Tampere university / ITC / Computing sciences / Jukka Koskinen / Comp.Sec.100-02

Maps and lists for Cyber Security 1

Version 2, 19.10.2020

This text was written to support learning, but its intention is not to provide direct solutions to problems that appear in the materials of exams or exercises at http://sec.cs.tut.fi/maso/ and https://sec.cs.tut.fi/harpo/. The latter requires registration that is available only for students enrolled in this course. And the former shows that already Exam 1 includes more topics than the current document. On the other hand, some details in this document go beyond the course requirements.

Contents

Course objectives

The concise definition of objectives in the official syllabus is reproduced here in italics. More detailed explanations follow each line.

The aim is to learn basic skills about cyber security, needed by everyone who is studying information technology.

This objective must be understood in a wide sense. The studies themselves impose a need of some cyber security skills, but the main motivation comes from the professions where the students of information technology will end up. In addition, this covers skills needed in society and not only as citizens but also as technology-aware counselors of fellow-citizens who are not likely to be able to take sufficient care of their own security. Lastly, the student will also be prepared for further studies in the infosec curriculum.

The student

• *identifies security and privacy threats and responsibilities;*

This is an obvious but not simple starting point of anything you do in the field of information security, and privacy can be considered just a branch of it. You must understand what bad can happen even if everything seems to be fine. In addition, in your work and to some extent in your private life it is not only good but obligatory to act properly in the face of security threats.

• *has a wide knowledge of the concepts, principles and mechanisms of information security;*

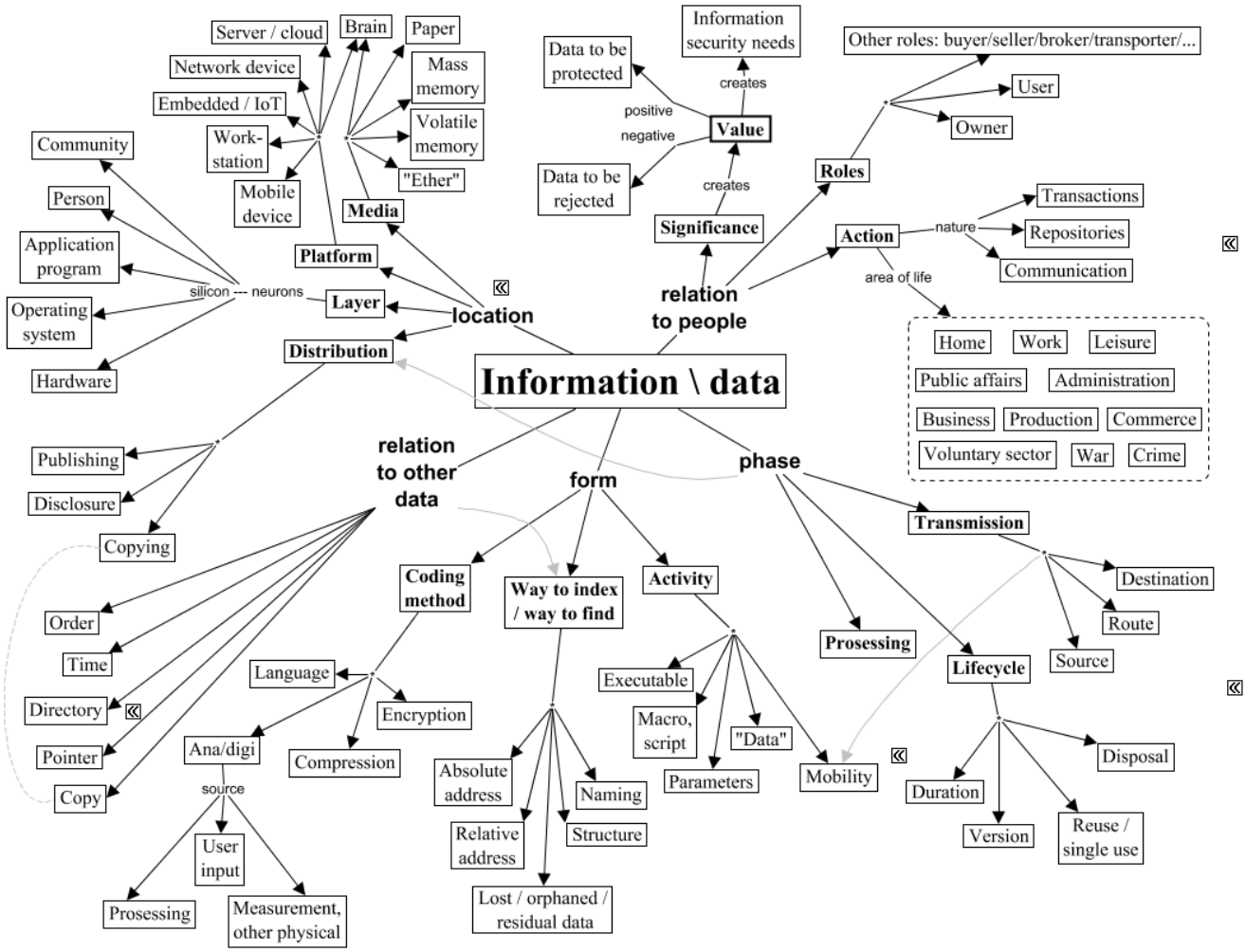
This course covers a very large spectrum of them. After passing this course you will not encounter very many completely new things even if you continue your studies in information security. There will always appear new kinds of vulnerabilities and attacks, but this is one of the facts that you also know

and. Most likely you will be able to categorize all new things in the spectrum. It is of course clear that you cannot obtain very deep knowledge over very wide range of things during a short course. This is why there is the third objective.

• *knows what kind of additional knowledge and skills is needed to perform various information security tasks in different application areas.*
This is a very important goal of a basic course, but it is difficult to measure. Your way of learning this will mainly be getting familiar with small parts of large documents, web sites that you found and judged yourself, and the constant flow of news during the process of doing an exercise on them. Thereby you are likely to start appreciating these sources of knowledge and understanding the depth of skills that is required in information security professions.

From information to information security

The concept map below presents a diversity of aspects and attributes that data can have. You may find more by asking how data becomes information, and what makes it knowledge is, or more simply, what is/are data like, where does it come from, where is it, how is it changed, where does it go, why, ...?

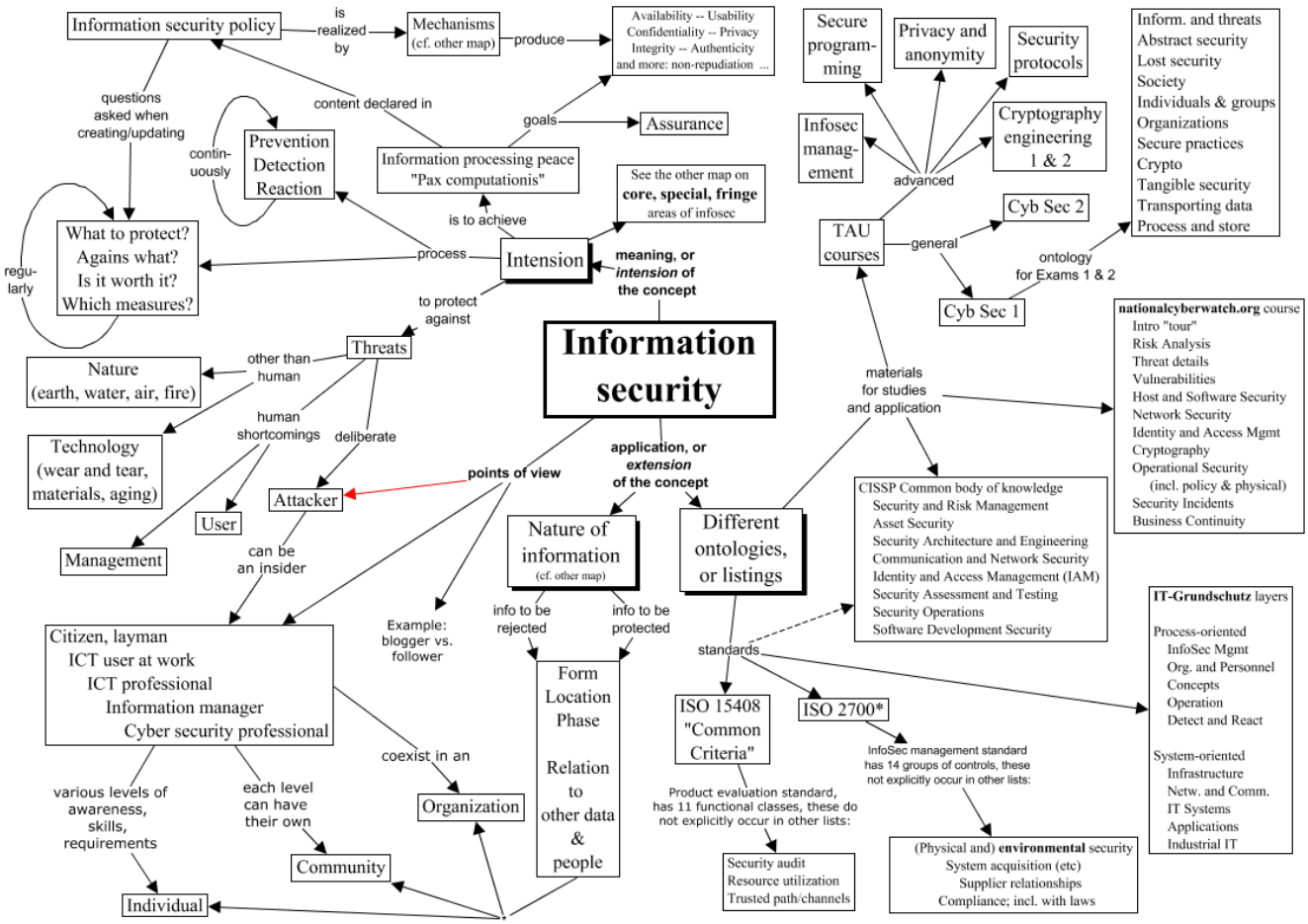


From each item in the concept map or from your own questions you can find a security perspective by asking, what can *go wrong*, or what *should not* happen. Of course, the question, what *should* happen, also raises security issues, but many of them are often perceived as the overall functioning of information systems.

Below is a concept map that characterizes information security in terms of *what* it is and *where* it is. The what is the intension and the where the extension of the concept. The latter topic refers to the map above and to various comprehensive administrative or educational breakdowns of information security. The ontology of Cyber Security 1 can be found below in a subsequent section. And further on, you will see ISO 27001 and the Common Criteria.

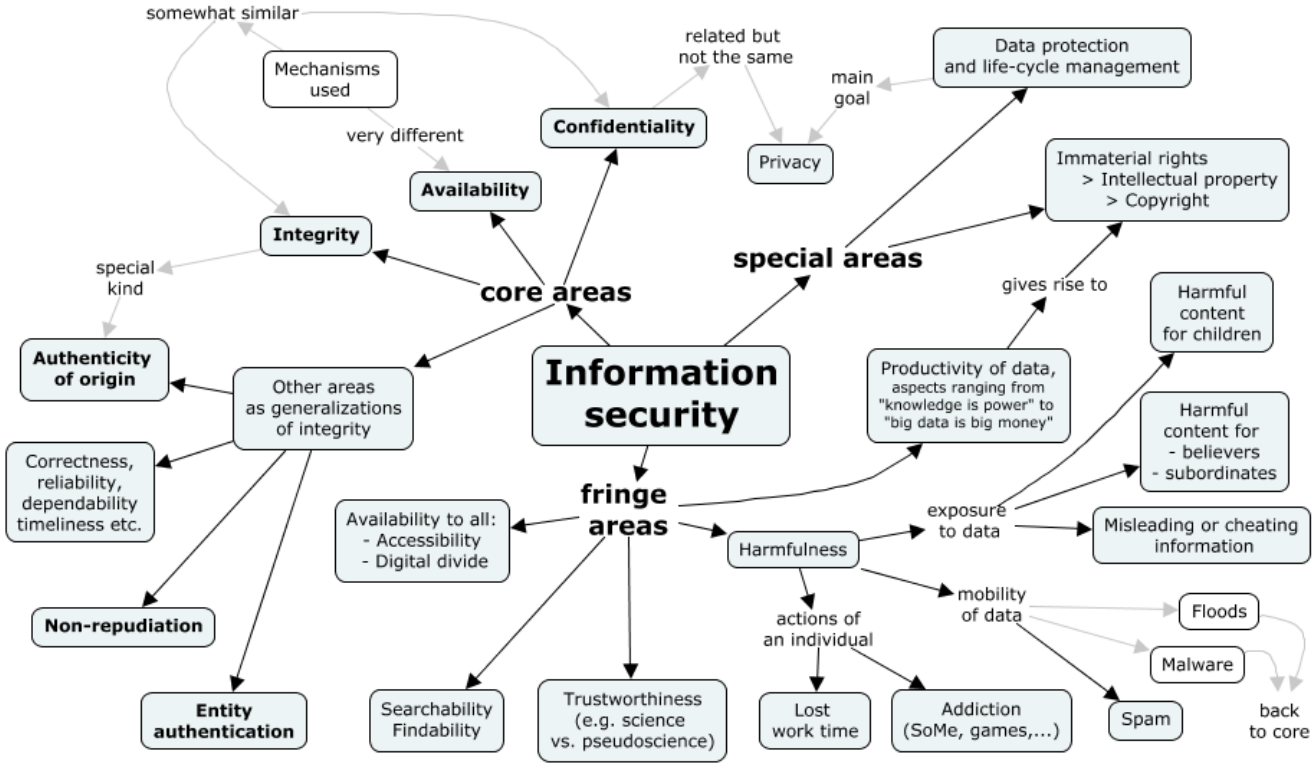
CISSP stands for certified information system security professionals. You may have seen some multiple choice question for that certification in a Harpo exercise. It is so well-known that it is partly linked to "standards". Another wide initiative in the USA is National CyberWatch Center. It is government-funded consortium of educational organizations.

IT-Grundschutz is a German standard, also available in English. It has two volumes explaining the management aspects in somewhat similar way as ISO 27001, and volume 3 for risk analysis. The security mechanisms and application areas can be found in the Compendium (final draft, 2019). It has 855 pages but you can fairly easily pick the things you are most interesting in. For instance, search the word Introduction and you can see definition of all sorts of vistas of information processing. If you search for the term "Basic Requirements" you will have a good supply of protection knowledge for each case. "Standard Requirements" strive for higher security and there are also provisions for "increased needs". Basic requirements are likely to be enough for your Essays and Exam 3.



Content areas of information security

The next concept map shows that there is much more to information security than C, I and A.



Content areas of the course, the ontology

The course contents follow an ontology of information security developed for this purpose. There are 3 major categories subdivided into 11 classes. This division is used to distribute the questions for Exams 1 and 2. The indented paragraphs below describe the classes more widely than the exam questions ask. For instance, there is nothing about p2p, voting systems or diffuse radiation in the Exams 1 and 2, but you can invent an essay topic for yourself out of such fields.

(1-3) What can happen if there is a lack of security, what can then be done, and how can everything be understood?

1. Information security defined by looking at information and threats to it.
What needs to be protected and why, what can go wrong, what or who is threatening, how to survey the threats, what is a threat model? (Detailed threats appear in later classes – especially such that are against protections.)

> Nature and vistas of information (incl. everything from human memory and hardware tokens to cloud services and internet as a whole). General definitions of information security (content of the concept: data processing peace, "CIA", etc.), threatening natural forces, technical problems, human error, administrative failures, intentional threats (from hacker ethics to malware economy to cyber warfare).

2. General features of security activities.
Most of the topics in this class can be understood without getting familiar with details in other classes, but a new level of understanding will follow that.

> General principles (also “design principles”), perspectives, ontologies, architectural models, organizations, standards, information sources, measurement, testing, assurance, theories, research, security industry, professions, education.

3. Addressing emerging threats

What can be done if a threat is realized? The better prepared you are, the more you can do. However, only in the later classes will the emphasis be proactive in the sense of trying to control what can happen. This class is positioned here to provide a "final frontier" in facing the threats, but many advance preparations are covered here.

> Response preparation (planning, procurement etc.), logging, duplication (including backup), detection (including antivirus and IDS), response, recovery, continuity, tracking (forensics) and follow-up.

(4-6) How are people situated in the field of security?

4. Security features of society

Societies, nations, economies and other large networks of people have structures, trends and missions that make some threats more possible than others.

> Cyber security (also from a corporate perspective), strategies, regulations (including cybercrime, copyrights, data protection), information society, cybercrime ecosystems.

5. Community and individual security aspects

People in their daily life and as members of local or network communities constantly face situations where information security comes to play.

> Infosec culture, protection from harmful information and scams, privacy in practice, usability, awareness, ethics and other human factors, plus identity and trust management.

6. How are security measures selected and managed within an organization?

This class covers administrative information security at the level where employees, plans, euros, etc. are discussed, but not usually bits, volts, or TCP ports (which are the topics in classes 11, 9 and 10 respectively).

> Strategies, policies, infosec management system, risk management, personnel, auditing.

(7-11) How do technologies help in the field of security?

7. What is the secure way to access and operate information and systems?

This class differs most from the traditional ones by putting together operational security, data security and other practices in information systems. This is the result of considering the title question in the case of normal use of information systems, where users and administrators must apply security-enhancing procedures. Some of them are quite independent security mechanisms, such as passwords or encryption, and they cover some topics that would otherwise fall into later classes, mainly cryptology.

> Access control, use of passwords, rules for different stages in the life cycle of information (including when to encrypt / sign e-mail), special systems such as healthcare, governmental systems.

8. Cryptologic methods

This class covers almost all cryptologic theory, but not any more practices like key management (in class 6) or usage of crypto (in class 7).

> Algorithms, implementations, breaking, protocols (including various forms of authentication), special systems like voting.

9. Physical and hardware security aspects

This class encompasses everything that is "tangible" except for people, though they are included in the biological sense.

Access control, facilities, power supply, anti-tampering, biometrics, security devices, trusted architectures, critical systems and industrial automation, diffuse radiation.

10. Securing the data network and mobile data

The data network is the basis of all kinds of data processing nowadays. This class applies in practice the theoretical knowledge from the classes above, especially from cryptology.

Secure structure (including hardware) and management of a network that is wired / wireless and of type enterprise / backbone / ad hoc; filtering; security protocols in theory and practice (e.g., IPsec and TLS); special systems like p2p, digital TV, WLAN, Bluetooth, 4G, RFID. Vulnerability testing.

11. Securing "stationary" data processing

Although the movement of data has already been covered in class 10, most phenomena in this class still move data in networks. Here, motion is less essential in implementing security than in class 10.

Databases, software, operating system, programming, embedded systems.

List of concepts and acronyms

The MCQs for Exam 1 have been made around these 90 concepts. The words here are links to Wikipedia, usually with some modification to the term. Some remarks are given in footnotes

2FA
Access control
Accountability
AES
Anonymity
Attack vector
Authentication
Availability
Backdoor
Backup
Block cipher
Botnet
Buffer overflow
CAPTCHA
Certificate authority
Challenge-response
Checksum
Copyright
Credentials
Cryptoprimitive
Cryptoprotocol
Cryptoalgorithm[1]
Cryptography
Cyber attack
Data protection
Defence in depth

Denial of service
DES
Dictionary attack
Distributed denial of service
DMZ
DRM
Eavesdropper
Encryption
Entropy
Exposure[2]
Firewall
GDPR
Hash
Hoax
Identity theft
Impersonation
Integrity
Intellectual property
IPsec
KEX
Key
Log in
Malware

Man in the middle
Non-disclosure agreement, NDA
Non-repudiation
One-time pad
Overwriting
Packet filter
Passive attack
Password
Password salt
PCI-DSS
Phishing
PKI
Plaintext
Policy [3]
Privacy
Pseudonym
Public – private key
Ransomware
Risk [3]
Risk analysis [3]
Risk management [3]
Sandbox
Script kiddie
Security control, mechanism, ...
Separation of duties
Shared secret

Signature
Single-sign-on
Social engineering
Spam
SSH
Steganography
Stream cipher
TLS
TOR
Trojan horse
virus
VPN
WPA
X.509 certificate
Zero-day

[1] Learn the concept from examples.

[2] This is a link to Wiktionary. Make the distinction to e.g. *disclosure* using ordinary meanings.

[3] Prefer the NIST-Intro (below)

Here is a list of names and acronyms appearing in both Exam 1 and 2, partly repeating the above list, but fairly complete in that matter..cmd

.com	CBC	DVV	IMEI	MFI	PIN	SIEM	UDP
.pif	CBC-MAC	ECB	IMSI	Microsoft	PKI	Siirto	UN
.scr	CPU	ESP	IoT	MobilePay	PPP	SIM	Unix
.tar	CRL	ESP	IP	NFC	PUK	SMS	URL
.txt	CSRF	EU	IPR	Nigeria	Rabin	Smurf	USB
A5	ctrl-Z	Excel	IPsec	NTP	RAM	SQL	USIM
ACID	CVE	GDPR	IPv6	OS	RBAC	SSH	VLAN
AES	DAC	HLR	ISO	OS	RC4	suid	VLR
AH	DDoS	HMAC	Java	OSI	RDF	SYN	VPN
AI	DEP	HR	JPEG	Oulu	ROM	TCB	Wifi
Amazon	DES	HTML	Kerberos	OWASP	RSA	TCP	Wikipedia
API	DH	HTTP	Kerckhoffs	PaaS	SaaS	TCP/IP	WLAN
ARP	DLL	HVAC	LAN	PayPal	SCADA	TLS	WPA
ASCII	DMZ	IaaS	Linux	PC	sgid	TOCTOU	X.509
Blowfish	DNS	IAM	MAC	PCI-DSS	SHA	TOR	XOR
Bluetooth	DNSSEC	IEEE	MAC	PGP	SHA1	Traficom	XSS
CA	DoS	IKE	MD5	PICS	SHA-256	Trojan	
CAPTCHA	DRM	IMAP	MDC	PIN	SIEM	Turing	

NIST-Intro

For the student to get a concise and holistic view of the course topics, *NIST Special Publication 800-12, Revision 1*, (2017) is recommended[4]. It bears the title *An Introduction to Information Security*. It suffices to study the main body of the document, pages 7–70 plus section 1.4. There are several topics that will not appear in the exams. The rest of this section gives advice on reading this document, the *NIST-Intro*.

The NIST-Intro is a brief text written in a comprehensive handbook style. The course is not dependent on it or even its vocabulary. For instance, the course does not use the terms adversary and adversarial, but instead attacker and malicious. More notably, the course does not include authenticity and non-repudiation under the concept of *integrity* (cf. sections 1.4, 10.20 and Glossary). The term *authorization* is used in the course mainly in the narrow meaning of granting access rights, while the NIST-Intro deploys the term in more complicated managerial contexts (cf. 6.5 and 7.1).

From chapter 1 only section *1.4 Important Terminology* is needed now, but it is good to be aware of the U.S. context given in the rest of chapter 1. Similar legislation and documents exist in other countries. The rest of the NIST-Intro makes frequent referrals to the documents outlined in chapter 1, mainly from the NIST or FIPS[5] publications. They are of course not the topic of this course.

The roles described in chapter 3 are not to be taken too "seriously" (i.e. learned verbatim), but each of them shows something that needs to be done with respect to information security. This also includes the supporting roles in section 3.17.

Note that the insider threat is much more than explained in the example of 4.1.2. It is almost common sense that it is more often related to some gain to the insider than just retaliation through sabotage when being terminated. Section 10.16 Personnel Security helps complementing the view.

Chapter 5 Information Security Policy is administrative but very important also for an engineer.

A sidestep to ISO 270**

It is worth noting here that ISO has published a whole set of standards concerning various administrative aspects, together specifying an *information security management system*, ISMS, for an organization. The standard series spans 19 documents in the numbering range 27000–27021 whereby it is called the ISO 27000 series. It is outlined in the freely available introduction ISO 27000 (Overview and vocabulary). You must learn from what you are reading just now that **ISO 27001** is the

[4] NIST=National Institute of Standards and Technology (USA)
[5] Federal Information Processing Standard (USA).

most important among these, because organizations can (and sometimes must) get their ISMS certified against it. That is, they get audited, pay, and are given a "badge": 'We are managing our IS well enough.'

This sidestep is worthy also because the NIST-Intro does not mention ISO 27000 series at all. This little fact is worth learning also as such because it shows how different standard bodies and different nations can see things differently and do things in their own way. The ISO 27000 document does not, of course, refer to any NIST publication. As a final detail the vocabulary in it defines *integrity* simply as "property of accuracy and completeness", i.e. without requiring authenticity and non-repudiation.

Back to NIST-Intro

The brief introduction to the NIST *Risk Management Framework* in chapter 6 is good to read, but as a framework (among many others) it will make more sense on later courses that delve deeper into risk management.

Chapter 7 *Assurance* provides more insight into its topic than is required in the exams. Assurance does not appear on the course in one place but dispersed into various places and usually without being named but just linked to issues that need assurance. (Note also that *assurance* does not appear as a term in the ISO 27000 vocabulary.)

The title of chapter 8, *Security Considerations in System Support and Operations* reminds concisely that there are many activities needed to run an information system and most of these activities are not focused on security. And still security needs and opportunities must be considered in everything. The needs are easy to understand for instance in software support or media control (sections 8.2 and 8.5), but spotting opportunities for improvement may require more insight. Examples are shown for instance in sections 8.1 and 8.6 on user support and documentation.

Users are central in several sections in chapter 10 that describes categories of security controls. Section 10.8 *Individual Participation* may seem surprising as a security mechanism or control, but this may not be so if you come from the U.S. or somewhere else where GDPR is not in power; General Data Protection Regulation of the European Union, since 2018, has 99 articles dealing with many aspects of "PII", personally identifying information.

There are considerably more technical details on cryptography on the course, especially in Harpo, than in the NIST-Intro, but chapter 9 gives a very good context for the techniques. Together with a certain amount of details such contextual understanding is likely to become of lasting value for most professions in cyber security. And at the infosec program at TAU it is definitely useful, because the advanced courses have a strong emphasis in cryptography engineering.

Chapter 10 *Control Families* gives a breakdown of security controls into 20 families. As you become more familiar with information security, and partly on this course already, you may find this division interesting by comparing it to other points of view. One influential viewpoint is the Common Criteria standard that firstly divides security into functions and assurance. The functions (given as *functional requirements* in part 2 of the CC) mainly correspond to what is meant by controls in the NIST-Intro. There are 66 families of functional requirements in the CC standard. So, it is better for you to stick to the NIST-Intro. But you must know the existence of CC and its purpose of facilitating assessment of information security – and its idea of doing this by checking various *assurance* requirements to know that the security *functions* (required by the type of system) are effective. The same idea is of course present in the NIST-Intro.

Pay attention to the cost considerations at the end of various chapters. If you enter the profession you need to cause expenses that your employer might think are not justified because they do not bring any income.

Cyber

It is time for a final word on NIST-Intro, ISO 27000, and CC, the three documents treated above. The word is *cyber*. It does not appear at all in the latter two (except in a name of an institute in CC). In the

NIST-Intro it appears 8 times on pages 7–70. Only one of the occurrences is *cyber security*, the name of this course, and even that mention comes from a referred document. The other mentions are mainly about cyber attacks and criminals. The name of the previous versions of this course was still *information security* in 2018. The terminological change is anticipated in the Glossary of NIST-Intro on page 79: "Note: DoDI 8500.01[6] has transitioned from the term information assurance (IA) to the term cybersecurity. This could potentially impact IA related terms."

Why is this course called cyber security? It is trendy, one could say, but it is true that no information security professional should any more ignore the security landscape outside his or her own organization. Any attack can also be part of something more serious that affects or is outright targeted at the critical infrastructure. This matter is not emphasized in the course contents which is rather traditional "InfoSec". You may wish to take up this sort of topic as one of your Exam 3 essays.

To mitigate the lack of *cyber* in Exams 1 and 2, a quote follows introducing the critical sectors and the way forward with learning "real" cybersecurity, including ENISA's role in it. ENISA is the European Union Agency for Network and Information Security. It has conducted a study, Stock taking of information security training needs in critical sectors (Dec 2017). This is the executive summary of that document:

> The European Union's Directive on security of network and information systems (NIS Directive[7]) asserts that "*network and information systems and services play a vital role in society*", and that the "*magnitude, frequency and impact of security incidents are increasing, and represent a major threat*". Given that urgency, the NIS Directive goes on to argue that "operators of essential services" need to identify "which services have to be considered as essential for the maintenance of critical societal and economic activities". This is in fact referring to the operators in the so-called **critical sectors** […]:
>
> - **energy,**
> - **transport,**
> - **banking,**
> - **financial market infrastructures,**
> - **health sector,**
> - **drinking water supply and distribution, and**
> - **digital infrastructure.**
>
> The protection of these seven critical sectors should have the highest priority, because when they are under threat, the functioning of society itself and the well-being of its citizens are at stake. As part of this effort, it is extremely important to increase the competences of cyber security personnel. This requires the availability of high quality trainings across the board, available to all critical sectors.
>
> Within the critical sectors, there are significant differences regarding the maturity level of cyber security. Therefore, some of the critical infrastructure operators will not be as ready as others, to counter the risks resulting from new cyber security threats in a timely and adequate manner.
>
> With the emphasis that the NIS Directive places on the importance of the seven critical sectors, this study aims to identify the current situation in these sectors in regard to the available cyber security trainings, and if there are any training needs specific to each of the sectors, beyond the generic needs for such trainings.
>
> Over the past years, ENISA has developed a wide range of cyber security trainings, and also delivered the training content to several national and governmental CSIRTs (Computer Security Incident Response Teams) as well as their constituents. The next important question that this study set out to answer is if and how the ENISA training portfolio actually is useful for the seven critical sectors – and what could be done to improve the suitability of that portfolio to the existing training needs.

[6] The DoDI reference is to a Department of Defense Instruction from 2014.
[7] Adopted 2016. In 2020, this official ENISA page on NIS refers *in future tense* to the national actions due in 2018.

The main general conclusions are:

- the cyber security training field is extensive and diversified, but does not sufficiently address the issue of raising the cyber security resilience of critical infrastructure: CIP-related trainings are still a niche
- there is a shortage of specialised trainings in the field of ICS/SCADA[8] systems cyber security – which is an essential element in countering operational threats (e.g. in the energy sector)
- there are very few trainings specialising in the specific threats encountered in the different (sub)sectors
- cyber security awareness raising trainings are lagging behind
- there is a shortage of trainings in regard to decision making as a result of data leakages or privacy incidents
- there is a pressing need for trainings related to GDPR, since this will affect every sector, and could have an operational impact on the organization.

Information security is a process

Information security is not the result of an activity in the way a computer or some other product can be. Nor can it be compared to a process in the sense of industrial processes or human metabolism. A person's physical condition is a useful metaphor to explain in what way information security is a process:

- Both good security and physical fitness require constant efforts to build and maintain them. This *ongoing activity* is largely administrative (for humans: awareness, self-discipline, couching etc). It is dominant and guiding in relation to the following:
- Physical exercise and healthy habits are needed for fitness and they involve *momentary activity*. The analogue in information security is application of different forms of protections. Main types of them are :
    - Prevention (in a wide sense). Most security are related to this, and there is a common subdivision around prevention (in a narrow sense). A human analogue is given in the parenthes.
        - avoidance, (don't smoke; don't go to dangerous places)
        - deterrence, (carry a visible weapon with you; not good for most hazards)
        - prevention, (vaccination, contraception – neither is perfect, unlike avoidance)
        - mitigation, (wear a cycling helmet; use vitamins and other antioxidants)
        - transfer of responsibilities, mainly through an insurance (helps humans, too)
    - Detection. An obvious example is virus scanning. In some cases, this is a process in the sense that a problem has to be monitored for some time before it can be handled (e.g. a suspected attack on a computer network)
    - Response. If the detection mechanisms have been in time, the response begins with
        - stopping. After that, the next steps may be
        - recovery (to continue the operation),
        - remediation (damage and / or the cause of the problem) and
        - possibly punishing the culprits.

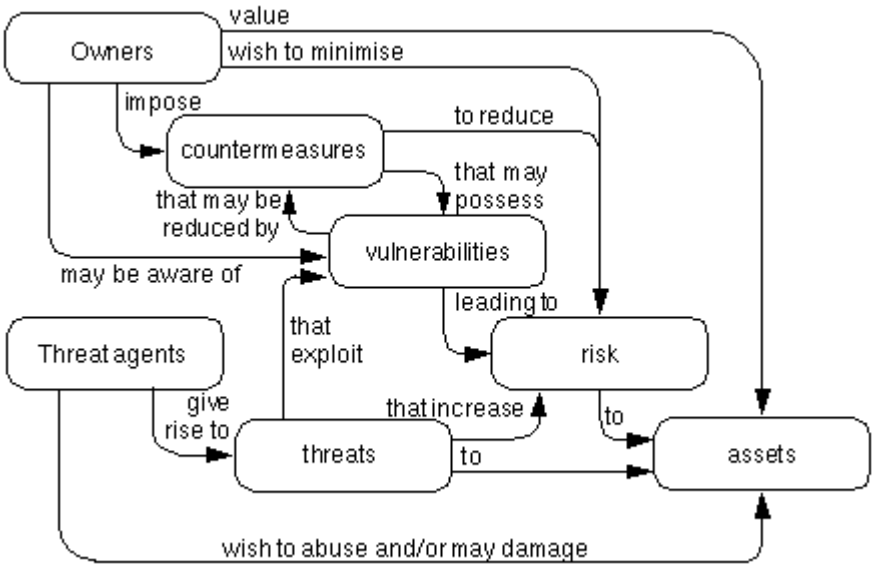
The constant need to build and maintain security stems from the fact that there is something going on all the time that can change the target of protection or its environment. Early version 2 of the Common Criteria standard summarized the main actors and actions in the diagram below.

For instance these two sentences can be ready from the the nonlinear diagram: Owners impose countermeasures to reduce risk to assets. Threat agents give rise to threats that exploit vulnerabilities leading to risk to assets.[9] In version 3 of the standard, the diagram was simplified by removing the

[8] ICS = Industrial control system, SCADA = Supervisory control and data acquisition.

[9] Such a sentence also appears and is explained in OWASP Secure coding practices quick reference guide v2.0, on page 4.

vulnerabilities. It makes sense, because there are more vulnerabilities elsewhere than in (good) countermeasures. Of course this diagram does not claim otherwise, and the extracted sentences avoid that route. However studying such a diagram too strictly would give you an equally distorted impression as taking the analogue to human fitness too far. The diagram introduces one further concept to include in the analogue, however, namely the owner. Data and information systems are like your body: not loose entities but belonging to someone who is supposed to take care of them.



As there were already two different terms used above for the abstract methods that create security, and a third one appears below, here is a list of them all roughly in the order of generality. The word security can be omitted when the context is clear. The last two terms are the least generic. A security provider is already an implementation of services, but the term security services can be used in quite an abstract sense.

- safeguard
- security control,
- security mechanism,
- protection,
- security measure, or countermeasure,
- security service,
- security provider

The process of getting any of those "security creators" into use in an information system can be divided into steps, such as the following:

1. Find out what is valuable and worth protecting (usually called assets).
2. Identify threats, vulnerabilities, attacks.
3. Assess the risks.
   If they are low enough, stop and return to the topic again at an appropriate time.
4. Prioritize vulnerabilities.
5. Select and install security measures, return to step 3.

The return instructions in steps 3 and 5 extend this one-time process to continuous. If there have been changes in the system, the return from step 3 must be made to step 1. In fact, it is the changes that give a reason to return to the topic. If no internal changes happened and the "appropriate time" just comes from the good practice of regular checking – once per year for instance – then that return may jump to step 2 instead of 1.

This division into iterative steps can be considered as a basic model of security work. In slightly more abstract terms, the same process can be described by the following steps:

(i) evaluate
(ii) plan
(iii) implement
(iv) maintain and monitor.

Step (iv) represents partly step 1 above, partly it is within the countermeasures in step 4.

Costs must be taken into account. They are unduly increased by both

- very strict security measures: the cost of acquisition and maintenance rises; and
- lack of sufficient security measures: the cost of losses is rises.

A compromise has to be found somewhere in the middle, but it is unlikely that there can ever be an exact numerical basis for optimization.

In steps 1–4 of the basic model you need to know your own system, but in step 5 you need to have a lot of information about security mechanisms. The goal of security courses is to tell you about some of them and provide references to others. While many of the mechanisms are in principle simple, the challenge is the diversity, specificity, and detail of these mechanisms, and still a degree of uncertainty, especially when multiple mechanisms are combined.

How do you know which costs are too high or what risks are low enough? In order to answer such questions, you need to know what you want. Someone has to decide how secure a system or organization is wanted and how much it is agreed to pay for. This decision is called an information security policy and is typically made by the management of the organization, even if the managers would not be (nominated) the data owners. The policy also shows how the responsibilities of the security process are shared.

Generally about threats

The need for security is easiest to realize by looking at what bad can happen. On the other hand, information security is defined through some positive features. These qualities, integrity, confidentiality, and availability naturally tell us something about what threats are avoided through them. However, here we take a very abstract look at what bad can happen to information or its processing.

Information can

1. disappear or be destroyed.
2. be transformed, or replaced with other data. In such cases the information may seem intact but its integrity is still lost.
3. be revealed to parties who should not see it. Here the information can be revealed either
    a) as such, like a secret document on paper or as a digital file – even if erased, or
    b) in such a way that an observer makes inferences about sensitive activities, like who communicates with whom, or the interval of keyboard presses (by recording the sound).
4. spread without proper compensation – through copying or theft. These are usually followed by use, which is then unauthorized (see next).
5. to be used without permission, which, in addition to the previous one, also means situations where the information does not (any longer) spread, but its usage exceeds the rights. In place of information here can also be a data processing resource.
6. be initially false, e.g., invented or lied, or just accidentally wrong. Computer programs in particular can be wrong and cause a lot of other annoyances on this list. Fake news is also a very widespread example.
7. be misinterpreted.
8. be irrelevant, or its use may otherwise fail. This is about data not making sense, because it is in wrong (or deprecated) context, format or language or the computing resource is unusable. The last

point is a very common failure, and it is not only due to denial of service attacks. Machines and software can fail for many reasons.
9. be rejected, e.g., when it is not received ("heard but not listened") or it is denied (active misinterpretation). A special case is the denying of information to which one has previously committed in one way or another, e.g., an agreement or activity from which observational information has been accumulated.
10. be abused – that is, used for harmful purposes, such as cheating, extortion, harassment. Such malice with information is, in principle, independent of whose the information is or who uses it, i.e., abuse here originally means other than unauthorized, although it may also be that, like in identity theft. Also originally you may think of this category as abusing *correct* information. However, a large area of malice can be found when combining #10 and #6. Besides the rather personal harms mentioned here, the combination enters the field of fake news, disinformation, information influencing and information warfare. And here it may depend on the point of view whether it is abuse or justified use – it can be lawful in a country to mislead its citizens or people in other countries.

The list reveals the very wide spectrum of issues that one must be aware of when identifying risks to information security. Many special systems may have ready-made checklists, but your own creative thinking is always an important complement to them. In addition, some active threats, i.e. those caused intentionally by someone, can be structured and even slightly anticipated according to what motivates the attacker.

Wrong kind of information seems to appear in the list in several places: broken integrity, false, misinterpreted, incomprehensible, rejected, abused. Of these six, only the first two are such where the information itself is wrong. In others, it is handled somehow in a wrong way or in a wrong context, and the processing fails or otherwise produces bad results. The difference between information that lacks integrity and initially false information may not be large, when you look at it, but the breakdown is worthwhile because the control is usually different. Maintaining integrity by fairly technical means is enough, but it does not help against falsehoods.

Any of the above can happen to the information if the person who owns or guards it acts improperly. Even if he tries his best, his own knowledge or perception of what is right can suffer from similar problems. It may happen that there is no guiding information, it is wrong, or he does not understand it. It has already been mentioned above how an attacker can make his victim act wrongly. Directly targeting an individual's thinking and emotional activities can be a *social engineering attack*. It is a personal influence, that results in disturbing the behaviour-controlling knowledge and emotions in the human brain for at least a moment (for instance by shifting the limit values of regulatory mechanisms). Thus, for its immediate effect, social engineering can be categorized at #10 above, but concerning data outside the human brain the attacker can achieve anything else from the list.

There is a scary visualization, originally from 2013, of large data breaches, especially disclosures of personal data, which are gateways to other data. When looking at threats against information, it must not be forgotten that the impact can extend far beyond any e-commerce server or storage cloud. A more primitive but much more painful example of these was provided by a 14-year-old boy in Łódź in 2008.

Taxonomy of attacks

Computer security has been around since the 1960's and a lot of research was done in the 70's. The Internet is younger, and also a great source for security research, for instance *An Analysis of Security Incidents on the Internet 1989-1995*. It is a dissertation by John D. Howard. He presented a

taxonomy[10] of attacks, which is still useful. It is given in the table below, where the attacker of a computer or network can be classified according to the first column and the goal (motivation) he is aiming for is one of those mentioned in the last column. The goal is achieved through some knowledge-related result. The result is that information can be accessed by some means. Access is divided into four stages.

The idea, then, is that different attacks can be modeled by selecting one (or more) options from each column. Because these are attacks over a computer network, one step is always the "process," that is, the fact that an attacker has launched a program on the target machine.

Attackers		**Tools**		**---------- Access ----------**						**Results**		**Objectives**
Hackers		User Command		Implementation Vulnerability		Unauthorized Access		Files		Corruption of Information		Challenge, Status
Spies	==>	Script or Program	==>	Design Vulnerability	=>	Unauthorized Use	=> Processes =>	Data in Transit	==>	Disclosure of Information	==>	Political Gain
Terrorists		Autonomous Agent		Configuration Vulnerability						Theft of Service		Financial Gain
Corporate Raiders		Toolkit								Denial-of-service		Damage
Professional Criminals		Distributed Tool										
Vandals		Data Tap										

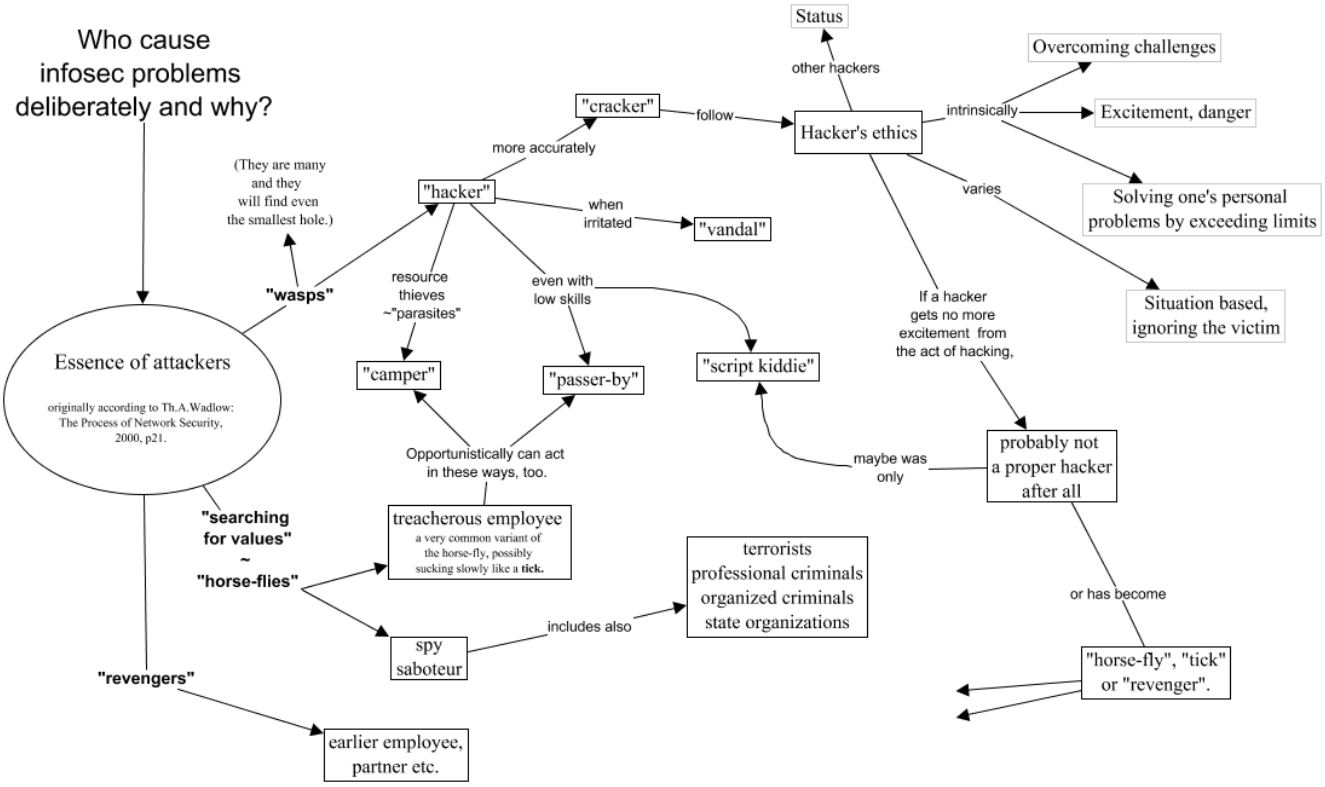
Since this study, the Internet has grown exponentially and an increasing proportion of machines have a high-speed, constantly open connection. These factors have contributed to the fact that the "Distributed tool" in the tools column of the table has gained considerable weight with botnets. These are programs run on hacked machines and used remotely via a command channel (e.g., traditionally IRC and now more generally http) and suitable for all types of attacks mentioned in the results column. The large quantity of machines have made botnets resemble armies. They have also introduced a new kind of motivation, or rather a new meaning to “Financial gain” in the last column of the table. Namely, a bot army commander can benefit financially by hiring it for various purposes. Bot networks also make it easier than ever to raise money through blackmail based on the threat of an attack.

The concept map below parses the attackers and their motives in a slightly different way. The term "script kiddie" is a dismissive designation for attackers who just use attack scripts compiled by more advanced ones, without necessarily understanding much about what is going on. The defender should not underestimate such attackers either.

A kind of contrast to script kiddies are the so-called APT attackers. The term APT, advanced persistent threat, has become more common around 2010. The name is quite illustrative. Persistence means roughly the same as the horse-flies in the concept map in the sense that an APT is motivated specifically for a particular target and does not give up easily. In many cases, there are state actors in the background.

Also non-state actors may aim for the "political gain, because it can include all kinds of ideological goals. On the other hand private hackers can participate in state-motivated cyber-attacks (e.g., from Russia to Estonia in 2007), or they can “organize” themselves on the basis of other types of ideas into activist groups, such as the famous Anonymous.

[10] A taxonomy is a classification that divides an entire field into mutually exclusive categories. Such a classification can be used, for example, to compile statistics on actual security threats, in which case conclusions can be drawn about the frequency of their occurrence, for example on the probabilities of the threats. This information, in turn, can be applied to security planning through risk analysis.



Security problems in a computer network

A computer network is a computing environment with multiple independent computers that are able to communicate with each other. Each machine can have multiple users, but otherwise the size of the machines and the distance between them are irrelevant to the operating principles of the network.

You should have some understanding of these concepts: Host, server, client, local area network (LAN), topology, gateways between networks, router, networks connecting networks (=internets), addresses, names, protocols and their layered structure (OSI model and TCP/IP model).

This is a list of network resources to be considered in the risk analysis, ordered by distance from the user: local nodes and links between them, the local network, its equipment, processes and stored data, e.g. control information; a gateway to an external network, and links from the gateway to the outside, external network routers, and resources such as databases.

The network has many useful features that also cause insecurity:

- Sharing resources is one of the benefits of networking, but it means that (i) there are more potential users (and at the same time abusers) and (ii) everyone has access to more machines than before, which may not always be the justified on the basis of earlier access rights.
- Complexity: the operating system of a single machine already is so large and complex that it is difficult to make it secure, at least so that it can be really trusted. When there are several machines and each has a slightly different configuration, the complexity of the entirety is closer to the product of the complexities of the individual systems than to their sum. This is not only a disadvantage, as different systems are not likely to be vulnerable in the same way. If there happen to be several firewall machines on some path of packets in the same network it may be beneficial to run different operating system on them. However, on workstations of an organization, already differences between versions of the same operating system can do more harm than good.

- Vague boundaries. Network extensibility is one of the advantages of a network, but especially interconnecting networks means that a user may not know what other parties – and of what characteristics – may be connected to it at any given time.
- Multiple attack points. While the effort of handling data (processes and files) is spread across multiple locations (which is a good thing), the number of locations where attacks can occur is increasing, and the user has to trust on access control mechanisms and other security measures of many machines.
- Anonymity: An attack can take place from very far away and through several intermediate stations, making it difficult to catch or even identify the perpetrator. Authentication between machines is also not directly as reliable as between people. It can be made more reliable, but it requires carefully designed protocols. There are also protocols that strengthen anonymity. They can be used both for good and bad: safely reporting problems in real life, or planning criminal actions.
- Unpredictable paths: The availability of computing is enhanced by the use of alternative servers or traffic routes, but at the same time you may end up using less secure nodes or links than you would like.

The threats to communication in a networked environment are similar to those that threaten information more generally. They can be summarized in four points: exposing a message, altering a message along the way, blocking of a message and appearing of a forged message. As a fifth problem there is unauthorized use of a remote resource over the network.

Let's take a closer look at what can go wrong in a network and how:

- Eavesdropping, "listening on the wire – or in the air". The question is whether it is worth sending anything without encryption.
  - **Cable**. In a local network (in the same segment) everyone hears everything, but in wired switch- and router-based LANs, the situation is not the same. However, in such networks, eavesdropping on the traffic of others with "packet sniffers" (such as Wireshark) is possible by means of port mirroring or e.g. ARP spoofing / poisoning. In addition, the electromagnetic field around the conductor allows the use of induction, but the difficulty of this operation varies greatly between different types of conductors. Optical cables do not have this problem, and they cannot either be wiretapped in a similar way as copper wires. Outside LANs, transmissions are multiplexed and therefore more difficult but not impossible to disentangle.
  - **Radio path**. Transmission cannot be completely focused, and it travels anyway through space where someone else can set up a receiver. Satellite transmission has very wide spread, but the problem is not very serious due to multiplexing.
- Wrong identity, appearing as someone else (impersonation) through
  - guessing or eavesdropping passwords.
  - avoiding authentication due to some error or design weakness, e.g. buffer overflow, getting a fake IP address validated.
  - getting access as trusted from a "neighbour" (`.rhosts` file and `rlogin` command) or as a `guest` from an open door or by a known / standard means (installation password).
- In addition to the above, confidentiality can be broken by
  - incorrect address in a message (due to a technical error or misspelling)
  - exposed data because of temporary decryption on intermediate stations (switches, routers, bridges, intermediate servers) or in working memory areas of programs that handle the message.
  - traffic analysis: the existence of a message or the frequency of communication already reveals something, the routing structure may also be revealed, or the names used in a LAN.

- Message integrity may be broken by
  modifying a message or part thereof, replacing a message with another, using an old message ('replay'), changing the sender's name, redirecting, deleting a message. There can also be (unintentional) noise, which is controlled quite well by error-correcting codes.
- Code integrity:
  The services, especially in the web, abound with situations where code is downloaded or used directly over the network. The user is not usually familiar with the code, does not always even realize that this is happening, and often cannot influence on the download.
- Denial of Service (DoS):
  Networks usually have redundancy in routing and services as well, but failure of critical paths crashes the network. This can happen because of a "flood": an attacker sends a large quantity of messages to someone, maybe authentic-looking ones that get into processing or just rubbish that fills the channel, possibly still duplicated as acknowledgments or error messages. A different type of blocking situation arises when routing becomes cluttered due to incorrect or attacker-modified "road signs".
  - Distributed DoS (DDoS):
    All kinds of "regular" DoSs get very high power when an attacker can directly or through intermediaries use a large number (e.g. thousands, but sometimes millions) of DoS agents against a victim. Such DoS agents, often called bot agents, can be poorly maintained machines on which, for example, a worm may have installed an attack program.

Overview of security mechanisms

The means to promote information security can be classified to

• Physical: includes hardware-specific means. The main means are various forms of separation and detection, as well as duplication, i.e. copies and fallback systems.

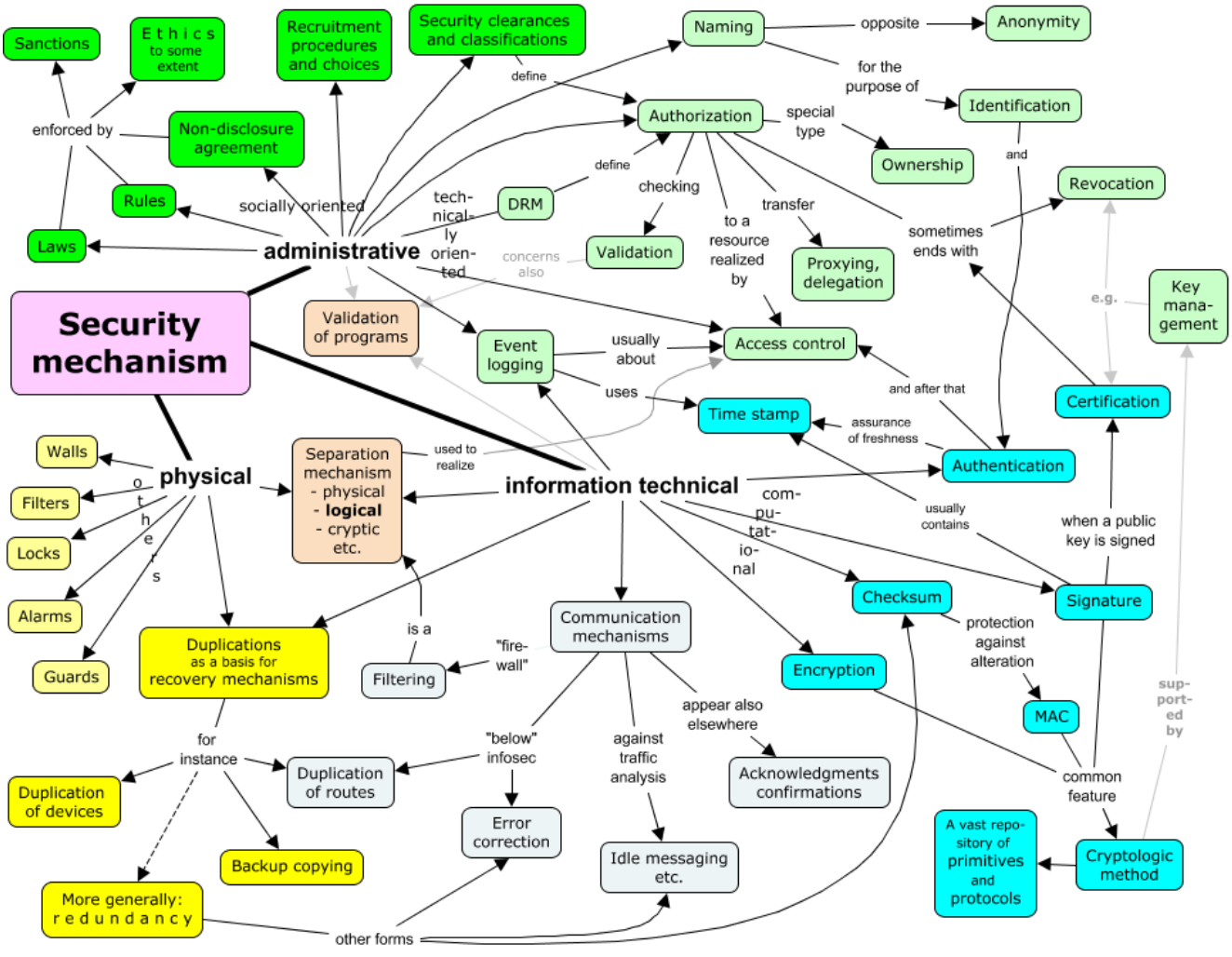
• Information technological or logical: realized with some level of programming. This is the most multifaceted category – as befits securing information.

• Administrative: regulatory and maintenance activities. This category has the most difficult challenges because it is closest to people.

A number of security mechanisms have been presented in the concept map below. They are related in many ways, but only a few connections are marked. Here is a brief explanation of the main logical mechanisms, i.e., those in the diagram that are directly linked to the term "information technological."

1. Encryption can be used to achieve confidentiality: the information will only be visible to those entitled to it. At the same time, integrity can usually be achieved quite well. In telecommunications, you have to decide whether to encrypt end to end, link by link, or something in the middle of these.
2. A checksum can be used to check for integrity: if the checksum is correct and *not tampered with*, then the information is likely to be in its original form.The probability of this gets higher the more complicated the checksum is. A *cryptographic* hash is needed if one wants to be sure. With a shared secret (a key) one can compute a MAC, message authentication code (often from a hash function), that prevents tampering of the check value itself to go undetected. A hash code can also be used to check software and other files for alteration by malware. In addition to hash codes, other integrity mechanisms are used in data communications (like CRC), but their purpose is not to protect against intentional modifications.
3. Signature provides a binding of data to the signer. The signing entity can also be a machine. Besides tying together the signatory and the data, a digital signature can support integrity of the data.
4. Authentication: verification of who or what an entity is – after or at the time identification. The entity can be a human or a system, e.g. a server machine.

5. Separation mechanisms implement access control, which is administrative, rights-based reasoning that restricts access to certain resources only for certain entities. There are physical separations, but mostly they are programmatic.
6. Event logging means appending each new event to a log, which can be used for instance as a basis of intrusion detection. Logs are sometimes called audit logs or audit trails, because checking them can also happen by auditors, sometimes external to an organization. Security audit is generally an inspection whether all these mechanisms are set up properly and according to a security policy.
7. The validation of programs includes both administrative and software-based means, like checking code signatures and running anti-virus, to ensure that the programs in a system are good and at least not harmful. Software license management belongs here or it can be thought of as part of DRM, digital rights management.
8. Communication mechanisms: error correcting codes are a traditional method. Traffic analysis can be thwarted by occasional "idle traffic" or making data packets of equal size etc.
9. Recovery mechanisms: redundancy as the basic method, where data is backed up and alternative services and routes built. (Also “exotic” methods, like cryptographic secret sharing).



The administrative and physical mechanisms are quite obvious. Here is an explanation of some other points in the map.

• Timestamp indicates when the related data were created, at the latest.

• Acknowledgment is a notification of receiving data and confirmation of receiving service. These are usually just parts of other mechanisms.

• Anonymity may not seem to be a mechanism, as special methods are needed to implement it in various contexts. The starting point for such methods, however, is to allow activity without name, and this makes anonymity itself a mechanism.

• Ownership is similarly not so mechanism-like as anonymity. The ownership of data by users and processes and the fact that this is taken into account, for example in access control, is a security mechanism,
• and the same goes for authorization, which is about granting rights.

Guidelines for private use of IT

Security of personal or home computing can be promoted a little more simply than following all the above mechanism. It is likely that your bank or insurance company offers good security guidelines on their websites. A minimal set instructed in 2004 by the Finnish communications authority was just to take care of

- software updates,
- anti-virus and
- firewall.

The first two of these fall into the box "validation of programs" in the concept map, but anti-virus also shares features of filtering and so do firewalls. As a logical separation mechanism filtering can also do other kind of pruning, discarding for instance spam and adult content.

An ordinary user of a computer, also of the handheld one, the smartphone, does not read instructions and could not care much less about the mechanisms on his or her devices. And even if one cares about one's own security and the instructions are read, they may go unaffected. This can easily happen to any of the following measures that have been condensed from a newer (2019) version of instructions of the same Finnish authority, nowadays called Traficom:

- Whenever you can, protect your account with multifactor authentication.
- Ensure that your devices have appropriate security already when you take them into use.
- Check before regretting, even for a second and third time.
- Install updates automatically.
- Make sure that all of your backup copy is actually saved and that you can recover data from it.

These fine-tune authentication, program validation and backup procedures from the concept map, and introduce two measures that are not in the concept map. Secure initialization of devices and services, especially of the IoT type, actually means almost everything in the map, but as things should be prepared by the manufacturer, it is again about validation. Proper security audit is too demanding for others than professionals. Checking before regretting, on the other hand, is mainly about using common sense, at least not getting cheated by offers that are too good to be true.

The last remark above is also the last item in the General online shopping checklist, that fills page 18 of the document Fraud in cross-border e-commerce, published by the European Consumer Centres Network in 2017.

Guidelines for developers

As mentioned in the previous section, the main responsibility of enabling secure IoT should be on manufacturers, developers and service providers. These are the industry audience to whom the Australian government has designed its Code of practice for securing the IoT for consumers, (2020). It has 13 points that are easy to understand and justify, but the fact that they must be reminded also tells that they have been sources of vulnerabilities. (*Only a few explanations are added here.*)

1. No duplicated default or weak passwords, *includes secure credential management*
2. Implement a vulnerability disclosure policy, *i.e. manage reporting and actions on reports.*
3. Keep software securely updated
4. Securely store credentials
5. Ensure that personal data is protected

6. Minimise exposed attack surfaces
7. Ensure communication security
8. Ensure software integrity, *i.e. against an unauthorized change.*
9. Make systems resilient to outages
10. Monitor system telemetry data, *that is, if it is collected, use it for anomaly detection*
11. Make it easy for consumers to delete personal data
12. Make installation and maintenance of devices easy
13. Validate input data

The last item in the above list corresponds to the first title in the OWASP Secure coding practices quick reference quide v2.0. While the above list only has short explanations in the document, the OWASP guide has altogether 214 points to check under 14 titles. The purpose of the two lists is of course nearly the same, but the vantage point is different. Most of the items in the above list relate to security needs of the application, albeit quite detailed ones, like credentials (#4) and personal data deletion (#11). Item #6 in the above list (attack surface) is a very general principle, and it is served by several of the OWASP list, which is given below. The number in parentheses shows the length of the checklist for each item, and numbers with # are main references to the above list. They do not mean inclusion in either direction, and especially #6 could be related to more items than now marked.

1. Input Validation (16)	relates to #13
2. Output Encoding (6)	
3. Authentication and Password Management (35),	relates to #1, #4
4. Session Management (19)	
5. Access Control (24)	relates to #1, #6
6. Cryptographic Practices (6)	
7. Error Handling and Logging (24)	relates to #10
8. Data Protection (12)	relates to #2, #5, #6, #11
9. Communication Security (8)	relates to #7
10. System Configuration (16)	relates to #3, #4, #8
11. Database Security (13)	relates to #6
12. File Management (14)	
13. Memory Management (9)	
14. General Coding Practices (12)	relates to #3

It is worth noting that OWASP is *Open Web Application Security Project*, but the secure coding principles are good for non-web applications as well. From the IoT listing only number #9 is not served by the OWASP checklists. It is about providing availability, and it is slightly surprising that the OWASP checklists do not have much to say about this. Of course, redundant servers and secure restoration of service is somewhat at a different level than coding. But outages can happen also by attackers. Denial-of-service is mentioned at item 3 of the OWASP list in relation to authentication.

System security design principles

What was said above concerned secure IoT and Web development, but those were just the target contexts of the referenced documents. Their applicability is wider, and on a basic course like this they are valuable in opening the eyes to all sorts of details that building secure systems requires.

One of the Harpo exercises on this course deals with the NIST Special Publication 800-160, Systems Security Engineering. As the exercise is not available outside the course, here is a copy of part of the instructions:

> The 800-160 is a fairly abstract and verbose document with the subtitle: Considerations for a Multidisciplinary Approach in the Engineering of Trustworthy Secure Systems. This is not only about information security, but the content is relevant to infosec professionals, as well.

This is true even if you cannot find for instance the terms denial of service, malware, phishing, backup, signature, phone, www, client, server.

and the introduction to question 4:

In the taxonomy of Appendix F there are 32 principles, whereas in Rick E. Smith's book *Elementary information security*) the list of security design principles contains only these 8 items (see his page):

- Continuous Improvement
- Least Privilege
- Defense in Depth
- Open Design
- Chain of Control
- Deny by Default
- Transitive Trust
- Separation of Duty

So, it is recommended now and in the exercise for you to check what kinds of design principles those 32 and 8 are.

The 8 principles are worth understanding and remembering, but it is questionable whether the same can be true of 32 principles. Their number is small, however, in comparison to security checklists, like the 214 items in the OWASP list mentioned above. The German handbook mentioned in the first section of this document is naturally far beyond memorizing with its 855 pages, but it is still much better than the "internet" where you can continue searches without getting a holistic picture.

This document presents pictures and listings, and more can be found in various standards as already mentioned. Some of them can be useful for you to create your own set of principles for your work or study. This could be done by abstracting ideas from lists and organizing them in one or more concept maps that reflect your own way of understanding and remembering. If not the maps in this document then possibly the layered design of Common criteria or IT-Grundschutz can give a starting point for this. A further example of layers is in the a U.S.-defence-oriented document Cybersecurity Maturity Model Certification (CMMC) where pages 10–11 give a couple of graphs showing how the practices accumulate when maturity increases. The total number of practices is 171.

Example: Trusting a program can be complicated

Assume your laptop is running Microsoft Windows and you are editing some files that you need to copy over the internet to a computer at the university. How can you make such file transfers securely?

What you are reading now is an exercise that presents the solution in the third paragraph and then spends 2,5 pages making questions – and hardly providing answers. The intention is to familiarize you with all sorts of issues that are related to trusting a program. There are no other exercises of this kind on the course, but this example may lead you to think of some other situation as a possible topic for your essay in Exam 3.

Start by finding the correct site to download WinSCP. Is it winscp.net?

Download the program file, and start investigating it. You can do this even if you don't use Windows. The file is less than 11 megabytes and doesn't do any harm.

There are three checksums given on the download page:

MD5: 1dd9535e9f9ee30679b14a6fc16c87b9
SHA-1: b93eb228de3293f74b54974296c72eb6e4fab046
SHA-256: 79de2d5cba143cba220ecf6c76d9e07407243e554ba524a78365ccd881b80214

What coding is this, with abcdef?

You may guess that SHA-256 has 256 bits. How many have SHA-1 and MD5?

The site indeed calls these checksums, but their calculation is much more complicated than mere summation. What is a more precise term for these “fingerprints”?

How can you calculate the fingerprint of the downloaded file? In Windows the keyword can be FCIV. Find the fingerprint of the file and compare it with the one given.

What does a successful comparison prove?

If the file were not ok, why would the page give a correct fingerprint?

So, a matching fingerprint only proves that the file is the same as on the page.

Could you find support elsewhere for the fingerprint, and what would it help?

Now see more closely into the file, but not into its contents but its metadata. In MS-Windows go to >Properties >Digital signatures. Or find a certificate decoder in the web, like https://www.sslshopper.com/certificate-decoder.html to decode this


```
-----BEGIN CERTIFICATE-----
MIIFeTCCBGGgAwIBAgIQAjJGbclbQOydIdkymr/NXTANBgkqhkiG9w0BAQsFADBs
MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3
d3cuZGlnaWNlcnQuY29tMSswKQYDVQQDEyJEaWdpQ2VydCBFViBDb2RlIFNpZ25p
bmcgQ0EgKFNIQTIpMB4XDTIwMDIxMDAwMDAwMFoXDTIzMDIxNzEyMDAwMFowgZcx
EzARBgsrBgEEAYI3PAIBAxMCQ1oxHTAbBgNVBA8MFFByaXZhdGUgT3JnYW5pemF0
aW9uMREwDwYDVQQFEwg4NzMzMTUxOTELMAkGA1UEBhMCQ1oxDzANBgNVBAcTBlBy
YWdlZTEXMBUGA1UEChMOTWFydGluIFByaWtyeWwxFzAVBgNVBAMTDk1hcnRpbiBQ
cmlrcnlsMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEArjgofJe/Yqml
CDbY+bsHzq5lKVtLZiHMP4vf6u9wNfdsZ93Zr4z7BS85U/cac0ab7v+zfQSb9SAp
v8K9heu/s02D/5ulSt2NFrW9uztFfZeE9L5Vy7IHhqFAdklZDTgM5zkuulDsK/g8
1vjYHFulKdCM/4DVO8dXNQiQfUlolmMMTIh+ygudKvZtRwQdpJUTkKu26Y4HSDYw
GTtB/BwzTX1T3TCCCOmGSXaqBNrXgQgfgjAs9Zfa5GCoaQOUo8Sl H8emNiVuHzJ4
yDixNeiTOLQvJNfLGHAPZShXnrf2BSbYrpBxKCDmucuhMQfgI3VUboRlohPzZUQ6
eRYSLltk7wIDAQABo4IB6TCCAeUwHwYDVR0jBBgwFoAUj+h+8G0yagAFI8dwl2o6
kP9r6tQwHQYDVR0OBBYEFB6VJN3DGK95aOAEc/7thaBFoj6lMCYGA1UdEQQfMB2g
GwYIKwYBBQUHCAOgDzANDAtDWi04NzMzMTUxOTAOBgNVHQ8BAf8EBAMCB4AwEwYD
VR0lBAwwCgYIKwYBBQUHAwMwewYDVR0fBHQwcjA3oDWgM4YxaHR0cDovL2NybDMu
ZGlnaWNlcnQuY29tL0VMQ29kZVNpZ25pbmdTSEEyLWcxLmNybDA3oDWgM4YxaHR0
cDovL2NybDQuZGlnaWNlcnQuY29tL0VMQ29kZVNpZ25pbmdTSEEyLWcxLmNybDBL
BgNVHSAERDBCMDcGCWCGSAGG/WwDAjAqMCgGCCsGAQUFBwIBFhxodHRwczovL3d3
dy5kaWdpY2VydC5jb20vQ1BTMAcGBWeBDAEDMH4GCCsGAQUFBwEBBHIwcDAkBggr
BgEFBQcwAYYYaHR0cDovL29jc3AuZGlnaWNlcnQuY29tMEgGCCsGAQUFBzAChjxo
dHRwOi8vY2FjZXJ0cy5kaWdpY2VydC5jb20vRGlnaUNlcnRFVkNvZGVTaWduaW5n
Q0EtU0hBMi5jcnQwDAYDVR0TAQH/BAIwADANBgkqhkiG9w0BAQsFAAOCAQEAkACN
0vCCRLPCJfpMyMIXIIYysXez4+ODX4xvUQfWZIrCqzvs+dl5rK+KrD2YYyQUdf+i
sBkVGX0jtTT3jP6KQXRZFPo+esnHKv2NPGh3feDRP7w8iSc//PHMD9uuUuGndvIl
YaRgqVNlZ4Ne1lPKXL8szRlb/6MbkpdK1cbx/zTnG5UY4/oeAIz1CBUqFh6OIVLP
O3KrJO5afSjnP0KH5Vq4ha9JzwIWMKgJY22CI8OFgRLMpajNbriLdoXCLWkiM4tv
oM/jzrqIOSATO6+cXRF1x0abIGl3mVCL1XfIqdYmrtuHx1m5FOG7uhBdE1sWer7y
EGdvHzDGZOCZFPW7Uw==
-----END CERTIFICATE-----
```


What sort of decoding happened? In other words what coding is it that starts like this `MIIFeTCCBGGgAwIBAgIQAjJGbclbQOydIdkymr/NXTANBgkqhkiG9w0BAQsFADBs` ?

You will see also a person's name. Whose? Is he the attacker who modified the program and digitally signed it? Who gave him the certificate for the code signing key?

(Go to >Details and >View Certificate.) What evidence is given to support the validity of the signature?

(Go to >Issuer Statement.) Which CP and CPS are valid for the certificate in question? See how these documents look like and what they tell about. (CP=Certificate Policy, CPS=Certification Practices Statement)

So, if you trust the certificate and the signature on the program, what evidence do you have that the signer has authority to sign the code?

You apparently get to evaluate the download site. There you get to a similar chain of investigation that first leads to a certificate authority. Then it leads to evaluating the trustworthiness of the site itself. Did they write the code or are they just distributing it? How would you trust that the site is not spreading malware?

Can you find support through some kinds of reputation checks – on the publisher and on the program itself? Does the publisher provide evidence that code was made securely?

Can you see the software license somewhere before starting to install the program? Yes, but can you find the page, without this direct link? When you see it is a GNU GPL, are you relieved that you don’t need to read more? Did you notice that there are some picture elements that are not under the GPL, and there are a few extra paragraphs in the license concerning them? And in the end there is also a note on privacy.

You need not install the program, but what is your conclusion, aided with your antimalware software, whether it is safe to install it?

Because the WinSCP file is digitally signed the operating system automatically attempts to verify the signature during installation. Most likely it accepts the signature because it regards the certificate as valid. So, you may think you did the above pondering in vain. Not really, the operating system is no wiser than you, but it just has the CA certificate preinstalled. And it probably still asked you to accept the installation. And furthermore, it probably asked you to provide admin credentials before proceeding.

This is just one example – and quite high in the hierarchy between hardware and humans – of the security tasks of an operating system. This is essential in the modern networked computing, but the more profound security services are lower in the hierarchy. Some of them are almost as old as operating systems themselves, for instance keeping user's code from poking or even peeking into the OS code. Some of the low-level controls are newer, like ASLR, about which you may find out yourself now, or when facing the MCQ on the Maso pages.

Let us get back to WinSCP. If you install and start using it, some important questions still remain.

The program does file transfer between hosts over the internet. How do you know what protection is needed and whether that protection is given? Firstly, how is the destination host authenticated? Secondly how is your transmission encrypted?

Knowing that WinSCP, or some similar program on your own machine, handles the security task properly is largely based on trust and the observation that many others have trusted without any problems. But crowds are not always right. And if you are one the early adopters, you might need to do your own testing by seeing what sort of traffic the program generates. A packet sniffer will help in this, but probably you would need several other tools, also such that are designed to do cryptanalysis or other sorts of attacks.

New programs to be installed would start this story over, and go beyond the course objectives. It is obvious though that you must not see any cleartext in the traffic, and you must not see repeating patterns when you modify the input, i.e. the file to transport. And of course, you should see an SSH handshake. SSH, the Secure SHell, originally a Finnish design from 1995 by Tatu Ylönen, will appear as a basis of WinSCP in what follows.

The main reason to trust the operation of WinSCP is that it provides support for SFTP and SCP and not just FTP. What does this mean? In other words, what are these protocols?

It is apparent that the authors and publishers of WinSCP are honest and try to do their work well. Although WinSCP can be considered a secure program, it has also had bugs, such as an integer overflow. It is described on https://www.cvedetails.com/cve/CVE-2013-4852/ in a single sentence, which can be structured as follows:

Integer overflow
in PuTTY 0.62 and earlier,
WinSCP before 5.1.6,
and other products that use PuTTY
allows remote SSH servers to
cause a denial of service (crash) and
possibly execute arbitrary code in certain applications that use PuTTY
via a negative size value
in an RSA key signature
during the SSH handshake,
which triggers a heap-based buffer overflow.

Here PuTTY is the SSH client program on which WinSCP is based, that is, WinSCP talks to the SSH server at the remote machine through PuTTY.

What stage does the vulnerability described above affect when using WinSCP:

a) Verification of the installation file signature discussed at the beginning of this text,
b) server program authentication to the client program,
c) authentication of the client user to the server or
d) the stage at which the program checks the integrity of the file transferred between machines?

How did you reach this conclusion?