Post-Quantum Cryptography: tomorrow's security

Jean-Christophe Deneuville[1]

[1] LIFO, INSA-CVL, 18 000 Bourges, France
jean-christophe.deneuville@insa-cvl.fr

Abstract. In this paper, we review state of the art quantum attacks against existing protocols, and study alternatives to existing protocols. As quantum computing emerges, existing cryptosystems are getting less secure. Time has come to think about quantum-safe transition.

Keywords: Cryptography, Security, Computer Science, Quantum Computing.

1 Introduction

1.1 Historical context

For millenaries, cryptography – also known as the science of secrets – has meant to provide high instances (military generals, kings, governments, …) with a secure way to communicate. In a nutshell, it allows to securely store sensitive data or transmit it across an insecure channel (messengers, mail, internet, …) so that only the intended recipient (who knows some secret) can read it.

More recently, new developments in public key cryptography have spurred worldwide applications such as e-commerce. The number-theoretic based tools that ensure the security have received a lot of attention and are therefore believed sustainable against "regular" adversaries (with standard machines).

Meanwhile, lots of progress have been made in the area of quantum computing, and quantum algorithms to efficiently solve number theory problems have been designed. As they drastically improve on their classical conterparts, the security tools are becoming vulnerable, and it is of prime importance designing quantum-safe primitive to replace them.

1.2 Existing security mechanisms

E-commerce applications require several cryptographic primitives for security. The first of them is *authentication*, which provides the client with strong guarantees on the server's identity. *Confidentiality* ensures that communications between them are secure. Then *integrity* guarantees that their exchanges are not modified over the wire. Finally, using *non-repudiation*, neither the client nor the server can deny a transaction they have issued.

Authentication and confidentiality are usually provided by encryption, whereas integrity and non-repudiation are ensured using digital signatures. Most public key encryption and signature schemes used for internet applications rely on number theory problems, such as *integer factorization*, or the *discrete logarithm problem*.

Both of them are long-standing problems, for which decades of research have only yield exponential time classical algorithms to solve them. That explains why they are believed to be hard hence safe for cryptographic usage.

1.3 Quantum algorithms for cryptanalysis

In the meantime, an alternative model for computer science was proposed in the early 80s: *quantum computing*. In this model, the atomic information is no longer a bit 0 or 1, but rather a *superposition* of the previous states, called a quantum bit (or qubit for short). Using another quantum mechanics phenomenon (entanglement), quantum computing allows to parallely explore all possible solutions to a problem, whilst classical algorithms inherently operate sequentially.

In 1994, Shor managed to take advantage of this quantum parallelism and proposed two *polynomial-time* quantum algorithms for the integer factorization and the discrete logarithm problems [1]. The complexity drop is illustrated in Fig. 1.

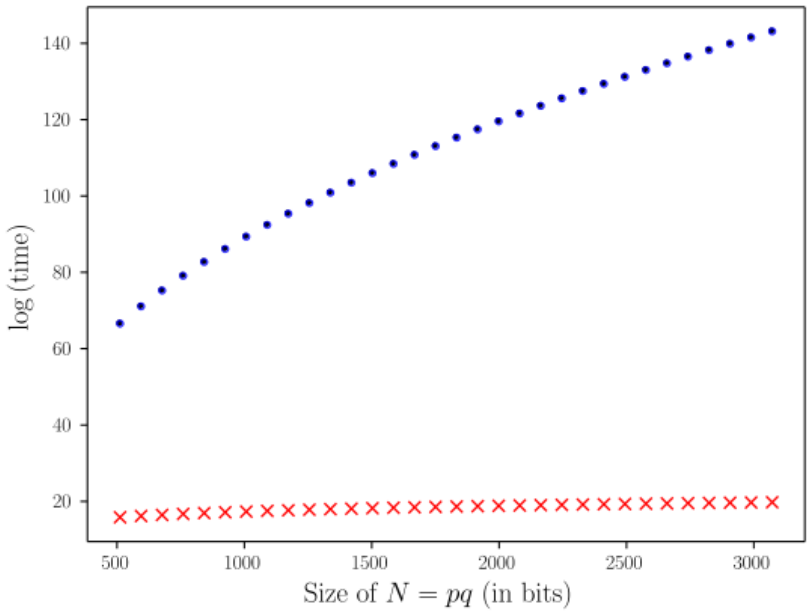

Fig. 1. Classical (blue dots) versus Quantum (red crosses) complexity of the Integer Factorization problem, as a function of the size of the inputs. Notice that the complexity for the discrete logarithm problem roughly behaves similarly.

As shown in Fig. 1, the computational cost of solving either problem is significantly reduced, especially for large input sizes. In 2018, most websites use 2048 bits keys in their certificate to authenticate. Given the above complexities, these certificates are presumably secure for approximately 2 years, against classical adversaries.

Now imagine a quantum adversary, running a quantum computer to break a 2048 bits RSA keys. The computational time required for him to succeed collapses from 2 years to within a week!

1.4 The race to quantum supremacy

As previously demonstrated, quantum computing enables such speedups that it becomes conceivable to quantumly solve previously classically intractable problems. Fortunately (from a cryptographer point of view), there seems to be caveats specific to quantum computing.

The first of them is the so-called number of qubits. Indeed, Shor's algorithm requires about the same amount of qubits as the bit length of the input (*e.g.* 2048). Technical improvements regularly double the number of available qubits, similarly to Moore's law for classical computers.

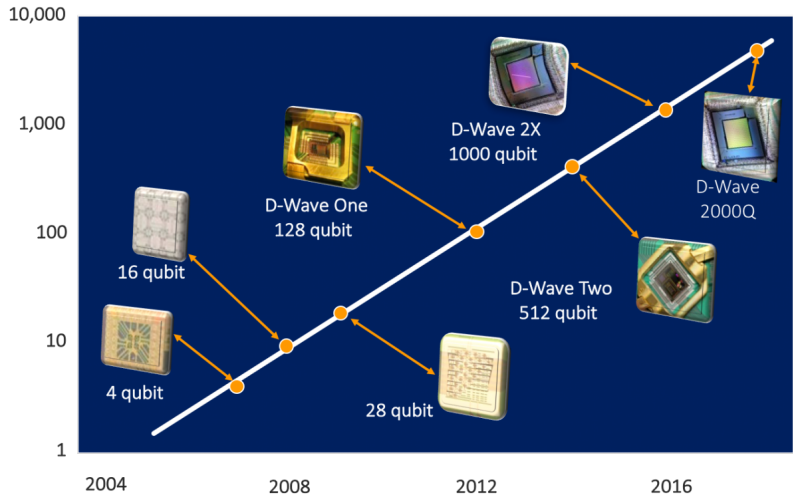

Fig. 2. Analog to Moore's law for quantum computing: the number of qubits (y-axe, logarithmic scale) approximately doubles every year (x-axe). (Source: D-Wave)

But the aforementioned qubits are not general purpose, they suffer from the second caveat: *fault-tolerance*. In quantum computing, errors usually arise due to uncontrolled interactions between qubits and their environment, making medium/long computations such as Shor's algorithm impossible.

These two engineering problems probably constitute the main hurdle to quantum cryptanalysis: scaling up the *number of fault-tolerant qubits*. On the bright side, this leaves cryptographers a moratorium to design and experience quantum-safe primitives. In 2018, the most advanced quatum computer features 72 fault-tolerant qubits, already allowing to solve previously inaccessible problems related to combinatorics, optimization and more, but not yet cryptographic problems.

2 Quantum-safe candidates

As explained in the previous section, quantum cryptanalysis is becoming a real threat for well established cryptography. Fortunately, alternative mathematic tools exist, for which no quatum speedup is known. We review the most promising of them for building tomorrow's security (the following list is non-exhaustive due to space restriction).

2.1 Code-based cryptography

Coding theory encodes information using redundancy in a way such that it is possible to detect – and even correct – errors that could arise during the message transmition over a noisy channel. In 1978, McEliece proposed the first public key encryption scheme based on coding theory. The idea is to encrypt a message by adding noise to it, and decrypt it using an efficient decoding algorithm, known only to the legitimate decrypter. More formally, the encryption scheme consists of the following algorithms:

- Key generation: Alice choses an error correcting code C, generated by a matrix G, for which she knows an efficient decoding algorithm D that can correct t errors. She scrambles G into $\hat{G} = SGP$ in order to hide D. The public key is $(\hat{G}, t)$ and the private key is (S, G, P, D).
- Encrypt: To encrypt a message m, Bob samples a uniformly random binary error of weight t (all zeros except for t coordinates, equal to one) and sends to Alice the ciphertext: $c = m\hat{G} + e$
- Decrypt: To recover m, Alice computes $D(cP^{-1})S^{-1} = D(mSG + eP^{-1})S^{-1} = m$, because $D(mG + e) = m$ provided that e has weight less or equal to t.

McEliece originally proposed to use binary Goppa codes. While other variants using other families of codes have been attacked due to their structure, the original proposal is still considered secure, and could be used as a replacement for encryption.

Unfortunately, code-based digital signatures would need more study: existing approaches require to be able to find small weight vectors for any message syndrome, which is relatively inefficient.

2.2 Lattice-based cryptography

Lattices are periodic sets of points in space. Similarly to code-based cryptography, messages are encoded into lattice points, then encrypted using a random noise of bounded (euclidean) norm. However, signature schemes are more efficient due to the regularity of the mathematical tool.

Lattice-based cryptography constitutes a serious post-quantum candidate, although most proposals are quite recent. An important aspect for security confidence is the comprehension of best known attacks. While sieving and enumeration are well studied and understood, lattice reduction only have loose worst-case upper bound, and algorithms tend to perform much better in practice than in theory. This can be problematic to set up tight parameters for constrained devices.

2.3 Hash-based cryptography

The main property of a hash function is *one-wayness*: it is easy to compute any element's image, but computationally infeasible to inverse the function. These objects are mainly used for authentication, but they can also be useful for building cryptography. However, such primitives do not benefit from security reductions to well known problems. Also, more study would be needed to get rid of the random oracle model.

2.4 Multivariate-based cryptography

Finding a common root to a set of non-linear multivariate polynomials has been proved to be a hard problem. Based on this observation, several cryptographic primitives have been proposed. Although there exists efficient multivariate-based signature schemes, most of them suffer from large key sizes. Additionally, appropriate cryptanalytic tools (namely Gröbner basis) lack an accurate evaluation of their complexity, making it a difficult task to set correctly parameters.

3 A post-quantum key exchange protocol

One of the most urgent primitive to design for quantum-safe security is key exchange. Indeed, if symmetric cryptography could still be used (at the price of doubling the key sizes) in a post-quantum world, protocols to exchange these keys need to be quantum-safe. We present such a protocol in this section, together with some parameters.

3.1 Overview of the protocol

The protocol presented hereafter is versatile in the sense that it can be adapted to several post-quantum candidates, such as coding theory (Hamming and rank metric) [2], or even lattice-based cryptography [3]. In Fig. 3, the exchanged secret is a sparse vector e.

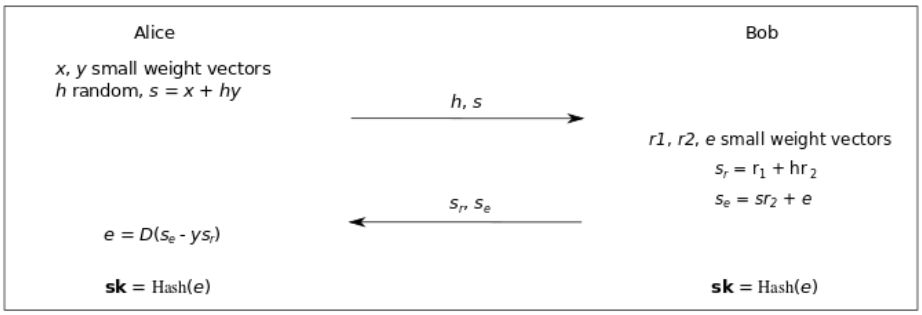

Fig. 3. Description of the generic Ouroboros key exchange protocol

This protocol has been proved secure under fairly hard problems (see original papers for more details), and relatively efficient (see Tab. 1).

3.2 Parameters for different candidates

Ouroboros can be instanciated using several metrics, such as the Hamming metric, the Rank metric or even the euclidean metric. Each of them features different security guarantees and parameters. Tab. 1 proposes different sets of parameters for the different metrics.

Table 1. Key sizes (in bits) for the Ouroboros key exchange protocol for different metrics.

Security	Hamming	Rank	Lattice	Pre-quantum
80	x	x	<1000	<1,024
100	x	x	<1300	<2,048
128	<23,000	<10,000	<1,600	<4,096

Conclusion

In this paper, we gave an overview of the rising quantum threat as well as hints to overcome the resulting attacks. While countermeasures exist, more work is required to experience their security, and make them even more practical. (Notice that the National Institute of Standards and Technology has began a process for standardizing post quantum primitives.)

References

1. Shor, P. W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on* (pp. 124-134). IEEE.
2. Deneuville, J. C., Gaborit, P., & Zémor, G. (2017, June). Ouroboros: A simple, secure and efficient key exchange protocol based on coding theory. In International Workshop on Post-Quantum Cryptography (pp. 18-34). Springer, Cham. ISO 690
3. Deneuville, J.-C., Gaborit, P., Guo, Q., Johansson, T. Ouroboros-E: An efficient Lattice-based Key-Exchange Protocol, to appear at the International Symposium on Information Theory (ISIT'18)