Introduction to cybersecurity

> Note: *pdf version may not be up to date*

Concepts

Information security

> Other names: *data security, cyber security*

Information security consists of three independent branches:

- information availability
- information integrity
- information confidentiality

Each of these three branches must be maintained for all information (data).

Availability

Data availability ensures the timely and convenient access to data for all authorised entities.

Integrity

Data integrity ensures that the information comes from a legitimate source and hasn't been altered, deliberately or accidentally, after its creation.

Confidentiality

Data confidentiality ensures that data, or the information carried by the data, can be accessed only by authorised subjects.

Security of data

Security of data is achieved by securing the IT assets surrounding the data. These assets include

- IT equipment
- data communication channels
- software

and must take into account

- the organisation (structure and operating principles)
- personnel
- data carriers
- infrastructure

Standard model of harm

Threats influence the data and exploit the **vulnerabilities** of IT assets or components of the systems. Threats in conjunction with vulnerabilities determine and pose the **security risk**.

When a risk is not mitigated, a **security breach** may happen. Therefore, to minimise risks and avoid breaches, **safeguards** or **security measures** must be set in place.

Security and planned residual risk

No matter the amount of safeguards set in place, absolute security can never be achieved. Implementation of safeguards merely minimises the probability of a breach.

Acceptable residual risk

Instead of absolute security, the concept of acceptable residual risk is followed.

Acceptable residual risk means the situation where the total price of all implemented safeguards is approximately equal to the forecast of loss caused by a security breach.

Security threats

A threat represents a potential violation of security by an external element. It may consist of

- violation of availability
- violation of integrity
- violation of confidentiality

or any combination of those. A threat is always considered an external influence.

Classification

Threats can be classified by

- the target: availability, integrity, confidentiality
- the source: subject causing potential harm
- the type of IT asset
- the extent of potential danage

Generally the first two classifications are used.

Threats classified by source

Threats classified by source can be separated in two groups:

Spontaneous or accidental threats

such as environmental threats (Force Majeure), technical failures or human error,

Deliberate acts or attacks

in case the damage is intentional or the motives are clear.

It is worthy of note that in practice, accidental or spontaneous threats cause in general more harm than deliberate attacks, due to neglicence.

Attacks

Attacks or deliverate acts are always based on humans making deliberate actions to harm an entity's security.

Sources of attacks

- Authorised users of IT systems
- Intelligence agents
- Crackers
- Other criminal elements

Channels of attack

- Direct contact with the target system
- Networks
- Portable data carriers such as external memory devices

Attack classifications

- physical attacks
- mis-use of resources
- blocking of resources
- interception
- fabrication
- system manipulation
- attacks to security mechanisms
- attacks via software / malware

Physical attacks

Physical attacks mainly harm the integrity and availability.

- attacks on infrastructure
- vandalism
- trespassing / break-in
- theft
- manipulation or destruction of equipment

Mis-use of resources

Mis-use of resources harms all branches of security.

- unauthorised use of IT systems
- mis-use of user rights
- mis-use of administrator privileges
- theft of services (such as mobile services)

Resource misuse is susceptible to happen during the conversion, maintenance, repairing and/or upgrade tasks performed by external parties.

Blocking of resources

Blocking of resources mainly harms availability.

- overloading of network
- mass-execution of tasks
- filling quotas

The most common implementation of resource blocking are distributed denial of service (DDOS) attacks.

Interception

Interception or eavesdropping is an attack to confidentiality by any unauthorised subject.

- voice interception in rooms
- interception of calls
- unauthorised reading or copying of data
- reading of residual information (ex: from printers)
- wiretapping
- inappropriate deletion of data or destructuring with subsequent reading

Fabrication

Fabrication or faking consits in fake data inputs or propagation, which harms integrity.

- playback of recordings out of context
- masquerade attacks (phishing)
- social engineering
- denial of authorship

System manipulation

Manipulation consists in unauthorised changes made to an IT system, which mainly harms integrity. It has some overlaps with fabrication.

- manipulation of data or software
- manipulation of communication lines
- manipulation of data during transfers
- attacks via service ports

Attacks to security mechanisms

Attacks to security mechanisms can harm all three branches of information security.
The main attack targets are authentication systems and cryptosystems.

Attacks via software

- malware
- undocumented features of legitimate software
- special purpose software

Malware

- logical bombs
- trojan horses
- worms
- viruses
- droppers (trojan installers)

Vulnerabilities

Vulnerabilities are all properties of a system which can be exploited by threats. They can be divided in four main classes:

- infrastructure vulnerabilities
- IT vulnerabilities
- personal-related vulnerabilities
- organisational vulnerabilities

Infrastructure related vulnerabilities

- unfavourable physical locations of devices
- primitive or deprecated infrastructure solutions

IT related vulnerabilities

- limited resources
- improper installation of equipment or connection lines
- errors, defects and undocumented features of software
- shortcomings of protocols
- shortcoming of data management
- inconvenience of safeguards

Personnel related vulnerabilities

- incorrect procedures
- ignorance and lack of motivation
- ignoring of safeguards and protocol (intentional and negligence)

Organisational vulnerabilities

- deficiencies of work organisation
- shortcomings of resource management
- incomplete documentation
- incomplete implementation of safeguards
- shortcomings of safeguard management

Interaction between threats and vulnerabilities

Threats exploit vulnerabilities.

Security diminishes as the probability of threats increases and so does the number of exploitable vulnerabilities. The reciprocal stands as well.

Digital data

Securing digital data

Cryptography is essential for achieveing both confidentiality and integrity, which differs significantly from paper based data.

Authentication itself becomes essential and availability is often ensured by the network.

Role of cryptography

Encryption or cyphering is a technique where data is converted to a non human readable or device interpretable form. The conversion process uses a certain piece of data called a key to achieve the encryption.

Cryptography is used both for

- ensuring confidentiality: data cannot be deciphered without the key
- ensuring integrity: data cannot be changed without having access to the decryption key or leaving a permanent mark that data was altered

Availability of digital data

- regular backups
- reliable IT systems
- digital record management system
- transmission of data via networks

Integrity of digital data

- data and its carrier are permanently tied together (excludes network based applications)
- client-server techniques where interacrtions are logged
- usage of e-signatures to cryptographically associate data and its creator

Confidentiality of digital data

- secure storage and transport of plaintext data
- encryption of data and public handling

Encryption adds the **key management** problem. Additionally, encryption should be mandatory for any confidential information transfers over the common network.

Safegurads

Safeguards minimise vulnerabilities and the residual risk.

Classification of safeguards

- their purpose

- their security area
- the type of assset
- the means of implementation
- strength of security

Purpose divided safeguards

- preventive safeguards
- identifying safeguards
- reconstructive safeguards

Safeguards may also be polyfunctional.

Preventive safeguards

Preventive safeguards are used to prevent security incidents such as

- to minimise vulnerabilities
- to prevent attacks
- to minimise risk probabilities
- to decrease the potential damage of attacks
- the facilitate reconstruction

Preventive safeguards can further be divided into three subcategories

Reinforcable safeguards

which have the purpose of minimising the impact of spontaneous threats

- order or systematicness (standards, procedures, ...)
- working conditions (micro-climate, ergonomics, social climate, corporate hierarchy)
- preventive checks (verification and testing, monitoring, auditing)
- security awareness (training of employees, security drills, ...)

Deterring / scaring safeguards

which have the purpose of minimising the probability of attack attempts,

Separative safeguards

which fend off the attacks by taking care of different security aspects. They can further be divided into

- spatial isolation
- temporal isolation
- logical isolation (access control, service broker, securing)

Identifying safeguards

Operative identification

Involves methods which are able to identify security incidents as soon sa they occur and allow for immediate response.

Post-event identification

Is based on logging of events and later analysis of the logs for future improvement or proof and action taking.

Evidence-based identification

Is based on security elements integrated to IT assets which allows to check for integrity and/or confidentiality of data.

Reconstructive safeguards

After a security incident, it is necessary to restore the normal operation of impacted IT systems. This can be achieved via

- backups (data redundancy)
- renovation (removing of defects and errors)
- replacing (substitute for corrupted or damaged data)

Classification by IT assets

German BSI IT Baseline Security Method and Estonian ISKE standard distinguish the following categories:

- generic aspects
- infrastructure
- IT systems
- networks
- applications

Classification by implementation

Organisational safeguards

which include the security of administration, of system design, management and security incident handling activities and operations.

They should be implemented in the first order and beginning with security policies formulation, risk analysis and security plan creation.

They also should include four main components:

- activities a person must do
- activities which are prohibited for a certain person
- consequences of frobidden actions
- consequences of negligence

Physical safeguards

which involves the infrastructure, mechanical components, guards, etc...

IT-related safeguards

which are mainly used for performing logical separations and identifications of security incidents.

Two main branches include software-based access control and cryptographic means.

Risk management

The main goal of risk management is the exact implementation of a set of safeguards.

Symmetric cryptoalgorithms

Secret key cryptoalgorithms or symmetric cryptoalgorithms are methods where the same secret key is used both for ciphering and deciphering. They are considered practically secure if they satisfy both conditions:

- the key is at least 128bits long
- there aren't any known effectice cryptoanalytic methods

Role of symmetric keys

Keys should always be generated separately: they are not created by encryption or decryption algorithms. Key lengths are determines by the algorithms of key generation themselves. Finally, any bit sequence of predefined length can be considered as a cryptographic key.

> The minimal keylength should be at least 128bits (16bytes) to counter exhaustive search methods.

Fields of use of symmetric cryptoalgorithms

- Transmission of confidential information using unsafe networks
- Secure storage of confidential data
- Secure erasing of data (rewrites)
- Creation of secure key materials for cryptographic use

Major symmetric cryptoalgorithms

The DES key generation algorithm is considered insecure due to its keylength being only 56bits. 3DES or triple mode DES was used until 2005.

AES is still considered an excellent algorithm for mainstream use since it won the 2001 NIST commercial symmetric cryptoalgorithms competition. It is still a *de facto* standard and it is estimated that 80%-85% of symmetric cryptoalgorithm usage involves AES. Common keylengths are 128, 192 and 256bits.

IDEA (128bits) originated in the late 1080s in Switzerland. FOX and IDEA NXT were published in 2003 with keylengths ranging from 0 to 256bits.

Blowfish with a variable keylength of up to 448bits was made by Bruce Schneider in the 1990s.

RC4 is a stream cipher with a keylength between 40 and 256bits and was created in 1987. It is considered too weak for modern use however.

Block and stream ciphers

Symmetric cryptoalgorithms can be divided into block and stream ciphers, with the former being more widespread than the latter.

Block ciphers

Block ciphers are used in methods where plaintext is divided into blocks of certain length and these block are encrypted separately.

Popular modes include:

- electronic codebook mode: ECM
- cipher block chaining mode: CBC
- K-bit cipher feedback mode: CFB
- K-bit output feedback mode: OFB

Cipher and output feedback modes are used in situations where some form of feedback needs to be organised.

ECM

Plainext blocks are encrypted independently from each other using the same secret key. The disadvantage is that each ciphertext block depends on only one plaintext block, which can cause repeats in ciphertext.

CBC

Before encrypting subsequent blocks, the result of the previous block is XORed with the plaintext. As an advantage, one block of ciphertext depends on all previous plaintext: there will be no repeats in ciphertext.

CFB

The feedback loop involves both block ciphers and XORing.

OFB

The feedback loop involves only the cipher block which recursively originates from a certain value, using the initial key.

Stream ciphers

Stream ciphers are used in methods where a key sequence is generated from a given secret key, and a traditional XOR process is used to cipher all of the plaintext.

Usage of different block cipher modes

ECMs are the most convenient, albeit not secure enough modes to encrypt long plaintexts, therefore the most used method is the CBC mode.
Feedback modes are less frequently used but allow the usage of block ciphers as stream ciphers in order to produce key sequences, and are therefore mainly used for secure erasing of data.

Inner structure of block ciphers

Block ciphers usually involve numerous transformations of plaintext called **rounds**, where the output of a previous round is the input for the next one. The way different rounds use keys is determined by a **key sequence algorithm**. In case such an algorithm is missing, the original key gets used, however, if it does exist, the **round keys** are computed from the initial key.

Parameters of typical block ciphers

- length of a key
- length of a block
- number of rounds
- presence (or absence) of a key sequence algorithm
- number of round keys
- length of round keys

Main basic operations inside rounds

- substitution - replacing of original characters by other ones
- transposition / permutation - changing the order of characters

Most transformations within a block are comprised of a combination of the two basic operations.

AES

DES is a block cipher with a block length of 64bits and key length of 56bits. It was internationally standardised and made available form FIPS PUB 46-s, however, it was already weakening by the end of the 1990s due to the short keylength. A competition for a new standard was launched, and thus AES was created.

AES must be a block cipher with a block length of at least 128bits and having 3 different possible keylengths: 128, 192 and 256bits. In AES, they key length and the block length are the same. For a 128bit key, 10 rounds are involved, 12 rounds for 192bit keys and 14 rounds for 256bit keys. It is noteworthy that there is no key sequence algorithm used.

Each round consists of four subsequent different type of transformations:

- byte subs
- row shifts
- column mixes
- round key adding

Cryptanalysis of AES

An exhaustive search would need to perform from 2^128 to 2^256 operations, which is practically infeasible. Moreover, no effective cryptanalysis methods are currently known to break AES encryption, making it practically secure.

However, in 2002, the surfacing of algebraic cryptanalysis did offer a way to potentially break 128bit AES with only 2^87 operations. However, in practice, algebraic cryptanalysis has not been yet implemented. Moreover, **related key attacks** where different mathematically related keys are used to brute force the algorithm, have also been conceived in theory only. Lastly, a theoretical **side channel attack** that is based on getting internal information from within the block has been conceived, but again, not implemented in practice.

AES breaking machine

A *breaking machine* is a parallel computer that performes exhaustive searches where different key intervals are searched simultaneously with different chips. Such machine would break DES within one second, but would currently take thousands of millions of years to break AES. Moreover, such machines cost north of fifty thousand euros. Therefore, all three versions of AES are likely to remain practically secure for many more years.

Implementation of AES

AES can be realised both by software and hardware, however hardware computings are up to a few hundred times faster depending on the chips and programming languages used.

Asymmetric cryptoalgorithms

Asymmetric or public-key cryptoalgorithms make use of two keys: one key is used for encryption, and the other for decryption. The same key cannot be used for both tasks. These keys are generated using algorithms that matemathically link them together in a way that it is practically impossible to derive one from the other. These keys are called public and private key.

The public key is usually known for all parties involved in the exchange, and can be made available to the public. A private key should however only be known to the author or owner of a key-pair.

Asymmetric algorithms are often used for key-exchange purposes, as they allow to exchange symmetric keys in a secure and encrypted manner. They are also used to sensure integrity of data, and lay the foundation to digital signature technologies.

As opposed to symmetric keys, arbitrary bitstreams cannot be consered valid keys, and therefore a special key generation algorithm is always required. However, such algorithms induce some form of information redundancy, therefore causing the need for significantly longer keys than symmetric ones to ensure practical security.

RSA

RSA is the most widespread public-key cryptoalgorithm in the world today. It is considered practically secure for key-lengths no less than 2048 bits. For RSA, computing the public key from the private key is fairly easy, however it is practically infeasible to extract the private key from the public key. The security of RSA is guaranteed by the difficulty in factorising large numbers, solving the discrete logarithm problem.

RSA supports arbitrary keylengths, however the most widespread keys come in powers of 2 such as 2048, 4096, ... Keys less than 2048bit ones are however already considered to be weak.

Mathematical background of RSA

An algorithm is called of polynomial complexity if for a task of length *N*, the solution time is *N^k* proportional with some fixed integer *k*. Polynomial algorithms are generally considered to be good algorithms since as *N* increases, the solution time is not growing very fast.

Exponential complexity algorithms induce much worse solving times, as for a task of length *N*, the solution time is proportional to the value of *2^N*.

Non polynomial algorithms are considered practically unsolvable, which may either be good or bad, depending on usage case.

> Discrete logarithm problem: *find g, given a, n, p in a = g^n (mod p)*

Edmond's Postulate: an algorithm is considered to be good if its time complexity can be represented by a polynomial *O(n^k)* from an input, with *k* some integer. Polynomials are closed under addition and multiplication, in other words, the sum and/or product of polynomials is always a polynomial.

> Closure of polynomials: O(n^k) + O (n') = O(n^[max{k,l}]) | O(n^k) * O(n') = O(n^[k+l])

RSA key-pair generation

Two large primes *p* and *q* are generated. Their product, the RSA module, is generated: *n = p * q*. Then, *e* is chosen in a way that it is relatively prime to *(p-1)(q-1)*. Finally, *d* is chosen such that *d * e = 1 mod (p-1)(q-1)*.

The pair *(n, e)* is the public key, and the triple *(p, q, d)* is the private key.

RSA ciphering

It is possible to encipher texts that are less than *pq* bits. The enciphering process includes a discrete exponent: *Y = Cip(X) = X^d mod n*. Likewise for deciphering: *X = Decip(Y) = Y^e mod n* since *(X^d)^e = X mod n*.

Main concepts of RSA

Without having a private key, a plaintext cannot be encrypted in a way that it is decipherable using a public key. Furthermore, a message encrypted with a public key cannot be decrypted using the public key again.

Conceptually, *e* is a public exponent and *d* is a secret exponent. Functions for which an inverse cannot feasibly be found are called *one-way functions*. If however, the inverse can be found with some minimal additional information, the function is called *trapdoor one way function*. RSA is an example of a trapdoor one-way function.

Cryptanalysis of RSA

Factorisation of 70 digit numbers take only a few minutes for average personal computers. In general, 100 digit numbers can still be factored at home in less than a day.

In 1996, the factorisation of a 140 digit number took 5 years and required a joint computational effort of many computers. To date, the largest factored number (2009) was 232 digits long.

A 300 digit number (1024bit RSA) would take millions of years using current classical computing methods and hardware.

Other asymmetric algorithms

- elliptic curve based algorithms (P-384 or ED25519)
- El-Gamal

- DSS
- Paillier' system

Hash functions and cryptoprotocols

RSA, due to its relative slowness, is unsuitable for integrity purposes on its own (1000x slower compared to symmetric ones). Instead, cryptographic hashes are used.

A cryptographic hash or cryptographic message digest, but also called fingerprint or thumbprint is a digest with a fixed length which is computed from an arbitrary-length message using a one-way function.

In usage, if for a given message-hash pair, the hash corresponds to the one generated from the message (using the same known algorithm), we can be sure (with great probability) that the hash has originated from that message, and from nowhere else.

Old-type hashes

Old-type cryptographic hashes were computed using a commpression function, which is one way and guarantees a fixed-length output from a longer fixed length input. It was iteratively used on data blocks.

Mandatory properties of hash functions

- any change, no matter how minor, must cause a full change of the hash
- hashes must be easily computable
- hash functions must be one way
- hashes must be resistant to second preimage attacks
- hashes must be collision-free (statistically)
- compression functions used must also be collision free (pseudo-collision free)

Birthday paradox

The probability that for *N* people the birthdays of two different people coincide will grow proportionally to *N^2*.

As a conclusion, if the output of a hash is *N* bit long, then the probability that *K* trials will give two identical hashes is *K = 1.17 * 2^(N/2)*.

The simplest cryptanalytic attack (exhaustive search for hash functions) of a *N*-bit hash function needs to iterate over *2^(N/2)* options.

Secure hash functions

- SHA-2 (SHA-256): constructed on the basis of MD4 in 2001, and of a length 256bits (32bytes)
- SHA-3 (256 / 32, 384 / 48, 512bits / 64bytes): constructed in 2010 to resist newer hash attacks

Insecure hash functions

- MD family: hash lengths of 128bits (16bytes) are vulnerable to collisions and exploits
- SHA-1: hash length of 160bits (20bytes) isn't completely broken yet, but is deemed too weak for practical use

MD5

Consists of 4 different rounds which process a message by 512bit portions. During each round, the result of the previous round is mixed with the following 512bits.

In 2005, signatures based on MD5 were reliably practically broken. In 2017, collisions can be found by exhaustive search with the amount of *2^24,1* within a minute.

SHA-1

A structurally similar algorithm to MD5 however with a longer length. Collisions can be easily found in theory, however the collisions do not allow to reverse the hash itself. In 2012, it was estimated that a hash breaking would cost $1mln. The first actual collision was found in 2017.

SHA-1 is allowed for usage in emergency situations in high stakes protocols with the following clauses:

- key-strengthening mode: the usage of the function twice in a row
- salting: appending of a random bitstream to the source, which protects against dictionary attacks

SHA-256 and SHA-3

The SHA-256, based on MD5 structurally, is the current de facto standard in commercial cryptography.

In 2006, SHA-3 was created for the NIST hash competition. The runner-up to Keccak (SHA-3) was BLAKE. Later, BLAKE2 was developed, it being considerably faster than all previous hash functions.

Sponge structure of SHA-3

A sponge structure is a two-step action:

- absorbing: a large array of which the initial part is changed step by step with bits of material to be hashed with a compression function
- squeezing: conversions are made within the array with the compresssion function and the hash is "read out" between conversions.

Message authentication codes

Message authentication codes or MACs are hash functions with a key assigned: computing and verifying of a hash needs a secret key on top of the message. It differs from public-key cryptoalgorithms by the fact that both the computing and verifying process can be performed by the same key.

Cryptoprotocols

A protocol determines which information moves between subjects and who, how, when transforms it.

A cryptoprotocol is a protocol where the transformations include different cryptoalgorithms and/or key geneartions.

The most widespread cryptoprotocol (on the Internet) is TLS, Transport Layer Security.

TLS

TLS is constructed to be used in conjunction with TCP/IP protocols. It enables authentication of parties in the network. It is also included in higher level protocols, enabling:

- ssh instead of telnet
- https instead of http
- ftps instead of ftp

TLS channel

TLS establishes a secure channel over a network, having the following properties:

- the channel is private: all transferred data is encrypted after the initial key exchange
- the channel is authenticated
- the successful receiving of all packages can be checked

Main principles of TLS

TLS connections are comprised of two phases:

- handshaking phase
- message transfering phase

The connection must be established between two unequal parties: client and server. The server authentication is mandatory, while authentication of a client is voluntary.

TLS handshake

The client notified the server of their wish to connect, and provides information about the available cryptoalgorithms. It sends a generated **nonse** to the server and demands the server authenticates itself.

The server then generates a message claiming its origin, appends to it the nonse, hashes the package and signs the hash. It then sends to the client its certificate (public key), the message and the signed hash.
The client can verify the signature to identify the server, and then store the server's public key. The client has now authenticated the server.

The client now encrypts a symmetric key with the server's public key and sends it to the server. The server deciphers this symmetric key with its private key, and the shared key can be used to exchange further data.

TLS security

TLS is a successor to SSL (secure socket layer) and improves many of its flaws. However, it still has weaknesses.

The major concern is the impresonation of the server by a malicious agent. The handshake can still be carried out, and the client will assume a legitimate connection was established. This is why certificates need to be used on top of TLS.

Other cryptoprotocols

- DNSSEC: Domain Name Security Extensions
- IEEE 802.11: wireless local area protocol
- IPSec

- S/MIME: secure MIME
- SSH: secure shell

Electronic signatures

Time stamps

A time stamp is a data set that is added to some other original data. They are issued by time-stamp authorities and give the possibility to provably compare the creation time of different datasets.

The timestamps are calculated from the hash of the data and the previously issued timestamp.

Validity of approval

The validity of approval is established by a query to a certification authority, which is made immediately after giving a signature. A digital proof is then also added to a document that the signature was valid at the time of signing.

Such embedded proofs guarantee verification capabilities even in case online connections cannot be established, however, a valid signing process will always require an online connection.

Certification infrastructure

Public key infrastructure consists of the following mandatory components:

- non reverse-engineerable hardware-based public-key container
- certification authority (CA)
- validity of approval service (at CA)
- time-stamping authority
- organisation and coordination of services

Data format

As digital signature are bound to bitstreams, digital documents that are signed should carry meaning. Therefore, for signing, only data formats carrying unique meaning and with a publicly available description should be used, unless for a specialised application.

EU electronic signature regulations

EU Regulation 910/2014 "On electronic identification and trust services for electronic transactions in the internal market" (eIDAS) is in place.