cryptovision

An Introduction to Post-Quantum Cryptography

Anyone interested in information security currently has to deal with post-quantum cryptography - i.e. cryptographic methods that withstand a quantum computer. This is definitely a challenge, because there are a great many post-quantum crypto algorithms. The mathematics behind them is challenging and putting them into practice is far from a no-brainer. This whitepaper provides a generally understandable introduction to post-quantum cryptography.

Atos

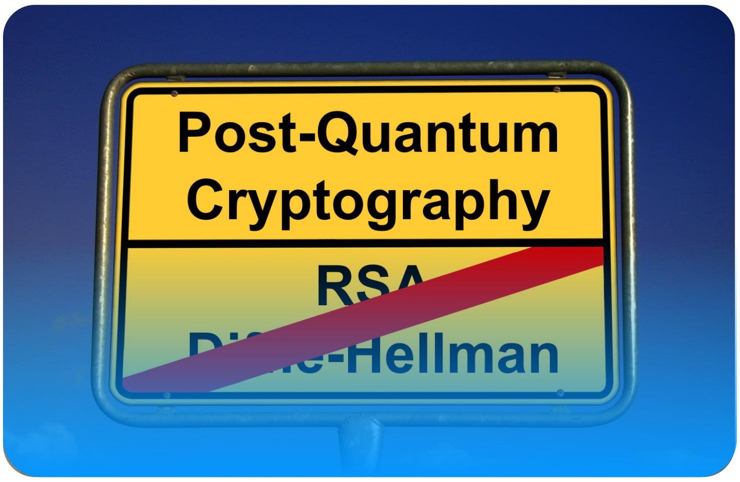
Preface: The post-quantum age is dawning

The quantum apocalypse is approaching. We therefore need to urgently look for quantum-safe alternatives to some of the current crypto methods.

Imagine that a hacker could access millions of online bank accounts and make transfers at will. Assume that the same hacker is able to read all encrypted emails that fall into his hands. And, in addition, imagine that this criminal person would be able to penetrate almost any corporate network in order to spy on it.

An unrealistic scenario? Not at all, because that's exactly how it could turn out if one day there are powerful quantum computers. Because these devices can be used to crack the RSA and Diffie-Hellman methods - two crypto methods that are used billions of times in web browsers, e-mail clients, smartphones and ATMs. The digital apocalypse would become reality.

Fortunately, we're not there yet. Although quantum computers already exist, they can so far only decompose smaller numbers into their factors. To threaten RSA or Diffie-Hellman, they would have to manage a similar operation with a 700-digit number. They won't be able to do that today or tomorrow.



But numerous experts are currently conducting intensive research on quantum computers and ensuring constant improvements - so the apocalypse is drawing closer. We must therefore look in good time for alternatives to RSA and Diffie-Hellman that are not vulnerable to quantum computers. Such methods do exist, and they are grouped under the term "post-quantum cryptography."

So far, post-quantum cryptography methods are hardly used in practice. On top of that, they are still too little researched to be used without hesitation. However, progress is being made here as well, and a number of post-quantum methods have now emerged with which we can venture into the post-quantum age.

In any case, we will have to deal with post-quantum cryptography in the coming years. This will not be easy. The various methods are diverse and mathematically extremely demanding. The descriptions of these algorithms available so far are mostly comprehensible only to specialists. And then numerous challenges await when it comes to implementing post-quantum cryptography - if only because one is usually dealing with particularly long keys and low performance. So there is a lot to do.

This whitepaper is intended to help bring post-quantum cryptography to a wider audience. Deeper mathematical knowledge is not necessary to read it.

Atos hopes you enjoy reading it!

Table of content

Chapter 1 - Introduction

Chapter 2 - Quantum computers

Chapter 3 – Post-quantum cryptography

Chapter 4 – What's next?

Chapter 1 – Introduction

How is encryption used today?

The legendary Enigma looked like a typewriter. The encryption device converted the typed letters into a jumbled mess of characters that could only be unraveled with an identical machine and a correctly set combination of numbers (key). The Germans used the Enigma, of which almost 40,000 were produced, to encrypt their Morse code during World War II.

Today, Morse code has been replaced by e-mails and Internet connections. These also have to be encrypted. This is done using methods such as the **Advanced Encryption Standard (AES)**, which, like the Enigma, processes a key without which it is impossible to unravel the encryption code.

The AES and the Enigma are examples of **symmetric encryption**. They each use the same key for encryption and decryption. The sender and receiver must agree on this key in advance.

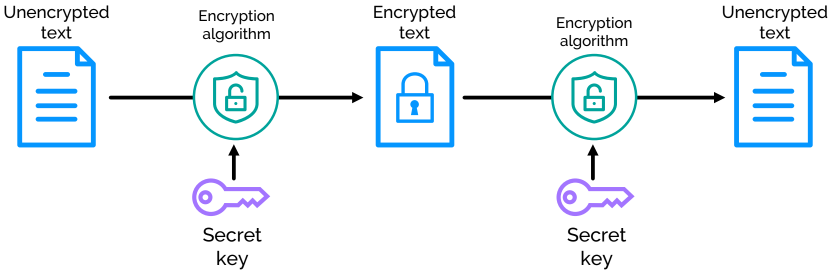
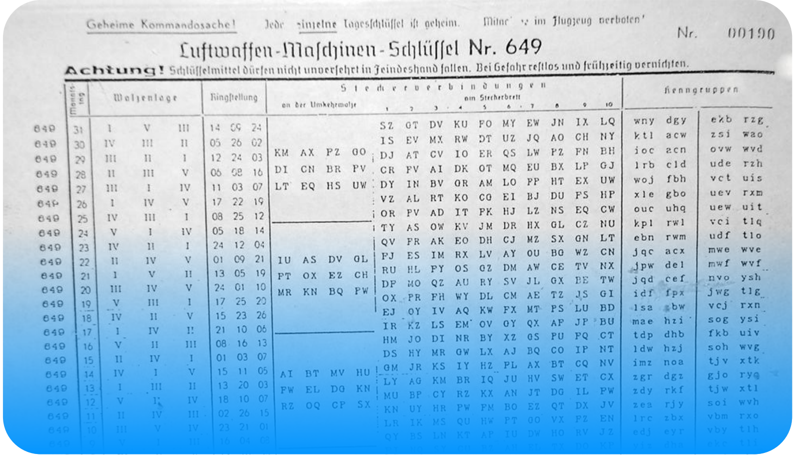


Figure 1: Symmetric encryption uses the same key for encryption and decryption. The sender and receiver must agree on this key in advance.

What is asymmetric cryptography?

The fact that the sender and receiver have to know the same key repeatedly causes logistical problems and security gaps - this is referred to as the "key exchange problem". In World War II, for example, submarines had to take key books with them on their voyages so that the radio operator knew the Enigma keys needed for each day. Of course, such books sometimes fell into the hands of the enemy, allowing them to decrypt without authorization. On the worldwide Internet, it can already be a challenge to agree on a key with each communication partner individually.



Geheime Kommandosache! Jede einzelne Tagesſchlüſſel iſt geheim. Mitnahme im Flugzeug verboten! Nr. 00190

Luftwaffen-Maſchinen-Schlüſſel Nr. 649

Achtung! Schlüſſelmittel dürfen nicht unverſehrt in Feindeshand fallen. Bei Gefahr reſtlos und frühzeitig vernichten.

Figure 2: Key lists and key books, as they were necessary for the Enigma, are no longer needed with asymmetric cryptography..

In the 1970s, mathematicians developed a surprisingly effective solution to the key exchange problem. This envisaged special procedures in which special keys are used in pairs. One key is secret, the other public. **Asymmetric cryptography** was born.

On one hand, asymmetric cryptography enables **asymmetric encryption**. This can be imagined like a mailbox with a snap lock: Anyone can drop a message in, but only the owner of the key can get it out. Mathematically, this is implemented with two keys belonging to one user: with the **public key**, anyone can encrypt a message for this person, and with the help of the associated **private key**, only this person can decrypt the message again. Of course, the user must keep his private key secret. The public key, on the other hand, should be accessible to everyone.

On the other hand, asymmetric cryptography enables **digital signatures**. This is not a scanned signature, but a document digest created with a private key. Only the owner of this private key can generate it, but anyone can verify it with the help of the public key.

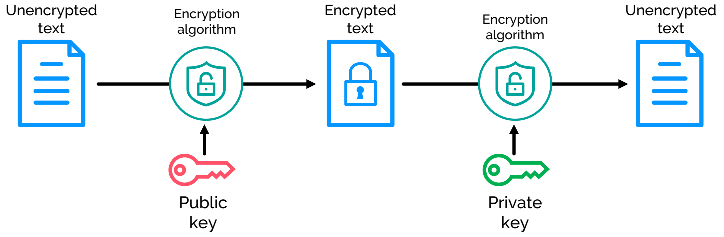
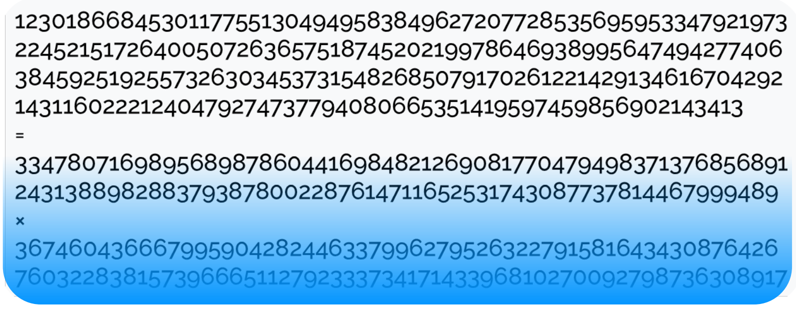


Figure 3: Asymmetric cryptography uses the public key for encryption and the private key for decryption.

How do RSA and Diffie-Hellman work?

The best-known and most widely used asymmetric encryption method is **RSA**. This was developed in 1978 and is named after the initials of its inventors Ron **R**ivest, Adi **S**hamir and Leonard **A**delmann.

RSA, like all asymmetric methods, is based on a one-way function. This is the name given to a mathematical function that is quick to compute, while the inverse requires a very large amount of computation. In the case of RSA, the one-way function is the multiplication of two prime numbers. Even if the numbers used have hundreds of decimal places, such an arithmetic operation can be done in seconds with a computer. The inverse, on the other hand, i.e., the decomposition of the prime number product into its factors (also known as factorization), is not nearly feasible within the lifetime of a human being, even with the best computers available today.



12301866845301177551304949583849627207728535695953347921973
22452151726400507263657518745202199786469389956474942774063
84592519255732630345373154826850791702612214291346167042921
431160222124047927473779408066535141959745985690214341 3

=

33478071698956898786044169848212690817704794983713768568912
431388982883793878002287614711652531743087737814467999489

×

36746043666799590428244633799627952632279158164343087642676
032283815739666511279233373417143396810270092798736308917

Figure 4: This 232-digit prime number product was decomposed into its two factors in 2009 after several years of computation. An entire cluster of computers needed 1500 processor years for this. The prime number products used for the RSA process typically have over 600 digits.

The exact operation of RSA is not for discussion here. However, it is important to know: In the RSA process, the private key consists of two prime numbers (in practice, these have 300 to 700 digits), while the product of these forms the public key. It is therefore quite easy to calculate the public key from the private key, but this does not work the other way around.

Some other asymmetric methods - among them **Diffie-Hellman** - are based on the fact that the calculation of the exponential function is simple in certain mathematical structures, while the inverse (i.e. the logarithm) is very complex. This is called the discrete logarithm. The exponential function in question is a one-way function.

Diffie-Hellman is not suitable for encryption, but two communication partners can use it to securely agree on a common secret key. They can then use this key for AES, for example. The Diffie-Hellman method thus solves the key exchange problem.

Factorization and the discrete logarithm are mathematically related. If it is possible to solve one problem - that is, to invert the corresponding one-way function - then the other problem is also solved. This means that all common asymmetric crypto methods ultimately depend on the same one-way function.

The RSA method can also be used for digital signatures, and it is even possible to use the same pair of keys. The private RSA key is used for signing, the public RSA key for verification.

Chapter 2 – Quantum computers

What is a quantum computer?

Conventional computers, as they are used today, function according to the laws of classical physics. A bit can assume two states in such a computer, either 0 or 1 (see Figure 5).

A **quantum computer**, on the other hand, is based on quantum mechanical phenomena. Such a device uses quantum bits (qubits) that can assume the states 0 and 1 simultaneously. Quantum computers can therefore perform certain computational steps in parallel rather than sequentially. This quantum effect allows computing power to increase significantly and ensures that quantum computers can - at least in theory - perform some tasks orders of magnitude faster than conventional computers.

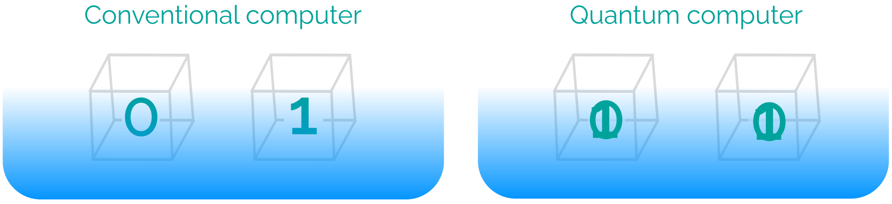


Figure 5: A bit of a conventional computer can only ever assume the value 0 or 1. In a quantum computer bit (Qubit), on the other hand, both states are possible at the same time. Qubits can therefore be used to perform several calculations simultaneously.

For example, quantum computers are able to search huge databases in a short time or pick out a particularly advantageous one from a large number of processes.

However, quantum computers have one disadvantage: although they can perform numerous calculations simultaneously, they can only ever deliver one result - for example, a database entry or an optimized operation. A quantum computer is therefore not suitable for sorting a list alphabetically, for example, since the result here is not a single list entry but the entire list.

What encryptions can be solved with quantum computers?

One of the tasks that a quantum computer can handle particularly effectively is the decomposition of a prime number product into the two associated primes. Since the RSA algorithm is based on precisely this mathematical principle, it is true that RSA can be cracked with a quantum computer. Diffie-Hellman and some other asymmetric crypto algorithms are also vulnerable to quantum computers.

Considering that RSA and Diffie-Hellman are used billions of times in web browsers, smartphones, VPN clients and elsewhere, the current development is alarming. For example, a hacker with a quantum computer could empty online accounts at will or decrypt encrypted emails - to name just a few examples. A catastrophe of apocalyptic proportions is looming.

Figure 6: Quantum computers could one day cause a global crypto catastrophe. However, the current models are still far too weak for this.

But there is no reason to panic yet. The quantum computers that have been realized so far are not particularly powerful and are also error-prone. To break an RSA key, a quantum computer needs about twice as many qubits as there are bits in the key. So with a key length of 2,048 bits, about 4,096 qubits are needed. However, these are error-free qubits, which do not exist in practice. The number of qubits needed in real terms for an RSA key could be 10 to 100 million. Today's quantum computers do not even come up with 100 qubits.

So we are still a long way from a quantum apocalypse. But that can change, because intensive research is being carried out. For example, the NSA is working on quantum computers. The European Union has announced a "Quantum Technology Flagship Project," while the German government has included two billion euros in its budget to promote quantum technology. By 2025, there should be the first quantum computer made in Germany. Google has even already succeeded in building practical quantum computers, even if they are not suitable for factorization and therefore do not pose a threat to cryptography. Hundreds of quantum computer start-ups have been founded.

Symmetric encryption schemes such as the AES can also be solved with quantum computers. However, the advantage that quantum technology brings here is much smaller than for asymmetric methods. The AES, for example, has a minimum key length of 128 bits - an order of magnitude that a quantum computer could just about manage in the distant future. If, on the other hand, 192 or 256 key bits are used, which the AES also supports, then even the best quantum computer will probably never stand a chance. So those who switch to longer AES keys in the next few years don't have too much to worry about. Many AES implementations have long since taken this step.

Chapter 3 – Post-quantum cryptography

What is post-quantum cryptography?

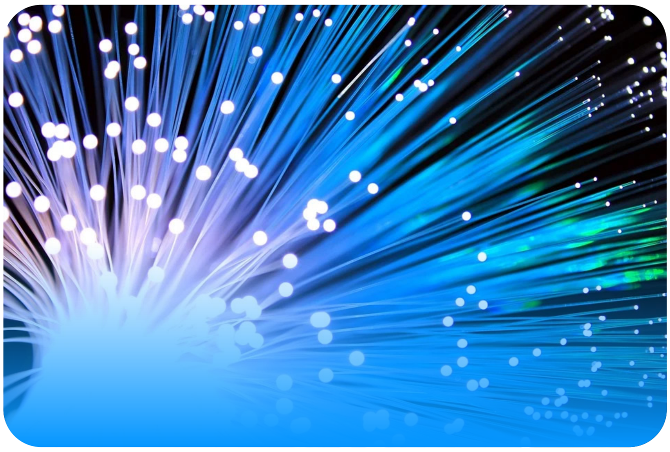
Fortunately, there are numerous other asymmetric crypto methods besides RSA and Diffie-Hellman. Some of them are not susceptible to quantum computers according to current knowledge. These are grouped together under the term **post-quantum cryptography**.

To begin with, the choice of post-quantum methods is large. However, virtually none of them has yet been put to practical use. Moreover, many methods of this kind are still comparatively poorly studied. This is urgently needed, however, because most post-quantum methods that were proposed at some point turned out to be insecure upon closer inspection.

The first important task for the next few years is therefore to further investigate post-quantum cryptography methods and to filter out the best of them. The second step is to put these methods into practice. The goal must be to replace RSA and Diffie-Hellman as completely as possible.

What is quantum cryptography?

Post-quantum cryptography should not be confused with **quantum cryptography**. The latter aims to use laser light to agree a secret key between two stations without the possibility of eavesdropping on the line - a concept also known as **quantum key distribution**. Quantum cryptography thus offers a solution to the key exchange problem. The data is usually transmitted using optical fiber.



Quantum cryptography has nothing to do with quantum computers, except that quantum physics is the base in both cases. In particular, quantum computers are not suited to perform or attack quantum cryptography. However, as quantum cryptography cannot be attacked with a quantum computer, it can be regarded as a part of post-quantum cryptography.

Compared to quantum computers, quantum cryptography is much easier to implement and is already being offered commercially in some cases. However, its usefulness is controversial. Since key exchange can be performed securely without quantum cryptography thanks to asymmetric cryptography, the former is sometimes referred to as a "solution without a problem".

What are the families of post-quantum crypto methods?

Over the past decades, well over 100 crypto methods have been developed that are considered quantum secure. Many of them showed security vulnerabilities that can be exploited without quantum computing, or proved impractical. Some other post-quantum methods, on the other hand, have so far withstood all attempts at attack.

It turned out that almost all post-quantum methods that are to be taken seriously belong to one of six families that differ in their mathematical foundations:

- **Lattice-based algorithms**: These methods operate in high-dimensional lattices.
- **Code-based algorithms**: Methods from this family use error-correcting codes.
- **Hash-based algorithms**: These algorithms are based on cryptographic hash functions.
- **Isogeny-based algorithms**: Methods from this family use isogenies between elliptic curves.
- **Multivariate algorithms**: Multivariate polynomials form the base of these algorithms.
- **Non-commutative algorithms**: Here, non-commutative groups form the base.

Post-quantum cryptography is currently a very active area of research. It is therefore not surprising that there has been considerable upheaval in recent years. Noncommutative methods, for example, have now largely disappeared from the scene after all too many methods from this family were broken.

Figure 7: Non-commutative crypto methods can be executed using a Rubik's cube, among other things. However, this family of post-quantum methods has not been able to gain acceptance due to security concerns.

Most multivariate crypto methods did not stand up to the critical eye of the experts, either. This post-quantum family is therefore now also considered a dead end. And then, in August 2022, SIKE, by far the most important isogeny method, was completely unexpectedly broken.

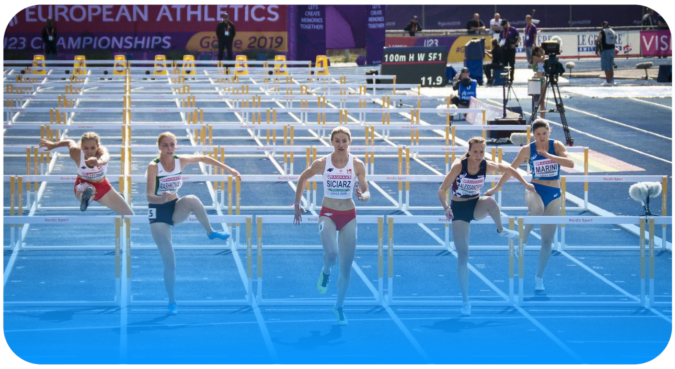
So there are still three of the post-quantum families in the running. The most promising are undoubtedly the lattice-based methods, some of which have proven to be equally secure and practical. The keys of these methods are usually significantly longer than those of RSA and Diffie-Hellman.

Code- and hash-based methods also have good security properties, but they are considered unwieldy because they require extremely long keys or generate extremely long signatures and are sometimes quite slow. As things stand, these post-quantum methods are likely to serve primarily as alternative solutions in the next few years if one does not want to rely exclusively on lattice methods.

And of course, everything could turn out differently in the end - after all, no one knows which weak points in which methods the experts will discover tomorrow.

What is the NIST post-quantum competition?

The U.S. authority NIST (National Institute for Standards and Technology) has held several competitions in recent decades to find the best possible cryptographic method for a specific purpose. The goal in each case was to standardize the winning algorithm.



The NIST competitions always had a great influence on the development of cryptography. For example, the aforementioned AES spread worldwide after it emerged as the winner of a NIST competition in 2000.

In 2017, NIST launched another competition. This time, the aim was to pit post-quantum crypto methods against each other. Experts from all over the world were invited to submit suitable algorithms for this purpose, from which the best methods were to be selected in a process lasting several years. The aim was to identify a portfolio of high-quality post-quantum methods for different purposes and with different mathematical foundations. Both signature and encryption or key exchange methods were eligible to participate.

NIST admitted 69 of the submitted methods to the competition. Many of the methods proved to be insecure or unsuitable on closer examination and were therefore eliminated from the race.

After three rounds of evaluation, NIST finally announced four winners in July 2022:

- **CRYSTALS-Kyber**: This is a lattice method for asymmetric encryption.
- **CRYSTALS-Dilithium**:Anotherlatticemethod,it is used for digital signature.
- **FALCON**: This signature method is also based on lattices.
- **SPHINCS+**: The hash-based SPHINCS+ is another signature method.

The NIST jury has also identified four additional candidates to be evaluated in a fourth round that will begin soon. After the first group of winners contained three lattice systems, the additional candidates are all non-lattice-based: Classic McElliece, HQC and BIKE are code-methods, while SIKE is based in isogenies. The latter was broken in August 2022. Whether this means that SIKE will be eliminated from the competition, or whether developers will be able to make improvements, was not yet clear when this whitepaper went to press.

And finally, NIST announced a new competition: Since no signature method with short and quickly verifiable signatures could be recommended in the previous course, NIST intends to call for new submissions in this area.

As in the past, NIST's decisions are expected to have a major impact in the IT world. Undoubtedly, the various winning procedures will also be incorporated into numerous standards and products outside the United States. However, NIST advises against implementing the four algorithms now, as minor details may change between now and standardization.

What do the BSI and the IETF say?

The German Federal Office for Information Security (BSI) is also keeping a close eye on current developments in quantum computing and post-quantum cryptography. Of course, the experts there are initially guided by the NIST competition, the outcome of which will also have a major impact in the German-speaking world. The BSI has not yet commented on the first four winning methods, but that will change.

For the time being, the BSI in its document "Kryptographische Verfahren: Recommendations and Final Lengths"* recommends Classic McEliece and FrodoKEM. The former is a code procedure and is one of the four algorithms currently being evaluated by NIST in the fourth round. FrodoKEM, another lattice method, on the other hand, has been eliminated from the NIST competition for the time being.

Classic McEliece and FrodoKEM are considered conservative choices. Neither method is among the most practical, but they have performed very well in safety considerations.

Classic McEliece is already over 40 years old, making it one of the oldest asymmetric methods ever. Since no one has found a weak point in over four decades, it can be assumed that it is secure. For this, one has to accept that the public keys are almost 700 times as long as with RSA.

FrodoKEM is also considered secure, but could not hold its own in the NIST competition due to a lack of efficiency.

The Internet Engineering Task Force (IETF), the Internet's standardization body, will undoubtedly also take its cue from the NIST competition. However, there are already two "Requests for Comments" (RFCs) that specify post-quantum procedures. These are the hash-based procedures XMSS (RFC 8391) and Leighton-Micali (RFC 8554), discussed below.

The IETF's choice is also considered conservative. Hash-based methods are least likely to have security vulnerabilities discovered at some point. In return, one accepts a low level of efficiency.

** BSI TR-02102 Kryptographische Verfahren: Empfehlungen und Schlüssellängen. January 2022*

How does lattice-based cryptography work?

Lattices

Figure 8 shows what is meant by a **lattice** in mathematics. In two-dimensional space, the definition of a lattice requires two vectors, which are called *A* and *B* here. *A* and *B* together are also called the **base** of the lattice. Points that can be reached with the help of the vectors are called **lattice points**.

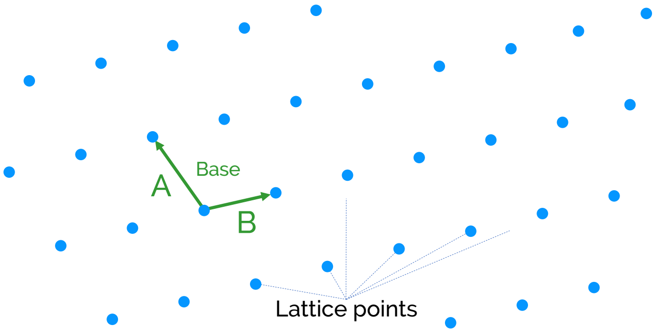


Figure 8: This lattice is defined with the two vectors *A* and *B*, which together are also called the base. Points that are rea-

As shown in Figure 9, there are always several bases that produce the same lattice. If the base vectors are nearly perpendicular to each other,

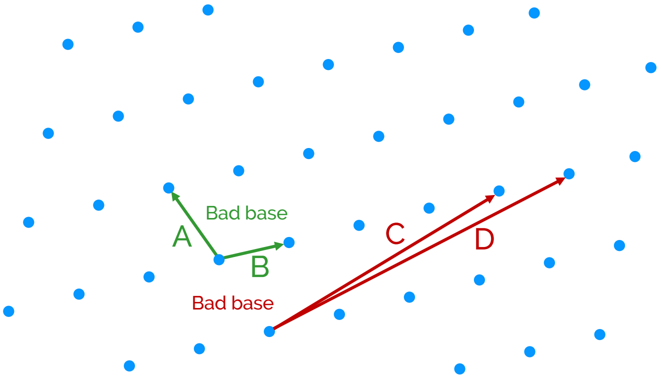


Figure 9: A lattice can always be defined with different bases. If the vectors of a base are approximately perpendicular to each other, it is called a good base; if the vectors are approximately parallel, it is called a bad base.

Of course, you can also define a lattice in three dimensions. For this, one needs a base with three vectors, whereby the third one must not lie in a plane with the other two.

In mathematics one is not satisfied with two or three dimensions, but knows for example also four-, five- or six-dimensional spaces. One cannot imagine anything under it, but one can calculate in such spaces quite well.

In post-quantum cryptography, we even have to deal with several hundred dimensions. For example, lattices in 500-dimensional space play a role there, for whose definition one needs accordingly a base with 500 vectors. In such a high-dimensional environment, one can simply calculate a bad base from a good one. The opposite way, on the other hand, is so complex that it would take billions of years even with the best computer.

How does CRYSTALS-Kyber work?

Encryption with CRYSTALS-Kyber

CRYSTALS-Kyber is one of the four post-quantum methods declared winners of the 2022 competition by NIST. It is the only encryption method in this quartett and intended as a post-quantum alternative to RSA.

CRYSTALS-Kyber is a lattice-based method and uses the so-called closest-vector problem (see Figure 10). In this, one assumes that a point *P* is given within a lattice, but it is not a lattice point. The question is: What is the closest lattice point to *P*?

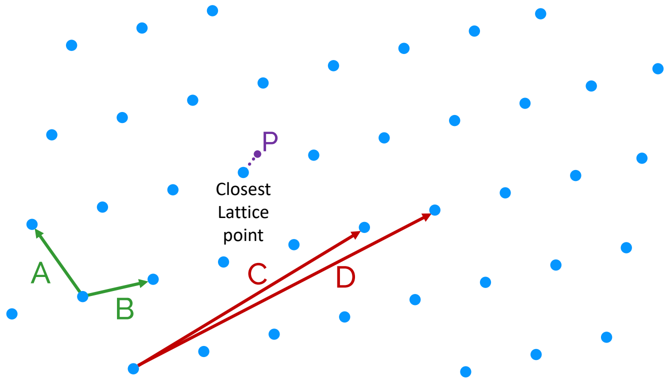


Figure 10: In the closest vector problem, a point *P* is given. The goal is to find the closest lattice point to *P*. In two-dimensional space, this is very simple. In 500-dimensional space, however, such a search is only practical with the help of a good base.

In the two-dimensional case, the closest vector problem is very easy to solve - *P* has only four neighboring points, and one of them must be the closest. In a 500-dimensional lattice, however, things look different. Here, an off-lattice point has no less than 2500 neighbors - a number with 150 digits. Fortunately, you don't have to try all of them to find the next one, because there are more effective methods.

The following applies: If a good base of the lattice is known, a computer can find the closest lattice point in a fraction of a second, even in high-dimensional spaces. If, on the other hand, only a poor base is available, then even the strongest computer will fall to its knees.

CRYSTALS-Kyber uses this principle. A good lattice base serves as the private key, while the public key is given by a bad base of the same lattice. To encrypt, the sender chooses a non-lattice point *P* in the immediate vicinity of a lattice point. The vector between the two points is the message. In 500-dimensional space, this vector has 500 components, which is enough to encode a 256-bit message, for example. The non-lattice point *P* is the ciphertet.

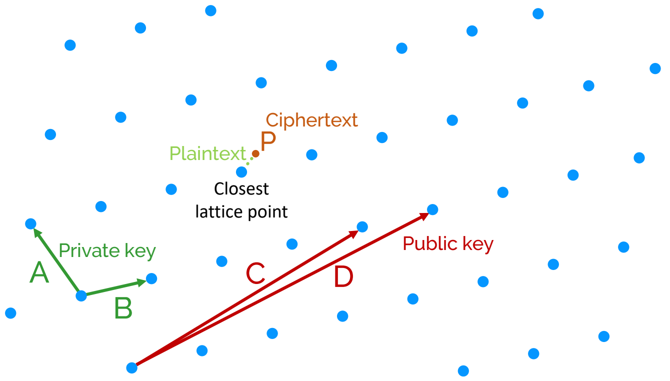


Figure 11: To encrypt with CRYSTALS-Kyber, the sender chooses a non-lattice point *P* next to a lattice point. The vector between the two points is the message. *P* is the ciphertext.

The receiver can easily reconstruct the message and thus decrypt the ciphertext, knowing a good base with which to compute the lattice point closest to the ciphertext. An attacker, on the other hand, has only a poor base available for this purpose, which makes it almost impossible to determine the lattice point in question.

How does CRYSTALS dilithium work?

Signing with CRYSTALS-Dilithium

CRYSTALS-Dilithium is also one of the four winners of the NIST competition. It is a digital signature method that, like CRYSTALS-Kyber, belongs to the lattice methods. However, the differences between CRYSTALS-Kyber and CRYSTALS-Dilithium are greater than the common name suggests.

CRYSTALS-Dilithium is also based on the closest-vector problem. The public key of the receiver is a non-lattice point P that is close to a lattice point. The latter forms the private key. Note that an attacker can compute the public key from the private key only if he solves the Closest Vector Problem, which is de facto impossible.

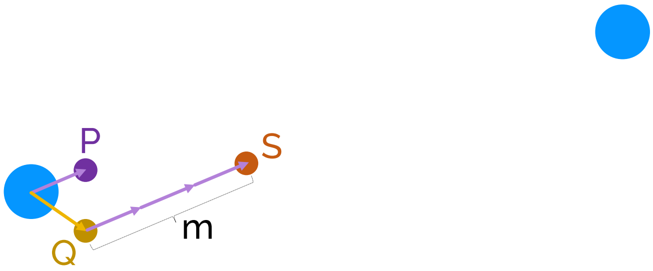


Figure 12: In this case, the message to be signed is a number, for example 3. For signing, the sender chooses a second non-lattice point Q, which is also close to the said lattice point, and calculates $S=Q+m^{*}P$. The resulting point S is the signature.

In this case, the message m to be signed is a number, say 3. To sign, the sender chooses a second off-lattice point Q (see Figure 12), which is also close to said lattice point, and computes $S=Q+m\cdot P$. The resulting point S is the signature.

To verify the signature, the receiver first checks whether $S=Q+m\cdot P$ holds. This is possible with a bad base. Furthermore he measures the distance between S and the lattice point - if this is so small that the lattice point must be the closest to S, the signature is genuine.

In the figure, this scheme works only with small numbers for m. If you take $m=10$, for example, you get too close to other lattice points. Therefore, one should think of the distances between lattice points in this case as being on the order of several kilometers, while points P and Q are only millimeters away from the lattice point in question. In this case, the near density m to be signed can also have a value of 100 without coming too close to other lattice points. Furthermore, it is relatively easy for the receiver to judge whether the distance between S and the lattice point is short enough. For example, if the distance is less than 50 centimeters, then the signature is most likely genuine, since a randomly chosen point would have been expected to be several hundred meters away.

In practice, the differences are not between centimeters and kilometers, but several dozen orders of magnitude more.

How does FALCON work?

Signing with FALCON

FALCON is the third lattice-based method to be selected as the winner in the NIST competition. It is a signature method.

FALCON is based on a problem described in Figure 13. In addition to the base vectors *A* and *B*, vectors *X*, *Y* and *Z* are added here. The number of the additional vectors must be larger than the number of the base vectors.

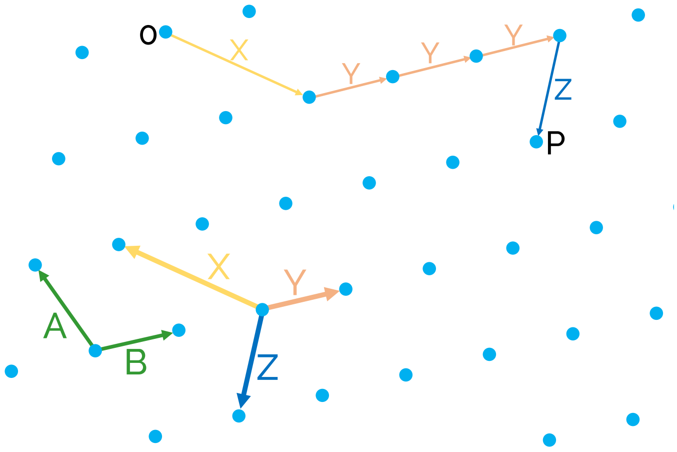


Figure 13: Here the question is: What is the shortest connection between *0* and *P* if only the vectors *X*, *Y* and *Z* are allowed as intermediate steps?

If one considers now a starting point *0* on the lattice as well as another lattice point *P*, then the question is: Which is the shortest connection between *0* and *P*, if only the vectors *X*, *Y* and *Z* are permitted as intermediate steps? In two-dimensional space, the answer is easy to find. In the 500-dimensional case, on the other hand: With a good base, the shortest path is relatively easy to calculate. With a bad base, even the strongest computer needs billions of years and more.

This principle can be used to explain the FALCON signature procedure (see Figure 14). In this case, the sender's private key is a lattice with an associated good base. The sender's public key is given by a bad base of the same lattice. Now, if the sender wants to sign a message, he converts it into a lattice point *P* . To create the signature, the sender calculates the shortest path between *0* and *P*, which is quite easy since he knows a good base. In the example, this shortest path is -*X*+3*Y*+*Z*, so the signature is -1, 3. 1.

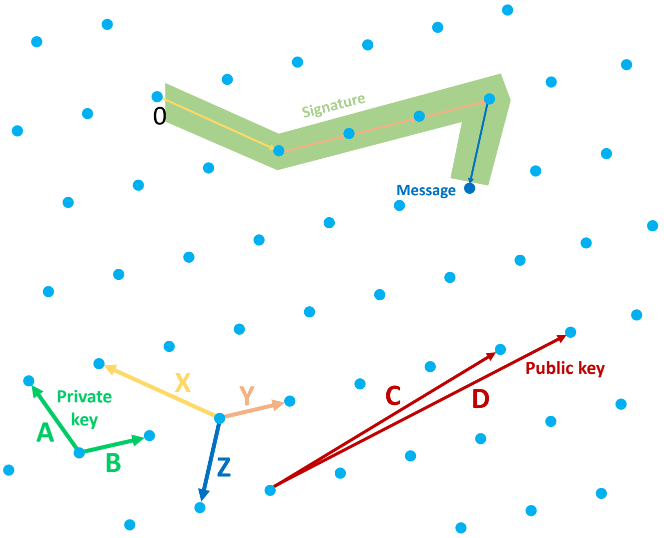


Figure 14: To create the signature, the sender calculates the shortest path between *0* and *P*, which is quite easy since it knows a good base. This shortest path results in the example from -*X*+3*Y*+*Z*, so the signature is -1, 3. 1.

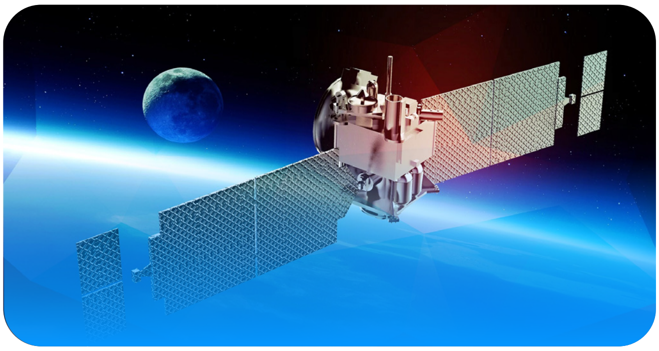
The receiver of the message can use the bad base to verify that the signature does indeed lead from *0* to *P* . Unfortunately, he cannot directly verify that it is indeed the shortest path. However, he can estimate the length of the shortest path based on an easy-to-compute formula and compare it with the determined actual length of the path present as a signature. If the difference is small enough, the signature is genuine.

How do hash-based methods work?

The family of hash-based crypto methods differs in several respects from the other five post-quantum families described.

Hash-based methods are comparatively simple mathematically. However, they can only be used for digital signatures, while they are not suitable for encryption. Another special feature is that a part of the private key must be made public for each signature, which limits the number of signatures per key.

Security considerations are the main arguments in favor of hash-based methods. Since the methods in this family originated back in the 1970s and are thus among the oldest asymmetric methods, they have been well studied. They are even provably secure under realistic conditions.



For this, hash-based signatures are considered unwieldy. Either the length of the signature or the length of the keys or the computing effort is too great for everyday use. Such methods are therefore particularly suitable when signing is performed infrequently, but particularly high and long-term security is required - for example, as a security anchor for communication with satellites.

The basic principle of hash-based signatures is comparatively simple:

1. The sender specifies two arbitrary constants *X* and *Y*, each consisting of 256 bits, for example. *X* and *Y* form the private key.

2. The sender applies a cryptographic hash function H to *X* and *Y* respectively. The results *A*=H(*X*) and *B*=*H*(*Y*) form the public key.

3. The sender now signs a bit as follows: If the bit has the value 0, he publishes *X*; if, on the other hand, it has the value 1, *Y* is published.

The receiver verifies this signature as follows: If the signed bit has the value 0, then it checks whether *A*=H(*X*). In the other case it checks whether *B*=H(*Y*).

However, this process is quite time-consuming - for the fact that only a single bit is signed. For example, if 256 bits are signed, then the sender must generate 256 values each for *X* and *Y*, apply the hash function to each of them, and publish the 512 results as a public key. In our example, this would give a length of over 131,000 bits each for the private and public keys, and this may then only be used for this one message. The signature itself is half as long as the key, i.e. about 65,500 bits. By comparison, RSA gets by with 2,048 bits of key and signature length, and a key can be used any number of times. There are various tricks to make this procedure more effective. However, these usually drive up the required computing time.

Another disadvantage: Since each value of *X* and *Y* may only be used once, the sender must remember which values have been used up and therefore keep a corresponding list. There is no such list for the crypto methods currently in use.

How do SPHINCS+, XMSS and Leighton-Micali work?

Signing with SPHINCS+

The hash-based signature method **SPHINCS+** is another winner in the NIST competition. SPHINCS+ uses the principle described in the previous chapter to generate signatures. Through various optimizations, the developers have succeeded in reducing the size of the public and private keys to a few hundred bits. In return, the signature is two orders of magnitude longer than with RSA, and the performance is among the worst of all post-quantum methods.

As a hash-based method, SPHINCS+ would actually need a used-key list. However, NIST wanted to avoid the difficulties associated with such a list from the outset and therefore only allowed signature algorithms without used-key lists for the competition.

SPHINCS+ therefore works with an additional trick: It provides a very large number of values for X and Y and provides that these are each selected randomly. If the number is large enough, the probability of a key being used twice becomes negligible. As a result, no used-key list is needed. SPHINCS+ therefore complies with the NIST specifications.

Signing with XMSS and Leighton-Micali

The Internet Standards Body IETF has now also published two hash-based signature schemes:

- **XMSS**: The eXtended Merkle Signature Scheme (XMSS) is described in RFC 8391.
- **Leighton-Micali**: This procedure is specified in RFC 8554.

Both procedures were published as Informational RFC, which means that they do not have official standard status. However, they can be considered as quasi-standards.

XMSS and Leighton-Micali require a list on which already consumed keys are noted. The two methods would therefore not have been eligible for the NIST competition.

Chapter 4 – What's next?

What needs to be done?

In the coming years, it will continue to be important to study the procedures of post-quantum cryptography. Cryptologists will undoubtedly find many more improvements, and they will discover vulnerabilities in procedures previously considered secure.

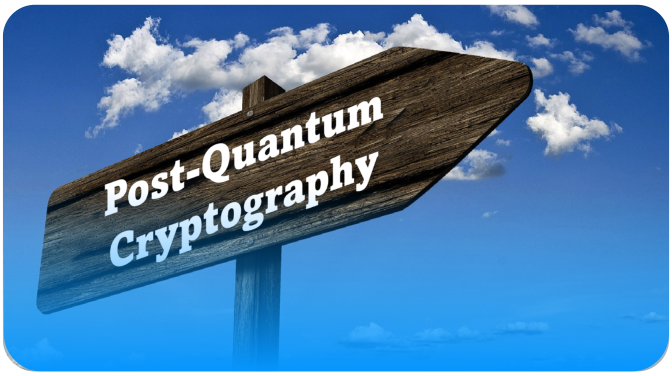
Standardization has already gained momentum with the four NIST winners and the two RFCs, but this development is still in its early stages. Standardizing the procedures is only the first step. In the second, they must be integrated into the corresponding formats and protocols.

Many experts, including the reponsible specialists from the French ANSSI* and the German BSI**, advocate using conventional encryption techniques and post-quantum processes in parallel during a transition phase. This would make it possible to live with any security gaps in the latter methods. After a few years, one could then switch over completely.

Some companies and organizations have already responded to the quantum threat. The NSA, for example, announced back in 2015 that it would tackle the migration to post-quantum algorithms in the near future. Before the appropriate standards are in place, however, such steps carry the risk of having backed the wrong horse.

Moreover, a central task will be to make the diverse and mathematically extremely demanding post-quantum methods accessible to as large an audience as possible. Since the descriptions of these algorithms that have been available so far are mostly only comprehensible to specialists, it is necessary to develop better descriptions. This white paper is intended to make a humble contribution to this.

And then numerous challenges await when it comes to implementing post-quantum cryptography. Current smart card chip architectures, for example, are mostly designed for RSA or Diffie-Hellman keys and have a corresponding coprocessor. In contrast, they are not designed to perform lattice or, code operations, certainly not with the necessary key lengths. The revision of current chip architectures is therefore an important challenge for the coming years.



There have long been numerous research projects investigating the use of the new methods in practice. Among the most important is the Aquorypt project ("Applicability of Quantum-Computer-Resistant Cryptographic Methods"), which is concerned with the implementation of post-quantum methods on chip cards and in embedded systems. It is supported by the German Federal Ministry of Education and Research. Meanwhile, at the IETF, there are several activities aimed at integrating post-quantum methods into Internet protocols.

Public key infrastructures (PKI), including X.509 and card-verifiable certificates, must also become post-quantum capable. The long keys alone make this a challenging undertaking. Several research projects are also underway in this area.

*www.ssi.gouv.fr/en/publication/anssi-views-on-the-post-quantum-cryptography-transition

**www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Krypto/Post-Quanten-Kryptografie.pdf

What is crypto agility?

Even when the first encryption formats for the Internet were developed more than three decades ago (it was initially a matter of e-mails), it was clear: the encryption methods used must be interchangeable. The manufacturers of crypto software were expected to enable the user to select his preferred crypto algorithms via a configuration setting and to being able to incorporate additional procedures into their products in a simple manner. In this way, it was possible to react quickly, especially if a procedure turned out to be insecure.

This principle is now widely used and known as **crypto-agility**. For instance, most protocols used on the Internet, such as TLS or IPsec, are crypto-agile, which means that many implementations allow switching from one crypto method to another at the click of a mouse. Crypto-agility is achieved primarily by the fact that the crypto processes used are not a fixed component of the respective solution, but are implemented in independent modules and called via precisely defined interfaces.

In the age of post-quantum cryptography, crypto-agility is more important than ever. In view of the threat posed by quantum computers and the fact that many a vulnerability has been discovered in post-quantum procedures in recent years, it must be possible to switch from one procedure to the other without major effort.

.

However, in the age of post-quantum cryptography, it is also more challenging than ever to ensure crypto-agility. This is because the various post-quantum methods have different properties than RSA and Diffie-Hellman. This is most noticeable in the keys, which are often many times longer in post-quantum cryptography than in conventional methods. Many a protocol cannot handle this so far. For a crypto manufacturer, therefore, it is not enough just to include an another library function.

In resource-poor environments, the low performance of post-quantum methods also stands in the way of crypto-agility.

The aforementioned used-key lists required by some of the hash-based methods are another challenge. This is because many crypto solutions are not set up to store data that will be queried in future crypto operations. This could affect the proliferation of such methods.

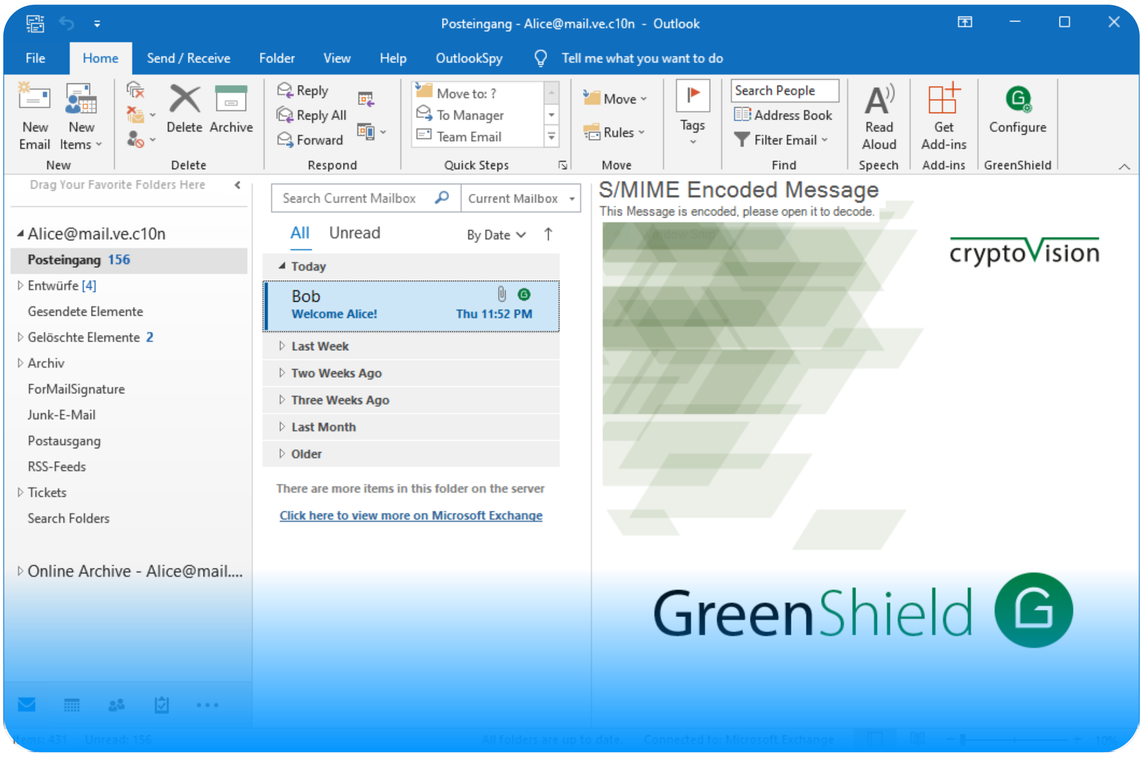
The use of post-quantum cryptography is thus inextricably linked to the crypto-agility paradigm. Undoubtedly, the market will more and more demand crypto-agile solutions, and manufacturers will have to adapt to this. Standards, benchmarks, and certifications need to be developed.

In addition, the increasing neessity for crypto-agility will make enterprise-wide key management even more important. If a new crypto algorithm shall be introduced or an existing one abolished, it is crucial to have an overview on which application uses which methods and to have the means to easily change this.

Who is Atos?

Atos is a global leader in digital transformation with 107,000 employees and annual revenues of over 11 billion euros. Atos is the number one worldwide in managed security services and Europe's number one in cybersecurity, cloud and high-performance computing. Atos offers tailored end-to-end solutions for all industries in 71 countries.

Since its founding in 1999, cryptovision, which was acquired by Atos in 2021, has focused exclusively on encryption technology and has developed, among other things, the proven, VS-NfD-approved GreenShield e-mail and file security solution (see Figure 15). The company has made a name for itself worldwide as an expert in secure yet user-friendly encryption solutions.



S/MIME Encoded Message

This Message is encoded, please open it to decode.

There are more items in this folder on the server

Click here to view more on Microsoft Exchange

Figure 15: The GreenShield software from Atos makes it possible to encrypt e-mails in a user-friendly way.

What is Atos doing in the area of post-quantum cryptography?

Atos is already preparing for the next generation of encryption technologies and is therefore also looking at post-quantum cryptography. Traditionally, the company has placed great emphasis on crypto-agility. Thus, the company's products typically support multiple crypto methods for the same purpose, with the ability to switch between them at the click of a mouse (see Figure 16).

In addition, obsolete methods can be easily deactivated and new ones incorporated. In this way, cryptovision managed the transition from RSA to ECC and from DES to AES. The transition to quantum-safe cryptography can be carried out using the same mechanisms. As soon as the first post-quantum methods are standardized and ready for use, Atos will immediately integrate them into the existing products in the manner described.

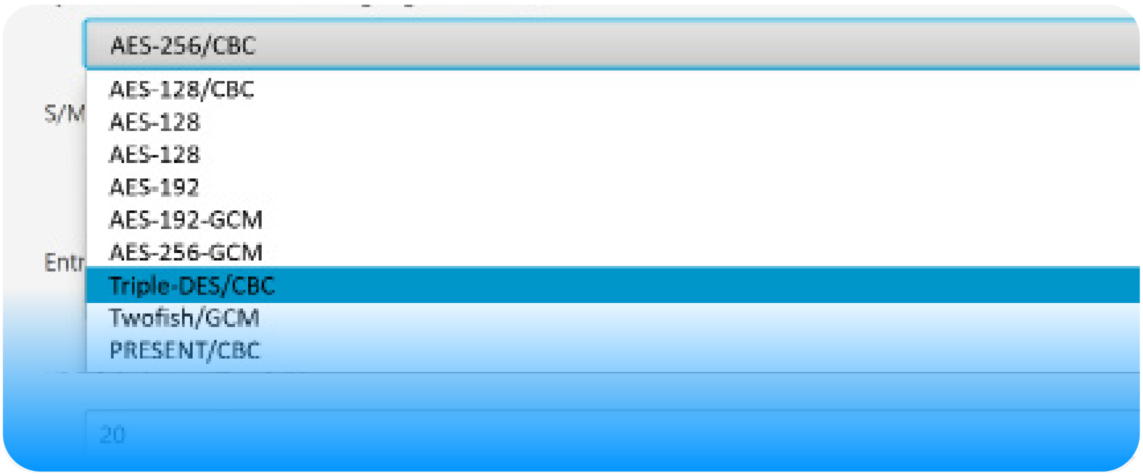


Figure 16: Atos places great emphasis on crypto agility. The company's solutions typically support multiple crypto methods for the same purpose, with the user able to switch between them at the click of a mouse.

How does Atos explain post-quantum cryptography?

Atos is aware that post-quantum cryptography can only become established if, in addition to specialists, as many developers, consultants, IT managers, administrators and IT executives as possible get to grips with it. This is not a foregone conclusion, because the mathematics behind the corresponding processes is complex and differs significantly from the principles that have prevailed in cryptography to date.

Even before the acquisition, cryptovision was involved in projects in which post-quantum cryptography is explained in an understandable way. The explanatory models developed by the company based on comics and everyday analogies are unique worldwide and have already been presented at numerous events - including the RSA Conference in San Francisco, Dragon Con in Atlanta and 44CON in London.

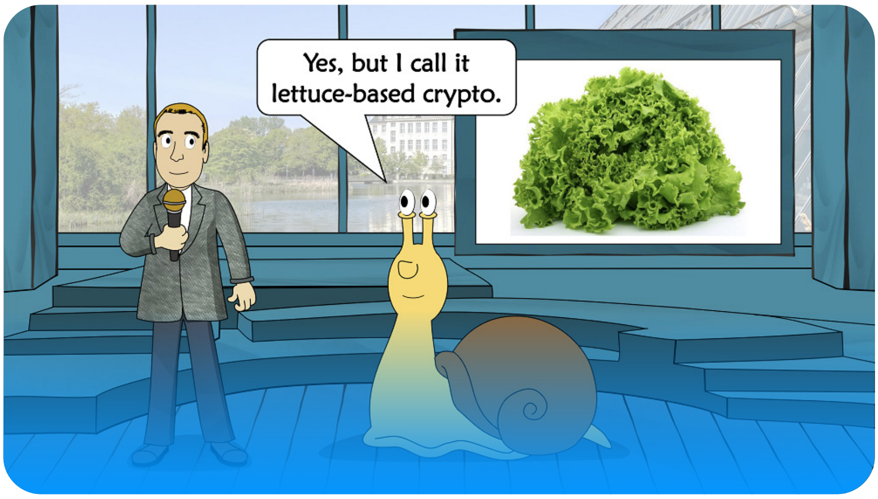


Figure 17: Atos works with models based on comics and everyday analogies that vividly explain post-quantum cryptography.

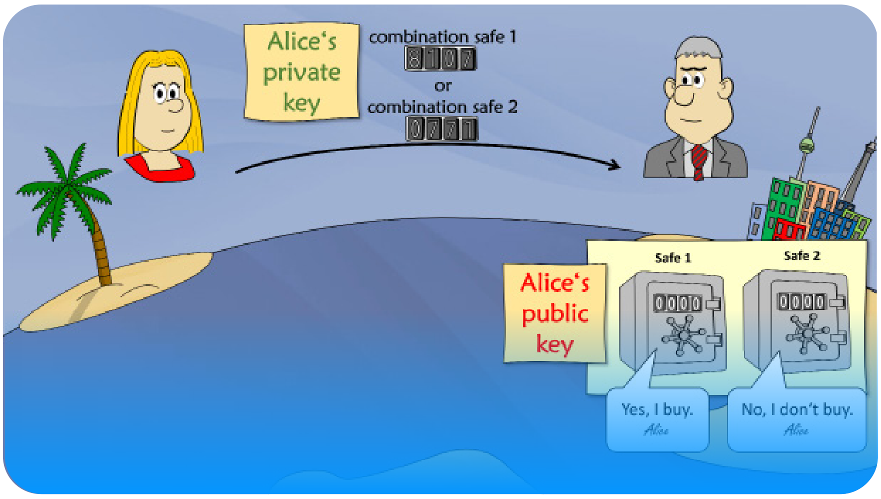


Figure 18: Atos' comic explanations have already been presented at numerous international events with great success.

Appendix 1: Further reading

Introduction to cryptography

Joachim von zur Gathen: *Crypto School*. Springer 2015

NIST competition

Official website: csrc.nist.gov/Projects/post-quantum-cryptography

Bruce Schneier: *NIST's Post-Quantum Cryptography Standards*. schneier.com/blog/archives/2022/08/nists-post-quantum-cryptography-standards.html

Lattice-based cryptography

Vinod Vaikuntanathan: *Lattices and Cryptography: A Match Made in Heaven*.
youtube.com/watch?v=5LGwaICJ5sw

Editor:
cv cryptovision (an Atos company) GmbH,
Veronica von Preysing

Source of supply:
cv cryptovision GmbH (an Atos company)
Munscheidstr. 14
45886 Gelsenkirchen, Germany

Published in fall 2022

Layout: studio ypsilon

Concept and editing: cv cryptovision GmbH (an Atos company)

Figures: cryptovision GmbH,
Bundesamt für Sicherheit in der Informationstechnik (BSI)

Note: This whitepaper is part of Atos' public relations efforts.
It is distributed free of charge and is not for sale.

www.cryptovision.com

Appendix 2: Post-quantum algorithm overview

Algorithm	Family	Use	Status	Advantage	Drawback
CRYSTALS-Kyber	Lattice	Encrypt	NIST winner	Secure	
CRYSTALS-Dilithium	Lattice	Sign	NIST winner	Secure	
FALCON	Lattice	Sign	NIST winner	Secure	
SPHINCS+	Hash	Sign	NIST winner	Very secure	Inconvenient
XMSS	Hash	Sign	RFC	Very secure	Inconvenient
Leighton-Micali	Hash	Sign	RFC	Very secure	Inconvenient
Classic McEliece	Code	Encrypt	NIST round 4, BSI	Secure	Long keys
HQC	Code	Encrypt	NIST round 4	Secure	
BIKE	Code	Encrypt	NIST round 4	Secure	
SIKE	Isogeny	Encrypt	NIST round 4, broken	Unsecure	Broken
FrodoKEM	Lattice	Encrypt	BSI	Secure	Inconvenient

cv cryptovision GmbH (an Atos company)
Munscheidstr. 14
D 45886 Gelsenkirchen

T: +49 209 16724-50
F: +49 209 16724-61