Friedrich-Alexander-Universität
Technische Fakultät

CISO Sprechstunde
04.09.2024

Offensive Informationssicherheit

Bewertung der Informationssicherheit

Informationssicherheit testen und prüfen:

- Sicherheits-Audits und IT-Revisionen dienen der generellen Überprüfung der IT-Infrastruktur hinsichtlich Ordnungsmäßigkeit, Effizienz, Effektivität etc. und sind nicht zwingend auf die Aufdeckungen von angreifbaren Schwachstellen fokussiert.

  → Trotz erworbenen Zertifizierung, wissen wir nicht, ob unser System eine Schwachstelle hat.

- Offensive Sicherheit simuliert reale Angriffe, und hat das Ziel, die Sicherheitslücken und Schwachstellen in System zu identifizieren.

  → Betrachtet dennoch nicht die Einhaltung von Sicherheitsstandards (So wird z. B. in diesem Rahmen nicht geprüft, ob bestimmte Daten im Falle eines Hardwareschadens durch regelmäßige Datensicherung wiederherstellbar wären, sondern nur, ob auf diese Daten Zugriff erlangt werden könnte)

→ Kombination der beiden Methoden

BSI „Durchführungskonzept für Penetrationstests“

Informationssicherheit: Rollen

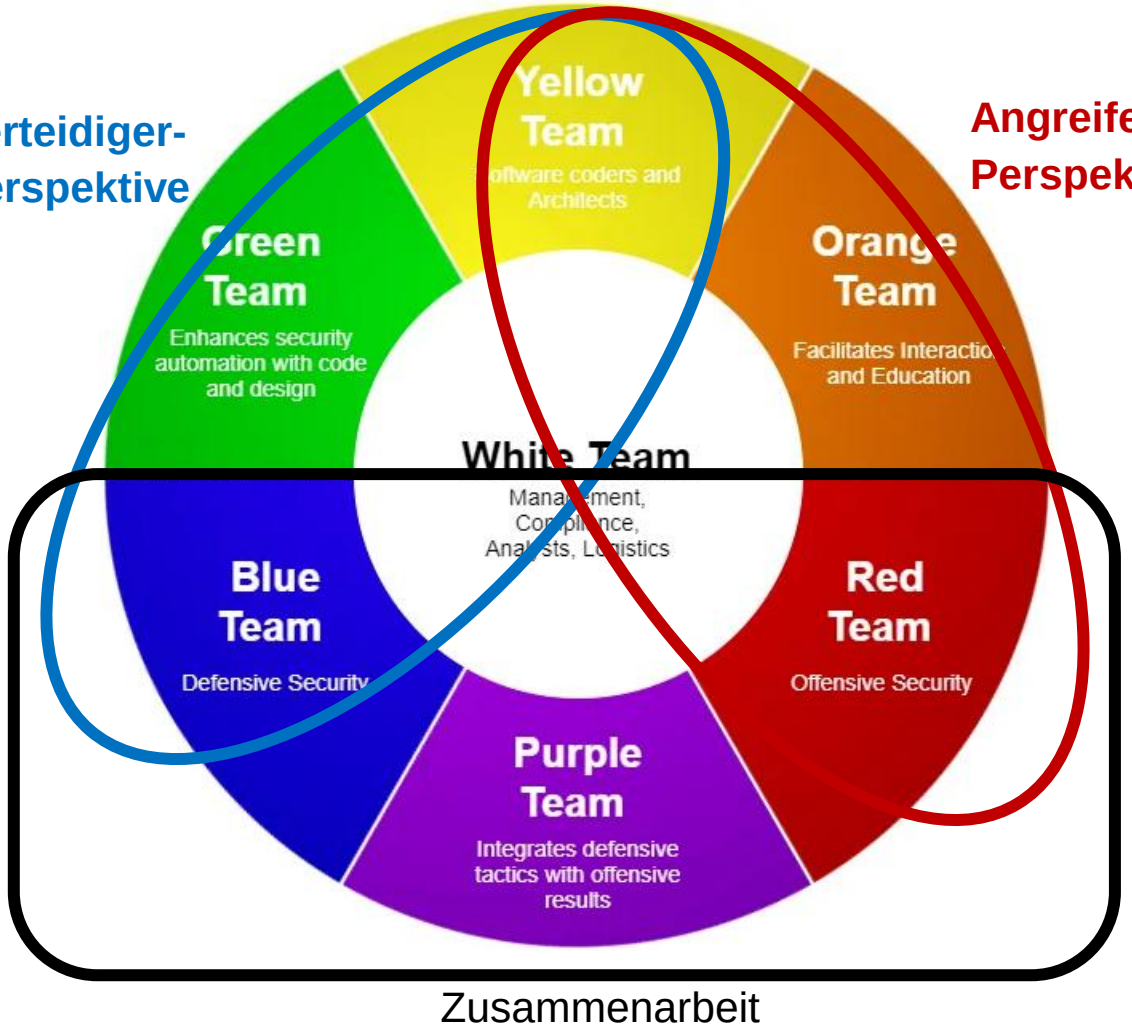
Offensive | Defensive | Architekten

https://medium.com/@dancovic/the-infosec-color-wheel-7e52fd822ae4

Informationssicherheit: Rollen

Verteidiger-Perspektive

Angreifer-Perspektive



Zusammenarbeit

Offensive | Defensive | Architekten

https://medium.com/@dancovic/the-infosec-color-wheel-7e52fd822ae4

Informationssicherheit: Rollen

- CISO
- SOC: Risiko- Notfall- Schwachstellenmanagement
- ISMS
- Richtlinien

https://medium.com/@dancovic/the-infosec-color-wheel-7e52fd822ae4

Red Team vs Pentesting

Abgrenzungen

Pentesting (= ethical hacker):

Fokus: Identifizierung von Schwachstellen in ausgewählten Systemen

- Sicherheitsbewertung, bei der ein erfahrener Tester eine Kombination aus Tools und manuellen Ausnutzungstechniken verwendet, um Schwachstellen zu identifizieren, die ein potenzieller Angreifer ausnutzen könnte
- Einmaliger, hoch-fokussierte Aufwand für die Bewertung einzelner Systeme

Red Teaming (= Nachahmung des Angriffes)

Fokus: Simulierung des kompletten Cyberangriffes auf die gesamte Infrastruktur

- Ansatz mit der Anwendung von Angriffsmodellierungstechniken
- Kontinuierliche strategische Planung aus Angreifer-Perspektive

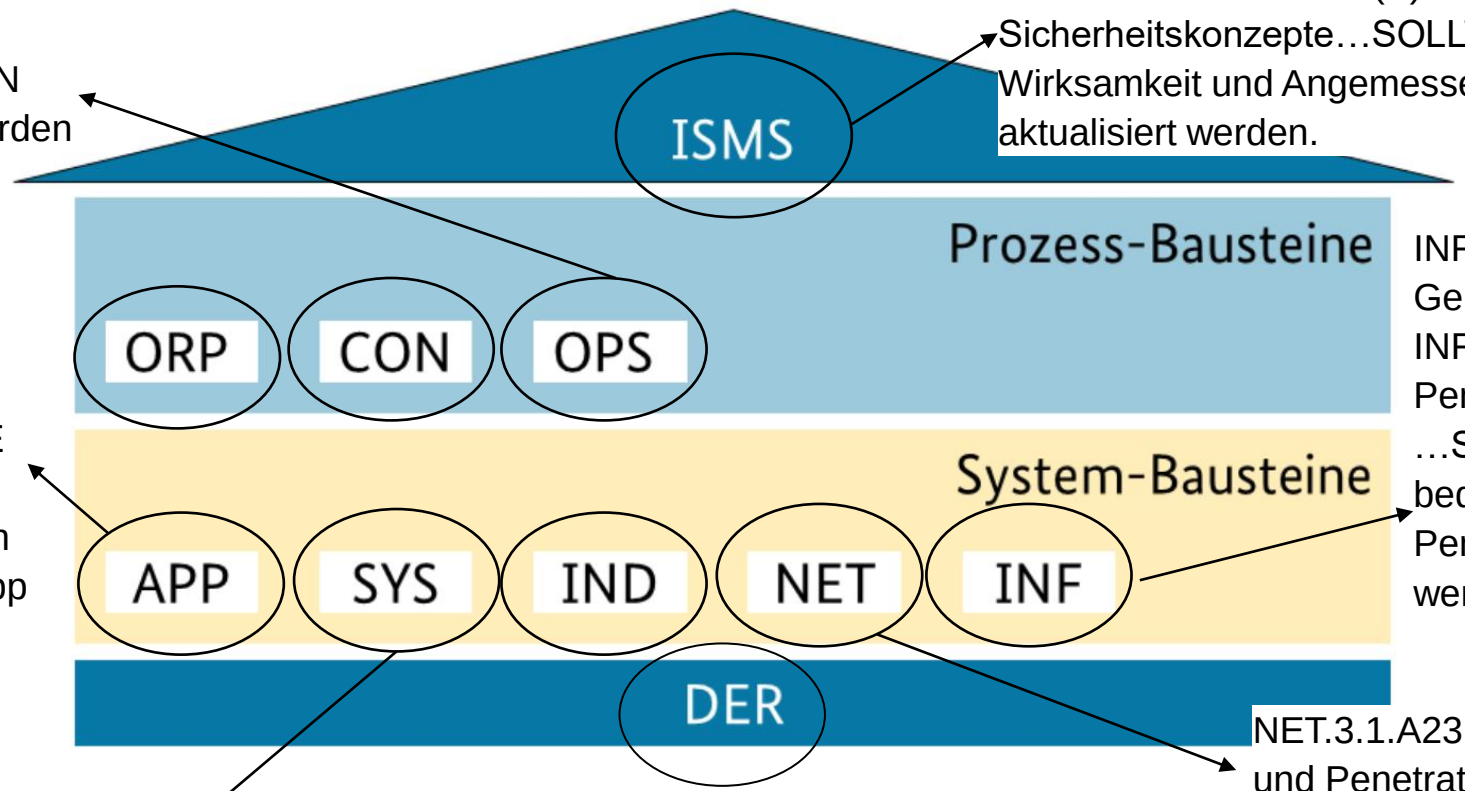
Teichmann, Fabian M., and Sonia R. Boticiu. "An overview of the benefits, challenges, and legal aspects of penetration testing and red teaming.", 2023

Pentesting und Red Team in BSI IT-Grundschutz

BSI IT Grundschutz: Bausteine

OPS.1.1.1 Allgemeiner IT-Betrieb
→ OPS.1.1.1.A23 (H)
Für alle IT-Komponenten SOLLTEN Penetrationstests durchgeführt werden

ISMS.1.A11 Aufrechterhaltung der Informationssicherheit (S) Der Sicherheitsprozess, die Sicherheitskonzepte…SOLLTEN regelmäßig auf Wirksamkeit und Angemessenheit überprüft und aktualisiert werden.



APP.1.4 Mobile Anwendungen (Apps)
APP.1.4.A15 (H)
Bevor … freigegeben wird, SOLLTE ein Penetrationstest durchgeführt werden. ..regelmäßig und zusätzlich bei größeren Änderungen an der App wiederholt werden

INF.13 Technisches Gebäudemanagement
INF.13.A30 Durchführung von Penetrationstests im TGM (H)
…SOLLTEN bedarfsorientiert Penetrationstests durchgeführt werden.

SYS.1.1.A24 Sicherheitsprüfungen für Server (S)

DER.1.A13 Regelmäßige Audits der Detektionssysteme (S)

NET.3.1.A23 /NET.3.2.A24 Revision und Penetrationstests (S)
Router, Switches, Firewall Struktur SOLLTEN regelmäßig auf bekannte Sicherheitsprobleme hin überprüft werden.

Pentest

Ziele

Ziele des Pentests:

- Erhöhung der Sicherheit der technischen Systeme
- Identifikation von Schwachstellen
- Bestätigung der IT-Sicherheit durch einen externen Dritten
- Erhöhung der Sicherheit der organisatorischen/personellen Infrastruktur

→ Penetrationstest stellt immer nur eine Momentaufnahme dar und kann daher keine Aussagen über das Sicherheitsniveau für die Zukunft geben

→ Die gründliche Durchführung kann zwar einen erfolgreichen Angriff nicht völlig ausschließen, sie reduziert jedoch die Wahrscheinlichkeit für einen erfolgreichen Angriff beträchtlich: Erweiterung der etablierten Prüfungshandlungen

Pentesting Klassifikation

Nach BSI

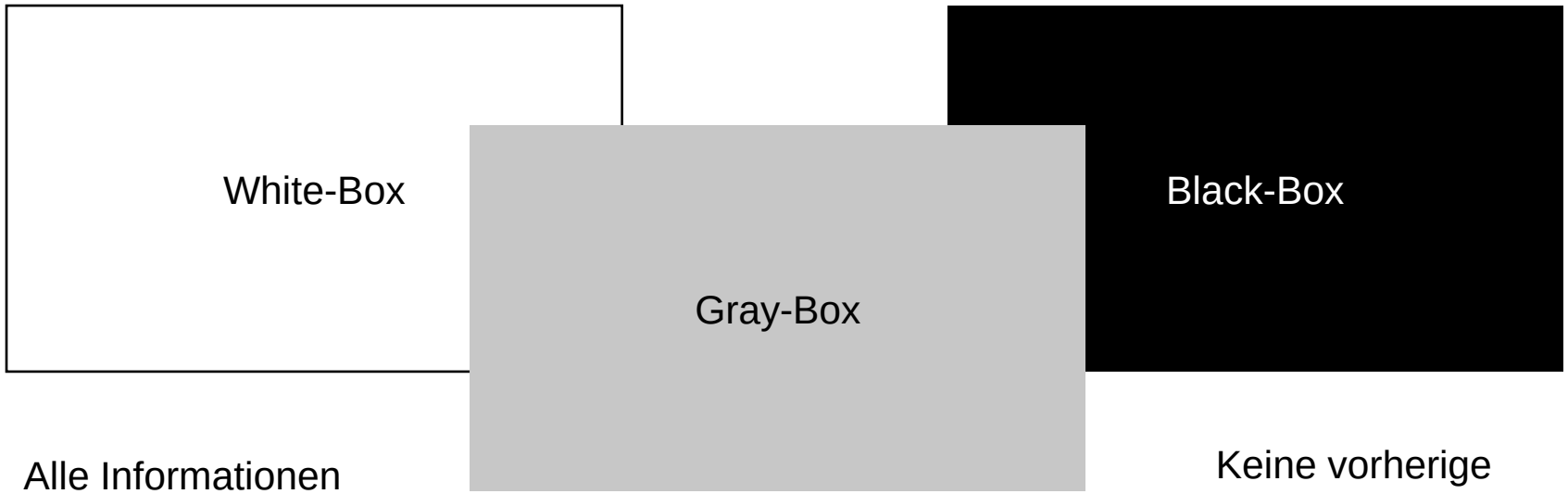
Klassifikationskriterien:

- **Informationsbasis**: Von welchem Wissensstand über das anzugreifende Netz bzw. Objekt geht der Penetrationstester aus?
- **Aggressivität**: Wie aggressiv geht der Penetrationstester beim Testen vor?
- **Umfang**: Welche Systeme sollen getestet werden?
- **Vorgehensweise**: Wie „sichtbar“ geht das Team beim Testen vor?
- **Ausgangspunkt**: Von wo aus wird der Penetrationstest durchgeführt?
- **Technik**: Welche Techniken werden beim Testen eingesetzt?

BSI „Durchführungskonzept für Penetrationstests“

Pentesting Kriterien

Informationsbasis



Shah, Sugandh, and Babu M. Mehtre. "An overview of vulnerability assessment and penetration testing techniques." , 2015

Pentesting Kriterien

Aggressivität

Aggressivitätsstufen:

- **Passiv**: mögliche Schwachstellen werden nicht ausgenutzt
- **Vorsichtig**: gefundene Schwachstellen nur dann ausgenutzt, wenn nach bestem Wissen eine Beeinträchtigung des untersuchten Systems ausgeschlossen werden kann
- **Abwägend**: es wird auch versucht, Schwachstellen auszunutzen die unter Umständen zu Systembeeinträchtigungen führen könnten. Allerdings wird vorher abgewägt, wie wahrscheinlich ein Erfolg ist und wie stark die Konsequenzen wären
- **Aggressiv**: es wird versucht, alle potentiellen Schwachstellen auszunutzen, wird auch bei nicht eindeutig identifizierten Zielsystemen eingesetzt oder Sicherungssysteme werden durch gezielte Überlastung deaktiviert.

Pentesting Kriterien

Umfang

Umfang des Pentestings:

- **Fokussiert**: nur ein bestimmtes Teilnetz, System oder ein bestimmter Dienst
- **Begrenzt**: eine begrenzte Anzahl von Systemen oder Diensten
- **Vollständig**: alle erreichbaren Systeme. Dabei ist zu beachten, dass auch bei einem vollständigen Test u. U. bestimmte Systeme, z. B. ausgelagerte und extern gehostete dennoch nicht geprüft werden dürfen

Pentesting Kriterien

Vorhegensweise

„Sichtbarkeit“ des Pentestings:

- **Verdeckt**: können nicht direkt als Angriffsversuche erkannt werden.

→ Prüfung von sekundären Sicherheits-Systemen und der vorhandenen Eskalationsprozeduren

- **Offensichtlich**

Pentesting Kriterien

Ausgangspunkt

Ausgangspunkt:

- Von **außen**: Überwindung von Zugangskontrollen (Firewall und Systeme in der DMZ, RAS-Verbindung..)
- Von **innen**: Zugriffsmöglichkeiten durch Personen mit Zugang zum internen Netzwerk

Pentesting Kriterien

Technik

Unterscheidung nach Techniken:

- „**klassisch**“: Nutzung von Funktionalitäten der eingesetzten Netzwerkprotokolle auf Netzwerkkomponenten, Computersysteme und oder Applikationen. Diese Art von Angriffen macht sich Schwachstellen oder Unzulänglichkeiten in Hard- und Software zu nutze.
- **KI**: Manipulation von KI-basierten Algorithmen durch Datenfehlklassifikation um false-posivite Ergebnisse zu produzieren und dabei authentifizierten Zugriff zu erlangen oder Schaden zu erzeugen.
- **Social Engineering**: Menschen mit privilegiertem Wissen insofern zu manipulieren, dass sie dem Angreifer sicherheitsrelevante Informationen, preisgeben.
- **Physisch**: Überwindung von physischen Sicherheitsmaßnahmen, um physischer Zugriff auf die IT-Systeme zu erlangen. Die Motivation ist dabei, auch ein Zugriff auf bzw. die Manipulation der gespeicherten Anwendungen und Daten.

„Klassisch“

Pentesting

Netzwerk:

- Simulierte Angriffe auf das Netzwerk, mit der Ausnutzung der Eigenschaften von: Netzwerkarchitektur und Topologie, Netzwerkdienste und Ports, Authentifizierung, Netzwerkprotokolle, Firewall- und Router, IDS und SIEM Konfigurationen, Datenverkehr

Software:

- Simulierte Angriffe auf Softwareanwendung: Eingabevalidierung und Benutzeroberfläche, Authentifizierung, Implementierungslogik, Anwendung von APIs, Server-Client Konfigurationen, Datenaustausch

Hardware:

- Physische Sicherheit der Komponente: Gehäuseschutz, physikalische Seiteffekte, Firmware, Schaltkreisen, Spannung- und Taktsignale, IoT Geräte, RFID und NFC Angriffe

Shah, Sugandh, and Babu M. Mehtre. "An overview of vulnerability assessment and penetration testing techniques." , 2015

Pentesting: Vorgehen und Methodik

Pentesting: Vorgehen

Phase 1: Vorbereitung

- Absichtserklärung: Umfang, Ziele, Zeitpunkt und Dauer
- Rahmenbedingungen
- Geheimhaltungsvereinbarung
- Vertraulichkeitsvereinbarung
- Risiko durch Straffreiheit und Haftungsbeschränkung
- Verhaltensregeln

….

Phase 2: Test

Anwendung von Techniken zur Erkennung und Ausnutzung der Schwachstellen

1. Informationsbeschaffung
2. Scannen
3. Schwachstellen-Mapping und Analyse
4. Angriffsgenerierung
5. Ausnutzen der Schwachstellen

Phase 3: Bericht

- Dokumentation der Untersuchung und Validierung der Testergebnisse
- Minderungsplan

Shah, Sugandh, and Babu M. Mehtre. "An overview of vulnerability assessment and penetration testing techniques." , 2015

Pentesting: Phase 2

Informationsbeschaffung

Verständnis fürs zu testende System – Erstellen eines möglichst vollständigen und genauen Profil des Sicherheitsstatus:

- Netzwerk
- Subsysteme
- Technologien
- Systemressourcen
- Anwendungen
- Kommunikationsinfrastruktur

Shah, Sugandh, and Babu M. Mehtre. "An overview of vulnerability assessment and penetration testing techniques." , 2015

Pentesting: Phase 2

Informationsbeschaffung

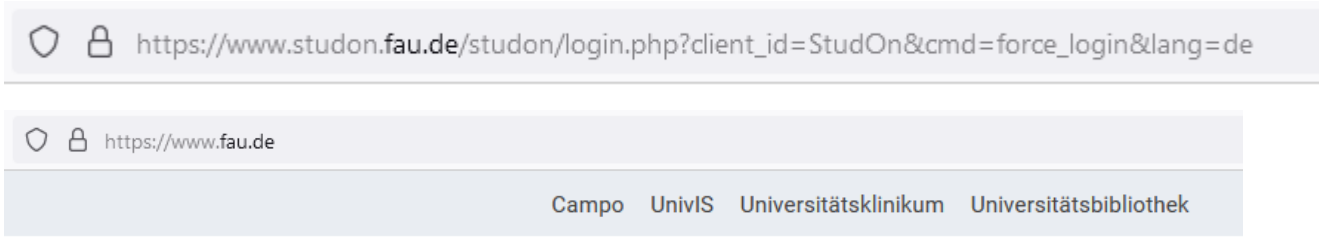
Domain Information

Domain:	fau.de
Updated On:	2018-02-19
Status:	connect
Name Servers:	dns-3.dfn.de ns1.rrze.uni-erlangen.de ns2.rrze.uni-erlangen.de tuminfo1.informatik.tu-muenchen.de

related domain names

denic.de dfn.de uni-erlangen.de tu-muenchen.de

https://www.whois.com



→ Recherche nach öffentlich zugänglichen Informationen

→ **Nmap (Network Mapper)** (und ähnliche Tools):

- Identifizierung der aktiven Geräte im Netz
- Suche nach offenen Ports
- Service- und Betriebssystemerkennung

Pentesting: Phase 2

Scannen

Suche nach Sicherheitslücken und Identifizierung von potentiellen Schwachstellen:

Manuell

Mit dem Einsatz von Tools:

- **Netzwerk:**

  Nessus, OpenVAS … Schwachstellenscanner:

  - Zuordnung zwischen gefunden Services und bekannten Schwachstellen für diese Services
  - Überprüfung der Konfigurationseinstellungen

- **Anwendungen**:

  Web: OWASP ZAP, Burp Suite, BeEF, Nikto, w3af ..

  Desktop: Statische (Flawfinder, Pychecker) und dynamische, binäre Analyse (Ghidra, IDA), Fuzzing…

  KI Anwendungen: Adversarial Robustness Toolbox , Python-Bibliotheken: CleverHans, Foolbox, SecML

Pentesting: Phase 2

Schwachstellen-Mapping und Analyse

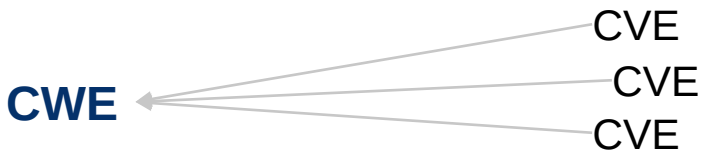
Analyse von potentiellen Schwachstellen:

- Überprüfung der potentiellen Schwachstellen
- Ermittlung von Schwere und Auswirkungen der gefundenen Schwachstellen
- Priorisierung der Schwachstellen

Shah, Sugandh, and Babu M. Mehtre. "An overview of vulnerability assessment and penetration testing techniques." , 2015

Common Weakness Enumeration (CWE)

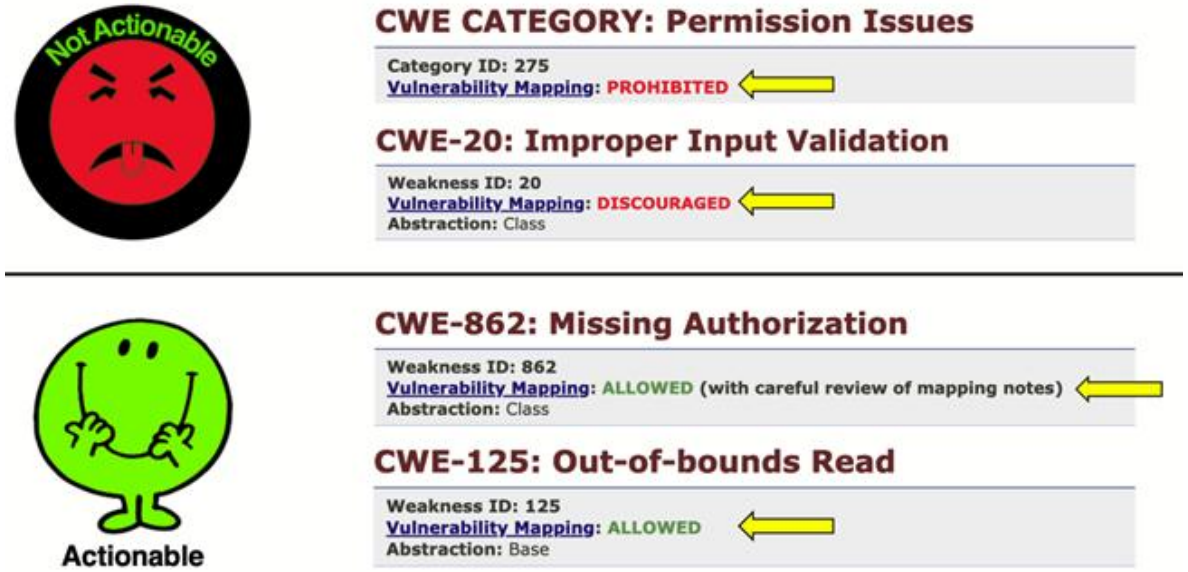
Von MITRE



→ Identifizierung der Grundursachen („Root Cause“) von Schwachstellen

→ Spezifisch & umsetzbar

→ Zuordnung zwischen CWEs und CVE



https://cwe.mitre.org/

Root Cause Mapping

Beispiel

Reference	Description
CVE-2001-1555	Terminal privileges are not reset when a user logs out.
CVE-2001-1514	Does not properly pass security context to child processes in certain cases, allows privilege escalation.
CVE-2001-0128	Does not properly compute roles.
CVE-1999-1193	untrusted user placed in unix "wheel" group
CVE-2005-2741	Product allows users to grant themselves certain rights that can be used to escalate privileges.
CVE-2005-2496	Product uses group ID of a user instead of the group, causing it to run with different privileges. This is resultant from some other unknown issue.
CVE-2004-0274	Product mistakenly assigns a particular status to an entity, leading to increased privileges.
CVE-2007-4217	FTP client program on a certain OS runs with setuid privileges and has a buffer overflow. Most clients do not need extra privileges, so an overflow is not a vulnerability for those clients.
CVE-2007-5159	OS incorrectly installs a program with setuid privileges, allowing users to gain privileges.
CVE-2008-4638	Composite: application running with high privileges (CWE-250) allows user to specify a restricted file to process, which generates a parsing error that leaks the contents of the file (CWE-209).
	alls some programs as setuid when they shouldn't be.
	dangerous procedures (Accessible entities).
	hod gets access to clipboard (Accessible entities).
	llows unprivileged users to modify source address of tities).
	an prevent setuid program from dropping privileges tions).

CWE-269: Improper Privilege Management

Weakness ID: 269
Vulnerability Mapping: DISCOURAGED
Abstraction: Class

Vulnerability Mapping Notes

Usage: DISCOURAGED *(this CWE ID should not be used to map to real-world vulnerabilities)*

Reason: Frequent Misuse

Rationale:

CWE-269 is commonly misused. It can be conflated with "privilege escalation," which is a technical impact that is listed in many low-information vulnerability reports [REF-1287]. It is not useful for trend analysis.

Comments:

If an error or mistake allows privilege escalation, then use the CWE ID for that mistake. Avoid using CWE-269 when only phrases such as "privilege escalation" or "gain privileges" are available, as these indicate technical impact of the vulnerability - not the root cause weakness. If the root cause seems to be directly related to privileges, then examine the children of CWE-269 for additional hints, such as Execution with Unnecessary Privileges (CWE-250) or Incorrect Privilege Assignment (CWE-266).

https://cwe.mitre.org/

Root Cause Mapping

Beispiel

CWE-250: Execution with Unnecessary Privileges

Weakness ID: 250
Vulnerability Mapping: ALLOWED
Abstraction: Base

Reference	Description
CVE-2007-4217	FTP client program on a certain OS runs with setuid privileges and has a buffer overflow. Most clients do not need extra privileges, so an overflow is not a vulnerability for those clients.
CVE-2008-1877	Program runs with privileges and calls another program with the same privileges, which allows read of arbitrary files.
CVE-2007-5159	OS incorrectly installs a program with setuid privileges, allowing users to gain privileges.
CVE-2008-4638	Composite: application running with high privileges (CWE-250) allows user to specify a restricted file to process, which generates a parsing error that leaks the contents of the file (CWE-209).
CVE-2008-0162	Program does not drop privileges before calling another program, allowing code execution.
CVE-2008-0368	setuid root program allows creation of arbitrary files through command line argument.
CVE-2007-3931	Installation script installs some programs as setuid when they shouldn't be.
CVE-2020-3812	mail program runs as root but does not drop its privileges before attempting to access a file. Attacker can use a symlink from their home directory to a directory only readable by root, then determine whether the file exists based on the response.
	Product launches Help functionality while running with raised privileges, allowing command execution using Windows message to access "open file" dialog.

Vulnerability Mapping Notes

Usage: ALLOWED *(this CWE ID could be used to map to real-world vulnerabilities)*

Reason: Acceptable-Use

Rationale:
This CWE entry is at the Base level of abstraction, which is a preferred level of abstraction for mapping to the root causes of vulnerabilities.

Comments:
Carefully read both the name and description to ensure that this mapping is an appropriate fit. Do not try to 'force' a mapping to a lower-level Base/Variant simply to comply with this preferred level of abstraction.

https://cwe.mitre.org/

CWE: Root Cause Mapping

2023 CWE Top 25

Rank	ID	Name	Score	CVEs in KEV	Rank Change vs. 2022
1	CWE-787	Out-of-bounds Write	63.72	70	0
2	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	45.54	4	0
3	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	34.27	6	0
4	CWE-416	Use After Free	16.71	44	+3
5	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	15.65	23	+1
6	CWE-20	Improper Input Validation	15.50	35	-2
7	CWE-125	Out-of-bounds Read	14.60	2	-2
8	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.11	16	0
9	CWE-352	Cross-Site Request Forgery (CSRF)	11.73	0	0
10	CWE-434	Unrestricted Upload of File with Dangerous Type	10.41	5	0
11	CWE-862	Missing Authorization	6.90	0	+5
12	CWE-476	NULL Pointer Dereference	6.59	0	-1
13	CWE-287	Improper Authentication	6.39	10	+1
14	CWE-190	Integer Overflow or Wraparound	5.89	4	-1
15	CWE-502	Deserialization of Untrusted Data	5.56	14	-3
16	CWE-77	Improper Neutralization of Special Elements used in a Command ('Command Injection')	4.95	4	+1
17	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	4.75	7	+2
18	CWE-798	Use of Hard-coded Credentials	4.57	2	-3
19	CWE-918	Server-Side Request Forgery (SSRF)	4.56	16	+2
20	CWE-306	Missing Authentication for Critical Function	3.78	8	-2
21	CWE-362	Concurrent Execution using Shared Resource with Improper Synchronization ('Race Condition')	3.53	8	+1
22	CWE-269	Improper Privilege Management	3.31	5	+7
23	CWE-94	Improper Control of Generation of Code ('Code Injection')	3.30	6	+2
24	CWE-863	Incorrect Authorization	3.16	0	+4
25	CWE-276	Incorrect Default Permissions	3.16	0	-5

https://cwe.mitre.org/

CWE:Top 1 Beispiel

CWE-787: Out-of-bounds Write

Weakness ID: 787
Vulnerability Mapping: ALLOWED
Abstraction: Base

Relevant to the view "Software Development" (CWE-699)

Nature	Type	ID	Name
MemberOf	C	1218	Memory Buffer Errors

Applicable Platforms

Languages

C *(Often Prevalent)*

C++ *(Often Prevalent)*

Class: Assembly *(Undetermined Prevalence)*

Technologies

Class: ICS/OT *(Often Prevalent)*

Detection Methods

Automated Static Analysis

This weakness can often be detected using automated static analysis tools. Many modern tools use data flow analysis or constraint-based techniques to minimize the number of false positives.

Automated static analysis generally does not account for environmental considerations when reporting out-of-bounds memory operations. This can make it difficult for users to determine which warnings should be investigated first. For example, an analysis tool might report buffer overflows that originate from command line arguments in a program that is not expected to run with setuid or other special privileges.

Effectiveness: High

Note: Detection techniques for buffer-related errors are more mature than for most other weakness types.

Automated Dynamic Analysis

This weakness can be detected using dynamic tools and techniques that interact with the software using large test suites with many diverse inputs, such as fuzz testing (fuzzing), robustness testing, and fault injection. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Common Consequences

Scope	Impact
Integrity Availability	**Technical Impact:** *Modify Memory; DoS: Crash, Exit, or Restart; Execute Unauthorized Code or Commands*

https://cwe.mitre.org/

Pentesting: Phase 2

Angriffsgenerierung

Identifizierung von Exploits, mit denen die priorisierte Schwachstellen gezielt angegangen werden können:

- Manuell/ Automatisiert
- Selbst erstellt/verfügbare Exploits/Tools
  - ExploitDB (https://www.exploit-db.com/)
  - Frameworks: Metasploit…
  - Tools mit dem Einsatz von KI: PentestGPT, DeepExploit…
  - Passwortcracker: John The Ripper, Hashcat, Cain and Abel…

Pentesting: Phase 2

Ausnutzen der Schwachstelle

Einsatz von Exploits mit dem Ziel, Zugriff auf die Systemressourcen zu erhalten:

- Kommunikation mit C2 Server
- Ausführung von Skripten/Tools..

Vorgehen nach OSSTMM

OSSTMM 3.0

The Open Source Security Testing Methodology Manual (OSSTMM):

- Entwickelt von Pete Herzog (Institute for Security and Open Methodologies (ISECOM))

> In art, the end result is a thing of beauty, whereas in science, the means of reaching the end result is a thing of beauty. When a security test is an art then the result is unverifiable and that undermines the value of a test. One way to assure a security test has value is to know the test has been properly conducted. For that you need to use a formal methodology. The OSSTMM aims to be it.

Definition von Security Test nach OSSTMM

OSSTMM 3.0

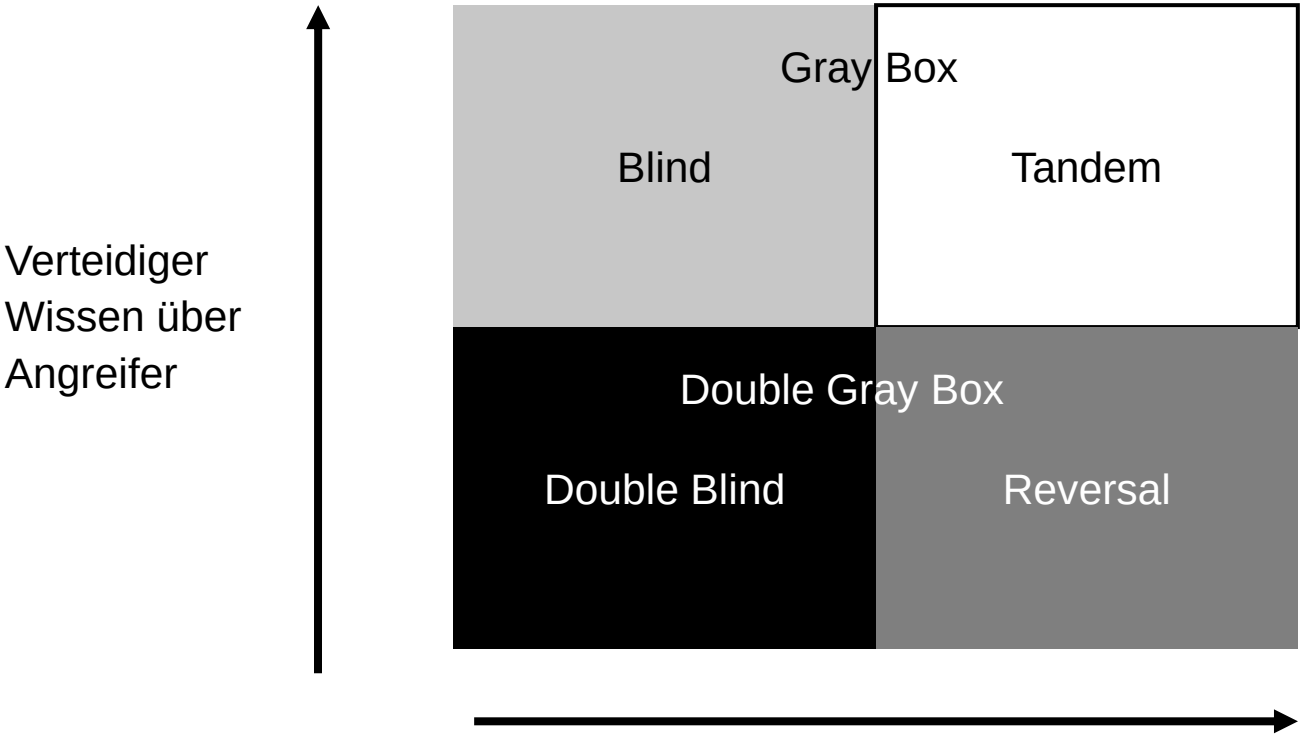
Sicherheitstest besteht aus 7 Schritten:

1. Definiere Zielsysteme (**Assets**). Assets haben Verteidigungsmechanismen (**Controls**). Sicherheitstest hat das Ziel, Schwachstellen in Verteidigungsmechanismen zu entdecken (**Limitiations**)
2. Definiere die zu testenden Zielsysteme (**engagement zone**)
3. Definiere alles außerhalb der zu testenden Systeme, die diese für Ihre Funktionalität brauchen (**scope**)
4. Definiere wie die zu testenden Systeme mit scope interagieren (**vectors**)
5. Definiere auf welchem Niveau (**channel**) passiert die Interaktion und teste alle Niveaus pro vector:
    1. Mensch
    2. Physisch
    3. Kabellos
    4. Telekommunikation
    5. Datennetzwerk

Definition von Security Test nach OSSTMM

Fortsetzung

6. Definiere Test-Typ



7. Definiere Regeln

→ **Berechne Attack Surface**

Security Metriken nach OSSTMM

OSSTMM 3.0

OPSEC

Visibility	1
Access	3
Trust	0
Total (Porosity)	4

CONTROLS

Class A

		Missing
Authentication	7	0
Indemnification	0	4
Resilience	0	4
Subjugation	0	4
Continuity	0	4
Total Class A	7	16

Class B

		Missing
Non-Repudiation	0	4
Confidentiality	0	4
Privacy	1	3
Integrity	0	4
Alarm	9	0
Total Class B	10	15

		True Missing
All Controls Total	17	31
Whole Coverage	42,50%	77,50%

LIMITATIONS

		Item Value	Total Value
Vulnerabilities	4	8,750000	35,000000
Weaknesses	5	5,000000	25,000000
Concerns	8	4,750000	38,000000
Exposures	0	5,025000	0,000000
Anomalies	0	4,250000	0,000000
Total # Limitations	17		98,0000

Limitations
15,930239

Security Δ
-17,72

True Protection
81,13

OPSEC
6,776361

True Controls
3,837843

Full Controls
4,986272

True Coverage A
20,00%

True Coverage B
25,00%

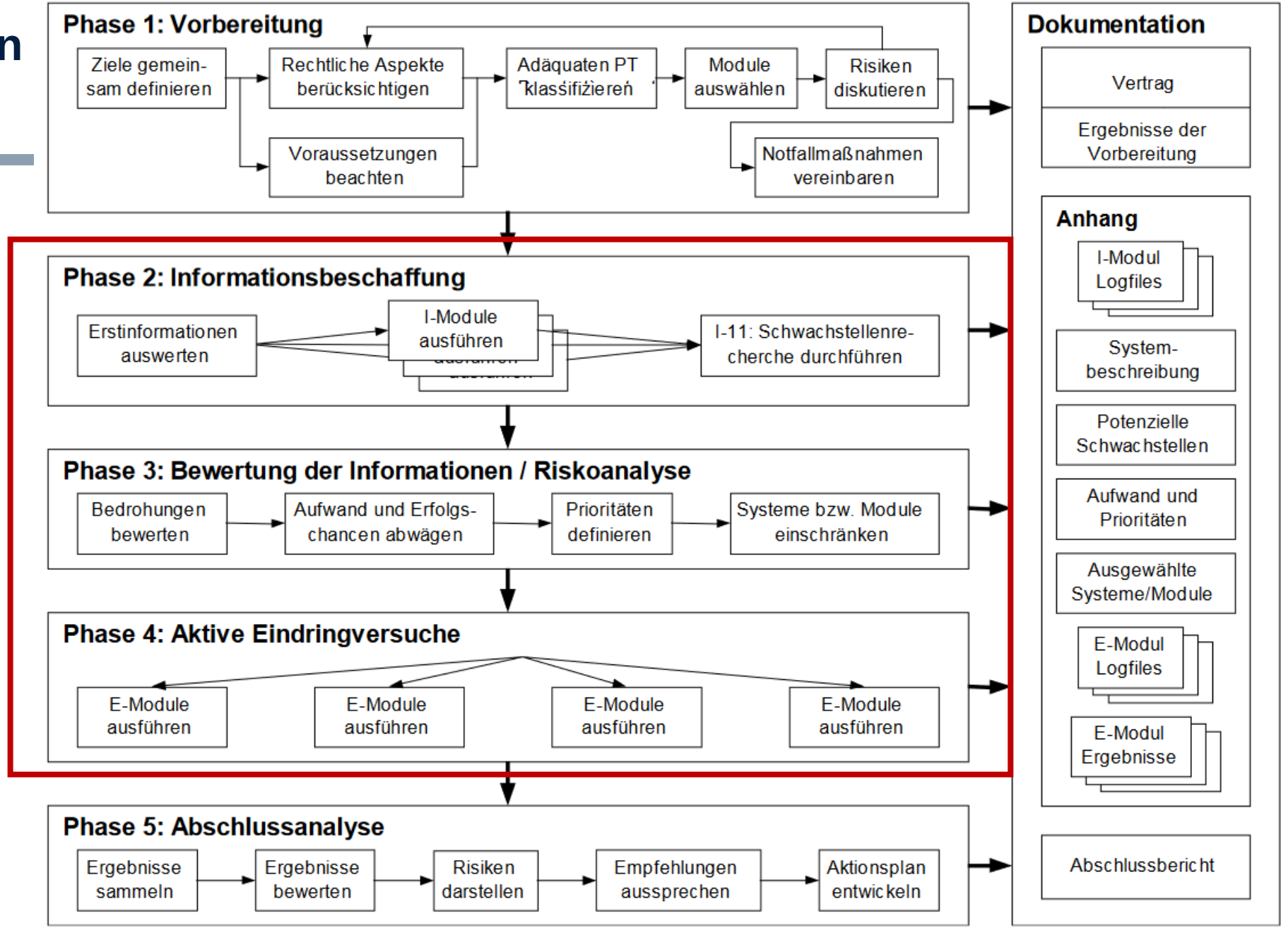
Total True Coverage
22,50%

Actual Security: 82,2269 ravs

Vorgehen

Pentesting nach BSI

Test Phase



BSI:Studie „Durchführungskonzept für Penetrationstests“

Module zur Informationsbeschaffung

Nach BSI

I 1 Auswertung öffentlich zugänglicher Daten

I 2 Verdeckte Abfragen von Netzwerkbasisinformationen

I 3 Offensichtliche Abfragen von Netzwerkbasisinformationen

I 4 Verdeckte Durchführung von Portscans

I 5 Offensichtliche Durchführung von Portscans

I 6 Identifikation von Anwendungen

I 7 Identifikation von Systemen

I 8 Verdeckte Identifikation der Router

I 9 Offensichtliche Identifikation der Router

I 10 Verdeckte Identifikation der Firewalls

I 11 Offensichtliche Identifikation der Firewalls

I 12 Recherche nach Schwachstellen

I 13 Identifikation von Anwendungsschnittstellen

I 14 Sammlung von Informationen für Social-Engineering

I 15 Sammlung von Informationen für computerbasiertes Social-Engineering

I 16 Sammlung von Informationen für persönliches Social-Engineering

I 17 Überprüfung der drahtlosen Kommunikation (nur scannend)

I 18 Test der Telefonanlage (Identifikation)

I 19 Test des Voicemailsystems (Identifikation)

I 20 Test des Faxsystems (Identifikation)

I 22 Identifikation von Zutrittskontrollen

I 21 Analyse der physischen Umgebung

I 22 Identifikation von Zutrittskontrollen

Module zur Informationsbeschaffung

Nach BSI; Beispiel

I 2. Verdeckte Abfragen von Netzwerkbasisinformationen

Es werden die grundlegenden Informationen über das zu überprüfende Netz als Basis für einen Penetrationstest durch unauffällige bzw. verdeckte Abfragen in Erfahrung gebracht.

Erwartete Ergebnisse:	**erledigt**
• Domain-Namen	☐
• IP-Adressbereiche	☐
• Hostnamen	☐
• IP-Adressen	☐
• Beschreibung der Server-Funktionen	☐
• ISP-Informationen	☐
• Ansprechpartner (Admin-C)	☐

Voraussetzungen:

IP-Adresse bzw. IP-Range oder Domain- bzw. Servernamen

Prüfungsschritte:	**Aufwand**
• Abfrage von öffentlichen Datenbanken (Whois, Ripe, Arin)	gering
• Abfrage von Name-Servern (Vorsicht: Versuch eines Zonentransfers könnte erkannt werden)	mittel
• Untersuchung der Informationen von E-Mail Headern	gering
• Untersuchung des HTML-Informationen der angebotenen Webseiten auf interne Links oder Kommentare	mittel
• Untersuchung von Newsgroups auf Postings von Mitarbeitern der Zielorganisation	gering
• Untersuchung von Stellenanzeigen der Zielorganisation auf Informationen zum IT-Umfeld	gering

Risiken:

Keine

Module für aktive Eindringversuche

Nach BSI

E 1 Verdeckte Verifikation tatsächlicher Schwachstellen

E 2 Offensichtliche Verifikation tatsächlicher Schwachstellen

E 3 Verifikation tatsächlicher Schwachstellen in Anwendungsschnittstellen

E 4 Verdeckter Test der Router

E 5 Offensichtlicher Test der Router

E 6 Test von Vertrauensbeziehungen zwischen Systemen

E 7 Verdeckter Test der Firewall von außen

E 8 Offensichtlicher Test der Firewall von außen

E 9 Beidseitiger Test der Firewall

E 10 Test des IDS-Systems

E 11 Abhören von Passwörtern

E 12 Test von Passwörtern

E 13 Test von „Denial-of-Service“ Anfälligkeit

E 14 Computerbasiertes Social-Engineering

E 15 Direktes, persönliches Social-Engineering mit physischem Zutritt

E 16 Indirektes, persönliches Social-Engineering ohne physischen Zutritt

E 17 Überprüfung der drahtlosen Kommunikation

E 18 Test der administrativen Zugänge zur Telefonanlage

E 19 Test des Voicemailsystems

E 20 Test der administrativen Zugänge zum Faxsystems

E 21 Test von Modems

E 22 Aktiver Test der Zutrittskontrollen

E 23 Überprüfung der Eskalationsprozeduren

Module für aktive Eindringversuche

Nach BSI; Beispiel

E 10. Test des IDS-Systems

Es wird getestet, ob ein evtl. vorhandenes IDS die potentielle Angriffe registriert und Alarme auslöst.

Erwartete Ergebnisse:	**erledigt**
• Typ des IDS	☐
• Verhalten des IDS auf verschiedene Angriffstypen	☐
• Aussage zur Performance des IDS	☐

Voraussetzungen:

Detaillierte System- und Firewallinformationen. Möglichkeit, die Alarmauslösung des IDS-Systems zu überwachen.

Prüfungsschritte:	**Aufwand**
• Durchführung von schrittweise offensichtlicheren Angriffsarten auf das Netzwerk der Zielorganisation	gering bis sehr hoch
• Evaluation der Reaktion des IDS auf die Angriffe	sehr hoch
• Abgleich der Angriffs- und IDS-Logfiles	hoch

Risiken:

Durch die Prüfungsschritte kann die Funktionsfähigkeit des Netzes der Zielorganisation beeinträchtigt werden.

Penetration Testing & Red Team: Gesetzliche und Ethische Normen

Gesetz als Motivation für Pentesting

Zwar existieren keine Gesetze, die eine Firma oder Behörde unmittelbar dazu verpflichten, Penetrationstests durchführen zu lassen, doch existieren verbindliche Vorschriften bezüglich

- der Handhabung der Sicherheit und der Verfügbarkeit von steuerrechtlich und handelsrechtlich relevanten Daten,
- des Umgangs mit personenbezogenen Daten,
- der Einrichtung und Ausgestaltung eines internen Kontrollsystem

BSI:Studie „Durchführungskonzept für Penetrationstests“

Gesetz als Motivation für Pentesting

Gesetz zur Kontrolle und Transparenz im Unternehmensbereich (KonTraG)

§ 91 Abs 2 AktG: „Der Vorstand hat geeignete Maßnahmen zu treffen, insbesondere ein Überwachungssystem einzurichten, damit den Fortbestand der Gesellschaft gefährdende Entwicklungen früh erkannt werden. [...]“

Bundesdatenschutzgesetz (BDSG)

§ 64 Abs 1 [**Anforderungen an die Sicherheit der Datenverarbeitung**]: „ Der Verantwortliche [..] haben unter Berücksichtigung des Stands der Technik, […] die Rechtsgüter der betroffenen Personen die erforderlichen technischen und organisatorischen Maßnahmen zu treffen, um bei der Verarbeitung personenbezogener Daten ein dem Risiko angemessenes Schutzniveau zu gewährleisten, [...] Der Verantwortliche hat hierbei die einschlägigen Technischen Richtlinien und Empfehlungen des Bundesamtes für Sicherheit in der Informationstechnik zu berücksichtigen.“

Telekommunikationsgesetz (TKG)

§ 164 Abs 1, 2 [**Technische und organisatorische Schutzmaßnahmen**] „Wer Telekommunikationsdienste erbringt oder daran mitwirkt, hat angemessene technische Vorkehrungen und sonstige Maßnahmen zu treffen […] Dabei ist der Stand der Technik zu berücksichtigen. […] angemessene technische und organisatorische Vorkehrungen und sonstige Maßnahmen zu treffen sofern diese Störungen durch äußere Angriffe und Einwirkungen von Katastrophen bedingt sein können, und zur Beherrschung der Risiken für die Sicherheit von Telekommunikationsnetzen und -diensten “

Zu beachtende Gesetze für Pentesting

Strafgesetzbuch (StGB)

§ 202a Abs 1 [**Ausspähen von Daten**]: „Wer unbefugt sich oder einem anderen Zugang zu Daten, die nicht für ihn bestimmt und die gegen unberechtigten Zugang besonders gesichert sind, unter Überwindung der Zugangssicherung verschafft, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft

§ 303a Abs 1 [**Datenveränderung**]: „Wer rechtswidrig Daten […] löscht, unterdrückt, unbrauchbar macht oder verändert, wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft“

§ 303b [**Computersabotage**]: „ Wer eine Datenverarbeitung, die für einen anderen/fremden Betrieb von wesentlicher Bedeutung ist, dadurch erheblich stört […] wird mit Freiheitsstrafe bis zu drei/fünf Jahren oder mit Geldstrafe bestraft. […] Der Versuch ist strafbar.“

Betriebsverfassungsgesetz (BetrVG)

§ 87 Abs. 1 Nr. 6 [**Mitbestimmungsrechte**]: „Der Betriebsrat hat [...], in folgenden Angelegenheiten mitzubestimmen: bei der Einführung und Anwendung von technischen Einrichtungen, die dazu bestimmt sind, das Verhalten oder die Leistung der Arbeitnehmer zu überwachen**.**

→ Eingriffe, deren Inhalt und Umfang müssen mit dem Auftraggeber abgestimmt sein und durch Einwilligung gerechtfertigt.

Etische Überlegungen

Offensive Sicherheit soll mit Respekt, Vertrauen, Professionalität und Ehrlichkeit durchgeführt werden:

- Transparenz
- Wahrhaftigkeit
- Verantwortungsbewusstsein
- Ausnutzung der gefundenen Schwachstelle
- Schutz der Privatsphäre
- Einsatz von Social Engineering und Ausnutzen von teilweise positiven „Schwächen“: Hilfsbereitschaft, Neugier, Pflichtbewusstsein