

# Risk analysis

Marcus Bendtsen

Department of Computer and Information Science (IDA)

Division for Database and Information Techniques (ADIT)

# Risk



- **risk = consequence \* probability**
- This is the classical definition that we will use in this lecture, but each of the factors can be decomposed:
  - Probability is a combination of the probability of a vulnerability and the probability of a threat.
  - Consequence can be of different types, money, goodwill, etc.

## Example:

Every row in our database is worth \$0.01 when it is protected. There are 10 000 rows in the database. The probability that somebody can steal our database is 0.5, thus the risk is:  $(\$0.01 * 10000) * 0.5 = \$50$ . If somebody is selling protection for \$100, then we would lose money by buying the protection, but if somebody is willing to sell protection at \$30, then it may be worth it to protect the remaining \$20.

# Risk analysis

- Risk analysis is a process of finding and quantifying **threats** using the aforementioned equation (risk = consequence \* probability).
- The challenge is in doing this in an **organised** manner, such that as many threats as possible are found, and that the quantification is done **as correctly as possible** (it is not always possible to use a quantitative risk measurement, sometimes a qualitative is necessary).
- It is not possible to find all threats, since no single individual or group, has complete and clear insight of all parts of a system.

# Some attacks

- An attack is the realisation of a threat.
- Automated attacks
  - Worms and viruses
  - Target low-hanging fruit
  - *Extremely* common
    - You have likely been exposed to these, but you may not be aware.
- Targeted attacks
  - Aimed at specific targets
  - Performed with a specific aim
  - Uncommon



# Some attackers

- **Curious attackers**
  - Computers were new, wanted to learn for fun.
- **Ideological attackers**
  - Defacing governments or businesses.
- **For-profit attackers**
  - Make money from breaking into systems. Targets systems that have value for them or their clients.
- **Corporate attackers, Terrorists and Nation states**
  - More exotic, significant resources. Most of the time not detected. Consequences can be disastrous, luckily very rare.
- **The type of attacker affects risk analysis.**
  - Protecting against *automatic attacks* is a lot different than protecting from *nation states*.
  - Motivation, risk adverseness, capabilities, patience, etc.

# Some purposes

- Break into systems
  - To steal information
  - To manipulate information
  - To use resources
- Take control over systems
  - To perform new attacks
  - To manipulate systems
- Disrupt service (Denial of Service)
  - To extort target
  - To discredit target
  - To facilitate other attacks

# Risk analysis - difficulties

- Risk analysis is difficult.
- Sometimes it is **not hard** to find the correct consequences, it is possible that one knows the consequence if a threat is realised.
- It is however **very hard** to estimate the probability:
  - A system can be targeted by attackers that test the same attack on millions of systems, or by somebody that is specifically targeting the system.
  - The probabilities for success are very different.
- Estimating incorrect probabilities can lead to one threat being judged as high risk, and thus resources are put towards mitigating this threat, however in reality another threat may actually have had a higher risk (which was not mitigated).

# Risk analysis methods in general

- The analysis needs to be constrained to a certain part of the system:
  - Not all details can be assessed in one single analysis, if one attempts this it often leads to a type of "analysis paralysis".
  - Another type of "analysis paralysis" comes from iterating the risk analysis indefinitely.
- **Depth of analysis needs to be constrained:** Are you only going to consider the programs running on a system, or are you also going to look at the source code of the programs?
- **Qualitative or quantitative?** Will you be using real numbers to quantify the risk equation or are you going to use qualitative values such as "high-mid-low"?



# Risk analysis methods in general

- There exists many methods for risk analysis.
- A common problem is that they expect the analyst to find all threats, vulnerabilities, etc.
- This will lead to **subjective opinions** being part of the analysis: different people will weigh the consequence and/or the probability of a threat differently.
- However, there is no closed-form mathematical formula to solve the problem and thus we must resort to heuristic methods, albeit that they are not globally optimal.
- Some methods use "*brainstorming*", such that the analysis is done by more than one person. The motivation is that you find more threats this way, however there are group dynamic issues (for instance, the one that speaks the loudest gets their opinion through).

# Why bother?

# Why bother?

*Just because a problem doesn't have a solution,  
doesn't mean that it isn't a problem.*

- *Timothy Geithner*

*(sort of, I didn't lookup the exact quote)*

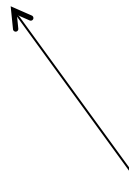
# Risk analysis methods– This lecture

- In this lecture we will look at three different risk analysis methods:
  - CORAS
  - Information Security Risk Analysis Method (ISRAM)
  - Attack Trees

# CORAS

# CORAS

- CORAS defines a language to model threats and risks.
- CORAS consist of 7 steps, where every step is in the direction of getting a quantification of the risks. (Sometimes CORAS is defined with 8 steps, but it is the 7 step method with an additional step that we skip).
- F. den Braber, I. Hogganvik, M. S. Lund, K. Stølen, F. Vraasen, *"Model-based security analysis in seven steps - a guided tour to the CORAS method"*

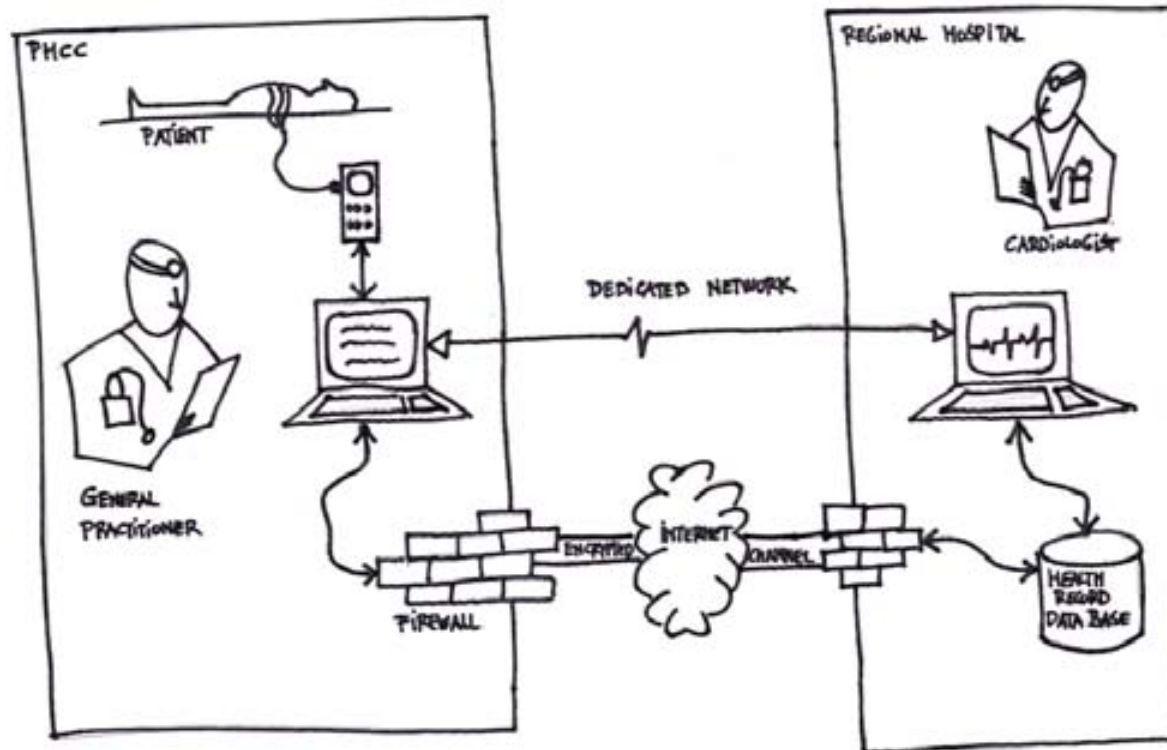


This is the CORAS that we use in this course, and the CORAS you should know.

# CORAS – Step 1

- **Customers** = They who own the system that is to be analysed.
- **Security experts** = They who perform the risk analysis (can be consultants or in-house).
- The initial meeting between the experts and customers is concerned with defining the scope (constraints)
  - It must become clear which assets are to be protected.
  - The boundaries of the analysis (depth and width) must be clearly defined, i.e. which parts and how deep of the system should be considered.

# CORAS – Step 1



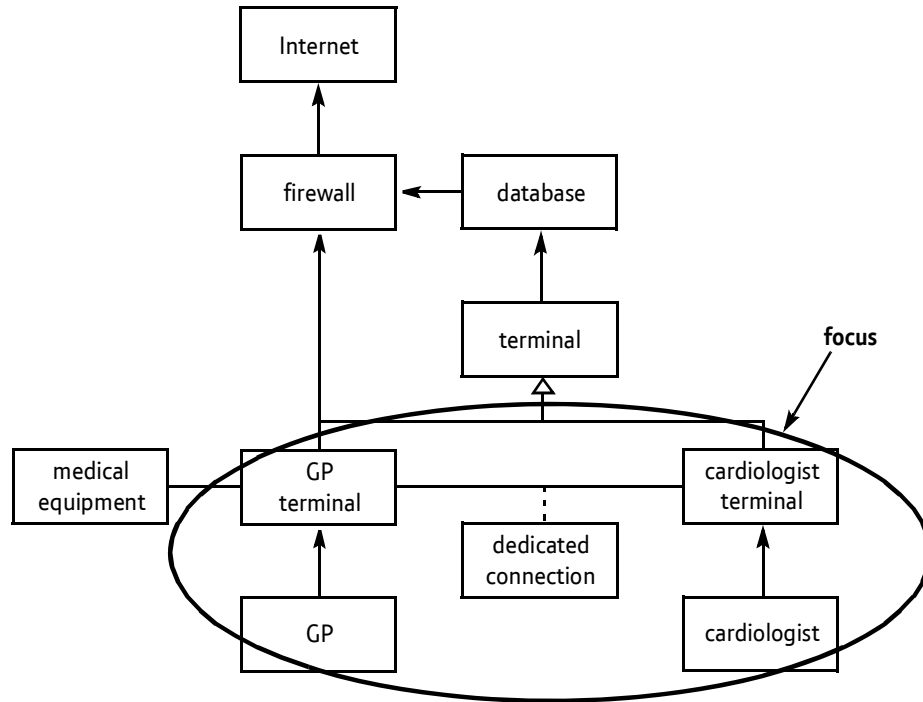
A “low-tech” picture of the system is drawn at the initial meeting in step 1. In this drawing it is ok to include parts of the system that should not be subject of the analysis. For instance, in this case the connection to the database should not be part of the analysis, yet it is in this picture for completeness.



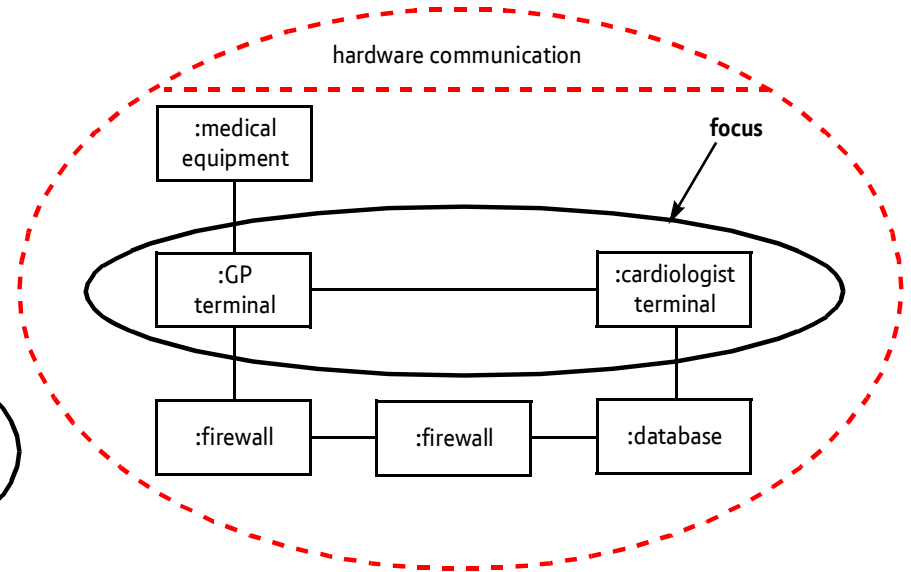
# CORAS – Step 2

- The system is formally defined using UML by the security experts (*class, collaboration, activity*).
- The experts also produce a **CORAS asset diagram**.
  - *Direct* assets and *indirect* assets:
    - Indirect assets are assets that are hurt due to a direct asset being hurt.
    - Arrows are drawn to show how damage to an asset affects other assets.
- A new meeting is set up with experts and customers where the experts show the diagrams and the customers can make amendments.

# CORAS – Step 2

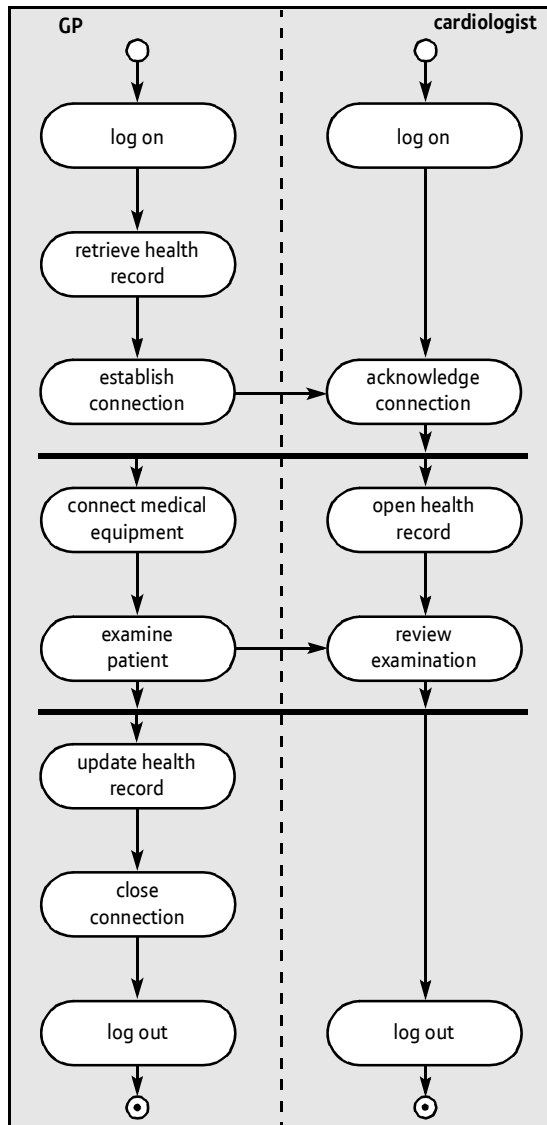


*Class diagram*

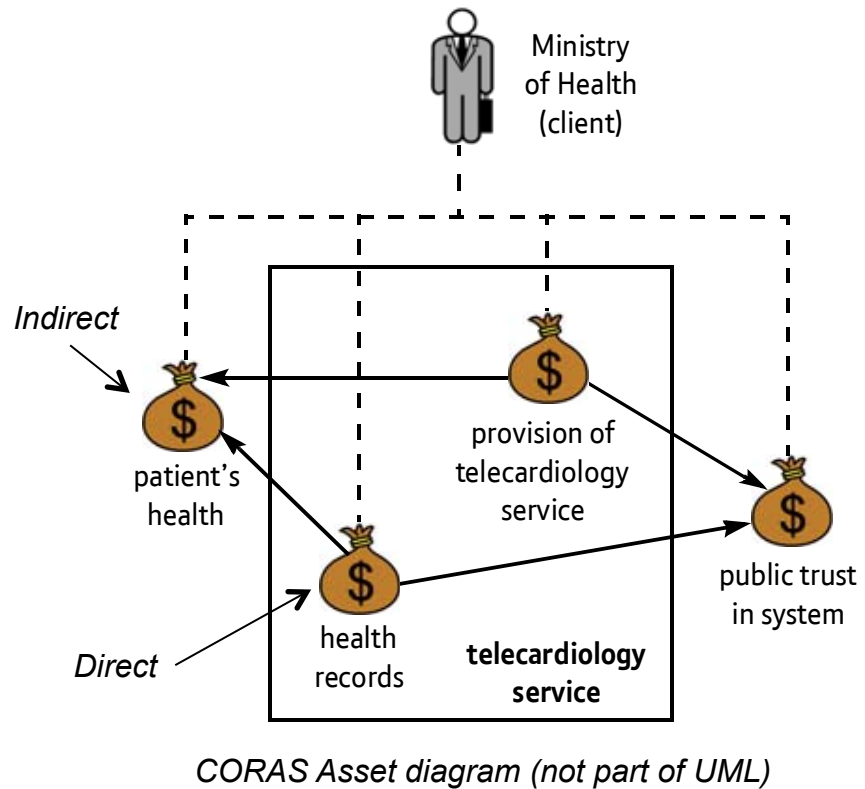


*Collaboration diagram*

# CORAS – Step 2










Activity diagram



# CORAS – Step 2

- Once the diagrams have been accepted by the customer, a brainstorming session is performed (with both customers and experts).
- Here, it is important to identify what threats the clients are worried about, e.g. that external person sees or hears something that is private, etc.
- These are not necessarily the most important threats, but they are a good starting point for the experts in depth analysis.
- The brainstorming leads to a risk table (next slide).

# CORAS – Step 2

 threat (accidental)  threat (deliberate)  threat (non-human)			 threat scenario  unwanted incident  asset	 vulnerability
Who/what causes it?			How? What is the incident? What does it harm?	What makes it possible?
Hacker			Breaks into the system and steals health records	Insufficient security
Employee			Sloppiness compromises confidentiality of health records	Insufficient training
Eavesdropper			Eavesdropping on dedicated connection	Insufficient protection of connection
System failure			System goes down during examination	Unstable connection/immature technology
Employee			Sloppiness compromises integrity of health record	Prose-based health records (i.e. natural language)
Network failure			Transmission problems compromise integrity of medical data	Unstable connection/immature technology
Employee			Health records leak out by accident — compromises their confidentiality and damages the trust in the system	Possibility of irregular handling of health records

Risk table

# CORAS – Step 3

- The last step of preparation.
- At the end of this step there are several documents that must be present and agreed upon by both customers and expert.
- Four more documents are authored:
  - **Sorting of assets** (which assets are most important)
  - **Consequence scales** (sometimes several scales are needed depending on the assets, it is easy to put numerical values for some assets and hard/impossible for others).
  - **Probability scales** (time: years, weeks, hours, etc. or probabilities: 10%,20%,1%).
  - **Risk evaluation matrix**

# CORAS – Step 3

Asset	Importance	Type
Health records	2	Direct asset
Provision of telecardiology service	3	Direct asset
Public's trust in system	(Scoped out)	Indirect asset
Patient's health	1	Indirect asset

Sorting of assets

Consequence value	Description
Catastrophic	1000+ health records (HRs) are affected
Major	100-1000 HRs are affected
Moderate	10-100 HRs are affected
Minor	1-10 HRs are affected
Insignificant	No HR is affected

Consequence scales (may need more than one)

Likelihood value	Description <sup>3</sup>
Certain	Five times or more per year (50-∞: 10y = 5-∞: 1y)
Likely	Two to five times per year (21-49: 10y = 2,1-4,9: 1y)
Possible	Once a year (6-20: 10y = 0,6-2: 1y)
Unlikely	Less than once per year (2-5: 10y = 0,2-0,5: 1y)
Rare	Less than once per ten years (0-1:10y = 0-0,1:1y)

Probability scales (may need more than one)

# CORAS – Step 3

		Consequence				
		Insignificant	Minor	Moderate	Major	Catastrophic
Frequency	Rare	Acceptable	Acceptable	Acceptable	Acceptable	Must be evaluated
	Unlikely	Acceptable	Acceptable	Acceptable	Must be evaluated	Must be evaluated
	Possible	Acceptable	Acceptable	Must be evaluated	Must be evaluated	Must be evaluated
	Likely	Acceptable	Must be evaluated	Must be evaluated	Must be evaluated	Must be evaluated
	Certain	Must be evaluated	Must be evaluated	Must be evaluated	Must be evaluated	Must be evaluated

Risk evaluation matrix

Must decide which risks have to be mitigated, and which risks can be ignored.



# CORAS – Step 4

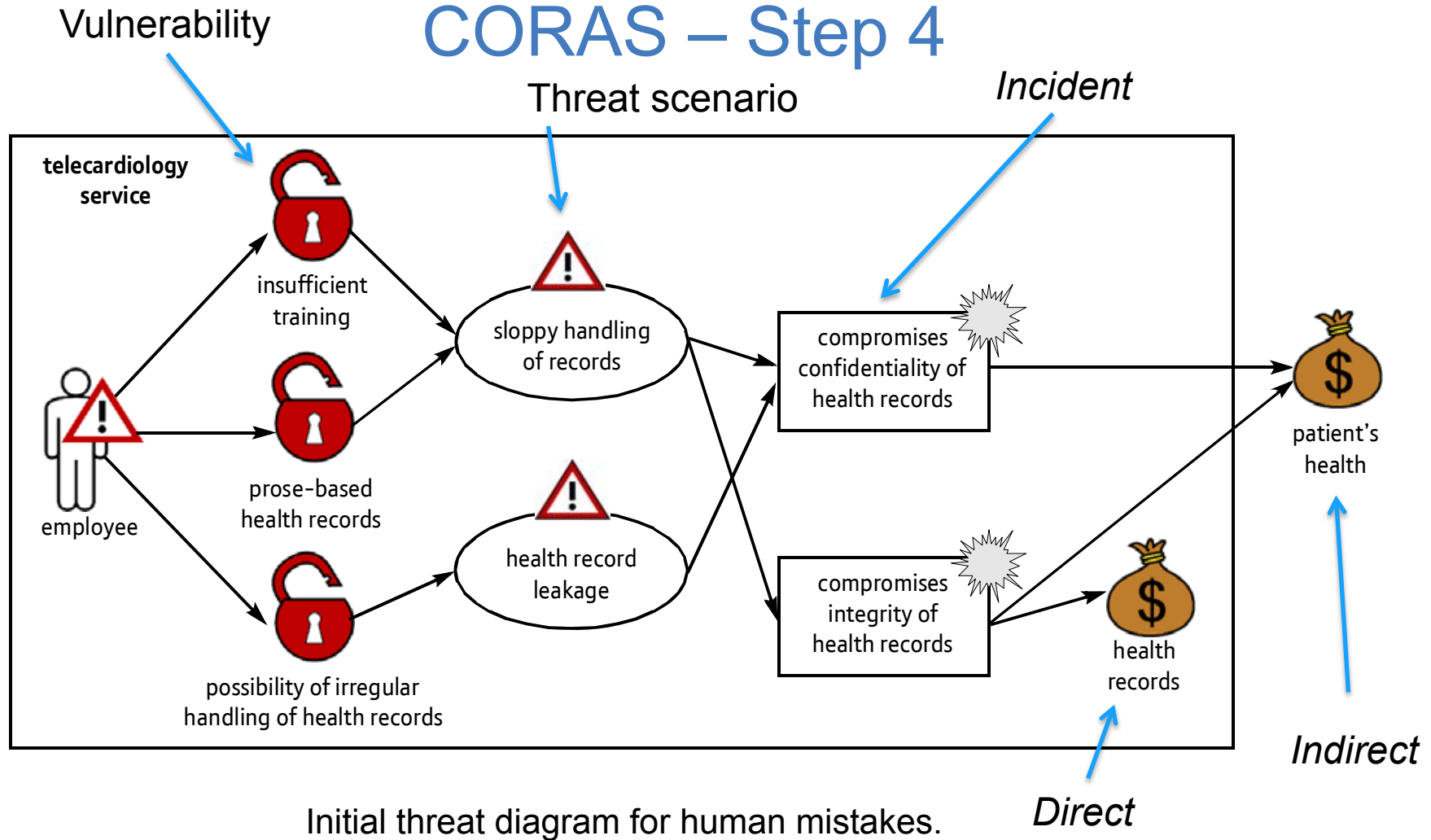
- Risk identification by structured brainstorming (**only experts**).
  - A thorough walkthrough of the system that is to be analysed.
  - People have different backgrounds and competences. (Does not necessarily have to be only IT-people).
  - The group will find more threats than a single person would find.
- Documented using **CORAS security risk modelling language**.



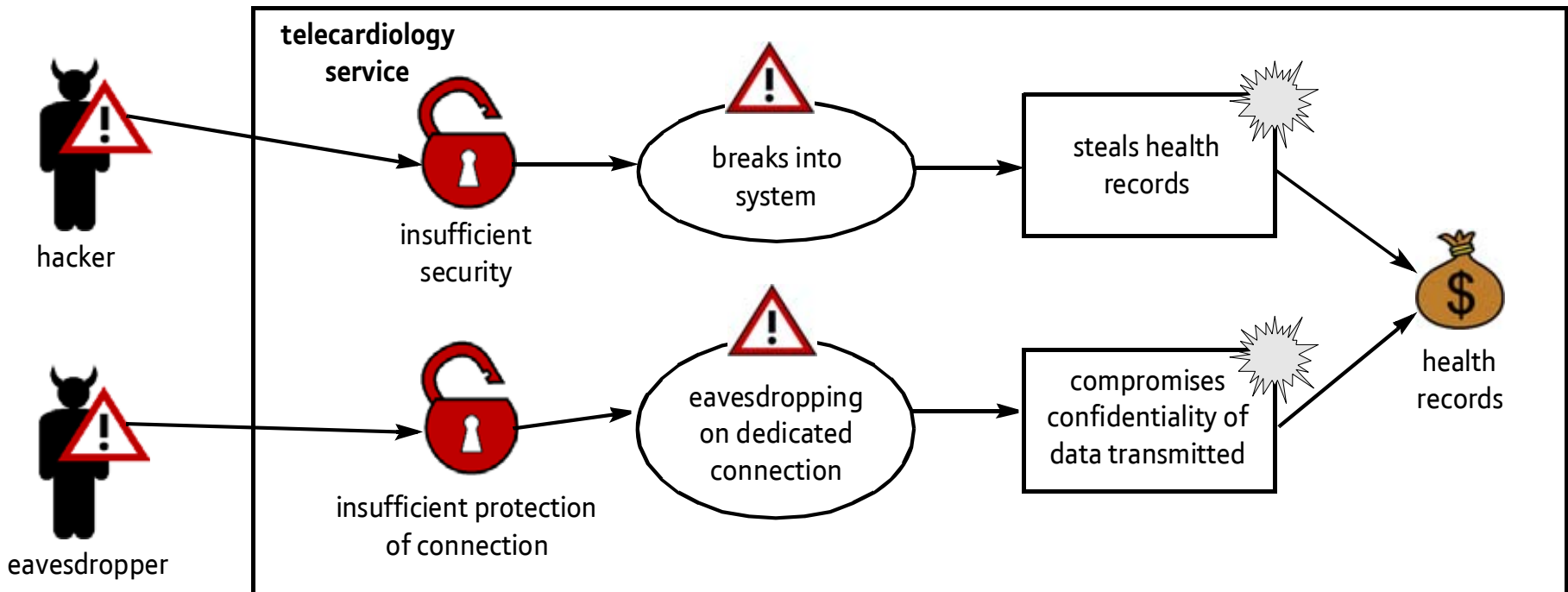
# CORAS – Step 4

- All documents created during step 1, 2 and 3 are used as input to the brainstorming session.
- One of the experts has prepared **threat scenario diagrams**
  - These initial documents are based on the threats that were pointed out by the customers in step 2.
  - These documents are updated and expanded during the session.

# CORAS – Step 4

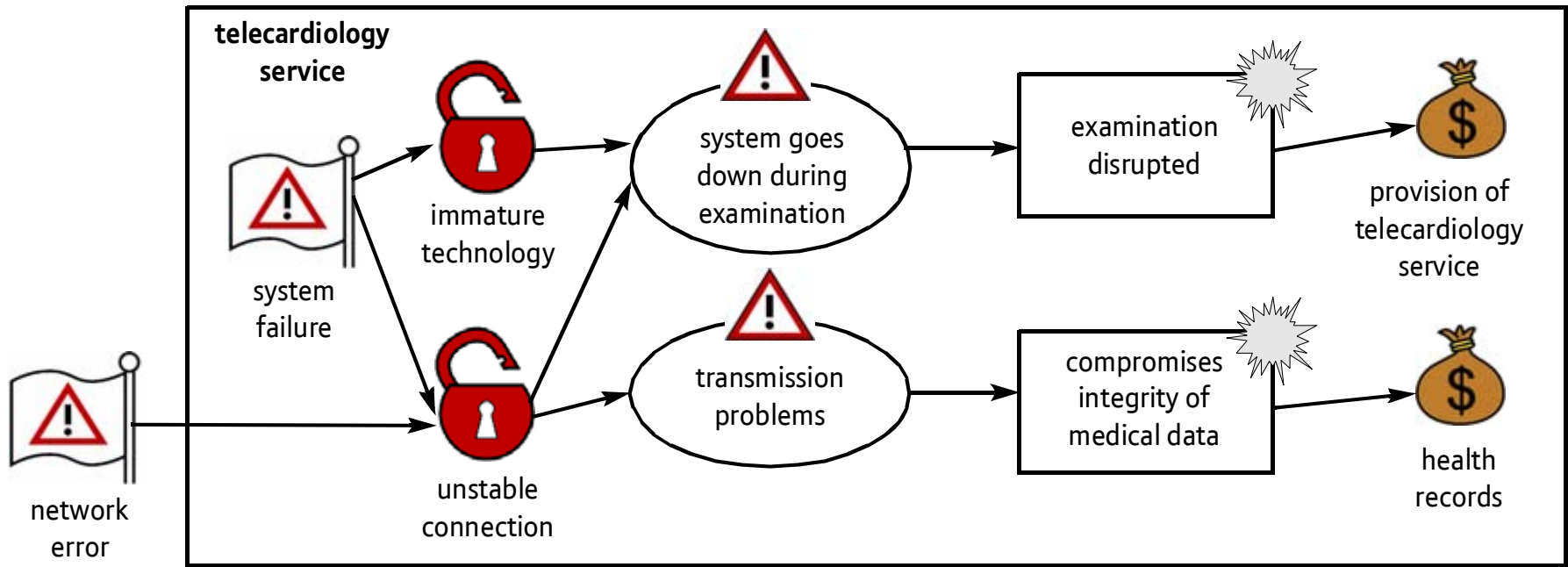


# CORAS – Step 4



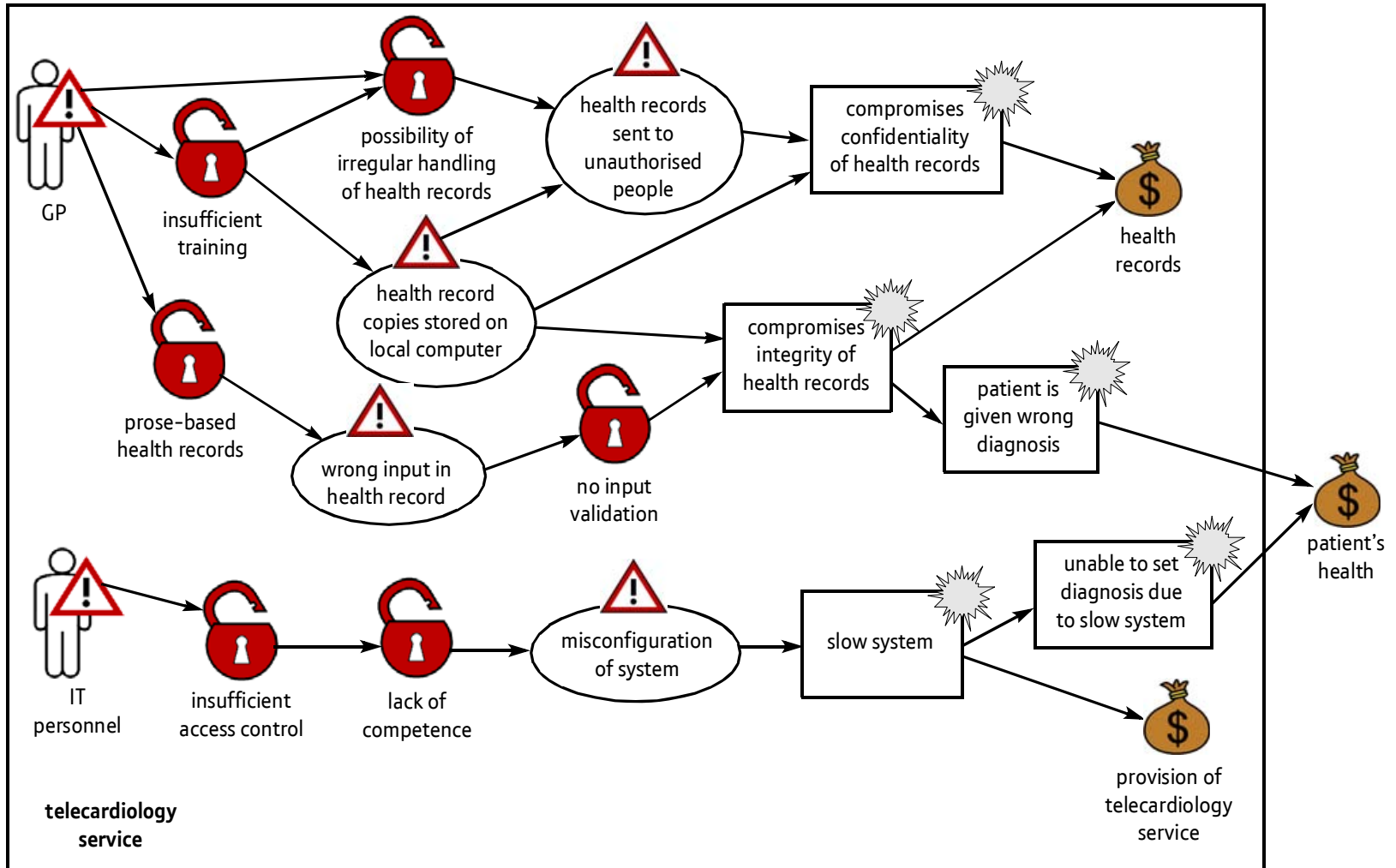
Initial threat diagram for human attacks.

# CORAS – Step 4



Initial threat diagram for "non-human" threats.

# CORAS – Step 4



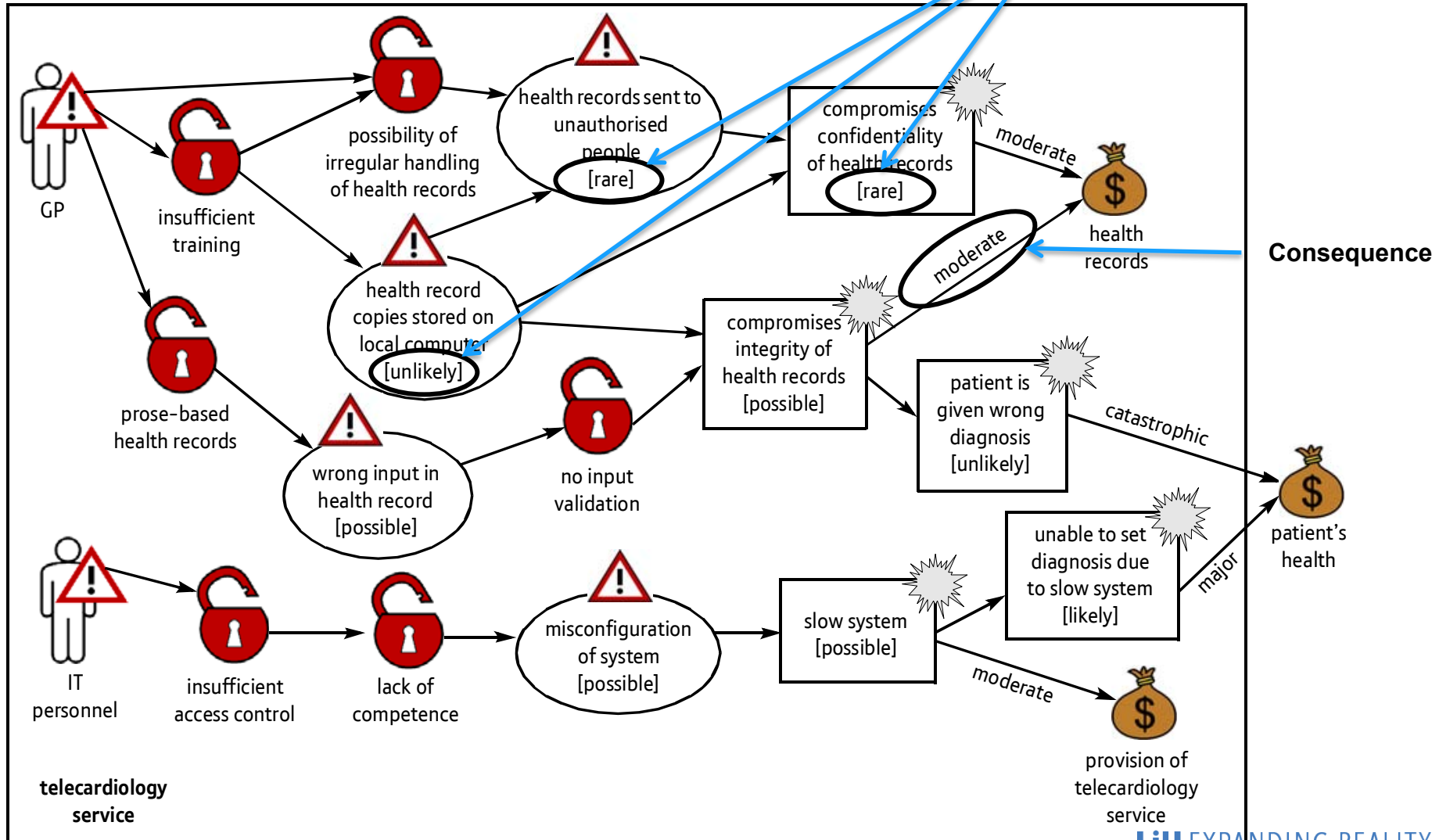
Updated and expanded threat diagram for human mistakes after the session. **LIU EXPANDING REALITY**

# CORAS – Step 5

- During another session (a workshop) the consequence and probability of every threat is estimated.
- Using the predefined scales from step 3:
  - Every participant of the workshop gives their probability and consequence estimate to every threat.
  - A consensus for estimates is found.
  - The estimated values are used together with the risk evaluation matrix to decide if the risk is worth analysing further (and finding mitigations) or if the risk should be accepted.

# CORAS – Step 5

Be careful!





# CORAS – Step 6

- Risk evaluation
  - Extract risks from the unwanted incidents (compromises confidentiality of health records CC1 = moderate / rare).
  - Place the risks in the risk evaluation matrix (defined earlier):

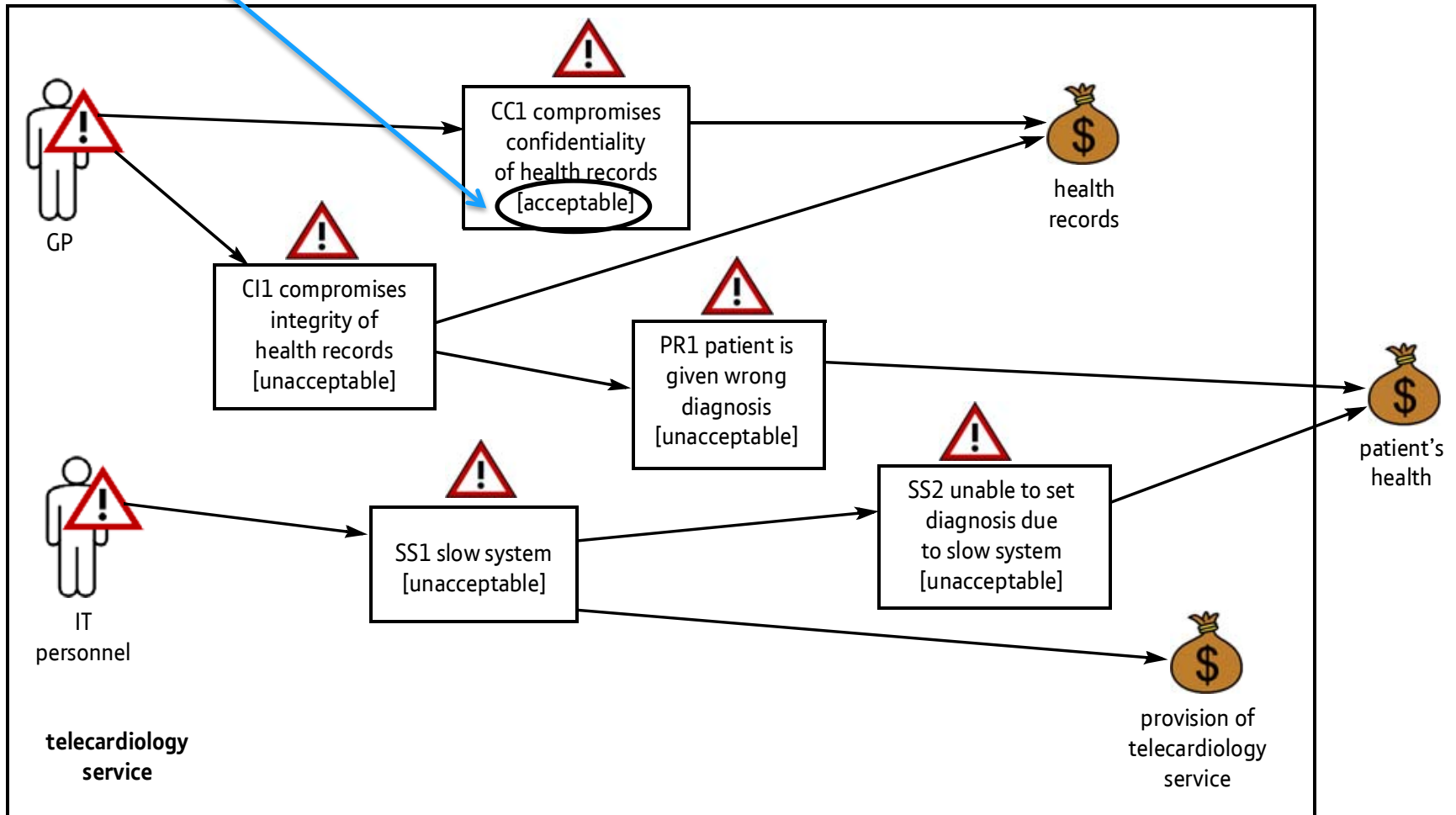
*This is outside the area that previously was defined as important.*

		Consequence				
Likelihood		Insignificant	Minor	Moderate	Major	Catastrophic
	Rare			CC1		
	Unlikely					PR1
	Possible			CI1, SS2		
	Likely				SS1	
	Certain					

- The customer must accept the matrix, and they may ask the experts to reconsider certain risk evaluations.
- A final diagram of the threats and the evaluated risk is presented.

# CORAS – Step 6

Will not be mitigated



# CORAS – Step 7

- All risks that fall into the grey area should be mitigated.
- In a new workshop, mechanisms are agreed upon that either lower the risk or consequence (or both) of a risk until it is acceptable.
- Some mechanisms can be more expensive than others, and therefore "cost-benefit" is partially weighed in.
- In the end, a plan is presented to the customers that include the mitigations.

# CORAS – Step 7

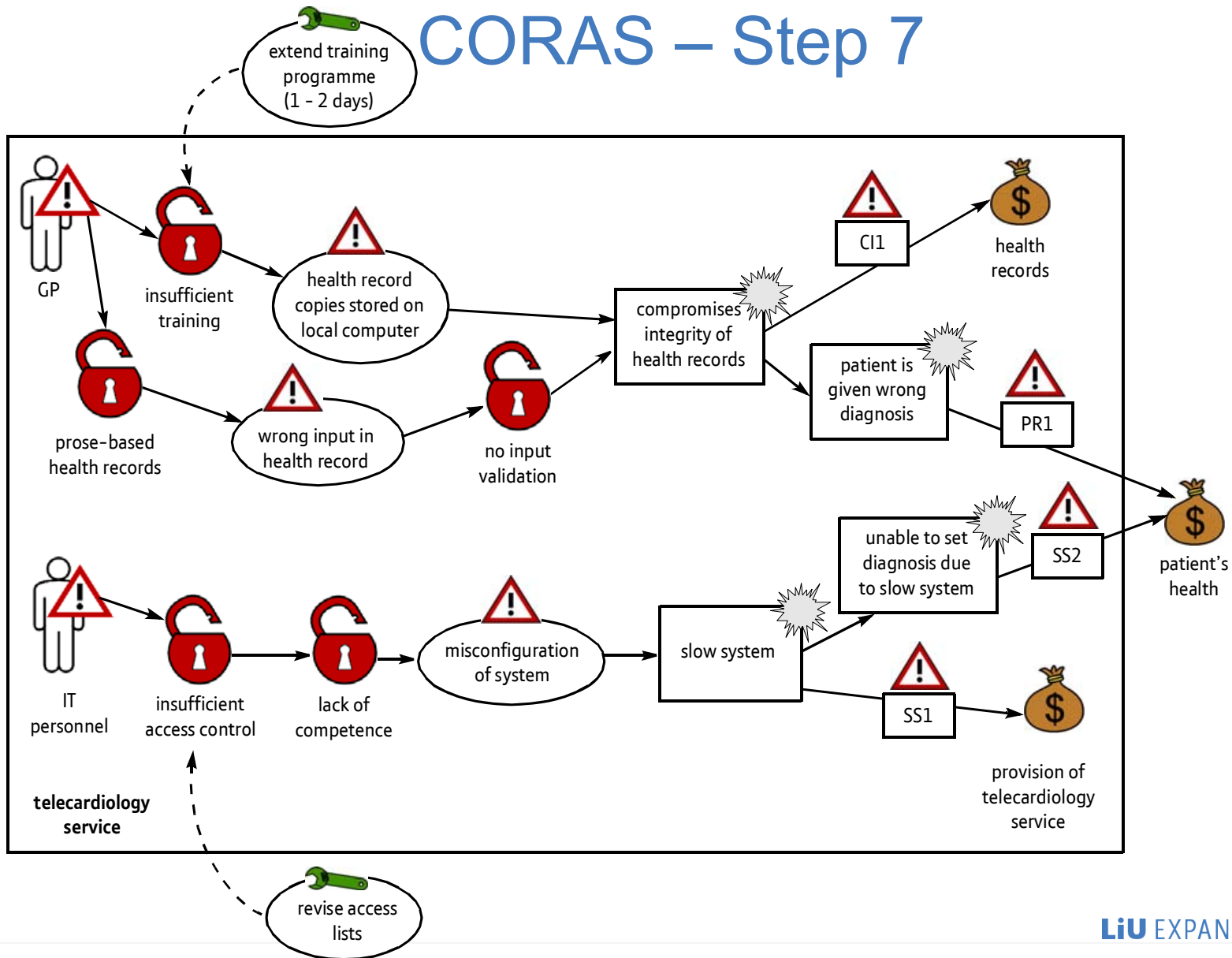
		Consequence				
Likelihood		Insignificant	Minor	Moderate	Major	Catastrophic
	Rare			CC1		
	Unlikely					PR1
	Possible			CI1, SS2		
	Likely				SS1	
	Certain					

We need to mitigate this.

**How?**

There are two options, reduce consequence (move to the left) or reduce probability (move upwards).

# CORAS – Step 7



# CORAS - Summary

- At first glance the method may feel a bit overwhelming, however once you have read and used it a couple of times it is quite straight forward.
- It is definitely a time and resource consuming method, and not all projects will benefit enough from CORAS to justify this cost.
- The strength lies in the constant connection with the customer and the use of brainstorming and workshops (where many voices and opinions can be heard).

Information Security Risk Analysis Method

**ISRAM**

# ISRAM - Introduction

- Focuses on one threat and tries to estimate the risk for this specific threat:
  - *The risk that a computer on a network gets infected by a virus.*
- Uses a specially crafted survey that is sent to users and experts.
- The answers to the survey estimates the risk of the threat (using probability and consequence).



# ISRAM – Step 1 and 2

- Step 1 – Identify the threat of interest: *virus infection*.
- Step 2 – Identify the factors that influence the *probability* and the *consequence* of the threat, and weigh these factors.

# ISRAM – Step 2

Probability factors	
The type of attachments in emails	3
Number of emails received per day	1
Number of downloaded files per day	1
The source of USB-drives	2

Weight	Explanation
3	The factor has a direct affect
2	The factor has some affect
1	The factor has an indirect affect

Consequence factors	
Backup of files	3
Physical location of files	2
Dependency on applications	1

- The number of factors for probability does not have to be the same as for consequence.
- More weights can be used, but it is hard to discern the difference between 3 and 4 on a 10 grade scale.
- The definition of the weights is not strictly defined.

# ISRAM – Step 3

- Step 3 – Convert factors to questions, create response options and give each option a score.

Question	A	B	C	D
How many emails do you receive per day?	0-10 (1)	11-30 (2)	31-40 (3)	41+ (4)
Where do you get USB-drives from?	From the company (0)	Bring them from home (4)		
How often do you backup your files?	Every day (1)	Every week (2)	Never(4)	

- The scores for the options are in parenthesis (they are removed when the survey is sent)
- The questions regarding probability and consequence are in the same survey.
- The possible scores for options are 0 through 4.

# ISRAM – Step 4

- Calculate the minimum and the maximum number of points that the questions regarding **probability** can give.
- Calculate the minimum and the maximum number of points that the questions regarding **consequence** can give.
- Create intervals (bins) such that scores can be translated to a scale of 1 to 5.

Points	Qualitative scale	Quantitative scale
29-48	Very low probability	1
49-68	Low probability	2
69-88	Medium probability	3
89-108	High probability	4
108-128	Very high probability	5

Poäng	Qualitative scale	Quantitative scale
47-68	Negligable consequence	1
69-90	Small consequence	2
91-111	Increased consequence	3
112-133	Serious consequence	4
134-160	Very serious consequence	5

# ISRAM – Step 4

- Create the final risk quantification table

Risk = Probability x Consequence					
	1: Negligible	2: Small	3: Increased	4: Serious	5: Very serious
1: Very low	1: Very low	2: Very low	3: Very low	4: Low	5: Low
2: Low	2: Very low	4: Low	6: Low	8: Medium	10: Medium
3: Medium	3: Very low	6: Low	9: Medium	12: Medium	15: High
4: High	4: Low	8: Medium	12: Medium	16: High	20: Very high
5: Very high	5: Low	10: Medium	15: High	20: Very high	25: Very high

# ISRAM – Step 5 and 6

- Step 5 – Complete the survey. It can be sent to users of the computers that are the subject of the analysis, and/or other experts.
- Step 6 – Use an equation to calculate the a value that represents risk (based on the factors for probability and consequence). Use the result of the equation in the risk evaluation table to get the final risk estimation.

## ISRAM – Step 6

$$Risk = \left( \frac{\sum_{n=1}^N [T_s(\sum_{i=1}^I \alpha_i s_{i,n})]}{N} \right) \left( \frac{\sum_{n=1}^N [T_k(\sum_{j=1}^J \beta_j k_{j,n})]}{N} \right)$$

- $N$  = number of respondents
- $I$  = number of questions regarding probability
- $J$  = number of questions regarding consequence
- $\alpha_i$  = the weight given to probability question  $i$
- $s_{i,n}$  = score for the option that respondent  $n$  choose for probability question  $i$
- $\beta_j$  = the weight given to consequence question  $j$
- $k_{j,n}$  = score for the option that respondent  $n$  choose for consequence question  $j$
- $T_s$  a function that translates an integer to the probability scale 1 through 5
- $T_k$  a function that translates an integer to the consequence scale 1 through 5

## ISRAM – Step 6

$$Risk = \left( \frac{\sum_{n=1}^N [T_s (\sum_{i=1}^I \alpha_i s_{i,n})]}{N} \right) \left( \frac{\sum_{n=1}^N [T_k (\sum_{j=1}^J \beta_j k_{j,n})]}{N} \right)$$

Respondent	Sum of probability questions	T <sub>s</sub>	Sum of consequence questions	T <sub>k</sub>
1	94	4	103	3
2	74	3	136	5
Mean: 3.5			Mean: 4	
Risk = 3.5 * 4 = 14 which is between medium and high risk, but closer to high risk				



# ISRAM – Step 7

- Step 7 – Evaluation of results
- The final risk estimation is the important outcome of the method.
- The estimation can be used to decide if new policies should be made or new mechanisms should be introduced.
- However, at the same time a lot of information has been gathered about the use of the systems analysed.
- For instance, it may be possible to get an idea of how often users update their software, if they are using administrative accounts properly, etc.
- This extra information is valuable when deciding between mitigations.

# ISRAM - Summary

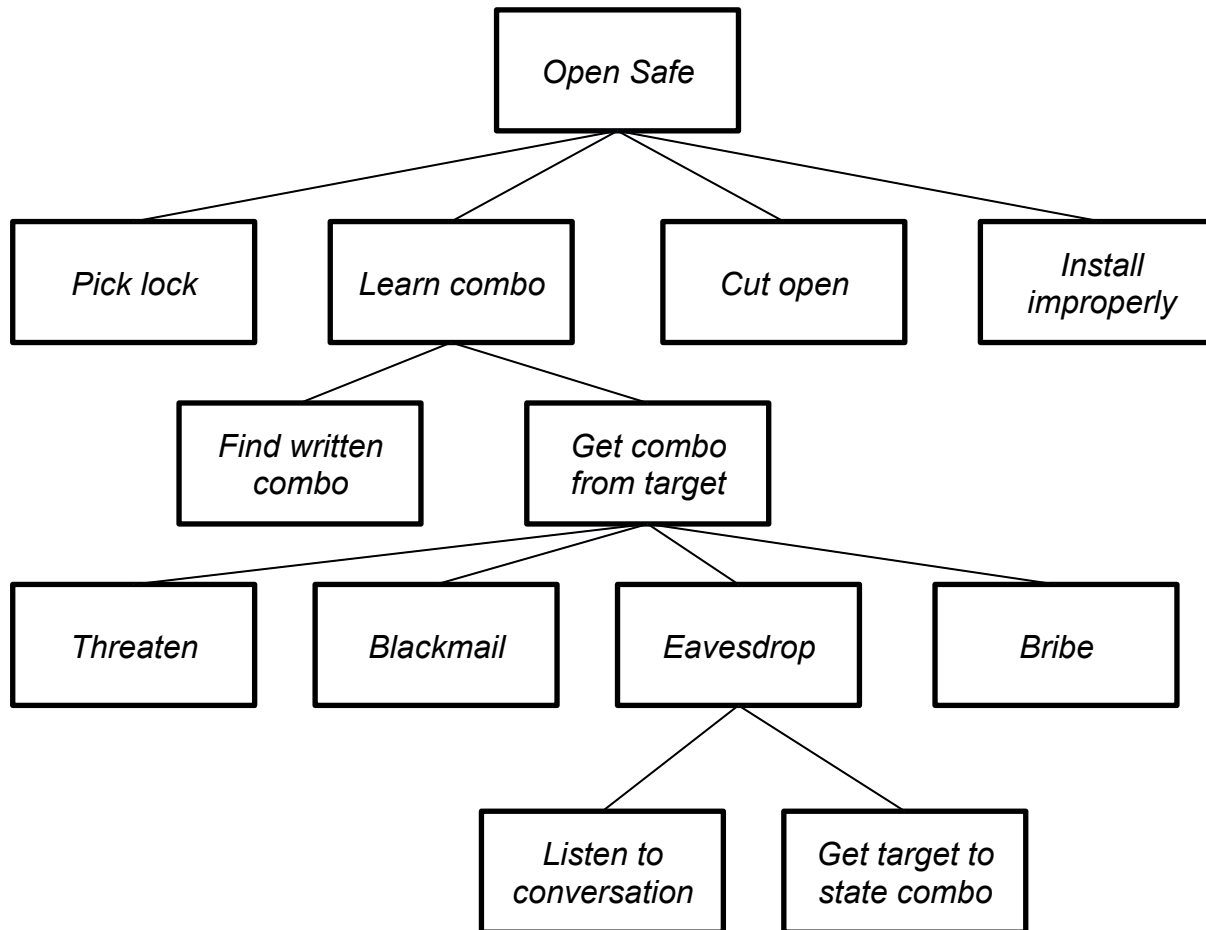
- ISRAM is useful when you want to estimate the risk for one specific threat.
- It only requires one person to administer the analysis (if there already are respondents). (It can be advantageous for more people to help with the choice of factors and weights).
- The outcome of the analysis is very dependent on the factors identified and the weights chosen. You cannot get answers to questions you did not ask.

# ATTACK TREES

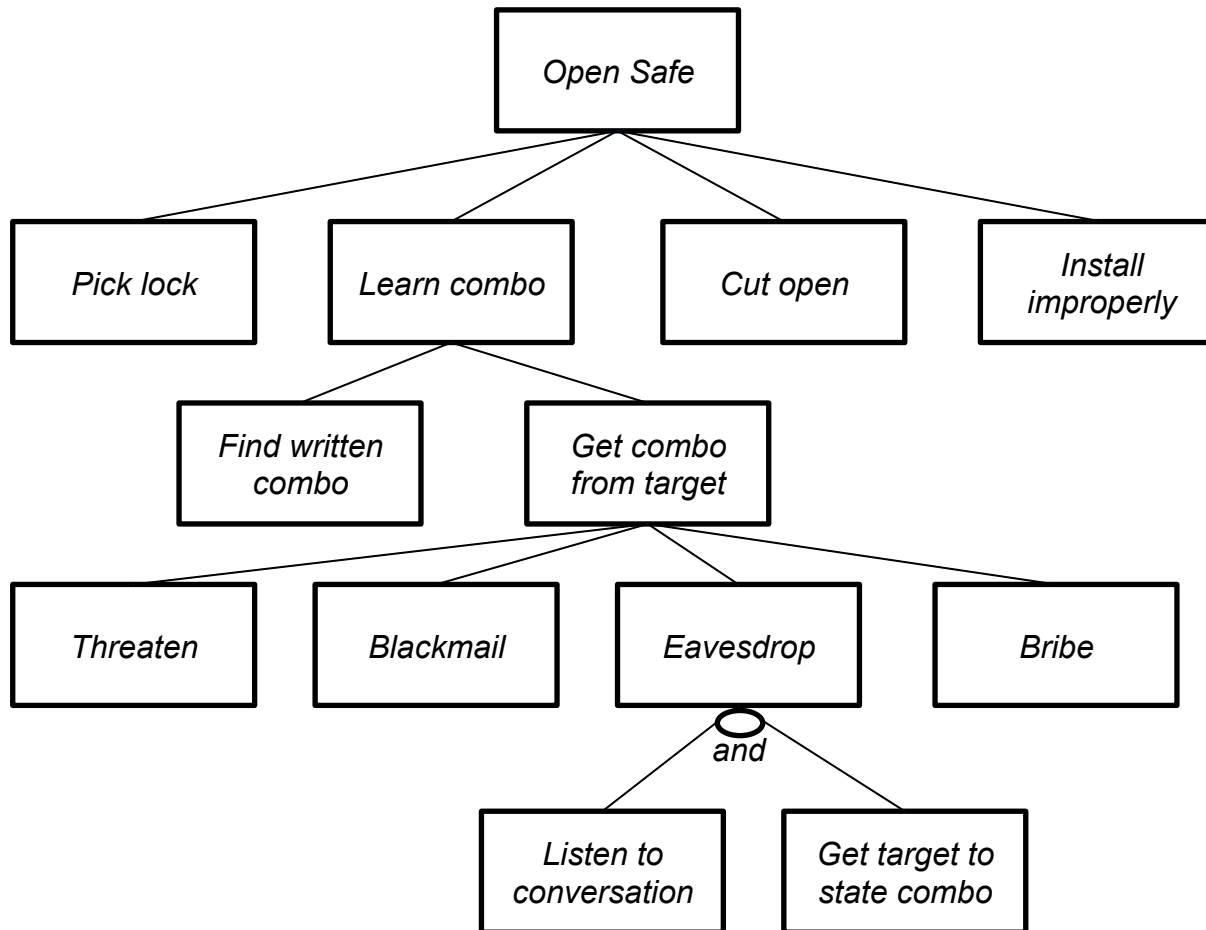
# Attack trees

Represent attacks against the system in a tree structure, with the goal as the root node and different ways of achieving that goal as leaf nodes.

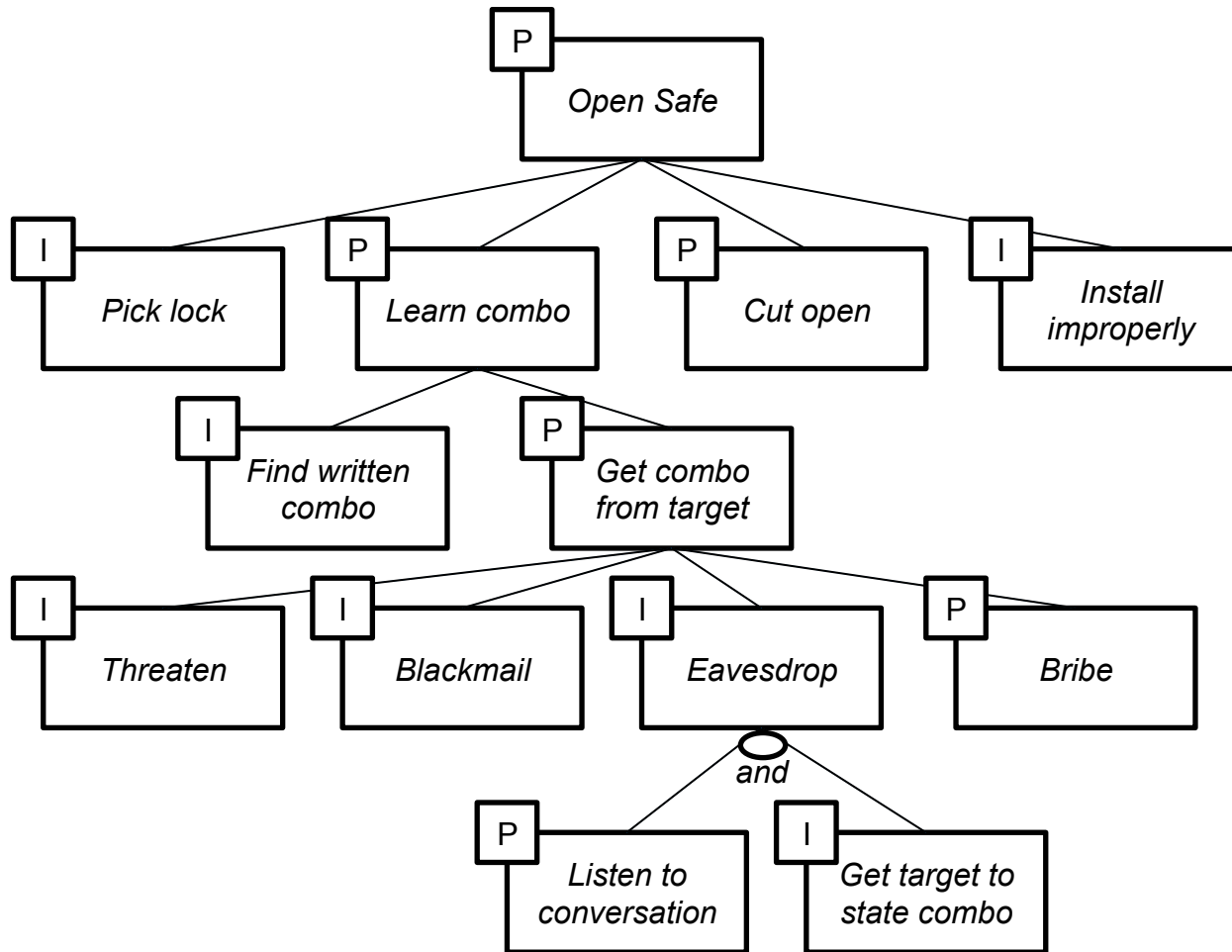
# Attack Trees



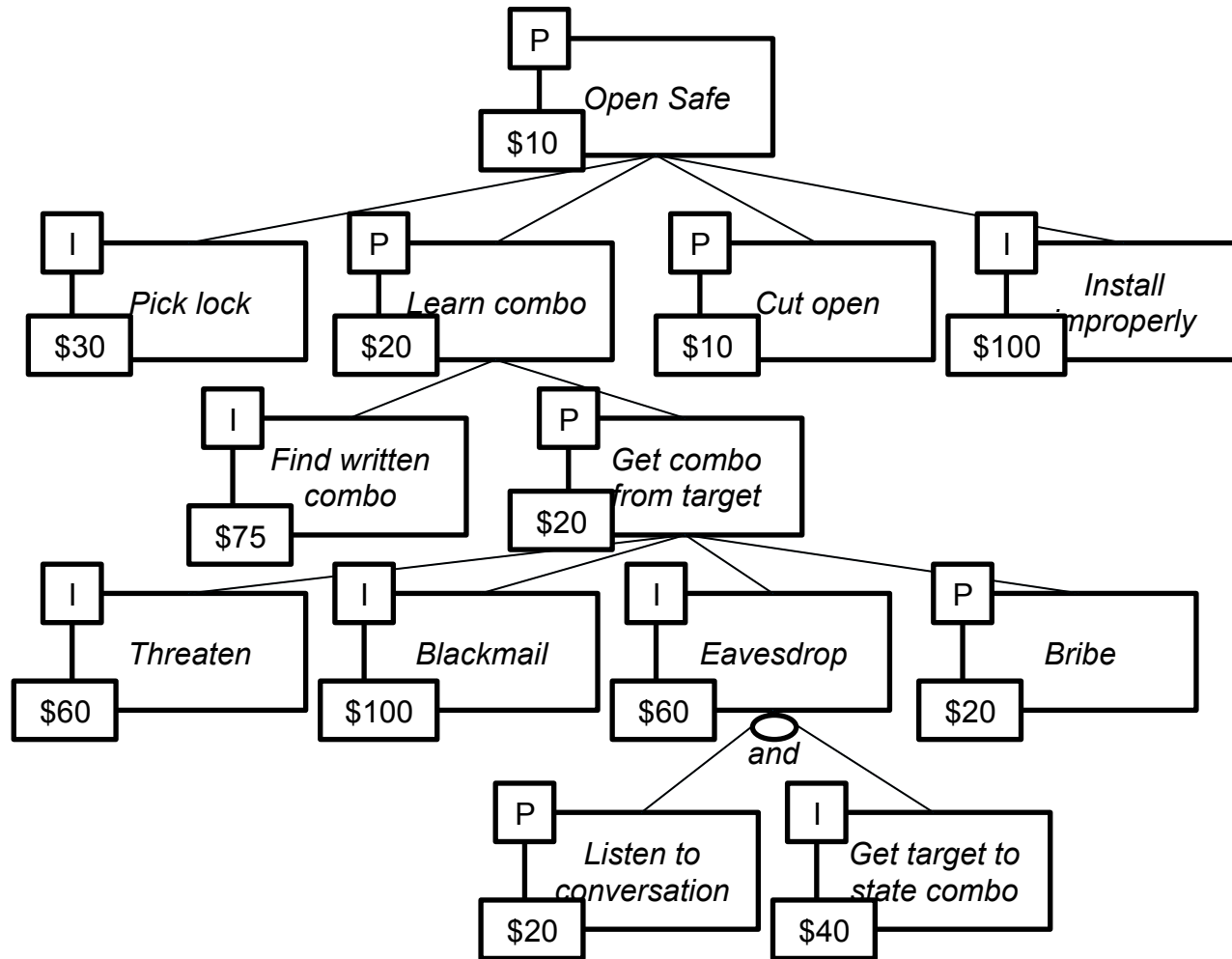
# Attack Trees



# Attack Trees



# Attack Trees





# Attack Trees

- We can annotate the attack tree with many different kind of Boolean and continuous values:
  - “Legal” versus “Illegal”
  - “Requires special equipment” versus “No special equipment”
  - Probability of success, likelihood of attack, etc.
- Once we have annotated the tree we can query it:
  - Which attacks cost less than \$10?
  - Legal attacks that cost more than \$50?
  - Would it be worth paying a person \$80 so they are less susceptible to bribes? (In reality you need to also consider the probability of success)

# Attack Trees

- First you identify possible attack goals.
- Each goal forms a separate tree.
- Add all attacks you can think of to the tree.
- Expand the attacks as if they were goals downwards in the tree.
- Let somebody else look at your tree, get comments from experts, iterate and re-iterate.
- Keep your trees updated and use them to make security decisions throughout the software life cycle.

# Risk analysis - Summary

## **Risk analysis is a cornerstone:**

- Development of new software may require a risk analysis prior to defining requirements and once the software has been developed.
- Expansion of a company to a new office may require a risk analysis of physical security and new business continuity planning.
- Changing the topology of a network system may require a risk analysis of how to break down the system in different security levels.

• ...

# Risk analysis - Summary

- Many methods exists – need to choose one that fits the current situation and available resources.
- CORAS, ISRAM and Attack Trees all have their advantages and disadvantages.
- Risk analysis is hard, *really hard*, and a successful analysis is dependent on the experts.
- Limiting the analysis, getting help from others and being organised are important common factors for any successful risk analysis.



# Linköpings universitet

expanding reality

[www.liu.se](http://www.liu.se)