IN5280 - Home exam 1

Security Analysis

University of Oslo

Spring 2019

1. Asset identification, categorization, and criticality assessment

a. Identify the information assets in this system

- an **asset** is any data, device, or another component of the environment that supports information-related activities. In this case, the system is Secure Common Messaging System. It makes it possible to send and receive information between government and citizens. Here are some of the assets which can be a part of the messages:

- Salary
- Tax returns
- Driver's license
- Hospital records
- Date of birth / Age
- Weight / height
- National identity number
- Passport number
- Address
- Bankcard information
- Telephone number
- eMailbox
- eMail-address
- Server
- Log in credential

b. Categorize the information assets

Asset	Description	Category
Salary	Salary information about the individual.	Financial
Tax information	Information about what the individual taxes.	Financial
Driver's license	Personal information - Can be replicated	Government Issued ID
Hospital records	Medical records of test etc.	Medical
Date of birth / Age	The date of when the individual was born	Personal Demographic
Weight / height	An individuals mass or height	Demographic
National identity number	Unique personal id	Personal Identification

Passport number	Unique number for passport	Government Issued ID
Address	Address of where an individual lives	Demographic
Bankcard information	Information about bankcard number, year, cvc2 etc.	Financial
Telephone number	The contact number for an individual	Contact (personal)
eMailbox	Mail organizer, both sent and received emails	Contact (personal)
eMail-address	Email address of an individual	Contact (personal)
Server	A server hosting the service	Technical
Log in credentials	Credentials used to log into the system for employees	Contact (personal)

The assets are mostly information that we think can be used or passed along with the system. The assets are either used as information to the system or information passing through with messages between the two parties. What we mean with this, is that each component in the system contains one of these assets as (meta)data. While some of these assets contain easily obtained data, most of them are sensitive personal data that the system and the surroundings are supposed to protect. For an attacker to obtain one of these assets or more, could mean that the systems concept of CIA is going to be damaged.

c. Assess the criticality of the assets with respect to confidentiality, integrity, and availability. Use the criticality definitions and scale that is included in the lecture slides from the first lecture.

- To assess the criticality of the assets we used the CIA criticality scale provided in lecture 1, from the course IN5280, see scale below.[1] The criticality is defined by the system with respect to the assets. We assume most of the assets are linked to the citizens/users information, this is what we take into account when deciding the “CIA”. The assets were categorized by the criticality they have with respect of CIA. There are

[1] https://www.uio.no/studier/emner/matnat/ifi/IN5280/v19/doc/in5280-security-by-design-lecture-1-introduction.pdf

three assets we have categorized as low and insignificant, due to them being trivial by our judgment. Assets that are sensitive and important to keep protected are classified as high and critical. In this table, we also decided to make the mailboxes and mail-addresses high in confidentiality, because the text specified that government employees should not be able to see the different addresses they send the messages to.

Looking at the assets that have been classified as critical:

- Hospital records are and should be highly private. No one else than the doctor and patient should be able to see what these records contain. This is also by law, and therefore we have chosen that it is critical in confidentiality.
- We came to an agreement that Unique Personal ID is something that should be kept available only to government employees, institutions and the citizen. Because it can be used to so much more than bypassing this system, we chose to classify this as critical.
- (Hva skal være på bankcard?)
- Discussing whether the availability of emailboxes should be classified as high or critical, we came to a conclusion that even tho it should not matter if the eMailboxes for some reason are down for a couple of minutes, there are going to be very important messages that may need to be sent and received at any time. Therefore being critical.
- One of the (may be the most important) key assets in the system is the server, which should hold the highest criticality with respect for CIA. As a response to this, we put Critical on all the points.
- To make sure that the right people log in and have access for the system, one needs to put Integrity on Critical. Debating whether the

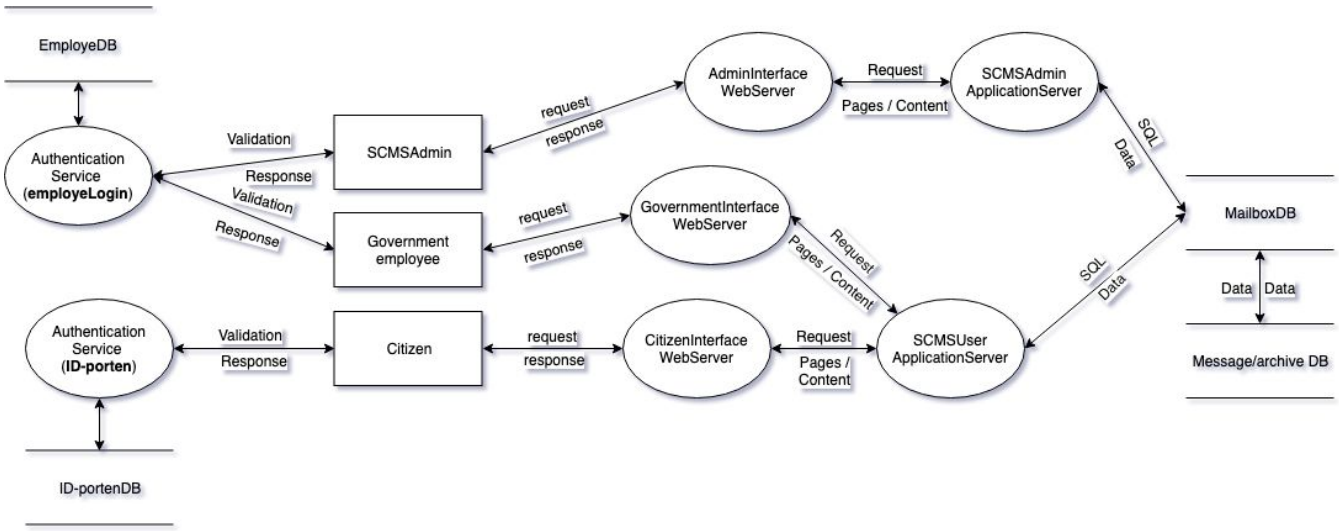
two others, Confidentiality, and Availability, should be marked as critical, led to compare the consequences of high and critical for the two. Depending on your role in the system, the confidentiality criticality should be set to critical, while availability remains on High.

Asset	Description	Category	System	C	I	A
Salary	Salary information about the individual.	Financial		High	Medium	Medium
Tax information	Information about what the individual taxes.	Financial		High	High	High
Driver's license	Personal information - Can be replicated	Government Issued ID		High	Medium	Low
Hospital records	Medical records of test etc.	Medical		Critical	Critical	High
Date of birth / Age	The date of when the individual was born	Personal Demographic		Low	Insignificant	Insignificant
Weight / height	An individuals mass or height	Demographic		Low	Insignificant	Insignificant
National identity number	Unique personal id	Personal Identification		Critical	High	High
Passport number	Unique number for passport	Government Issued ID		High	High	High
Address	Address of where an individual lives	Demographic		Medium	Insignificant	Insignificant
Bankcard information	Information about bankcard number, year, CVC etc.	Financial		High	High	Critical
Telephone number	The contact number for an individual	Contact (personal)		Low	Low	Low
eMailbox	Mail organizer, both sent and received emails	Contact (personal)		High	High	Critical
eMail-address	Email address of an individual	Contact (personal)		High	High	High
Server	A server hosting the service	Technical		Critical	Highly	Critical
Log in credentials	Credentials used to log into the system for employees	Contact (personal)		Critical	Critical	High

	Confidentiality	Integrity	Availability
Critical	Irreparable damage to society and potential loss of life if the information becomes known to unauthorized. Business penalties with critical financial consequences.	Information where errors will lead to irreparable harm to society and loss of life. Very high financial losses.	Information with very high availability requirements. Unavailability for more than 5 minutes causes critical damage.
High	Serious damage to company or individuals if the information becomes known to unauthorized. Business penalties with major financial consequences. Serious reputation loss.	Information where errors directly affect decisions that can cause harm to individuals and / or society. Big economic consequences	Information with high availability requirements. . Unavailability for more than 1 hour causes critical damage.
Medium	Some harm to company or individuals if the information becomes known to unauthorized. Some reputational losses, moderate economic consequences.	Information where errors can cause some harm to individuals and / or society, some reputation loss and moderate economic consequences.	Information with moderate availability requirements. Unavailability for more than 1 day causes critical damage.
Low	Company-internal information. Only minor harm to company or individuals if the information becomes known to unauthorized.	Information where errors affect decision making to a small extent. Negative consequences are very limited.	Unavailability has little significance. The information may be unavailable for 1 week without any consequences.
Insignificant	Public information. No harm to the business or individuals if the information becomes known outside company.	Information where errors do not have any negative consequences. Does not affect decision making.	Unavailability of information does not have any negative consequences.

2. Threat modeling

a. Create a data flow diagram to document the attack surface of the system

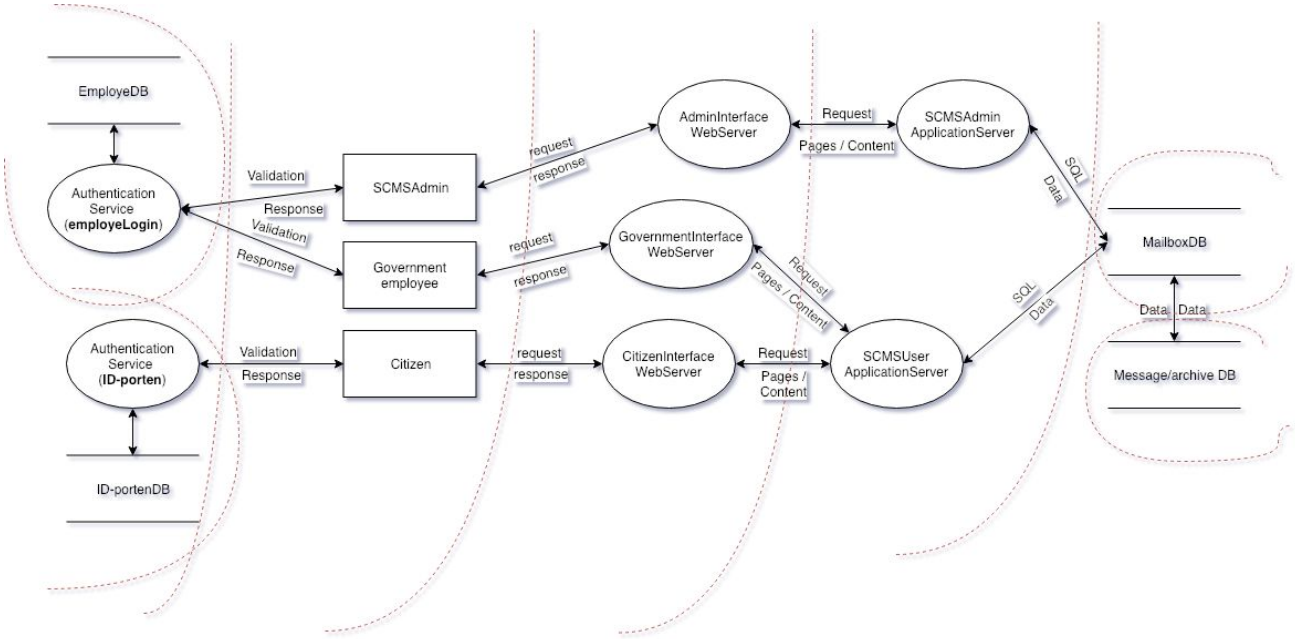


For the Secure Common Message System, we created a data flow diagram that maps the data flow of the system, as we interpreted from the assignment text. As it is more common, we begin designing the diagram from the moment the users log in. In this particular case, two roles were focused on. The sender and receiver, or as the text described them, government employees and citizens. In this case, the SCMS is used when the government employees send sensitive messages to the citizen, without anyone knowing the content or the receiver's information. From this, an assumption was made that the users first have to go through authentication, before being taken to their rightful interface. Every system needs an administrator (or more), and therefore we chose to add that role to the diagram. As there will be more receivers, than senders and admins, we split the authentication service into two different, where one handles the admin and sender operations, and the other the receivers'.

From here we have assumed that the two different authentications for the employees lead to two different interfaces, due to their roles being different, even tho they might both be government employees. The citizen has the interface where they will only be able to view and/or reply to the messages they receive. As we view the sender and receiver roles as users, we connect them to the SCMS user application server and the administrator to its own SCMS application server. For the users (sender & receiver), the user interface is different because they will use the SCMS service differently. The interface will display different information with respect to their role and access but use the same application server. While the administrators will have both different interface and application server because they can add and change content as well as services. The flow from all the roles will at the end connect to the mailbox database and the messages archive.

As a result, we get a data flow diagram that can easily be mapped to point out threat agents and attack surface of the system. The attack surface derived from this diagram is divided and points out the vulnerable components in the system. We chose to separate the different databases first. Then layer the authentication, roles, interfaces, and servers in groups to find indicators of exposure and compromise. Bypassing the authentication exposes the role of who you have identified yourself as in the system, hurting the concept of integrity. Every role has been given the authorization to do something; the citizen role can read incoming messages, the government employee can read and send messages and the admin is authorized to make changes and modifications in the system. The roles themselves can be exposed by going

through authorization and manipulating the users, or if it is the users who want to attack the system. The different interfaces offer different functionality, that each has its vulnerabilities that are prone to exposure. Availability of the system is mostly dependent on the server, which makes the application servers important to add to this attack surface as vulnerable components. This is how we meant it was necessary to map the attack surface and is visualized below



b. Identify threat agents

- All the users of the system and the admin can be possible threat agents, as well as external users.

Sender - usually the *government* worker who sends documents and letters, etc.
Receiver - the *citizen* who is receiving the information
Admin - who has control of the system, who has access and maintains the system.
External entity/Hacker - an external person with malicious intent.

c. **For each threat agent, identify 3-5 attack goals**

- Some of the goals are similar because these goals might overlap between the actors and some attacks can be achieved by the same method.

Sender (government):

- Change/delete information
- Crash the system
- Get access to off-limit emails/messages/cases
- Get personal information

Receiver (citizen):

- Gain admin privileges
- Trick/blackmail Sender (Get senders personal credentials)
- Get access to other receivers information

Admin:

- Change/delete information
- Crash the system to cause harm to the company
- Implement malware to sabotage the company

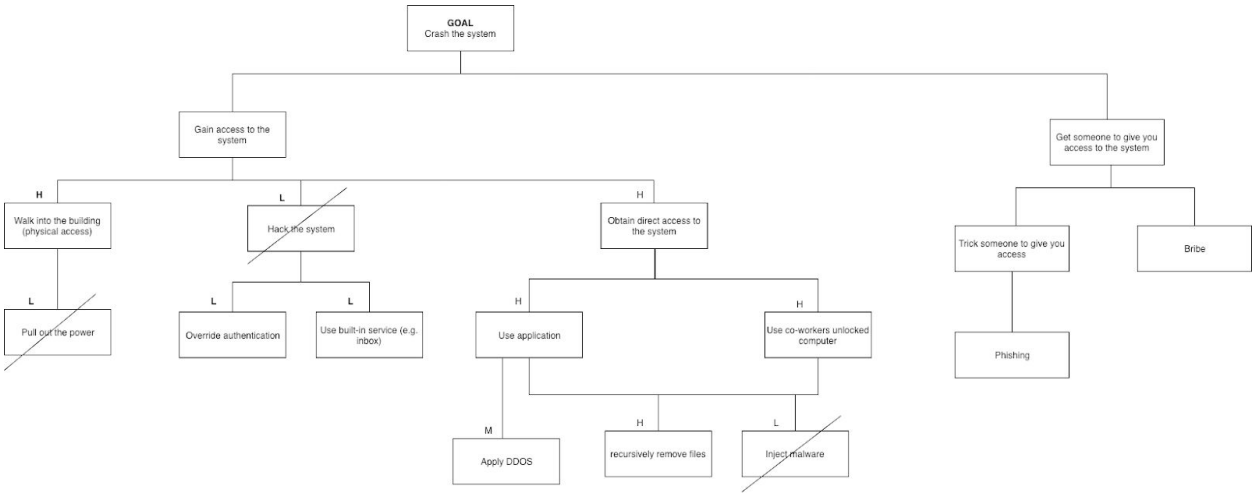
Hacker:

- Crash the system
- Find person credentials
- Find mail content

d. Select one attack goal for three different threat agents and create attack trees for these goal/agent combinations

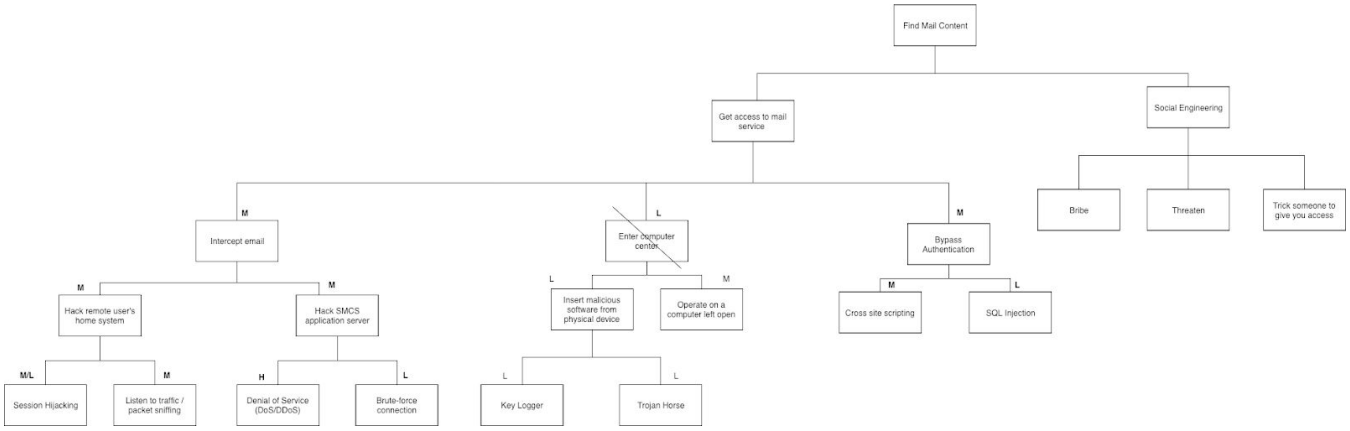
- As an alternative to STRIDE, we use attack trees to find threats. We focus more on the technical aspect rather than the social, and therefore all our attack trees barely mention social engineering methods. See larger image at the end of the file.

Sender:



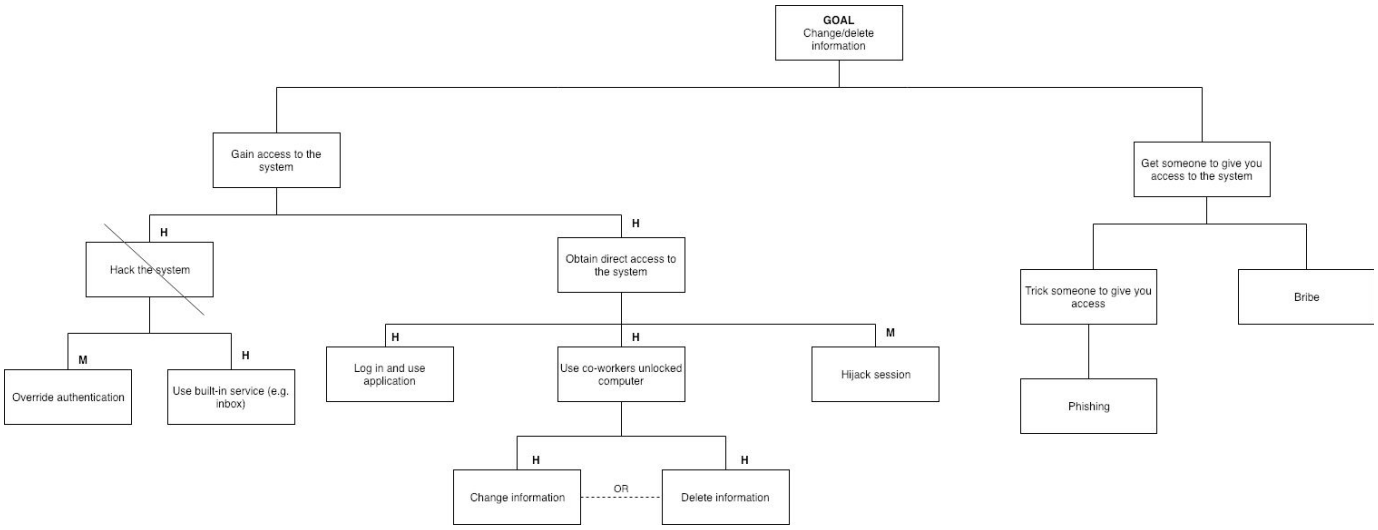
This attack tree focuses on the sender's goal of trying to crash the system. By pruning the tree, we mapped out the methods that were most likely to be carried out. As senders are government employees, they should be able to obtain direct access to the system, and the method that would most likely crash the system is to apply DDOS.

Hacker:



For our external role, we decided to create a tree for the goal of finding mail content. From the pruning, we came to the conclusion that the hacker would find the content from either intercepting mail or bypass the authentication, as walking into a computer center would be too hard.

Admin:



For our admin user which has access to everything in the system, we created a couple of scenarios. The scenario we think will be most likely where he deletes or changes either from using their own user or co-workers computer. We chose to prune “hack the system” option because from the admins' point of view we thought this option would be harder or less appealing.

3. **Risk analysis, using the RMF**

a. Identify business goals for this system

- The business goals are goals for the system which benefits either one or both of the actors.

Business goals	
BG1:	An easier way of communicating / faster reply/information
BG2:	Reduce cost - less manual handling of documents
BG3:	Better structure for cases and follow-ups
BG4:	More information and guidance available
BG5:	Trustworthy communication

b. From the business goals, identify business risks

- The business risks may either affect the goals directly or indirectly

Business risks	
BR1:	Information leaked
BR2:	Disappearing cases/messages
BR3:	Users might not trust the system
BR4:	System unavailable

c. From the attack trees and Data flow diagram, identify technical risks for the system

- These are the technical risk we thought about based on the attack trees and data flow diagram.

Technical risk	
TR1:	DDoS
TR2:	Broken Authentication
TR3:	Infected system
TR4:	Social Engineering
TR5:	Manipulation of Messages (basic editing)
TR6:	Server crash
TR7:	SQL injection
TR8:	Cross-site scripting
TR9:	Session hijacking
TR10:	Man in the middle attack

d. Link the technical risks to the business risks – what technical risks affect what business risks?

- There are some technical risks in this table that is not in the previous task because now we looked at technical risks related to the business risks. We have commented some mitigation on the technical risks were it was suited. Mostly on the sub-risks, because these will be the specific attacks. The probability, consequence, and risks are chosen based on what we think is suitable according to what we have heard and read about similar incidents. The technical risks with low probability is because we assume the servers or what is hosting the application is not located where natural disasters are common. Most of the probability is medium or high, because we think this can happen, it is just that no one says that they have been attacked openly or if it is a small attack. We also listed mitigation on all the specific technical risks so they can protect against them, even though the mitigations might not protect against everything.

Technical risks		Probability	Consequences	Risk	Mitigation
BR1:	**Information leaked**				
TR1:	*Cross-site scripting*	M	M	M	Validating Input
BR2:	**Disappearing cases/messages**				
TR1:	*Manipulation of Messages*				
TR1.1:	SQL injection	M	H	H	Input validation
TR1.2:	MitM attack	M	H	H	Add encryption or Verify TLS/SSL Setups

BR3:	**Users might not trust the system**				
TR1:	*Social Engineering*				
TR1.1	Employees	M-H	H	M	Have good training on privacy and security
TR1.2	Helpdesk	M	H	M	Help desk should only have read access and not write.
TR2:	*Broken Authentication*				
TR2.1:	Weak passwords	M-H	H	M	Improve password handling, stronger policy/requirements
TR3:	*Session hijacking*				
TR3.1:	Packet Sniffers	H	H	H	Authentication control
TR3.2:	IP Spoofing	H	H	H	Packet filtering, Use cryptographic network protocols
BR4:	**System unavailable**				
TR1:	*DDoS*				
TR1.1:	Botnet attack	M	H	H	Separate network usage
TR.1.2:	HTTP Flood	M	H	H	Implement a challenge to the requesting machine
TR2:	*Server crash*				
TR2.1:	Power outage	L	H	M	Back-up generators
TR2.2:	Natural disaster	L	H	M	Fortify physical environment and protection.

4. **Security requirements**

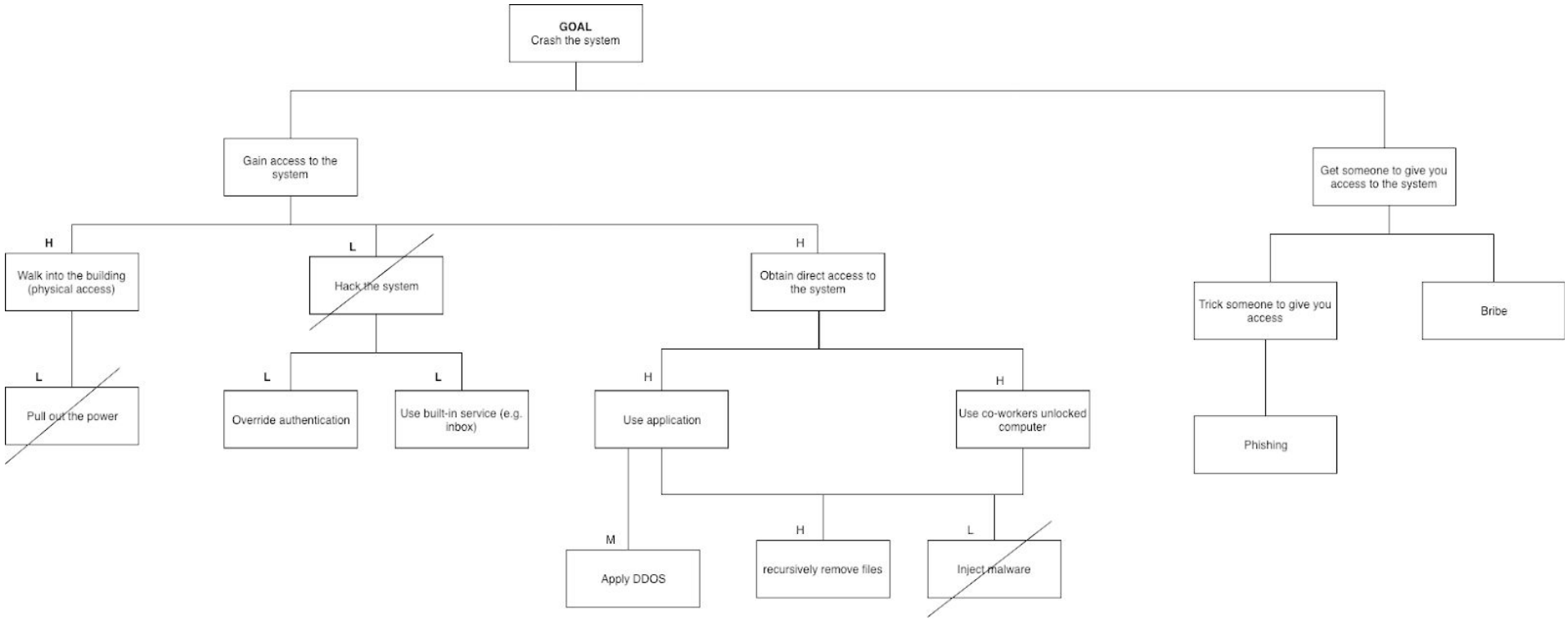
a. From the above analysis, create a set of security requirements for the system

- The system requirements created below are created with respect to both users, admin, employees and the system itself.

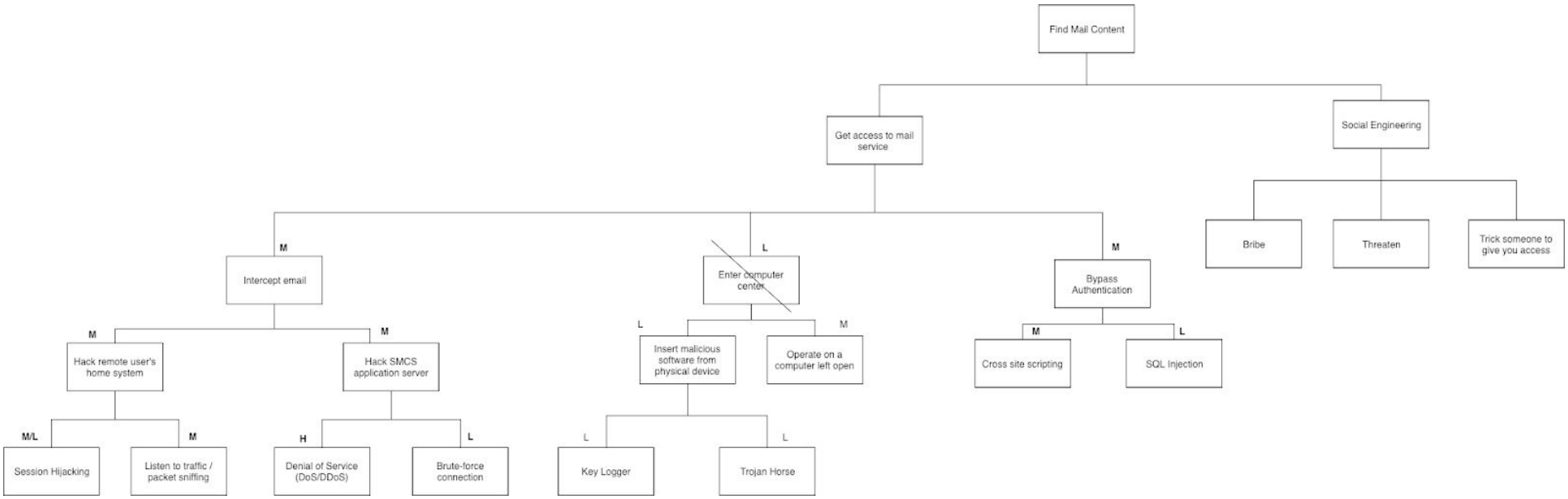
Security Requirements:

- The system should allow employees to log in with passwords according to company policies.
- The system should allow users to log in with a standardized log in method (i.e. *IDportalen*)
- The system should have input validation where input is required.
- The system should allow safe and encrypted transport of messages.
- The system should keep personal information confidential.
- The system should be available at all times (create back-up storage for example).
- The system should protect all data, especially sensitive data, in a way that preserves its integrity.
- The system should refresh, to ensure the user to be logged out if inactive for a longer period.
- The system should log interactions with cases (who replied, when and what).

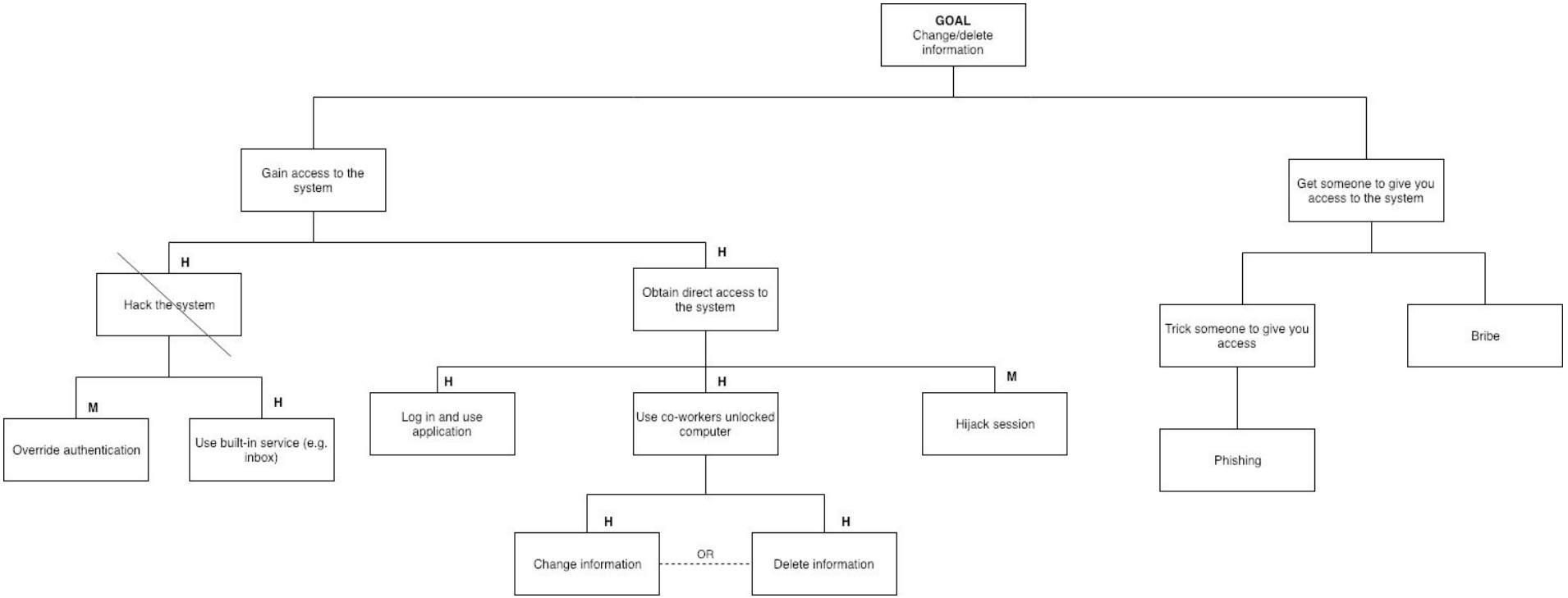
Sender:


GOAL
Crash the system
Gain access to the system
Get someone to give you access to the system
H
Walk into the building (physical access)
L
Hack the system
H
Obtain direct access to the system
Trick someone to give you access
Bribe
L
Pull out the power
L
Override authentication
L
Use built-in service (e.g. inbox)
H
Use application
H
Use co-workers unlocked computer
Phishing
M
Apply DDOS
H
recursively remove files
L
Inject malware

Hacker:


Find Mail Content
Get access to mail service
Social Engineering
M
Intercept email
L
Enter computer center
M
Bypass Authentication
Bribe
Threaten
Trick someone to give you access
M
Hack remote user's home system
M
Hack SMCS application server
L
Insert malicious software from physical device
M
Operate on a computer left open
M
Cross site scripting
L
SQL Injection
M/L
Session Hijacking
M
Listen to traffic / packet sniffing
H
Denial of Service (DoS/DDoS)
L
Brute-force connection
L
Key Logger
L
Trojan Horse

Admin:


GOAL
Change/delete information
Gain access to the system
Get someone to give you access to the system
H
Hack the system
H
Obtain direct access to the system
Trick someone to give you access
Bribe
M
Override authentication
H
Use built-in service (e.g. inbox)
H
Log in and use application
H
Use co-workers unlocked computer
M
Hijack session
Phishing
H
Change information
OR
H
Delete information