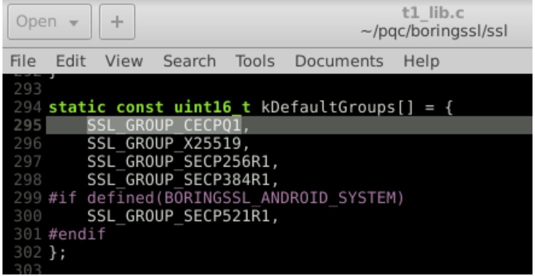
Description: Bimodal LattIce Signature Scheme (BLISS) with SHA-2-512 hash hunction

First Registration Authority

BoringSSL: The Google fork of the OpenSSL Library

BoringSSL provides a TLS stack for Google projects such as Android, Chrome Browser, Gmail, Google Search, ... *everywhere Google*. It has been largely written from scratch.

Latest development version implements **New Hope Ring-LWE key agreement** (Alkim et al: "Post-Quantum Key Exchange - a New Hope", USENIX Security '16, to appear.)

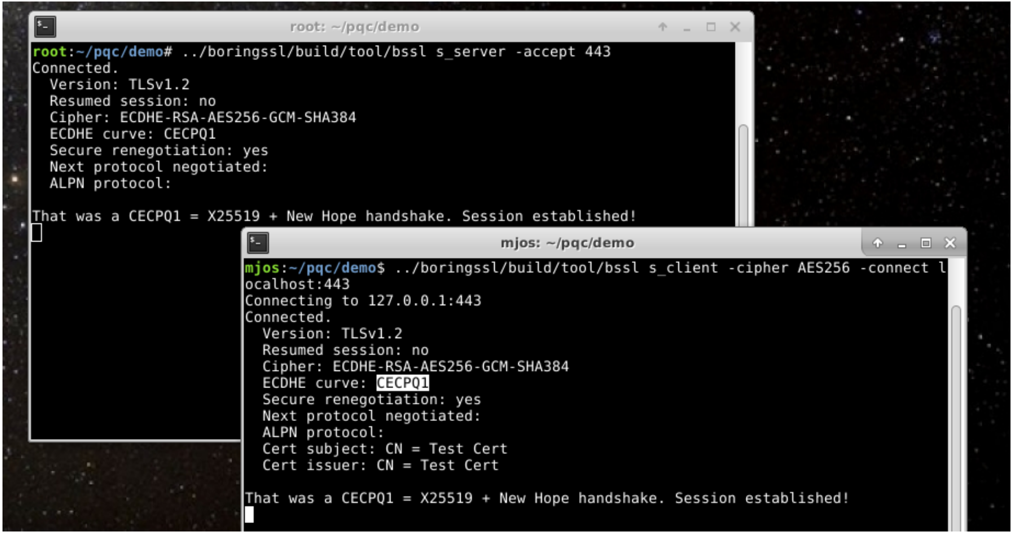


```c
static const uint16_t kDefaultGroups[] = {
    SSL_GROUP_CECPQ1,
    SSL_GROUP_X25519,
    SSL_GROUP_SECP256R1,
    SSL_GROUP_SECP384R1,
#if defined(BORINGSSL_ANDROID_SYSTEM)
    SSL_GROUP_SECP521R1,
#endif
};
```

The "CECPQ1" (New Hope + X25519) experimental key exchange can be enabled by adding a single line of code.

Authentication is performed with classical signatures. CECPQ1 is intended to provide "passive forward security" for current sessions.

BoringSSL TLS v1.2 session with CECPQ1 (New Hope + X25519)



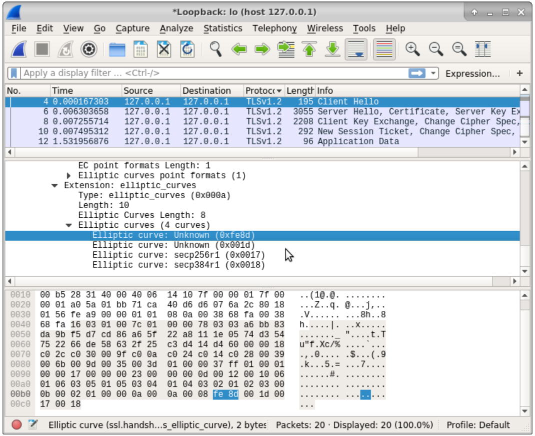
```
root:~/pqc/demo# ../boringssl/build/tool/bssl s_server -accept 443
Connected.
  Version: TLSv1.2
  Resumed session: no
  Cipher: ECDHE-RSA-AES256-GCM-SHA384
  ECDHE curve: CECPQ1
  Secure renegotiation: yes
  Next protocol negotiated:
  ALPN protocol:

That was a CECPQ1 = X25519 + New Hope handshake. Session established!
```

```
mjos:~/pqc/demo$ ../boringssl/build/tool/bssl s_client -cipher AES256 -connect l
ocalhost:443
Connecting to 127.0.0.1:443
Connected.
  Version: TLSv1.2
  Resumed session: no
  Cipher: ECDHE-RSA-AES256-GCM-SHA384
  ECDHE curve: CECPQ1
  Secure renegotiation: yes
  Next protocol negotiated:
  ALPN protocol:
  Cert subject: CN = Test Cert
  Cert issuer: CN = Test Cert

That was a CECPQ1 = X25519 + New Hope handshake. Session established!
```

BoringSSL TLS v1.2 session with CECPQ1 (New Hope + X25519)


*Loopback: lo (host 127.0.0.1)
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

```
No.  Time          Source     Destination  Protoc▾  Lengtł Info
   4 0.000167303   127.0.0.1  127.0.0.1    TLSv1.2   195 Client Hello
   6 0.006303658   127.0.0.1  127.0.0.1    TLSv1.2  3055 Server Hello, Certificate, Server Key Ex
   8 0.007255714   127.0.0.1  127.0.0.1    TLSv1.2  2208 Client Key Exchange, Change Cipher Spec,
  10 0.007495312   127.0.0.1  127.0.0.1    TLSv1.2   292 New Session Ticket, Change Cipher Spec,
  12 1.531956876   127.0.0.1  127.0.0.1    TLSv1.2    96 Application Data
```

```
        EC point formats Length: 1
      ▸ Elliptic curves point formats (1)
    ▾ Extension: elliptic_curves
        Type: elliptic_curves (0x000a)
        Length: 10
        Elliptic Curves Length: 8
      ▾ Elliptic curves (4 curves)
          Elliptic curve: Unknown (0xfe8d)
          Elliptic curve: Unknown (0x001d)
          Elliptic curve: secp256r1 (0x0017)
          Elliptic curve: secp384r1 (0x0018)
```

```
0010  00 b5 28 31 40 00 40 06  14 10 7f 00 00 01 7f 00   ..(1@.@. ........
0020  00 01 a0 5a 01 bb 71 ca  40 d6 d6 07 6a 2c 80 18   ...Z..q. @...j,..
0030  01 56 fe a9 00 00 01 01  08 0a 00 38 68 fa 00 38   .V...... ...8h..8
0040  68 fa 16 03 01 00 7c 01  00 00 78 03 03 a6 bb 83   h.....|. ..x.....
0050  da 9b f5 d7 cd 86 a6 5f  22 a8 11 1e 05 74 d3 54   ......._ "....t.T
0060  75 22 66 de 58 63 2f 25  c3 d4 14 d4 60 00 00 18   u"f.Xc/% ....`...
0070  c0 2c c0 30 00 9f c0 0a  c0 24 c0 14 c0 28 00 39   .,.0.... .$...(.9
0080  00 6b 00 9d 00 35 00 3d  01 00 00 37 ff 01 00 01   .k...5.= ...7....
0090  00 00 17 00 00 00 23 00  00 00 0d 00 12 00 10 06   ......#. ........
00a0  01 06 03 05 01 05 03 04  01 04 03 02 01 02 03 00   ........ ........
00b0  0b 00 02 01 00 00 0a 00  0a 00 08 fe 8d 00 1d 00   ........ ........
00c0  17 00 18                                           ...
```

Elliptic curve (ssl.handsh...s_elliptic_curve), 2 bytes · Packets: 20 · Displayed: 20 (100.0%) · Profile: Default

There is no RFC for the new key exchange method, so a temporary identifier 0xFE8D is currently used for CECPQ1.

However, the method is reasonably fast (2500 New Hope exchanges / second), implementation is robust, and CECPQ1 offers reasonable “future proofing”.

The TOR developers are also seriously considering adding New Hope to the TOR protocol, routers, and browsers.

Conclusions

1. New quantum resistant cryptography is required to handle national security information. About 4.5 million people hold security clearances in United States alone; this requirement affects millions of computers and information processing systems.

2. Government and civilian cryptographers see classical but "post-quantum" public key algorithms as a preferable solution to Quantum Key Distribution. NIST is organizing an international effort to standardize such algorithms in the same vein as AES and SHA3.

3. According to NSA estimates, current widely used public key algorithms such those based on RSA and Elliptic Curves may be made redundant during 2020s. Lattice-based Crypto (e.g. Ring-LWE) offers some of the most promising post-quantum alternatives.

4. Research in this area is strongly and officially encouraged by at least U.S. and U.K. COMSEC authorities – as security is not sufficiently well understood yet. However, vendors such as Microsoft and Google have already started implementing PQ Crypto.

..thank you!