Penetrationtest Report

DVWA

Table of Contents

1 Version Control

Version	Title	Author	Date
0.1	Initial Report	Niklas Bessler	01/01/2020
0.2	Quality Assurance	Niklas Bessler	01/01/2020
1.0	Final Report	Niklas Bessler	01/01/2020

2 Point of Contact

2.1 Contractor

Niklas Bessler
Welzhomer Str. 1337
63791 Karlstone
Germany

Niklas Bessler - project manager, penetrationtester - niklas@sani-sec.de

2.2 Client

DVWA
8468 Bellevue Lane
South Portland, ME 04106

James Smith - product owner - test@example.com
Juan Steele - system administrator - test@example.com

The final report will be handed out to James Smith. Technical issues (e.g., loss of network connectivity or a single system is not responsive) will be immediately reported to Juan Steele.

3 Project Details

3.1 Project Objectives

This is a sample report. The purpose of this sample report is solely to show the idea of how a penetration test report might look like.

3.2 Scope of Work

We have been engaged to perform a penetration test on **one system**. The system has the IP address 172.0.0.2. Other systems in the network should not be tested.

3.3 Period of Testing

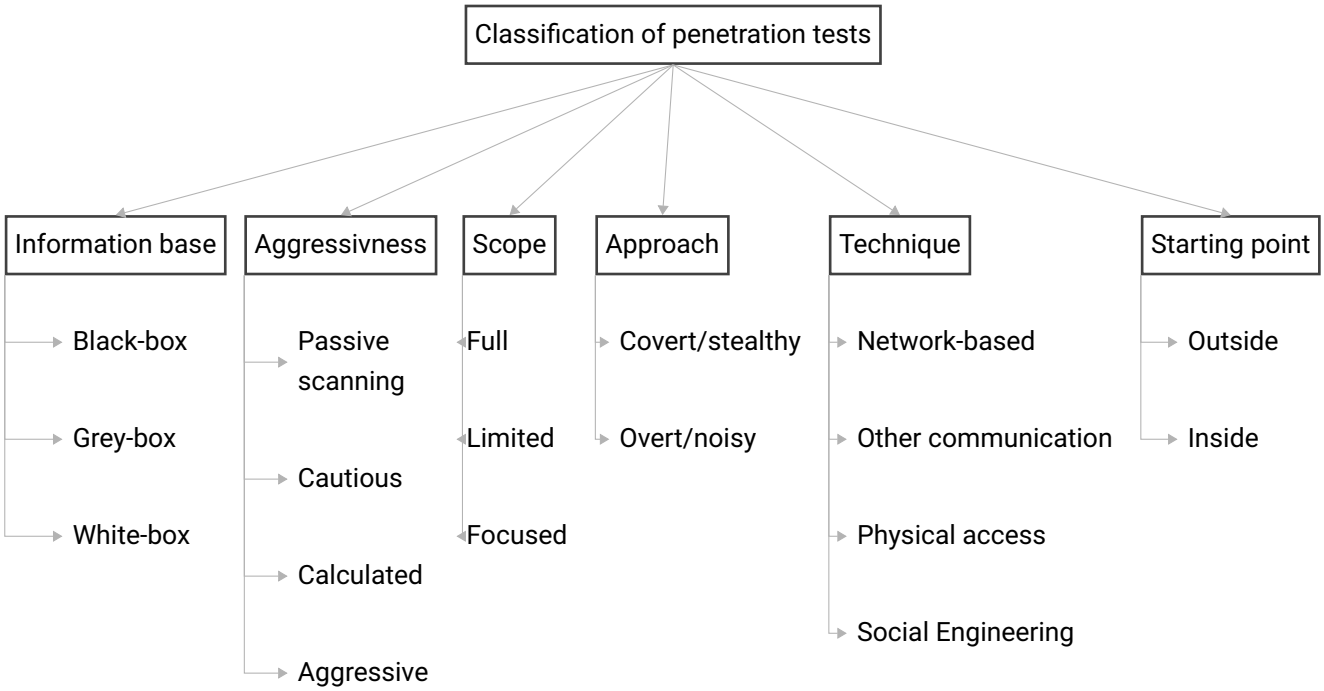
The penetration test was performed over the period from **January 1, 2020 to January 2, 2020** (full-day). Network access using a VPN was granted for the duration of the penetration test. The penetration tester was assigned the IP address 172.17.0.1.

All vulnerabilities found are usually provided with a date and time stamp when they were found/exploited. This is for reasons of comprehensibility, since the exploitation of the vulnerability can usually be tracked in the servers log file.

3.4 Test Classification

The approach of the specific penetration test is elaborated in coordination with the customer before testing. Our classification of penetration tests is oriented towards the whitepaper "Study: A Penetration Testing Model" published by the German Federal Office for Information Security (German: Bundesamt für Sicherheit in der Informationstechnik).

The penetration test is classified as cautious grey-box test within a limited scope. We do not take measures to be stealthy during the test. The penetration test is performed from inside the companies network. DoS (denial of service) attacks and social engineering techniques are not included. An overall methodology is described in the appendix.



3.5 Limitations

The informative value yield by a penetration test is always limited. There is no 100% security guaranteed, even if all findings are properly mitigated and recommended countermeasures are fully implemented. The penetration test, however, evaluates whether the systems and applications within the scope are based on a professional security level.

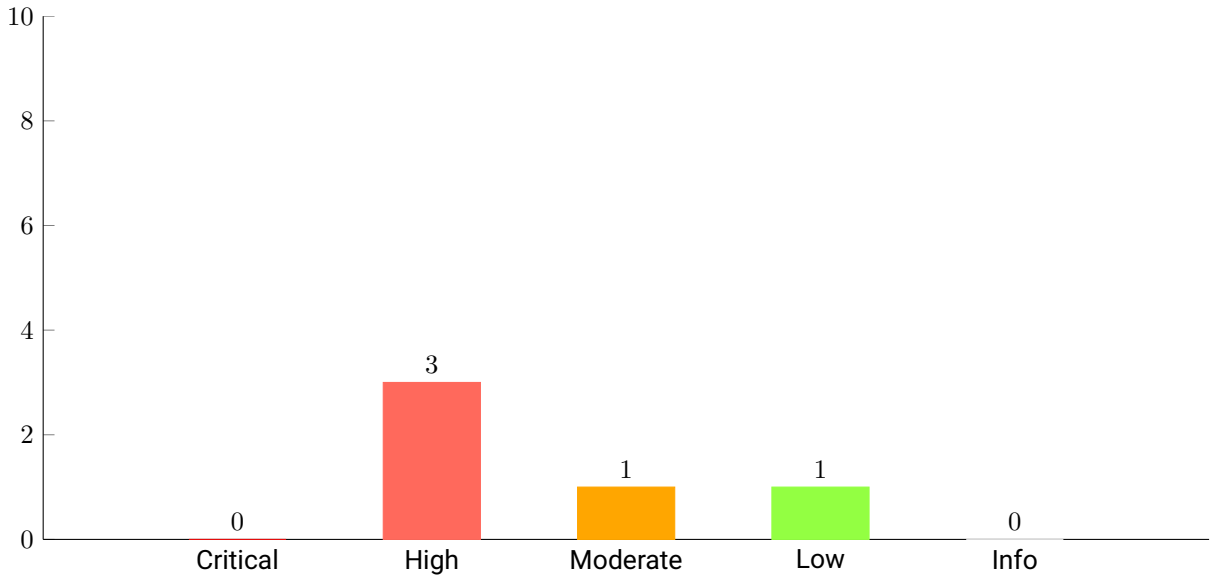
In general, security should be considered as a continuing process. Therefore, we recommend regular security tests, which can improve the security of technical systems, organizational and personnel infrastructure significantly.

Non-specific attacks that are based on automated scripts and attacks performed by so-called „script kiddies“ (solely use of known exploits) are most likely covered in the testing scope. Therefore, these attacks are deprived of any basis. In addition, the effort of a successful attacker is at least increased to the effort that the penetration tester has invested.

4 Executive Summary

4.1 Summary

In total, we have identified **5 vulnerabilities**. These include **high-risk vulnerabilities**. A severity distribution is shown in the diagram below.



4.2 Vulnerabilities

Nr.	Risk	Description
5.1	High	Lack of Transport-Layer-Encryption
Fixed: No	**Affected Systems:** 172.17.0.2	
5.2	High	Command Execution
Fixed: No	**Affected Systems:** 172.17.0.2	
5.3	High	SQL Injection
Fixed: No	**Affected Systems:** 172.17.0.2	
5.4	Moderate	Insufficient Password Policy
Fixed: No	**Affected Systems:** 172.17.0.2	
5.5	Low	Deprecated Hash Function
Fixed: No	**Affected Systems:** 172.17.0.2	

5 Technical Report

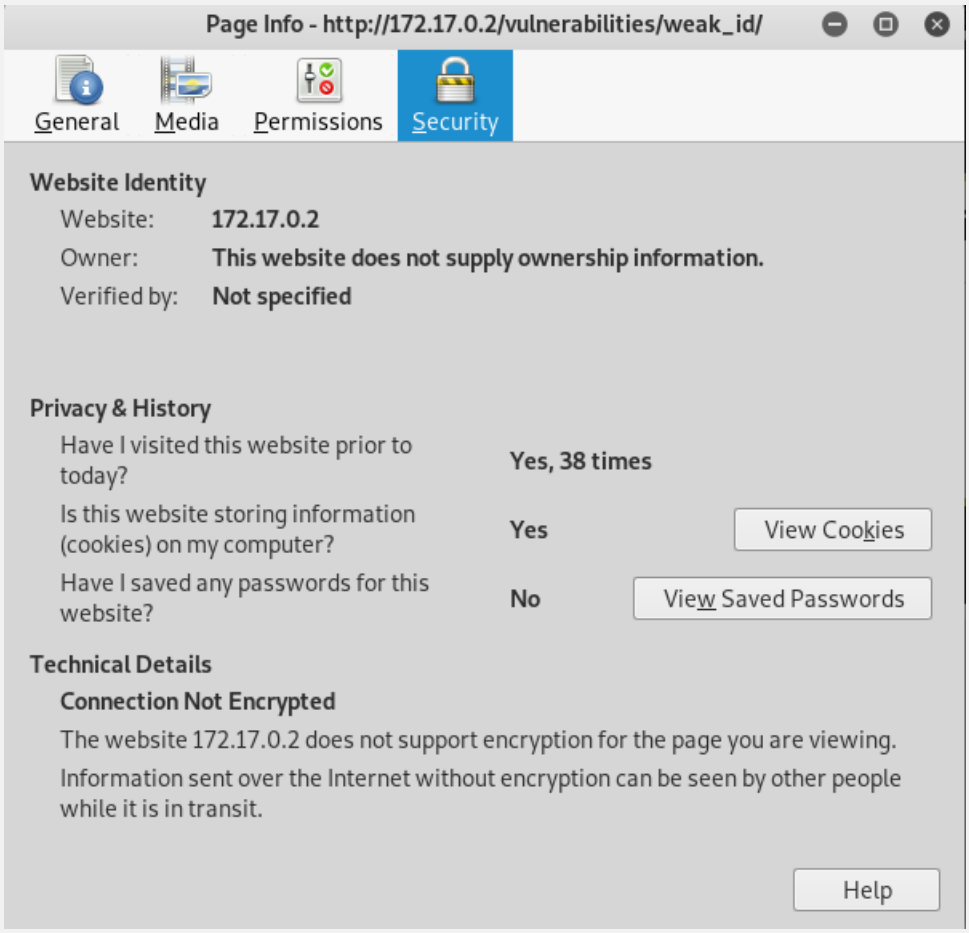
5.1 Lack of Transport-Layer-Encryption

Affected Systems: 172.17.0.2
Ressource: /
Risk: High
Time: 01/01/2020 15:00

Likelihood: Very Likely
Impact: Moderate

Vulnerability Description:
The web application does not implement transport layer protection. Lack of TLS leads to a lack of integrity which allows attackers to modify content in transit.

Proof of Concept:



Recommendation:
The web application should use HTTPS (Hypertext Transfer Protocol Secure) instead of HTTP. Additionally, HSTS (HTTP Strict Transport Security) should be enabled.

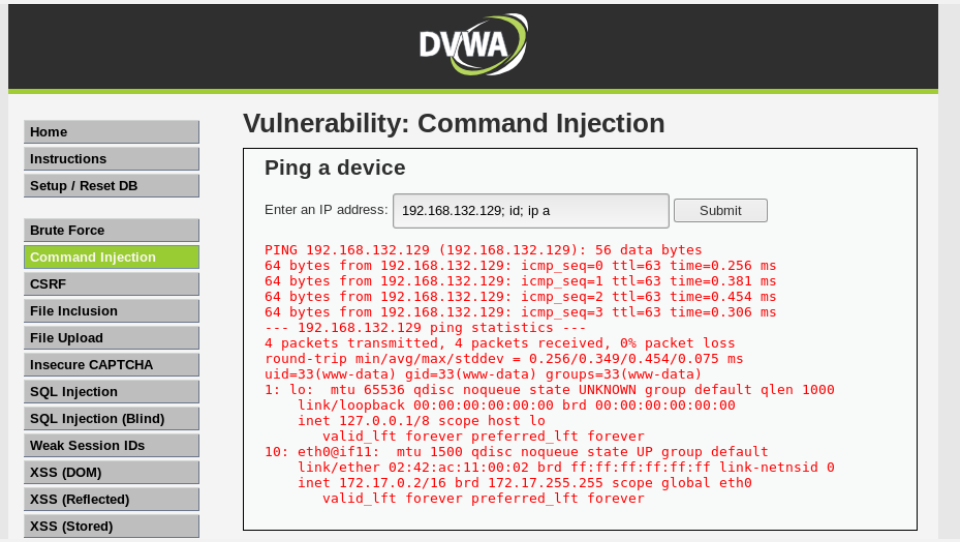
5.2 Command Execution

Affected Systems: 172.17.0.2
Ressource: /vulnerabilities/exec/
Risk: High
Time: 01/01/2020 15:00

Likelihood: Likely
Impact: Significant

Vulnerability Description:
The input field specifies a target for a ping command, e.g., an IP address. However, the user input can the concatenated with another system command that will be executed as user www-data. Eventually, this vulnerability can lead to code execution.

Proof of Concept:
The penetration tester executed two non-critical commands *id* (list all users) and *ip a* (list network configuration) via the code execution vulnerability. The output of the command is shown on the website.



Recommendation:
A whitelist to only allow valid user input should be implemented. At least, sanitize all user input for critical characters.

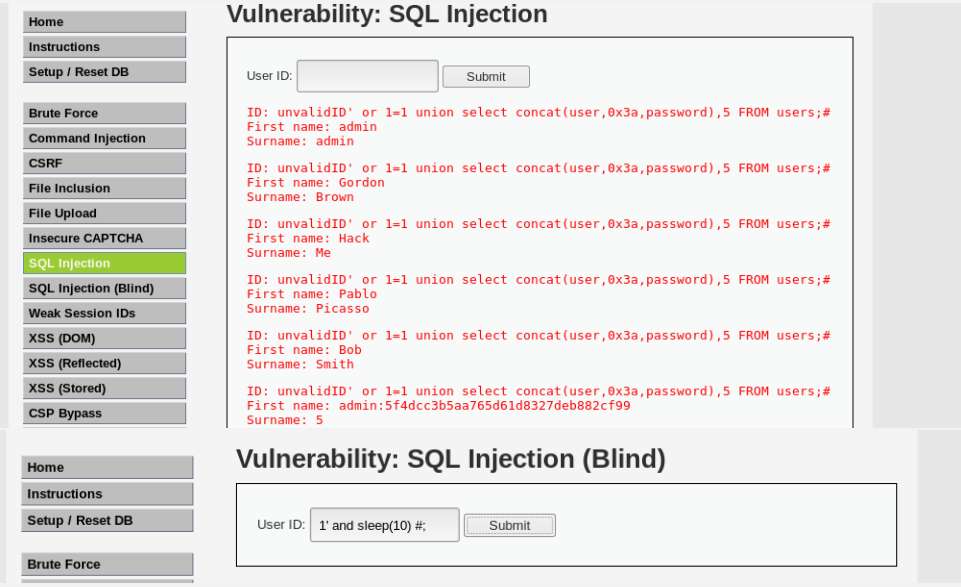
5.3 SQL Injection

Affected Systems: 172.17.0.2/vulnerabilities/exec/
Ressource: /vulnerabilities/sqli/ **&** /vulnerabilities/sqli_blind/
Risk: High
Likelihood: Likely
Time: 01/01/2020 15:00
Impact: Significant

Vulnerability Description:
The web application is prone to SQL injections. In consequence, all data of the database can be potentially accessed or manipulated by an attacker.

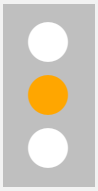
Proof of Concept:
The entered input and its meaning is listed bellow. The screenshot illustrates access to the database where all MD5-hashed passwords are printed and the execution of the SQL command sleep() as well.

```
Number of columns can be enumerated:
unvalidID' or 1=1 order by 3 ;#
Note: order by 3 or higher provokes an SQL error: Unknown column '3' in 'order clause'

Table names can be enumerated:
unvalidID' or 1=1 union all select table_name,5 FROM information_schema.tables;#

Column names of specific a table can be enumerated:
unvalidID' or 1=1 union all select column_name,5 FROM information_schema.columns where table_name='users';#

A table can be enumerated (table users in this case):
unvalidID' or 1=1 union select concat(user,0x3a,password),5 FROM users;#
```



Recommendation:
Consistently use Prepared Statements from the MySQL Improved Extension (MySQLi).

5.4 Insufficient Password Policy

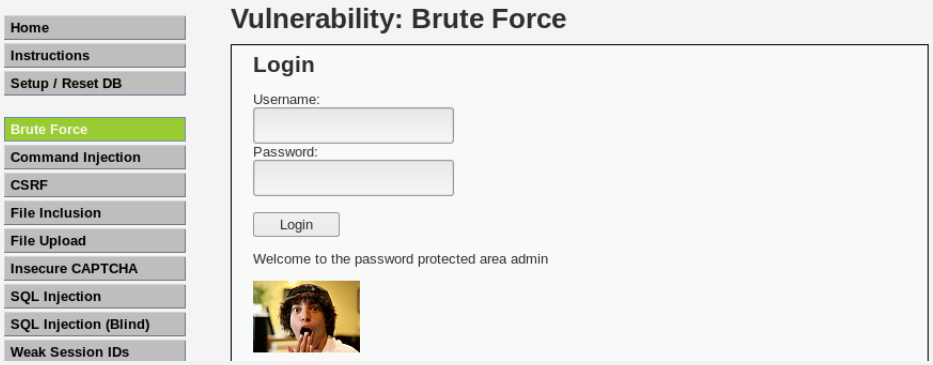


Affected Systems: 172.17.0.2
Ressource: /
Risk: Moderate
Time: 01/01/2020 15:00

Likelihood: Possible
Impact: Moderate

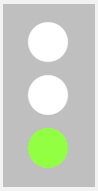
Vulnerability Description:
User of the application can set a weak password. These passwords are prone to brute-force attacks. The confidentiality cannot be ensured.

Proof of Concept:



Recommendation:
A password policy should be noth implemented and enforced. Strong passwords must meet the requirement of a minimum password length of eight characters. Longer passwords are generally more secure.

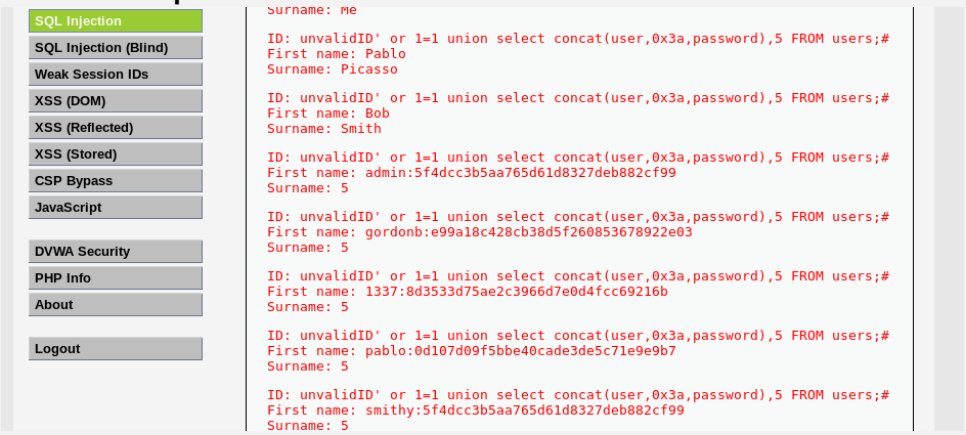
5.5 Deprecated Hash Function



Affected Systems: 172.17.0.2
Ressource: /vulnerabilities/sqli/ **&** /vulnerabilities/sqli_blind/
Risk: Low
Likelihood: Unlikely
Time: 01/01/2020 15:00
Impact: Moderate

Vulnerability Description:
The user passwords in the MySQL database are stored as MD5 hashes. MD5 is known to be deprecated and shouldn't be used anymore. If attackers could gain access to the database (compare Vulnerability 5.3), the hash values can be easily converted into cleartext user passwords.

Proof of Concept:



Recommendation:
Cryptographically strong hash functions should be used to generate hash values. The use of bcrypt is recommended.

6 Appendix

6.1 General Testing Procedure

Our approach is based on the methodology proposed in the paper "Study: A Penetration Testing Model" published by the German Federal Office for Information Security (BSI). The document was jointly authored with experienced companies in the IT security field. The methodology describes how to approach penetration tests efficiently and target-oriented.

Thus we also rely on the following scheme to achieve the best possible quality and significance of the penetration test.

1. **Research information about the target system**
   Computers that can be accessed over the internet must have an official IP address. Freely accessible databases provide information about the IP address blocks assigned to an organization.
2. **Scan target systems for services on offer**
   An attempt is made to conduct a port scan of the computer(s) being tested, open ports being indicative of the applications assigned to them.
3. **Identify systems and applications**
   The names and version of operating systems and applications in the target systems can be identified by "fingerprinting".
4. **Researching Vulnerabilities**
   Information about vulnerabilities of specific operating systems and applications can be researched efficiently using the information gathered.
5. **Exploiting vulnerabilities**
   Detected vulnerabilities can be used to obtain unauthorized access to the system or to prepare further attacks.

6.2 Webapplication Testing Procedure

Our penetrationtesting methodology of web applications is based on the "OWASP Testing Guide 4.0". The Open Web Application Security Project (OWASP) is a nonprofit foundation that works to improve the security of software.

We also test for the so-called "OWASP Top 10" (listed on the next page). With this, we want to ensure that vulnerabilities are tested in a systematically and reproducibly way. As a consequence, it is more unlikely to miss important aspects. The most important working steps involve:

1. **Business Processes & Application Logic Mapping**
   A profound understanding of all components, functionality and interfaces of the application that will be tested is essential. Therefore, we try to understand both the business logic and application logic. In order to do this, we read the documentation and interact with the application like a normal user in the first instance.
2. **Target Reconnaissance & Automated Scanning**
   Then, we manually connect to the application to confirm that it can be reached and tested. In this step we also perform so-called fuzzing to identify the potential attack surface, which is an automated scan of the application.
3. **Manual Web Vulnerability Testing**
   With all that information, a team of experienced penetration testers will manually evaluate results of the security scans. Then, the application will be manually tested for vulnerabilities that were not found using vulnerability scanners. In this phase we focus on the business critical components of the application. This is highly dependent on the application that has to be tested.
4. **Manual Web Vulnerability Exploitation (optional)**
   Identified vulnerabilities can be exploited to minimize false-positive results. However, if the availability or reliability of the application can be affected, we will not execute exploits without prior agreement.
5. **Risk Assessment**
   All findings will be assessed regarding to their likelihood and impact.

6.3 OWASP TOP 10

The OWASP Top 10 is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications. The newest publication from 2017 includes the following security risks:

A1 Injection

A2 Broken Authentication

A3 Sensitive Data Exposure

A4 XML External Entities (XXE)

A5 Broken Access Control

A6 Security Misconfiguration

A7 Cross-Site Scripting (XSS)

A8 Insecure Deserialization

A9 Using Components with Known Vulnerabilities

A10 Insufficient Logging & Monitoring

6.4 Risk Evaluation

A risk is always composed of the likelihood and the extent of damage.

The **likelihood** depends on the following factors:

- How easily can the vulnerability be identified?
- Are vulnerability scanners successful? (visibility)
- Are there public exploits available? (exploitability)
- Does the exploitation of the vulnerability require certain permissions? (privilege escalation)
- Is a vulnerability chain needed in order to exploit the vulnerability? (vulnerability chaining)
- Is the weakness due to social behaviour? (social engineering)

To determine the extent of the **potential impact** the following aspects must be considered:

- financial loss
- loss of reputation
- preparations of the attacker that could lead to a critical impact
- violation of the CIA Triad
  - **confidentiality:** the property, that information is not made available or disclosed to unauthorized individuals, entities, or processes
  - **integrity**: maintaining and assuring the accuracy and completeness of data over its entire lifecycle
  - **availability**: the degree to which a system or subsystem is in a specified operable and committable state

6.4.1 Classification Likelihood

When assessing the probability of occurrence, it is essential that in addition to "looking back" (empirical values, comparable events in other organizations, key figures, statistics, etc.), it is also essential to "look forward" in order to take previously unknown findings and developments into account (e.g., the emergence of new technologies or changed infrastructure).

The assessment of the factor likelihood is as follows, ordered by increasing frequency/probability:

Very Unlikely: 1x100 years or the exploitation of the vulnerability is almost impossible by experienced attackers

Unlikely: exploitation every 10-50 years or the exploitation of the vulnerability is only possible with considerable effort by experienced attackers

Possible: annual exploitation, or the exploitation of the vulnerability is possible by experienced attackers with reasonable effort

Likely: monthly exploitation or exploitation of the vulnerability is possible with little effort („script kiddies“ are able to exploit the vulnerability)

Very Likely: (multiple) daily exploitation or little effort to exploit the vulnerability (vulnerability is exploited during normal operation)

6.4.2 Classification Impact

When assessing the potential negative impact, the possible financial loss (fines, lost turnover opportunities) of a company must be taken into account. Gradual damage in the sense of a loss of reputation (loss of customers, decreased value of services or products), as well as violations of the CIA Triad (compromising a system) must be considered.

The assessment of the factor impact is as follows, sorted in ascending order:

Negligible: only informational or completely insignificant incident that can increase an attacker’s information base, however (normal business can be continued without any issues).

Minor: insignificant incident or a vulnerability that can significantly contribute to preparations for an attack.

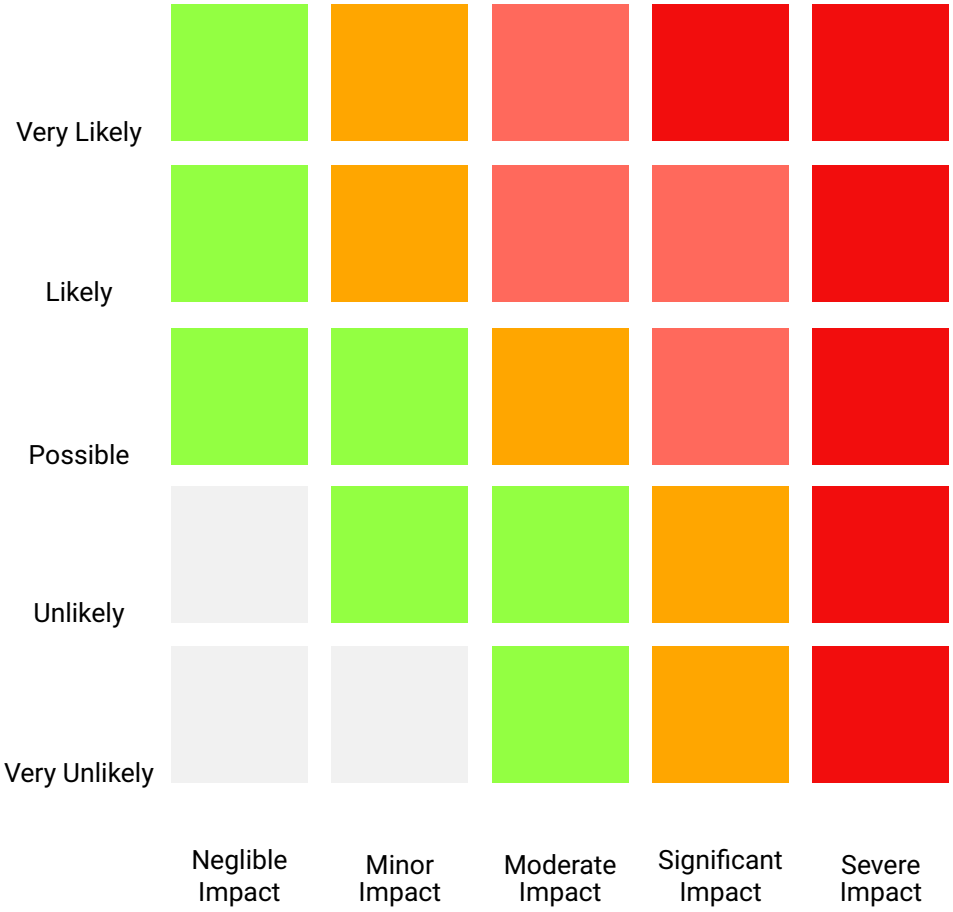
Moderate: significant incident. The company’s annual profit objectives may be at risk. A moderate loss of reputation is possible.

Significant: significant incident. The company’s profit objectives over a period of several years may be at risk or a significant loss of reputation is possible. Unprivileged access to a system.

Severe: the vulnerability can lead to an existential threat to the company or complete compromise of a system.

6.4.3 Risk Assessment Color Scheme

Boxes colored in dark red represent a critical vulnerabilitiy. Red boxes denote a high risk, while orange coloring means a moderate risk and green coloring means a low risk. Gray boxes mean just "informational".



The isolated assessment of one vulnerability often results in less severe estimates compared to the severity of a vulnerability chain. Therefore, we report always the more severe risk in the technical report to provide a holistic view. Vulnerabilities that are dependent on other vulnerabilities are referenced.

Risk treatment of a single (noncritical) vulnerability can reduce the aggregate risk of a vulnerability chain significantly. If you need assistance or have questions related to risk control strategies, please contact one of our experts, who will be glad to assist you.