Anne Frühbis-Krüger, Xenia Bogomolec

Of Hiding Data in Analogue and Digital Times

Cryptography is the science -- or maybe the art -- of hiding sensitive information while allowing an exchange thereof in plain sight. Since ancient times, there have always been settings, where it was vital to exchange information without a third party overhearing or intercepting it: The roman historian Pausanias reports that in the fifth century BC, during the Peloponnes war, spartans passed on vital information encoded by means of a skytale. The idea behind this encryption is to wind a narrow, but long strip of parchment around a cylinder and then write on it in lines parallel to the axis of the cylinder. The message is sent in the form of the parchment without the cylinder. The recipient can only correctly decipher the message, if he possesses a cylinder of the same diameter and winds the parchment around it. In other words, all letters are correct and present in this encrypted message, while the key is the knowledge about the dimensions of the cylinder.

Jules Cesar, on the other hand, sent his secret messages by replacing each character by some other character based on an exchange rule known only to sender and recipient. Over time such methods were improved and refined, as were the methods to break such encryptions. For a long time, it seemed like a never ending race culminating in the invention of encryption hardware like the famous Enigma machine used until World War II. But the advent of computers took this to a completely different level: storage and transfer in digital form continuously increased not only the sheer amount of data, but also the range, speed and power of attacks to break encryptions. Nowadays, cryptography and cryptanalysis are very active research areas of mathematics and computer science and yet again we find ourselves at a game-changing moment in the development of new technologies: Quantum Computing.

Basics of Cryptography

Let us first look at the tasks and solutions of classical cryptography: The main challenges to meet are making sure that a message is not read or altered in its passage from a sender to a recipient. In the figure 1 this is illustrated by Alice sending a message to Bob and Fake-Bob overhearing it or Fake-Alice altering it respectively. This scenario is known as a man-in-the-middle attack. To avoid such breaches in the confidentiality of the communication channel, Alice and Bob agree on using a cryptosystem, that is a software which allows Alice to encrypt a message and Bob to decrypt it with reasonable effort, while it is practically unfeasible for Fake-Alice/Fake-Bob to put their plan into action. Here Alice and Bob could agree on the same secret, called a key, for encrypting and decrypting the message and make sure that it is not known to anybody else -- the most classical strategy, but with the huge drawback that they have to have agreed on the key or passed it on from one to the other at some point in the past. This latter problem is called a key-exchange problem and always arises in this setting of a symmetric cryptosystem, i.e. in the situation of using the same key by Alice and Bob. A well known example of a symmetric cryptosystem is DES (Digital Encryption Standard) introduced in the late 1970s in the USA, which served its purpose well, until the ever increasing speed and power of computers made brute force attacks by searching through the space of possible keys (of length 56 bit) feasible. The system became too vulnerable in the 1990s so that it was replaced by its successor AES (Advanced Encryption System) in 2000.

The wish to avoid the need for secretly exchanging a key made asymmetric cryptosystems popular, in which Alice and Bob do not use the same key. Instead Alice and Bob each create a pair of two keys, one of which is their secret (the so-called private key), while they make the other one publicly available (the so-called public key). For ensuring that Bob is the only one to decrypt the message, she uses his public key to encrypt it and he subsequently uses his private key to decrypt it. If Alice wants to make sure that Bob can decide the authenticity of her message, Alice uses her private key to sign the message so that Bob can verify it using her public key.

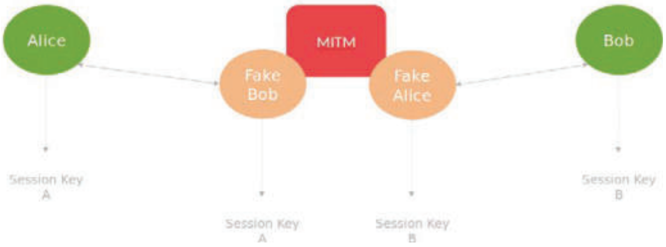

Figure 1

At the heart of any cryptosystem of this kind is a mathematical problem, which is known (or at least believed) to be hard to solve. For practical reasons, the problem should not require extremely large amounts of data. Classically, factorization of (large) integers and discrete logarithms are used as such problems. The first public key cryptosystem RSA (named after Rivest, Shamir and Adleman, who suggested it in the 1970s) depends on the factorization problem: it is known that finding the secret key can be reduced to the difficulty of factoring a

Prof. Dr. Anne Frühbis-Krüger
Carl von Ossietzky Universität Oldenburg
anne.fruehbis-krueger@uol.de

Xenia Bogomolec
Coding Services Hannover
xb@quant-x-sec.com

certain integer which has been fixed in the beginning. Increasing the size of chosen data effectively helped the system to withstand the evolution of computers for quite a while, but by now it has been replaced by newer systems.

The discrete logarithm, the other classical choice of a hard problem, needs a bit more explanation for non-mathematicians. From school, we recall that given two numbers a and b the logarithm of b with base a is the answer to the question $a^x = b$. The discrete logarithm problem is of a very similar nature, but does not necessarily deal with numbers. It should be formulated more generally with elements of certain groups (groups are sets of mathematical objects in which there is an operation · subject to some rather natural calculation rules). Given an element *g* from the group G, for which all other elements are of the form g^k for some integer *k*, the discrete logarithm problem seeks the smallest positive exponent *s* for a given *b* from G such that $b = g^s$. This problem is e. g. used in the El Gamal-Cryptosystem. In contrast to the factorization problem, the discrete logarithm allows a wider range of choices for the underlying data. In other words, it is not only possible to increase the size, but also to pass to a completely different kind of group, as has been done in elliptic curve cryptography, where the underlying group is the group of points of a fixed curve. Passing to more involved groups allowed to attain the same security level with smaller key sizes.

To sum up this brief discussion: The security level of any digital cryptographic algorithm reduces to the hardness of its underlying mathematical problem. If this underlying problem cannot be solved with available technologies within reasonable time, then the algorithm is considered secure. The involved secret, which is only allowed to be known to the communication parties, can be compared to a key for an unbreakable door to a room with the confidential information inside.

Gamechanger Quantum Computing

Technological advances always come with benefits and drawbacks. The declared aim of progress in computing is to make the computation of given problems less hard. Paradoxically, achieving this goal causes a fundamental drawback with global effect: It puts an expiration date on any cryptographic algorithm.

Considering this challenge, we want to name two main driving forces for the development of new technology:

1. We are confronted with a problem which we cannot solve. So we are driven to build a machine or a program which can solve this problem.
2. We have a problem which we can solve, but the solution is expensive. Then we are driven to find a more convenient solution.

An example for a product of an incentive of the first type is the predecessor of all our currently used binary computers: Collossus Mk 1. It was built by Alan Turing during the World War II in Bletchley Park in order to decrypt the German Enigma ciphers. The implemented decryption mechanism is a so called „brute force attack“. It is the least efficient attack type on cryptosystems: Every possible decryption key is tested once. The complexity of this attack equals the security level of the crypto algorithm itself: $2^{\text{key length in bits}}$. Colossus made this brute force attack feasible by automation. The 550 people who operated the machine would never have been able to decrypt as many messages by hand in the same time. The information from the decrypted messages is widely acknowledged to have shortened the war by many months, saving tens of thousands of lives.

The realization of Quantum Computing is a product of the second main driving force - at least initially. Exploitation of quantum mechanical phenomena for computing offers new solutions to problems which cannot even be solved by a supercomputer. It is the least advanced quantum technology besides Quantum Communication, Quantum Imaging, Quantum Metrology and Quantum Simulation. There are three known Quantum algorithms which will render our currently globally used cryptosystems weak and later even useless - as soon as Quantum processors with enough stable qubits will have been made a reality:

1. Shor‘s algorithm for integer factorization and discrete logarithm is an alternative solution to currently used binary algorithms. It applies to asymmetric cryptosystems which are primarily used for key exchanges and electronic signatures.
2. Grover‘s search algorithm can be used to perform database searches with lower complexity. Therefore, it also enables brute force attacks on symmetric cryptosystems of lower complexity than the algorithm itself. Symmetric cryptographic algorithms are used for static data encryption and for data transfer in hybrid crypto schemes (in combination with asymmetric algorithms).
3. The Quantum Algebraic Attack on Cryptosystems by Y.-A.Chen, X.-S.Gao, (https://arxiv.org/pdf/1712.06239.pdf) is the most recent paper on a quantum attack on cryptosystems which can be reduced to a so-called Boolean Multivariate Equation System. It potentially affects globally used symmetric and asymmetric cryptosystems and even some hashing algorithms.

Quantum processors with enough stable qubits to run the three above algorithms will have a global impact on science, industry and society. To perform a successful database search with Grover’s Search algorithm on AES we would need 2,953 stable qubits for a 128 bit key and 6,681 stable qubits for a 256 bit key. According to a recent analysis, a quantum computer with 4099 perfectly stable qubits could break the RSA-2048 encryption in 10 seconds[1]. “Perfectly stable” means that there are no additional qubits necessary for error correction. For example Google’s quantum processor Bristlecone has 72 qubits and an error rate of 0.6%, which results from the device specific coherence time (between 50-90 microseconds). Coherence time measures how long a qubit can survive its own quantum properties. Decoherence is caused by the interactions of a qubit with its environment, which cause disturbances and finally the superposition collapse. It defines the timeframe within which any calculation needs to finish. This timeframe can be extended by quantum error correction[2] realized by a multiple number of the originally needed qubits.

[1] https://www.quintessencelabs.com/blog/breaking-rsa-encryption-update-state-art/

[2] http://www-math.mit.edu/~shor/papers/good-codes.pdf

IBM has announced to create a 1000 qubit processor by 2023. Currently they are building a 400 qubit prototype. And D-Wave's quantum annealer Advantage runs with more than 5000 qubits. Quantum annealing is only applicable to certain classes of problems, such as combinatorial optimization problems and probalistic machine learning. Volkswagen optimizes traffic flow[3] with D-Wave processors. They solve problems which could not even be handled by a supercomputer. Quantum annealers are not considered a threat to classical cryptography. But they indicate how unpredictable technological evolvements can be. Peter Shor himself recently gave an interview in which he warned about the potential consequences of waiting too long with taking precautions in applied cryptography[4].

Figure 2

Researchers have been working on new cryptosystems which are resistant to these known quantum attacks. There are both physical and mathematical solutions:

Quantum Key Distribution (QKD) relies on physical quantum mechanics phenomena. Any eavesdropper who tries to measure the qubit stream, will alter it, and can subsequently bedetected. Furthermore, he cannot make perfect copies of the qubit stream. But still a man in the middle could measure the original qubit stream and re-initiate a new one for the originally intended message receiver - if there is no additional communication party authentication implemented.

Post-Quantum cryptography includes all cryptosystems based on mathematical problems which are resistant to the previously mentioned Quantum attacks. They can easily be integrated with our currently used hardware and systems and are therefore much less expensive. Furthermore, they can be used for authentication in dynamic Quantum Communication systems to avoid man in the middle attacks.

QKD networks have been established since 2002 (DARPA, SECOQC2 in Vienna, SwissQuantum in Geneva, etc.). The international quantum communication channel between China and the Institute for Quantum Optics and Quantum Information in Vienna launched by the QUESS1 space mission covered a ground distance of 4,700 miles enabling the first intercontinental secure quantum video call. Encryption keys are distributed via satellites. The challenge imposed by the instability of qubits is solved by network nodes and quantum repeaters.

Mathematical armament against quantum attacks on encrypted data has been under development for many years. The National Institute of Standards and Technology (NIST) is in the final Post-Quantum cryptography standardization round. Soon we will have Post-Quantum secure cryptographic algorithms in public crypto-libraries. But from industrial experience we know that migration to new technologies often takes a lot of time. Even if it only means to update a crypto-library and re-configure a system. The dependencies to other digital and business processes and compatibility with systems and legal requirements of communication partners often set long term blockers. To give a timeframe for such evolutions: The hashing algorithm SHA-1 is still active in critical infrastructures, 20 years after its official deprecation.

Anne Frühbis-Krüger studied Mathematics and Physics in Kaiserslautern, where she also did her PhD in Mathematics. After her Habilitation and shorter stays in Paris and Berlin, she moved to Leibniz Universität Hannover in 2007 and later to Oldenburg in 2019, where she now holds a full professorship. Her research interests cover a wide range of topics in the areas of symbolic computation, arithmetic and algebraic geometry and singularity theory. Currently she also serves the mathematical community as Chairperson of the Fachgruppe Computeralgebra of GI, DMV and GAMM.

Xenia Bogomolec is a Mathematician and Information Security Specialist with a background in Computeralgebra, system engineering, network programming and lawful telecommunication interception. Since 2016, she has been working as analyst and functional-technical coordinator with a strong regulatory focus in various IT and Information Security projects in the financial and biotech industry. Since 2017 she also works with various collaborators on post-quantum security analyses. This includes identification of post-quantum secure cryptographic algorithms wrt. known quantum attacks. Since autumn 2020, a small team of her recently founded company Quant-X Security & Coding is also working on quantum algorithm tests on quantum annealing processors.

[3] https://www.dwavesys.com/media-coverage/volkswagen-optimizes-traffic-flow-quantum-computers

[4] https://www.spektrum.de/news/peter-shor-erklaert-inwiefern-quanten-computer-gefaehrlich-werden-koennen/1795889?