# Security Risk Analysis

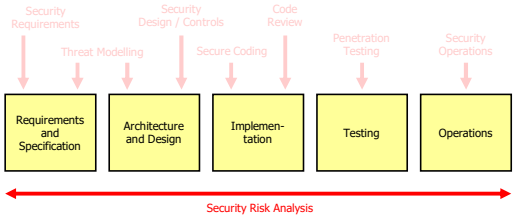
Prof. Dr. Marc Rennhard, Dr. Stephan Neuhaus
Institut für angewandte Informationstechnologie InIT
ZHAW School of Engineering
rema | neut @zhaw.ch

## Content

- Introduction to Security Risk Analysis
- The overall process to perform security risk analysis
- NIST 800-30: A relatively simple methodology to rate the risk of threats and vulnerabilities but that works quite well in practice – in particular if performed by an experienced person
- OWASP Risk Rating Methodology: A more structured approach towards rating the risk of threats and vulnerabilities that is well-suited for beginners, but also for experienced persons to increase confidence in the risk ratings
- Risk Mitigation: What can we do if we identify risks that are too high

# Goals

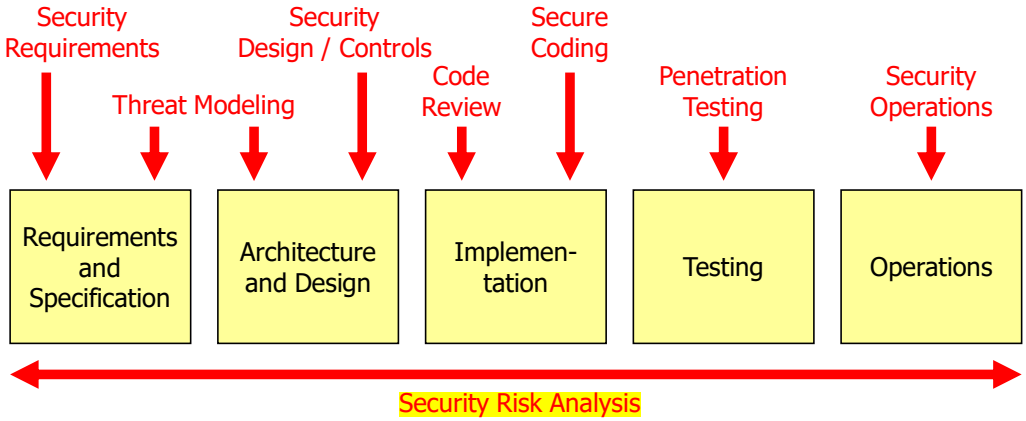
- You understand the purpose of security risk analysis during the secure software development process
- You know the overall security risk analysis process and can apply it
- You know the risk rating methodologies according to NIST 800-30 and OWASP and can apply them to rate the risk of threats and vulnerabilities
- Security activity covered in this chapter:



# Introduction to Security Risk Analysis

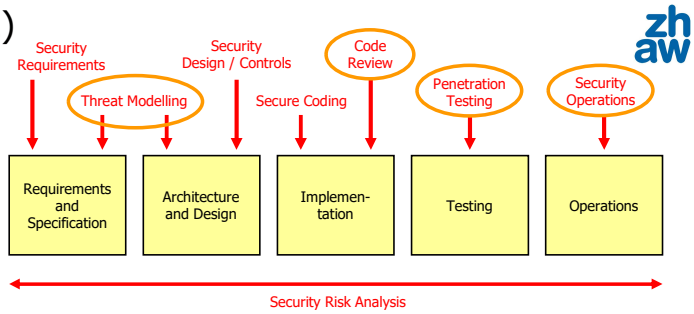
# Security Risk Analysis (1)

- The purpose of security risk analysis is to rate the risk (the criticality) of security threats and vulnerabilities
  - Based on this risk rating, we can decide whether a specific threat / vulnerability should be addressed or not
- In the secure development lifecycle, security risk analysis is «drawn» as a horizontal activity as it complements many of the other activities



## Security Risk Analysis (2)

Some examples where security risk analysis complements the other security activities:




- During threat modeling, to rate the risk of the identified threats
- During code reviews, to rate the risk of a discovered security bug
- During penetration testing, to rate the risk of a discovered vulnerability
- During operations, to rate the risk of operational issues
  - E.g., to decide whether redundant systems should be used and how frequently backups should be made

- Remark: Depending on the security activity, we rate different «things»: threats, bugs, vulnerabilities, operational issues... For the sake of simplicity, we will always talk about «rating the risk of vulnerabilities» in the remainder of this chapter – but keep in mind that everything we are discussing also applies to threats, bugs and operational issues.



**Risk Analysis vs. Security Risk Analysis**

Risk analysis in general should be an essential activity during any software project and goes beyond security risks, e.g.:

- Technical risks (are there technical risks that may endanger the project from working as specified?)
- Personnel risks (are there single persons in the project team that would be difficult to replace?)

With security risk analysis, the focus is on security risks, i.e., risks that could endanger one or more of the security goals: confidentiality, integrity, availability.

## Quantitative Risk Analysis

- One way to do risk analysis is to express risk as financial loss, e.g., as the amount of money lost during one year
  - This is identified as Annualized Loss Expectancy (ALE)
- It is calculated as follows:
  - SLE: Single Loss Expectancy
  - ARO: Annualized Rate of Occurrence

$$ALE = SLE \times ARO$$

- Example:
  - Assume we expect that our user database is compromised every 5 years and whenever this happens, this costs CHF 100'000
  - ARO = 1/5, SLE = CHF 100'000 → ALE = 20'000 CHF/year

+ Advantages: Shows the costs of risks and the maximum amount of money one should spend to fix the problem

− Disadvantages: Very difficult to make reasonable guesses for SLE and ARO, e.g.
  - What's the financial loss of a website defacement?
  - How many times will an SQL injection vulnerability be exploited in a year?



**Single Loss Expectancy**: Costs of a single successful attack

**Annualized Rate of Occurrence**: How many times per year will a successful attack happen

**Quantitative Risk Analysis**

In general, quantitative risk analysis works quite well when incidents happen, for instance, because a component of a machine breaks or because of natural disasters that happen with a certain regularity. For such cases, one has typically good data based on previous experience. This allows to compute e.g., the risk of a hurricane hitting a city or the risk that a manufacturing process is stopped because some machine breaks down. But with IT risks, we have the big unknown of the attacker, which extremely hard to quantify.

# Qualitative Risk Analysis

- As a result, a qualitative risk analysis is often preferred in security risk analysis
- Qualitative risk analysis works as follows:
  - For every security vulnerability, one estimates the likelihood of a successful attack and the resulting impact of the attack
  - Likelihood and impact are not expressed in numbers, but in a relatively small number of levels (e.g., 3 or 5)
    - More than 5 levels does hardly make sense due to the qualitative nature of the approach
- Based on the determined levels for likelihood and impact, a risk value is «calculated», using also only a few levels
- Risks above a certain level usually must be reduced with appropriate countermeasures

# Security Risk Analysis – The Process

# Security Risk Analysis – The Process (1)

We identify 4 steps in the (qualitative) security risk analysis process:

1. Identify security vulnerabilities that should be risk-rated, e.g., by doing threat modeling, penetration testing, code reviews,...
   - As a basis for the following steps, provide the following information for the identified vulnerabilities:
     - The actual attack that is used
     - The threat agent (the attacker who may carry out the attack)
     - The vulnerabilities that are involved / exploited
     - Security controls (if any) that are already in place that help protecting from the attack

This step is covered in other chapters in this module:

- Web Application Security Testing (chapter 6)
- Security Requirements Engineering and Threat Modeling (chapter 9)
- Security Testing Tools (lab 5)

## Security Risk Analysis – The Process (2)

2. For each vulnerability, estimate the likelihood of a successful attack and its business impact
3. For each vulnerability, determine the risk based on likelihood and impact
4. Risk mitigation: For each vulnerability and associated risk, decide which actions should be taken (if necessary), e.g., additional security requirements or specific security controls

Covered in this chapter

The corrective measures are covered in other chapters in this module:

- Software Security Errors and Java Security (chapters 3 & 4)
- Fundamental Security Principles (chapter 5)
- Developing Secure Web Applications (chapters 7 & 8)
- Security Requirements Engineering and Threat Modeling (chapter 9)
- Module IT-Sicherheit



**Business Impact**

Whenever possible, consider the business impact of an attack, i.e., what negative business consequences will occur if a particular attack happens. E.g., will there be an expected loss in profit, a loss in customers and so on. Sometimes, one also talks about technical impact of an attack, e.g., how many systems will no longer work for how long if the attack happens. However, it's often not obvious from the technical impact what this really means for business (e.g., are these systems that are not working business-critical or not?), and as result of this, one should always try to determine the business impact of a successful attack (i.e., ask yourself what the downtime of these systems truly means for the business). This is especially important if the recipient of the risk analysis is senior management (executive level). For them, it's always the business risk (based on the business impact) what justifies investments in fixing security problems.

Mature companies often have an asset classification guide and/or a business impact reference to help formalize what is important to their business. These standards can help you – when doing a risk analysis – to focus on what's truly important for business. If these aren't available, then talk with people who understand the business to get their take on what's important.

## Steps 2 and 3 of the Security Risk Analysis Process

- To core of the security risk analysis process are steps 2 and 3
  - Determine likelihood, impact and risk of vulnerabilities
  - There are plenty of guidelines/standards available from various organizations that can be used as a basis to determine these values
  - Here, we look at two specific guidelines/standards to do these steps
- NIST 800-30: Risk Management Guide for Information Technology Systems
  - A relatively simple methodology that doesn't provide lots of guidance to pick appropriate values for likelihood and impact
  - But it works quite well in practice – in particular if performed by an experienced person
- OWASP Risk Rating Methodology
  - Basically an extension of NIST 800-30 that uses a more structured approach to pick appropriate values for likelihood and impact
  - Well-suited for beginners, but also for experienced persons to increase confidence in the risk ratings

# Security Risk Analysis – NIST 800-30



**NIST** = US American *National Institute of Standards and Technology*

## NIST 800-30 – Likelihood Determination

- NIST 800-30 uses three levels to rate likelihood and impact: *High*, *Medium* or *Low*
- Basically, the standard just provides some definitions that should help to choose the right level
- To determine the likelihood of a successful attack, the levels are defined as follows:

| Likelihood Value | Definition |
|---|---|
| High | The threat agent is highly motivated and sufficiently capable, and controls to prevent the risk from occurring are ineffective. |
| Medium | The threat agent is motivated and capable, but controls are in place that may impede successful materialization of the risk. |
| Low | The threat agent lacks motivation or capability, or controls are in place to prevent or at least significantly impede the risk from occurring. |



**NIST 800-30: Risk Management Guide for Information Technology Systems**

*https://csrc.nist.gov/publications/detail/sp/800-30/archive/2002-07-01*

Note that there's a newer version of this standard available (*https://csrc.nist.gov/publications/detail/sp/800-30/rev-1/final*), which uses a much more detailed and also more complicated approach to pick reasonable values for likelihood and impact. One can of course also use this updated version, but experience shows that the more «precise» a standard tries to be by using various checklists and a complicated system to determine the actual likelihood and impact values, the less likely it will be used correctly and consistently in practice. Therefore, we continue to use here the previous version due to its simplicity and due to the fact that in practice, such simple approaches are often preferred over more complicated ones, in particular if the complicated approaches do not really work significantly better (as is the case here).

**«...controls are in place that may impede successful materialization of the risk»**

*impede* = to delay or stop the progress or movement of sb./sth.

An example for this could be a web application firewall (WAF) that protects a vulnerable web application (that contains, e.g., an SQL injection vulnerability). This certainly provides some protection to make attacks more difficult, but it's definitely not as good as truly fixing the issue in the web application as an attacker may find a way to defeat the WAF (e.g., by using an attack variation that is not detected by the WAF) or he may first try to compromise another system, from which the web application may be attacked directly, without the WAF in between.

# NIST 800-30 – Impact Determination

- To determine the impact of a successful attack, the levels are defined as follows:

| Magnitude of Impact | Definition |
|---|---|
| High | Exercise of the vulnerability (1) may result in the highly costly loss of major tangible assets or resources; (2) may significantly violate, harm, or impede an organization's mission, reputation, or interest; or (3) may result in human death or serious injury. |
| Medium | Exercise of the vulnerability (1) may result in the costly loss of tangible assets or resources; (2) may violate, harm, or impede an organization's mission, reputation, or interest; or (3) may result in human injury. |
| Low | Exercise of the vulnerability (1) may result in the loss of some tangible assets or resources or (2) may noticeably affect an organization's mission, reputation, or interest. |

## NIST 800-30 – Risk Determination

- The overall risk (which uses five levels) is then determined according to this risk matrix:

| | Impact | | |
|---|---|---|---|
| Likelihood | Low | Medium | High |
| High | Medium | High | Critical |
| Medium | Low | Medium | High |
| Low | Info | Low | Medium |

- Descriptions of the risk levels:
  - *Critical* – absolute need for corrective measures; an existing system should no longer be operated until corrective actions are implemented
  - *High* – strong need for corrective measures; an existing system may continue to operate if really needed, but corrective actions should be implemented as soon as possible (e.g., days to at most a few weeks)
  - *Medium* – indicates that corrective actions are needed and should be implemented within a reasonable time (e.g., next major release)
  - *Low* – indicates that the system's decision authorities must determine whether corrective actions are needed or decide to accept the risk
  - *Info* – the risk can be accepted



**Adjustments made to NIST 800-30**

We made a small adjustment to NIST 800-30. The original standard just uses three risk levels *high*, *medium* and *low*. Here, we changed *high* at the top right to *critical* and *low* at the bottom left to *info*. This adjustment is reasonable because most risk rating methodologies use five levels these days, because the updated version of NIST 800-30 also uses the two additional risk ratings at the top and bottom end of the spectrum, and because the other values of the risk table were not changed. The main reason for this change is to align the standard with the OWASP risk rating methodology, which also uses five risk levels.

## NIST 800-30 – Exercise (1)

- A tax consulting company with 1'000 employees develops its own financial accounting system (contains all financial data: invoices, salary,…)
- 10 financial accountants will have access to read / modify data
- The core components of the system have already been designed and you are given the task to do some threat modeling to find security-critical vulnerabilities before implementation begins
- You identify this threat: Accountants may modify the salary data in non-legitimate ways and can deny having done this
  - Based on this, they could offer their co-employees to «increase their salaries a bit» – of course for financial compensation
- You discover that there's absolutely no logging / auditing mechanism to track changes made by the accountants → vulnerability!
- As a side note, the spirit among the employees is not so great due to recent salary cuts…
- Your task: Determine the risk of this vulnerability based on NIST 800-30



**STRIDE**

Applied to STRIDE terminology, the threat above is a repudiation threat because authorized users can perform malicious actions and state later that «they didn't do it» as the actions are not logged.

## NIST 800-30 – Exercise (2)

- Likelihood can be rated *Medium*
  - The employees are somewhat motivated to carry out such an attack (*Low* to *Medium*)
  - Controls to discover who has performed the attack are missing, which means there's a high probability the attack will be carried out (*High*)
- Impact can be rated *Medium*
  - There may be some financial loss but likely not too much (*Low*)
  - Reputation may take a serious hit if this becomes public (*High*)
  - No injuries expected (*Low*)
- So the risk of this vulnerability is *Medium* and we need to do something «within a reasonable time»…

| | Impact | | |
|---|---|---|---|
| Likelihood | Low | Medium | High |
| High | Medium | High | Critical |
| Medium | Low | Medium | High |
| Low | Info | Low | Medium |




**Reputation Damage**

A tax consulting company is very dependent on its reputation. If it becomes publicly known that such an internal attack has taken place, it is very likely that the company will lose some clients.

**Why not Impact *High*?**

One could also come to the conclusion that the impact should be rated *High* because the NIST 800-30 description says *«Exercise of the vulnerability (1) may result in the highly costly loss of major tangible assets or resources; (2) may significantly violate, harm, or impede an organization's mission, reputation, or interest; or (3) may result in human death or serious injury.»* - so it's an OR-expression. Still, it is reasonable in this case to choose *Medium* impact only because the immediate financial damage is not very drastic (too much salary will be paid, but that won't be very significant compared to the total salary sum) and the reputation damage will result in monetary loss only over time (mid- to long-term), so there's time for compensation measures by the company (e.g., an advertisement offensive to restore trust in their company). If the immediate financial damage would be big as well (even threatening the existence of the company), then *High* impact should be chosen.

## NIST 800-30 – Discussion

- NIST 800-30 is a fairly simple methodology, but that works well in practice and that has several benefits:
  - It's fast: It's not uncommon in practice that you have to rate 100 threats / vulnerabilities (e.g., after doing threat modeling in a complex system) – NIST 800-30 allows to rate them relatively quickly
  - It usually leads to reasonable risk ratings, in particular if performed by an experienced person – because such persons usually have a good understanding of the ratings *High*, *Medium* and *Low*

- But it also has some limitations:
  - The provided guidelines to assess likelihood and impact are minimal – especially beginners often feel «a bit lost» and are unsure whether they have picked the right values
  - Even experienced persons sometimes have to rate threats / vulnerabilities where they are unsure when using NIST 800-30 (e.g., a vulnerability or attacker type or a specific system «they have never seen before»)
  - In such situations, it makes sense to use a more structured methodology that provides more guidelines → e.g., OWASP Risk Rating Methodology

# Security Risk Analysis – OWASP Risk Rating

## OWASP Risk Rating

- OWASP Risk Rating is basically an extension of NIST 800-30 that provides more guidance to determine likelihood and impact
  - Uses the same risk matrix and can therefore be used together with NIST 800-30 (e.g., use NIST 800-30 to rate 80% of the threats / vulnerabilities and use the OWASP method for the other 20% where you are unsure)
- With OWASP Risk Rating, one does not determine likelihood and impact directly, but instead one has to rate a set of factors
  - Each factor is rated from 0 – 9 (higher = bigger likelihood / impact)
  - For each factor, examples are provided to help choosing reasonable values
  - The ratings of the factors are then used to calculate the likelihood and impact, which again determine the resulting risk
- To determine the likelihood, eight factors are used:
  - Four threat agent factors to rate, e.g., skills and motivation of the attacker
  - Four vulnerability factors to rate, e.g., ease of exploitation of a vulnerability and difficulty to detect a successful attack
- To determine the impact, four factors are used to rate, e.g., loss of money or reputation



**OWASP Risk Rating**

*https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology*

**Microsoft DREAD**

Another points-based risk rating approach is Microsoft DREAD

*https://msdn.microsoft.com/en-us/library/ff648644.aspx*

## OWASP Risk Rating – Threat Agent Factors (Likelihood)

- Skill level: How technically skilled is this group of threat agents?
  - No technical skills (1), some technical skills (3), advanced computer user skills (4), network and programming skills (6), security penetration skills (9)
- Motive: How motivated is this group of threat agents to find and exploit this vulnerability?
  - Low or no reward (1), possible reward (4), high reward (9)
- Opportunity: What conditions and resources are required for this group of threat agents to find and exploit this vulnerability?
  - Full access or expensive resources required (0), special access or resources required (4), some access or resources required (7), no access or resources required (9)
- Size: How large is this group of threat agents?
  - Developers (2), system administrators (2), intranet users (4), partners (5), authenticated users (6), anonymous Internet users (9)

(Note: you can assign any value 0 – 9 for each factor, the descriptions are only examples for reasonable value selection)



**Threat Agent Factors**

The first set of factors are related to the threat agent involved. The goal is to estimate the likelihood of a successful attack by a group of assumed attackers. Note that there may be multiple threat agents that can exploit a particular vulnerability, so it's usually best to use the worst-case scenario (i.e., consider the most powerful attacker). Sometimes, it makes sense to rate a vulnerability multiple times, for different attack groups. For instance, for external cyber criminals and for internal attackers such as administrators (if you expect both these types of attackers), as they often have completely different access to the target systems, which may result in different risk ratings.

**Opportunity**

Full access means that an attacker must, e.g., be physically close to the target system (e.g., needs access to the server room). Some access can mean, e.g., that the attack can be executed from within the company, e.g., by a visitor who manages to plug in his computer within the company network. In general, this «access» identifies «how physically close» an attacker must be to the target, i.e., how difficult it is for an attacker to reach / to communicate with the system where the attack can be carried out (and it is not related to whether an attack requires prior authentication, as this is reflected in the next factor «Size»). With cyber attacks, specific resources are often not needed, but there are some exceptions. To carry out a denial of service attack, one possibly must rent a botnet. To break an SSH server, it may be necessary to buy an exploit against an (unknown) vulnerability on the cyber black market.

This is a factor which is sometimes not obvious to rate. Assume an administrator wants to attack his own company simply by abusing his normal access rights. Let's assume he can only do the attack in the server room, so «full access» would be needed, which would get a rating 0. But now one can argue that this attacker has access to the server room anyway as part of his job (it's not a big hurdle for him), so it should be rated higher. Which one is right? It depends. If the administrator can simply enter the server room at any time, e.g., with a physical key and no access to the room is logged, the factor should be high, maybe a 7. But if access is done with an electronic key card so that any entrance (and exit) is logged or if there's a guard who records this, then the administrator can no longer just sneak in at any time undetected, and this more complicated access should result in a lower rating (e.g., 3), because the attack opportunity is no longer easily given in the sense that the attack may easily be detected because one knows who was in the server room at what time. This shows you that you have to carefully balance this factor based on the attacker type you are considering.

**Size**

Make sure to rate this with care. For instance, if we expect an attack can be done by powerful cyber criminals over the Internet, this should not rated as 9, despite the explanation «anonymous Internet users». «Anonymous Internet users» means that «almost every Internet user could do the attack», which wouldn't be reasonable in this case. Therefore, it should probably be rated as maybe a 3 or a 4 in this case. So don't take the explanations here word for word, but consider them more as examples to reflect some sizes of attack groups. As another example, consider, «intranet users», which is a 4 according to the given examples. This is reasonable for «typical company sizes». But if you have a small company with 10 employees that can do the attack, it maybe should be rated as a 2, and with 10'000 employees it should maybe be a 7.

**Opportunity and Size**

These two factors seem to be somewhat related, because if an attack requires very special physical access (e.g., can only be carried out while being physically present in the server room which is access protected by a human guard), then this likely also means that the size of the attack group is small, as only few attackers will attempt such an attack at all (e.g., only system administrators that have legitimate access to this room or a few cyber criminals that will take the risk to try to get into the server room). But the two factors can also be quite independent. For instance, an attack may require expensive resources (e.g., an expensive tool that can be bought in the darknet) but can still be carried out by quite a lot of cyber criminals (that are willing to make this investment if the target is attractive enough) anonymously over the Internet (so Opportunity may be 0, but Size may be, e.g., 4). Or an attack can be executed from anywhere (e.g., from the Internet), but only the best cyber criminals have the knowledge and experience to execute the corresponding vulnerability (so in this case, Opportunity may be 9, but Size may be 3).

## OWASP Risk Rating – Vulnerability Factors (Likelihood)

- Ease of discovery: How easy is it for this group of threat agents to discover this vulnerability (just finding it, not yet exploiting it)?
  - Practically impossible (1), difficult (3), easy (7), automated tools available (9)
- Ease of exploit: How easy is it for this group of threat agents to actually exploit this vulnerability (once it has been found)?
  - Theoretical (1), difficult (3), easy (7), automated tools available (9)
- Awareness: How well known is this vulnerability to this group of threat agents («how much do the attackers suspect» that the vulnerability may exist)?
  - Unknown (1), hidden (4), obvious (6), public knowledge (9)
- Intrusion detection: How likely is an exploit to be detected?
  - Active detection in application (1), logged and reviewed (3), logged without review (8), not logged (9)



**Vulnerability Factors**

The next set of factors are related to the vulnerability involved. The goal here is to estimate the likelihood of the particular vulnerability involved being discovered and exploited. Assume the threat agent selected before.

**Awareness**

This factor rates how much the attackers suspect that the vulnerability may exist. The reason for including this factor is that if a potential attacker suspects or knows that a vulnerability exists, then this increases his motivation and he'll likely invest lots of time and energy to actually find the vulnerability so he can possibly exploit it.

As an example, assume someone has posted in a public web forum that he discovered an SQL injection vulnerability in a specific system, without providing addition details. If an attacker finds this, then this increases his awareness with with respect to the existence of this vulnerability, which increases the probability that he'll ty to find and exploit the vulnerability.

**Ease of Discovery**

This factor deals with the actual discovery of a vulnerability, e.g., of the SQL injection vulnerability of which the attacker may be aware (see above). Discovering a vulnerability includes the activities that are necessary to demonstrate / find out that the vulnerability indeed exists. In the case of the SQL injection vulnerability, this means detecting the actual vulnerability in a sense that the attacker can demonstrate that he can use the vulnerability to manipulate a query according to his wishes (proof of concept). It does not include the actual exploit, however, this is covered by the "ease of exploit" factor.

**Ease of Exploit**

The effort to perform the actual exploit (assuming the vulnerability has been detected) is also covered in a separate factor. This is reasonable because even when a vulnerability has been detected, actually exploiting it may require significant additional effort. For instance, exploiting a found SQL vulnerability may turn out to be difficult if the attacker does not know the database schema (a lot of trial and error may be required to find out the names of tables and columns so they can be accessed by exploiting the vulnerability).

**Intrusion Detection**

The reason for including this factor is that the more logging and reviewing of logs is done, the more likely it is an attack will be detected early, maybe even so early that the attack can still be stopped before any damage is done. In addition, especially in the case of internal attacks, having good intrusion detection can have the effect that an attacker won't try to do an attack as he fears that he will be detected.

## OWASP Risk Rating – Impact Factors

- Financial damage: How much financial damage will result from an exploit (direct damage by the attack and effort to recover from it)?
  - Less than the cost to fix the vulnerability (1), minor effect on annual profit (3), significant effect on annual profit (7), bankruptcy (9)
- Reputation damage: Would an exploit result in reputation damage that would harm the business (long-term damage)?
  - Minimal damage (1), Loss of major accounts (4), loss of goodwill (5), brand damage (9)
- Non-compliance: How much will regulations by governments (e.g., laws) or other companies be violated by an exploit?
  - Minor violation (2), clear violation (5), high profile violation (7)
- Privacy violation: How much personally identifiable information could be disclosed?
  - One individual (3), hundreds of people (5), thousands of people (7), millions of people (9)

**Impact Factors**

Looking at these factors, you can see that they determine the *business* impact of a successful attack. This is reasonable because – as discussed before – one should focus on the negative business consequences that will occur if a particular attack happens.

**Financial Damage and Reputation Damage**

These factors appear to be overlapping. To avoid confusion, use the two as follows:

- Financial damage is used to specify direct financial damage (costs) of the attack. For instance, if our e-shop is down for 6 hours and this results in a loss of revenue of CHF 10'000, then this is the direct financial damage. Or to recover from the attack, our administrators had to invest 200 addition working hours, which corresponds to CHF 16'000.
- Reputation damage is used to reflect long-term damage. In reality, this likely also means financial loss, in addition to the direct costs that were already taken into account above. For instance, in the e-shop example, it is likely that due to the unexpected downtime, some customers will prefer other e-shops in the future, which means a certain additional financial damage.

**Non-Compliance and Privacy Violations**

Both non-compliance and privacy violations can result in further financial damage, as non-compliance with regulations (e.g., Sarbanes-Oxley Act or regulations in the health care area) could mean the government, companies or other persons (of which private information was disclosed) may take the attacked company to court, which may result in getting a monetary fine (and don't forget costs for lawyers...).

## OWASP Risk Rating – Likelihood and Impact Determination

- The likelihood of successful exploitation is determined by taking the average of the threat agent and vulnerability factors, e.g.:

| Threat agent factors | | | | Vulnerability factors | | | |
|---|---|---|---|---|---|---|---|
| Skill level | Motive | Opportu-nity | Size | Ease of discovery | Ease of exploit | Aware-ness | Intrusion detection |
| 5 | 2 | 2 | 1 | 3 | 3 | 4 | 2 |
| Overall likelihood: 2.875 | | | | | | | |

- The impact of successful exploitation is determined by taking the average of the impact factors, e.g.:

| Impact factors | | | |
|---|---|---|---|
| Financial damage | Reputa-tion damage | Non-com-pliance | Privacy violation |
| 7 | 5 | 1 | 6 |
| Overall impact: 4.75 | | | |




**Business Impact vs. Technical Impact**

The impact as discussed here is used to rate the business impact of a successful attack, because this is typically what we want to know. However, the OWASP method also allows to rate the technical impact (and based on this the technical risk), which can be used in case the analyst does not have enough business understanding to rate business impact. In such a case, the appropriate business representatives should then take the technical risks and, based on them, make a decision about the actual business risks.

To rate technical impact, there are four technical impact factors (overall impact is then again determined by taking the average):

- Loss of confidentiality: How much data could be disclosed and how sensitive is it?
  - Minimal non-sensitive data disclosed (2), minimal critical data disclosed (6), extensive non-sensitive data disclosed (6), extensive critical data disclosed (7), all data disclosed (9)
- Loss of integrity: How much data could be corrupted and how significant is the damage?
  - Minimal slightly corrupt data (1), minimal seriously corrupt data (3), extensive slightly corrupt data (5), extensive seriously corrupt data (7), all data totally corrupt (9)
- Loss of availability: How much service could be lost and how vital is it?
  - Minimal secondary services interrupted (1), minimal primary services interrupted (5), extensive secondary services interrupted (5), extensive primary services interrupted (7), all services completely lost (9)
- Loss of accountability: Are the threat agents' actions traceable to an individual?
  - Fully traceable (1), possibly traceable (7), completely anonymous (9)

As can be seen, technical impact is broken down into factors aligned with the traditional security areas of concern: confidentiality, integrity, availability, and accountability. The goal is to estimate the magnitude of the impact on the system if the vulnerability were to be exploited.

Note that technical impact should really only be used if you cannot determine business impact, as it can be misleading. For instance, assume that exploiting a specific vulnerability may lead to disclosing all data, so the corresponding factor «loss of confidentiality» would be rated with 9, which may result in a relatively high rating for the technical impact. However, it may be that disclosing of this data is not really critical for the company, as the data has little value for outsiders (e.g., because it may be just anonymized statistical data about usage of a system that may be important for the company for system optimization but that is basically worthless for outsiders). So in this case, the true damage of disclosing the data would be overrated if one would only consider technical impact.

There's another limitation with technical impact factors. Assume you have identified a vulnerability that allows an attacker to read the data in your database (e.g., with SQL injection). But only reading is possible, not manipulating. Exploiting such a vulnerability may have a major business impact if the data is highly important for the company. But looking at the technical impact factors, the 2nd and 3rd factors would be rated with 0, as integrity and availability are not relevant here, which would result at most in a medium technical impact (see technical impact computation on one of the following slides). To cope with this to a certain degree, you can omit the technical impact factors that are not relevant, i.e., only use the 1st and 4th factors to determine the technical impact in this case (and use/omit other factors in other cases).

# OWASP Risk Rating – Risk Determination

- The computed values (likelihood, impact) are mapped to *High*, *Medium* and *Low* as follows:
  - Likelihood 2.875 → *Low*
  - Impact 4.75 → *Medium*

| Likelihood and impact levels | |
|---|---|
| 0 to <3 | Low |
| 3 to <6 | Medium |
| 6 to 9 | High |

- And the overall risk is determined according to the following risk matrix (which is the same as with NIST 800-30)

| | Impact | | |
|---|---|---|---|
| Likelihood | Low | Medium | High |
| High | Medium | High | Critical |
| Medium | Low | Medium | High |
| Low | Info | Low | Medium |

  - So we have a risk of *Low*

## OWASP Risk Rating – Example (1)

Let's again rate the risk of the vulnerability identified in the financial accounting system of the tax consultant company

- Threat agent factors:
  - Skill level: Attackers (accountants) have advanced user skills (4)
    - But thinking about this: The skill level of the accounts is not really relevant here as they simply abuse their legitimate system access – every accountant can do the attack, independent of the actual skill level
    - In practice, when using OWASP Risk Rating, this happens from time to time: You should rate a factor that does not really make sense
    - To cope with factors that are not relevant when rating a vulnerability/threat, it's best not to rate the factor at all, so instead of (4), we use (-)
  - Motive: Spirit among employees is not so great (salary cuts), there may be financial gain (4)
  - Opportunity: We assume physical access to the system is possible from within the company and at any time, no special resources are required (7)
  - Size: Only financial accounting staff have the required privileges to do the attack (2)

**Skill Level**

One could argue that the skill level still has a little effect on whether an attack will be done as accountants that have «security penetration skills» are maybe more likely to consider such an attack than accountants with very little «technical skills». This may be the case to a small degree, but a malicious accountant will mainly be driven by the motivation to do such an attack and by the fact the he has the opportunity to do this, and the skill level plays a minimal role and a high skill level also won't really help him to the attack in a better way. So overall, the influence of the factor is really small (at best) and makes little sense here so it's best to ignore it and to not rate it at all.

## OWASP Risk Rating – Example (2)

- Vulnerability factors:
  - Ease of discovery: Difficult to detect as the lack of logging is not obvious, maybe one could start with an «accidental» minor modification and see if anyone notices (3)
  - Ease of exploit: Once found, this is easy (7)
  - Awareness: We assume there have been rumors that only little logging is done (4)
  - Intrusion detection: No logging at all (9)
- Impact factors:
  - Financial damage: Some limited direct damage can be expected (people getting too much salary, effort to recover from the attack) (3)
  - Reputation damage: Significant brand damage to be expected if a successful attack became public (9)
  - Non-compliance: No impact as a successful exploit does not violate any compliance requirements for this company (0)
  - Privacy violation: No disclosure of personally identifiable information (0)



**Ease of Exploit**

One could argue that this factor also does not make a lot of sense here as there's no real exploitation involved because accountants simply abuse their legitimate system access. However, simply because no real exploitation is involved is no reason to ignore the factor, because the factor is part of the rating methodology (and rightly so) and the fact that exploitation is easy here has of course an influence on the likelihood because if exploitation were more difficult, likelihood would be lower. Therefore, in contrast to skill level before, which really has no influence on the likelihood of the attack and which correspondingly was omitted, the ease of exploit factor should definitely not be ignored.

# OWASP Risk Rating – Example (3)

- Likelihood determination (only average of the rated factors):

| Threat agent factors | | | | Vulnerability factors | | | |
|---|---|---|---|---|---|---|---|
| Skill level | Motive | Opportu-nity | Size | Ease of discovery | Ease of exploit | Aware-ness | Intrusion detection |
| - | 4 | 7 | 2 | 3 | 7 | 4 | 9 |
| Overall likelihood: 5.14 (*Medium*) | | | | | | | |

- Impact determination:

| Impact factors | | | |
|---|---|---|---|
| Financial damage | Reputa-tion damage | Non-com-pliance | Privacy violation |
| 3 | 9 | 0 | 0 |
| Overall impact: 3.0 (*Medium*) | | | |

- So, we still get a risk rating of *Medium* (just like with the NIST 800-30 method before)
  - But due to the more structured approach, there's more confidence in the result

## OWASP Risk Rating – Exercise (1)

- You are performing a penetration test of a custom e-shop web application with 50'000 registered users
- You discover an SQL injection vulnerability that allows to read the stored credit card information of all users from the shop's database
- Discovering and exploiting the vulnerability each required several days of skillful probing, automated tools did not help at all
- The vulnerability can be exploited «from the Internet», no login is required
- A user recently reported in a public web forum that he had found an SQL injection vulnerability in this e-shop, without disclosing any details
- The application employs extensive logging (web logs, DB logs,…) and log files are inspected regularly
- Your task: Assess the risk that cyber criminals can exploit this vulnerability to get all credit card information from the application

## OWASP Risk Rating – Exercise (2)

- Likelihood determination:

| Threat agent factors | | | | Vulnerability factors | | | |
|---|---|---|---|---|---|---|---|
| Skill level | Motive | Opportu-nity | Size | Ease of discovery | Ease of exploit | Aware-ness | Intrusion detection |
| | | | | | | | |
| | | | | | | | |

- Impact determination:

| Impact factors | | | |
|---|---|---|---|
| Financial damage | Reputa-tion damage | Non-com-pliance | Privacy violation |
| | | | |
| | | | |



**Limitations of the OWASP Risk Rating Methodology**

The methodology is in general well suited to achieve reasonable risk ratings. But it also has some limitations:

- Sometimes, some of the factors don't really make sense in some scenarios. For instance, considering again the the example of the tax accounting system where internal accountants are the attackers, the «skill level» does only make little sense as the accountants just abuse their legitimate access – so it does not really matter whether they have just «advanced computer user skills» or «security penetration skills». As a result, we ignored this factor when rating the vulnerability.
- Sometimes, the examples for the ratings are not a good fit for a particular scenario. For instance, looking at the «privacy violation» factor, then disclosing the information of 1'000 individuals should be rated with 6. However, depending on the company, this can be more or less serious. For instance, a relatively small company with 1'000 customers that discloses all this data will likely have a big problem and may not survive financially if it is sued by the customers. So, in this case, the rating should be a 9. Conversely, a company with one million customers may easily handle the loss of the data of 1'000 customers, so the rating here should maybe be a 3.

The methodology could have provided support to deal with such issues, but this would also have made the entire approach more complicated. Instead, the factors and the examples for the ratings were chosen based on practical experience so that they work well «on average» and it is therefore recommended that you just rate all the factors individually based on the examples provided for the ratings 0-9 without thinking too much whether the factors or ratings make perfect sense in the specific scenario and without thinking too much whether there are dependencies between the factors (see, e.g., the relation between the factors «opportunity» and «size» as discussed in the notes of a previous slide). This usually will give you good and reasonable results in most cases. But if you really think that the ratings of a factor have to be adapted so that they fit the scenario you are analyzing, then you can do this. This could make sense with the «privacy violation» factor if, e.g., the loss of the data of 1'000 customers would be devastating to you – so just rate this with 9. Or if you a factor is really not relevant (as with the factor «skill level» as described above), you can also simply omit the factor completely.

Another limitation is that when having fixed a vulnerability, the rating often is still relatively high. For instance, assume you fix the SQL injection vulnerability in the exercise above. As a result of this, many of the vulnerability factors will have a very low value of 0 or 1 (using 1 is maybe a good idea because there may always be implementation mistakes or code errors in the underlying DB access library, so attacks may still be possible). But the threat agent factors are unchanged because you cannot affect them, so the resulting likelihood is probably Medium, although it should – according to common sense – probably be Low. This shows that the methodology works usually well as long as there are exploitable vulnerabilities, but it tends towards too high likelihood ratings once the vulnerability has truly been fixed so that successful attacks are very unlikely. In those cases, you should then replace the «computed» likelihood value (here: Medium) with the one that makes more sense (here: Low). Or, you can just go back to the simpler methodology NIST 800-30, which would result in a low likelihood in this case.

# Risk Mitigation

## Risk Mitigation

- Once the risks have been determined, one must decide what further actions have to be taken → risk mitigation

- Risk mitigation means that the following must be done for the vulnerabilities included in the risk analysis:
  - Prioritize the vulnerabilities according to the risk rating
    - This should make sure that the most critical vulnerabilities are handled first – and not the ones that are easiest to mitigate
  - Decide what to do with the vulnerabilities (risk mitigation options)
  - If necessary, propose, design and implement corrective actions to reduce or avoid the risk of a vulnerability

- Once risk mitigation has been completed, update your risk analysis documentation
  - Determine the new risk values that take the corrective actions into account
  - Check if there are no unacceptable risk levels remaining

## Risk Mitigation Options

With every vulnerability and associated risk, you can decide to do one of the following:

- Risk Acceptance
  - Accept the risk, e.g., because it's so small that any corrective action would be pointless / financially not reasonable (e.g., risk rating is *Info* / *Low*)
- Risk Reduction
  - Implement corrective measures to either reduce the likelihood or the impact (or both) to reduce the risk to an acceptable level
- Risk Avoidance
  - Avoid the risk completely, e.g., by removing the functionality with which it is associated
- Risk Transfer
  - Transfer the risk to someone else, e.g., buying insurance
- Risk Ignorance
  - You know there's a (possibly high) risk, but you simply ignore it

**Risk Avoidance**

In the previous example (stealing credit cards), the risk could be avoided completely by not storing the credit card information at all. But this would also reduce usability, because users would have to enter this information during every payment process.

**Risk Transfer**

In the real life, this is done all the time. We have liability insurances for our cars and fire insurances for our houses. The reason is that there are potentially high risks associated with the threats covered by the insurance. So we transfer these risks at relatively moderate costs. In the information processing world, however, this is nor widespread yet, but it’s likely that there will be more and more desire to by insurance against cyber attacks.

**Black Swans**

This term is used for highly critical but rare incidents. E.g., those with likelihood (very) Low, but with impact High. According to the two discussed methods, the resulting risk is Medium, which seems «difficult to accept». But in practice, you still typically have to accept such risks if the corresponding functionality is needed in the system, but in such situations, you should really think hard about effective countermeasures to make the likelihood as small as possible (and uses not only preventive measures, but also measures (e.g., monitoring) to quickly detect an attack that has happened). And maybe you should also think about ways to bring down the impact to Medium. But in practice, you may eventually have to accept and live with such risks, even if they may be devastating for your company in case of a successful attack. Additional cyber insurance for such cases may also help you to cope with such risks.

## Risk Reduction

- With software projects, risk reduction is often the primary mitigation option
- The risk of a vulnerability can be reduced by either reducing the likelihood or the impact (or both)
  - Likelihood is usually easier to reduce than impact, as it can often be reduced using appropriate measures to make «the attack more difficult»
- It is important to select cost-effective strategies to reduce a risk
  - Obviously, the costs to reduce a risk should not be bigger than the expected financial damage from the vulnerability
  - It is not necessary to find the best solution to reduce a risk, but to find a cost-effective solution that reduces the risk to an acceptable level
  - Note that a selected solution may also have negative side effects (e.g., on usability) and may even result in new risks
    E.g., encrypting backup tapes may create problems during recovery



**Cost-Effective Solutions**

E.g., to achieve «better» authentication security, it may be enough to enforce a minimal password strength (which can easily be implemented) instead of replacing pure passwords with a multi-factor authentication method based on tokens or certificates (which is a costly solution).

**Negative Effects on Usability**

Switching from password-based authentication to a two-factor authentication solution that includes an additional one-time token increases security, but also has a negative impact on usability because users have to do «more work» during authentication.

## Risk Reduction – Example

- The OWASP Risk Rating factors are also helpful to think about ways to reduce a risk
  - The high values obviously provide the most reduction potential
- Example: Some options to reduce the likelihood of the vulnerability identified in the financial accounting system of the tax consultant company

| Threat agent factors | | | | Vulnerability factors | | | |
|---|---|---|---|---|---|---|---|
| Skill level | Motive | Opportu-nity | Size | Ease of discovery | Ease of exploit | Aware-ness | Intrusion detection |
| - | 4 → 2 | 7 → 2 | 2 | 3 | 7 | 4 → 1 | 9 → 1 |

Reduce motivation by treating employees «better»

Enforce 4-eyes principle to modify critical data

Improve intrusion detection by automatically sending an e-mail to the head of HR whenever a salary is modified

- With these adaptations, likelihood is reduced from 5.14 to 2.57



**4-Eyes Principle**

In this case, one could imagine a solution where an accountant that wants to modify critical data first has to request (using a paper document) access to the critical area, which must be approved by at least one member of middle management or higher. The accountant then only gets access during a clearly defined time window, e.g., between 14:00 and 15:00 on a specific day.

This results in a significant reduction of the value assigned to «opportunity», as the accountant no longer has write-access at any time, but only after having requested special access and only after having «exposed himself» and after having «presented a good story» to get write access.

**Ease of discovery & ease of exploit**

Assuming that a 4-eyes principle and good intrusion detection are added, should the ratings of these two factors be changed as well? One could argue that «ease of discovery» should be changed, because with the new safeguards, the main problems are solved and there's no longer «a vulnerability to be detected», so one could rate this factor with 0. «ease of exploit», however, should remain being rated as 7 because doing the actual attack – changing the salary – is still as easy as before (but the attack is more complicated to set up and the risk of being caught is much higher, but this is already reflected in «opportunity» and «intrusion detection»). So, to summarize, it would be reasonable to set «ease of discovery» to 0 and «ease of exploit» to 7 because of the new safeguards.

## Security Risk Analysis – Final Remarks

- Security risk analysis is always subjective – performing the analysis in a team increases the likelihood that the outcome is sound
- When using OWASP Risk Rating, don't try to be over-precise
  - It does not significantly matter whether you assign 4 or 5 to a specific factor, as we only do a coarse-grained rating *Low* – *Medium* – *High*
  - To be on the safe side, be a bit pessimistic when choosing the values
- To do security risk analysis in an efficient and effective way, it's recommended to combine the two methodologies: Use NIST 800-30 as a basis and supplement it with OWASP Risk Rating as needed
  - Most threats / vulnerabilities can usually be rated with NIST 800-30
  - If you are unsure with some threats / vulnerabilities, rate them with OWASP Risk Rating
  - The more experienced the analyst is, the more he'll rely on NIST 800-30 and the less frequently he'll use OWASP Risk Rating
- Of course, the methods presented here are just two examples that work well and that can be used well together
  - There are several other standards that can be used, and some companies also use their own internal standards

**Some further remarks**

- With iterative software development processes, risk analysis should be part of several iterations (just like, e.g., threat modeling)
  - Because any extension of the system (new requirements, use cases, design extensions, functionality etc.) can result in new vulnerabilities and therefore risks and can affect the risks of already known vulnerabilities.
  - For instance, assume an e-shop hasn't stored payment information of its customers so far. As a result, the threat that an attacker may read all user data from the database was not considered to be a big risk until now, simply because the threat agent factors were rated relatively low (especially the motivation was rated very low). But assuming that from now on, credit card information will also be stored in the database (a system extension), the risk of the threat is affected as the motivation for the attackers will now be much higher. In addition, the business impact will go up, as significant more damage can be expected (reputation, non-compliance, privacy violation).
- But even when a system is in operation, risk analysis should be reviewed periodically
  - E.g., because new attack methods (and therefore vulnerabilities) surface from time to time or because new powerful automated attack tools have appeared
- Security risk analysis is useful beyond secure software development
  - E.g., when performing a penetration test or in general when doing security assessments of systems / companies as an external security analyst

**When to use NIST 800-30 and when to use OWASP Risk Rating?**

There's no general answer for this. Once you have done several such analyses, then you'll probably feel confident and use NIST 800-30 to rate most threats / vulnerabilities. But if you are a beginner, OWASP Risk Rating well certainly help you to do a good risk analysis process so in this case, you may use the OWASP method for many or even all threats/vulnerabilities you have to rate. But even if you are experienced, there may be some threats/vulnerabilities that you have to rate where you are unsure about the rating based on NIST 800-30 (e.g., because you have to rate a vulnerability type you have never encountered before, because you are dealing with an attacker that is difficult to grasp (which often happens with internal attackers where some countermeasures are often less effective than with external attackers), because you are doing the risk analysis in the context of a type of system that you are not familiar with, because there are countermeasures involved that help to a certain degree but that are not really very effective etc.) or where you want to explain the reasons for the ratings to somebody else in more detail. For such threats/vulnerabilities, using OWASP Risk Rating is then a very reasonable approach to use.

## Summary

- The purpose of security risk analysis is to rate the risk (the criticality) of security vulnerabilities
- Security risk analysis complements other security activities such as threat modeling and penetration testing
- Risk is determined as a function of likelihood and impact
- The NIST 800-30 approach is a relatively simple methodology to rate the risk of vulnerabilities but that works quite well in practice – in particular if performed by an experienced person
- OWASP Risk Rating provides a more structured approach towards rating the risk of vulnerabilities that is well-suited for beginners, but also for experienced persons to increase confidence in the risk ratings
- The final step of risk analysis is risk mitigation, which means deciding about what options must be taken to reduce risks to acceptable levels (if necessary)