Penetration Testing

Strategies and Methods

Clemens H. **Cap**
ORCID: 0000-0003-3958-6136

Department of Computer Science
University of **Rostock**
Rostock, Germany
clemens.cap@uni-rostock.de

June 20, 2026

Master Course on
Cybersecurity and Applications of Cryptography

Contents

1. Foundations

Scope, ethics, and types of testing

What Is Penetration Testing?

Definition: Penetration testing is a cyberattack which is

1. **authorized** and
2. **simulated**
3. **to identify** vulnerabilities
4. **before** genuine adversaries do so.

Distinguishing Properties

- Includes actual exploitation, not only detection of possibilities
- Requires explicit **written** authorization
- Is deliberately time-boxed and scope-limited
- **Outcome** is an evidence-based risk assessment

Standards and Mandatory Pentesting

DORA **D**igital **O**perational **R**esilience **A**ct (EU)
EU based regulation for strengthening cybersecurity in the financial sector.
Requires pentesting for important entities on regular basis

NIS2 **N**etwork **I**nformation **S**ecurity regulation (EU)
Requires cybersecurity risk-management and their evaluation for clearly defined (1) essential and (2) important entities
But: No direct requirement for pentesting.

PCI-DSS **P**ayment **C**ard **I**ndustry **D**ata **S**ecurity **S**tandard (Int)
Requires pentesting annually and after significant changes.

SOC2 **S**ystem and **O**rganization **C**ontrols 2; US framework for accountants (US)
Recommends penetration tests.

HIPAA **H**ealth **I**nsurance **P**ortability and **A**ccountability **A**ct (US)
requires reisk analysis, risk management and security measures.
Proposes regular vulnerability scanning and pentesting

Why Does Pentesting Matter?

Business and Regulatory Drivers are as follows:

Compliance:	Regulations demand pentesting
Validation:	Confirms whether existing controls actually work
Evidence:	Supports risk-management and investment decisions
Detection:	Exposes gaps in monitoring and incident response

Example: Regulatory Trigger

A firm governed by PCI-DSS requirement 11.4 must perform external and internal penetration tests at least annually and after any significant infrastructure change.

Legal and Ethical Foundations

Authorization Is Non-Negotiable

Performing pentesting or similar methods without written permission is a criminal offense in virtually every jurisdiction.

US:	Computer Fraud and Abuse Act
Germany:	§202a of the StGB (criminal code)
UK:	Computer Misuse Act.

Ethical Obligations

1. **Confidentiality:** Treat all discovered data as strictly privileged
2. **Integrity:** Avoid destructive actions and protect stability
3. **Transparency:** Report all findings honestly and completely

RoE: Rules of Engagement

Purpose

The Rules of Engagement (RoE) form the contract that defines what the tester may do, against which targets, and under which constraints.

Mandatory Contents

- **Scope**: In-scope and out-of-scope ranges, domains, and applications
- **Techniques**: Permitted methods and escalation procedures
- **Timing**: Test windows and allowed intensity
- **Contacts**: Emergency contact chain and defined halt conditions
- **Handling**: Data storage, retention, and destruction obligations

Types of Penetration Tests

By Knowledge Level

Black-box	No prior information; simulates an external attacker
White-box	Full architecture and source access; maximizes coverage
Grey-box	Partial knowledge; balances realism against depth

By Target Domain

- Network and infrastructure testing
- Web, mobile, and API application testing
- Wireless, physical, and social engineering testing

Penetration Test versus Related Activities

Vulnerability Scan

- Broad, automated inventory of potential issues; no exploitation

Penetration Test

- Defined scope; vulnerabilities are exploited and proven

Red Team Engagement

- Objective-based and stealthy; measures detection and response

Comparison

The three differ in goal, not merely in effort.
Choose the activity that answers the client's actual question.

Example: Choosing a Test Type for a Scenario

Scenario

A retailer launches a new payment API and asks whether its security operations center would notice an intrusion.

Reasoning

- The question is about detection, so a stealthy red team fits best
- A grey-box scope accelerates progress while remaining realistic
- A pure vulnerability scan would answer a different, narrower question

Conclusion

Match the engagement type to the client's actual question: Here, resilience matters more than a vulnerability inventory.

Self-Test: Foundations (1)

Recall

1. State three properties that distinguish a penetration test from a vulnerability scan
2. Which document defines scope, permitted techniques, and halt conditions?
3. Name two statutes under which unauthorized testing is prosecuted
4. Differentiate black-box, grey-box, and white-box testing

Self-Test: Foundations (2)

Apply

1. A client wants to know whether its SOC detects intrusions. Which engagement type fits, and why?
2. During testing you find that an out-of-scope system is trivially exploitable. What is the correct course of action?
3. Explain why being time-boxed and scope-limited is a defining property rather than a weakness of penetration testing

2. Methodology

Frameworks that structure an engagement

Established Methodologies

Why Use a Methodology?

Methodologies ensure

1. repeatability
2. completeness, and
3. comparability

of results across engagements and across testers.

Common Standards for Pentesting Methodologies

1. **PTES:** **P**enetration **T**esting **E**xecution **S**tandard
   A seven-phase execution standard for full engagements
2. **OWASP:** **O**pen **W**orldwide **A**pplication **S**ecurity **P**roject:
   Web Security Testing Guide and Guidebook
3. **NIST:** National Institute of Standards and Technology
   Document SP 800-115
   Technical Guide to Information Security Testing and Assessment
4. **OSSTMM:** Open Source Security Testing Methodology Manual
5. **MITRE:** ATT&CK: An adversary technique knowledge base [Str+20]

The PTES Lifecycle

Abb. 1: The **seven phases** of the Penetration Testing Execution Standard, executed iteratively until the engagement objectives are met. Rechte s. Anhang.

NIST SP 800-115 and OSSTMM

NIST SP 800-115 Technical Guide

A technical reference that structures testing into planning, discovery, attack, and reporting, with emphasis on documentation and repeatability [Sca+08].

OSSTMM Open Source Security Testing Methodology Manual

A metrics-driven methodology centered on attack surface:
Quantifies operational security through a Risk Assessment Values score.

When to Combine

A regulated client may require NIST-aligned reporting.
The tester might use PTES and OSSTMM internally to organize the workflow.

MITRE ATT&CK as a Reference Framework

Structure	
Tactics	Adversary goals, such as Initial Access or Lateral Movement
Techniques	How a goal is achieved Example: Document T1566 on Phishing Techniques
Procedures	Concrete usage observed from real threat actors

Why It Helps

- It maps findings to documented adversary behavior [Str+20]
- It provides a shared vocabulary for offensive and defensive teams
- Separate matrices exist for Enterprise, Mobile, and ICS environments

The Cyber Kill Chain and ATT&CK

The Cyber Kill Chain

A seven-stage model of an intrusion, from reconnaissance through weaponization, delivery, exploitation, installation, command-and-control, and actions on objectives.

Relationship to ATT&CK

- The Kill Chain describes the high-level sequence of an attack
- ATT&CK enumerates the concrete techniques within each stage
- Used together, they connect strategic narrative to tactical detail

Threat Modeling with STRIDE

The STRIDE Categories

S Spoofing identity
T Tampering with data
R Repudiation of actions
I Information disclosure
D Denial of service
E Elevation of privilege

Role in Testing

Threat modeling is performed early, before active testing: It focuses effort on the highest-value attack paths.

3. Discovery

Reconnaissance, scanning, and enumeration

Passive Reconnaissance

Goal

Collect information about the target
without sending a single packet to the target's own infrastructure.

Common Sources

- WHOIS, DNS lookup tools, and certificate transparency logs
- Internet-wide scan databases such as Shodan
- Public profiles, code repositories, and job postings to understand used tech stack
- Wayback Machine for historical and forgotten content

Operational Security

Even passive recon leaves traces in third-party logs.
True passivity requires anonymizing infrastructure.

Active Reconnaissance

Goal

Interact directly with target systems
to enumerate live hosts, open ports, and running services.

Core Activities

1. **Host discovery:** ICMP sweeps, ARP scans on the LAN, and TCP probes
2. **Port scanning:** SYN scans for TCP, dedicated scans for UDP services
3. **Service detection:** Version identification and banner grabbing
4. **OS fingerprinting:** Inference from TCP/IP stack behavior

Trade-off

Aggressive timing finds more in less time
but is trivially detected by any competent **intrusion detection system**.

OSINT Tooling

Core Tools

theHarvester	Aggregates emails, hostnames, and IPs from public sources
Maltego	Builds graph-based relationship maps of entities
Amass	Enumerates subdomains via passive and active methods

Practical Value

Aggregated OSINT often reveals shadow-IT assets, leaked credentials, and the precise technology stack before any active scan is run.

DNS and Subdomain Enumeration

Techniques

- Zone-transfer attempts against misconfigured name servers
- Brute-force resolution against curated subdomain wordlists
- Passive harvesting from certificate transparency logs
- Permutation and alteration scanning for predictable names

Why It Matters

Subdomains frequently expose staging, admin, and legacy services that are weaker than the primary site and that the organization has forgotten it operates.

Network Scanning with Nmap

Essential Scan Types

`-sS`	TCP SYN (stealth) scan; the common default, needs root
`-sU`	UDP scan; slow; required for DNS, SNMP, and NTP
`-sV`	Service and version detection
`-sC`	Runs the default set of NSE scripts
`-A`	Runs aggressive set of detections

Practical Notes

- Nmap timing templates trade speed against stealth
- Output formats include normal, XML, and grepable for later parsing

Example: Scanning with nmap

```
nmap -sV 192.168.1.10
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 9.8
80/tcp  open  http    nginx 1.28
443/tcp open  https   nginx 1.28
```

Src. 1: Using nmap to find services and their versions

Examples: Service Enumeration

SMB on TCP 445

Enumerate shares, users, and domain information with `enum4linux-ng` or `CrackMapExec`; default and null sessions are common findings.

LDAP on TCP 389 and 636

Anonymous binds frequently expose Active Directory objects.
Useful for BloodHound analysis.

SNMP on UDP 161

Default community strings such as `public` are widespread.
A `snmpwalk` can reveal routing tables and installed software.

Fuzzing

Definition: Fuzzing

Automated software testing method that injects invalid, malformed, or unexpected inputs into a system to reveal software defects and vulnerabilities.

Monitors for exceptions such as crashes or information leakage.

Example: Web Application Enumeration

Discovery Activities

- **Content discovery:** Directory and file brute-forcing with `ffuf` fuzz faster u fool
- **Fingerprinting:** Technology detection from response headers
- **Parameter discovery:** Locating hidden inputs and endpoints
- **Client-side review:** Mining JavaScript for endpoints and secrets

Wordlist Quality Matters

Curated, scenario-specific wordlists dramatically outperform generic dictionaries.

Task: Build a Reconnaissance Plan

Scenario

You are engaged for a grey-box test of `example-corp.com`.
You receive only the primary domain and one low-privilege account.

Deliverables

1. List five passive sources you would consult first, and justify each
2. Define which active scans you would run, and in which order
3. Specify two measures that limit your operational footprint
4. Identify which findings would change your subsequent strategy

4. Exploitation

Confirming and leveraging vulnerabilities

Automated Vulnerability Scanning

Network-Level Scanners

- Nessus: Commercial, comprehensive CVE-based scanning
- OpenVAS: An open-source alternative with a maintained feed

Web Application Scanners

- Burp Suite: A proxy with active scanning and manual interception
- OWASP ZAP: An open-source, scriptable scanner for pipelines

Manual Vulnerability Analysis

Why It Is Indispensable

Automated scanners produce false positives and miss entire classes of flaws.

What Scanners Miss

- Business-logic flaws that depend on application semantics
- Chained, multi-step vulnerabilities
- Authorization edge cases and novel zero-day issues

Key Techniques

- Static source-code review and binary analysis
- Manual request manipulation through an intercepting proxy

CVE, CVSS, and Prioritization

Identifiers and Scores

Common **V**ulnerabilities and **E**xposures (CVEs) uniquely names disclosed vulnerabilities
Common **V**ulnerability **S**coring **S**ystem (CVSS) provides a metric

Score Is Not Risk

A critical CVSS score on an isolated, well-controlled system may pose less business risk than a medium score on a public API: Always apply contextual judgement.

Materials: CVE Database and Example of a CVE

The Exploitation Phase

Goal

Prove that an identified vulnerability is genuinely exploitable in the target environment, not merely theoretically present.

Operating Principles

- Obtain a shell, token, or data access that demonstrates real impact
- Document every step precisely for later reproduction
- Stay within the **R**ules **o**f **E**ngagement (RoE)
- Be Moderate: No destructive payloads, no excess exfiltration.
- Prioritize stability: Avoid crashing production systems

The Metasploit Framework

Architecture	
Modules	Exploits, payloads, auxiliaries, and post-exploitation tools
msfconsole	The primary interactive interface
Meterpreter	An in-memory post-exploitation agent

Typical Workflow

1. Search for a module by CVE or service name in Metasploit
2. Configure target and payload options
3. Verify reachability, then execute and validate the result

Common Network Exploitation Techniques

Credential-Based Attacks

- Password spraying against SSH, RDP, SMB, and WinRM
- Default credentials on network devices and IoT hardware
- Pass-the-Hash against NTLM-authenticated services

Protocol and Service Exploitation

- BlueKeep (CVE-2019-0708): A pre-authentication RDP flaw
- Log4Shell (CVE-2021-44228): JNDI injection through logging

Web Attacks: SQL Injection and XSS

SQL Injection

Unsanitized input is interpreted as SQL, altering the intended query.

Cross-Site Scripting

Stored	Persists server-side; executes for every visitor
Reflected	Travels in a crafted URL; needs social engineering
DOM-based	Lives in client-side JavaScript; the server never sees it

Web Attacks: SSRF and Authentication Flaws

Server-Side Request Forgery

The server fetches an attacker-controlled URL, granting access to internal resources unreachable from the Internet, such as a cloud metadata endpoint.

Authentication and Session Flaws

- Weak or guessable session identifiers and password-reset tokens
- Insecure JSON Web Tokens with weak or absent signatures
- Misconfigured single sign-on and open redirectors

Example: Proving Exploitability Safely

Situation

You suspect SQL injection in a login parameter and must demonstrate impact without harming production data.

A Proportionate Proof

- Extract only the database version string or a single dummy row
- Capture the exact request and response as evidence
- Stop short of dumping customer tables, which is unnecessary and harmful

Principle

Evidence must prove the vulnerability, not maximize damage.
Minimal proof is both sufficient and ethical.

5. Specialized Testing

Wireless, cloud, social, and evasion

Wireless Network Testing

WPA2-Personal

Capture the four-way handshake, then crack the pre-shared key offline.

Practical Notes

- An Evil Twin access point captures enterprise challenge-response credentials
- Tooling includes Aircrack-ng and hashcat
- Authorization must cover every access point in scope, including guests

Physical Penetration Testing

Objective

Test physical controls: Barriers, access systems, surveillance, and the response of security personnel.

Common Techniques

- **Tailgating:** Following authorized staff through a secured door
- **Badge cloning:** Reading and replaying low-frequency access cards
- **Lock bypass:** Defeating mechanisms without destructive entry
- **Dropped devices:** What do users do with devices they find?

Validate the security control measures in place!

Social Engineering and Phishing

Primary Vectors

Phishing	A mass email with a malicious link or attachment
Spear phishing	A targeted, personalized variant
Vishing	Voice-based manipulation, for example a fake helpdesk
Pretexting	A fabricated scenario that establishes trust

Ethics

Stay within the authorized personnel list.
Never impersonate law enforcement or emergency services.

Cloud Penetration Testing

Scope Considerations

Providers publish testing policies: Most service-level testing no longer needs prior approval, yet activities such as denial-of-service simulation remain prohibited.

Representative Attack Vectors

- SSRF against the metadata endpoint to steal identity credentials
- Misconfigured object-storage policies exposing data publicly
- Overly permissive identity roles enabling privilege escalation

6. Reporting & Defense

Communicating and remediating risk

The Penetration Test Report

Two Audiences

Executive summary	Business risk and strategy for leadership
Technical report	Findings and reproduction steps for engineers

Recommended Structure

1. Scope and methodology
2. Executive summary with an overall risk rating
3. Findings ordered by severity
4. Appendices with raw output and proof-of-concept evidence

Risk Rating and Severity

Qualitative CVSS Mapping

Severity	CVSS Range	Typical SLA
Critical	9.0–10.0	Immediate
High	7.0–8.9	Two weeks
Medium	4.0–6.9	Thirty days
Low	0.1–3.9	Ninety days

Source: CVSS v3.1 Specification [For19].

Always Adjust for Context

A finding reachable only from an isolated segment is rarely critical in practice, whatever its base score.

Defense, Remediation, and Hardening

Durable Defenses

- Defense in depth: Layered controls so one failure is not catastrophic
- Zero Trust: No implicit trust by location; continuous verification
- Foundations: Patching, hardening baselines, and least privilege

The Professional Mindset

- Enumerate thoroughly before exploiting, and document everything
- Authorization and scope compliance are non-negotiable
- Clear reporting connects technical findings to business risk

Appendix

Übersicht

Literaturverzeichnis

[For19] Forum of Incident Response and Security Teams. *Common Vulnerability Scoring System Version 3.1: Specification Document*. Online. 2019. URL: https://www.first.org/cvss/v3.1/specification-document (cit. on p. 50).

[Sca+08] Karen Scarfone et al. *Technical Guide to Information Security Testing and Assessment*. Tech. rep. NIST Special Publication 800-115. National Institute of Standards and Technology, 2008. DOI: 10.6028/NIST.SP.800-115. URL: https://doi.org/10.6028/NIST.SP.800-115 (cit. on p. 18).

[Str+20] Blake E. Strom et al. *MITRE ATT&CK: Design and Philosophy*. Tech. rep. MP180360R1. The MITRE Corporation, 2020. URL: https://attack.mitre.org/docs/ATTACK_Design_and_Philosophy_March_2020.pdf (cit. on pp. 16, 19).

Programmquellenverzeichnis

Verzeichnis aller Abbildungen

Rechtsnachweise

Rechtliche Hinweise (1)

Die hier angebotenen Inhalte unterliegen deutschem Urheberrecht. Inhalte Dritter werden unter Nennung der Rechtsgrundlage ihrer Nutzung und der geltenden Lizenzbestimmungen hier angeführt. Auf das Literaturverzeichnis wird verwiesen. Das Zitatrecht in dem für wissenschaftliche Werke üblichen Ausmaß wird beansprucht. Wenn Sie eine Urheberrechtsverletzung erkennen, so bitten wir um Hinweis an den auf der Titelseite genannten Autor und werden entsprechende Inhalte sofort entfernen oder fehlende Rechtsnennungen nachholen. Bei Produkt- und Firmennamen können Markenrechte Dritter bestehen. Verweise und Verlinkungen wurden zum Zeitpunkt des Setzens der Verweise überprüft; sie dienen der Information des Lesers. Der Autor macht sich die Inhalte, auch in der Form, wie sie zum Zeitpunkt des Setzens des Verweises vorlagen, nicht zu eigen und kann diese nicht laufend auf Veränderungen überprüfen.
Alle sonstigen, hier nicht angeführten Inhalte unterliegen dem Copyright des Autors, Prof. Dr. Clemens Cap, ©2020. Wenn Sie diese Inhalte nützlich finden, können Sie darauf verlinken oder sie zitieren. Jede weitere Verbreitung, Speicherung, Vervielfältigung oder sonstige Verwertung außerhalb der Grenzen des Urheberrechts bedarf der schriftlichen Zustimmung des Rechteinhabers. Dieses dient der Sicherung der Aktualität der Inhalte und soll dem Autor auch die Einhaltung urheberrechtlicher Einschränkungen wie beispielsweise Par 60a UrhG ermöglichen.
Die Bereitstellung der Inhalte erfolgt hier zur persönlichen Information des Lesers. Eine Haftung für mittelbare oder unmittelbare Schäden wird im maximal rechtlich zulässigen Ausmaß ausgeschlossen, mit Ausnahme von Vorsatz und grober Fahrlässigkeit. Eine Garantie für den Fortbestand dieses Informationsangebots wird nicht gegeben.
Die Anfertigung einer persönlichen Sicherungskopie für die private, nicht gewerbliche und nicht öffentliche Nutzung ist zulässig, sofern sie nicht von einer offensichtlich rechtswidrig hergestellten oder zugänglich gemachten Vorlage stammt.
Use of Logos and Trademark Symbols: The logos and trademark symbols used here are the property of their respective owners. The YouTube logo is used according to brand request 2-9753000030769 granted on November 30, 2020. The GitHub logo is property of GitHub Inc. and is used in accordance to the GitHub logo usage conditions https://github.com/logos to link to a GitHub account. The Tweedback logo is property of Tweedback GmbH and here is used in accordance to a cooperation contract.

Rechtliche Hinweise (2)

Disclaimer: Die sich immer wieder ändernde Rechtslage für digitale Urheberrechte erzeugt ein nicht unerhebliches Risiko bei der Einbindung von Materialien, deren Status nicht oder nur mit unverhältnismäßig hohem Aufwand abzuklären ist. Ebenso kann den Rechteinhabern nicht auf sinnvolle oder einfache Weise ein Honorar zukommen, obwohl deren Leistungen genutzt werden.
Daher binde ich gelegentlich Inhalte nur als Link und nicht durch Framing ein. Lt EuGH Urteil 13.02.2014, C-466/12 (Pressemitteilung, Blog-Beitrag, Urteilstext). ist das unbedenklich, da die benutzten Links ohne Umgehung technischer Sperren auf im Internet frei verfügbare Inhalte verweisen.
Wenn Sie diese Rechtslage stört, dann setzen Sie sich für eine Modernisierung des völlig veralteten Vergütungs- und Anreizsystems für urheberrechtliche Leistungen ein. Bis dahin klicken Sie bitte auf die angegebenen Links und denken Sie darüber nach, warum wir keine für das digitale Zeitalter sinnvoll angepaßte Vergütungs- und Anreizsysteme digital erbrachter Leistungen haben.
Zu Risiken und Nebenwirkungen fragen Sie Ihren Rechtsanwalt oder Gesetzgeber.
Weitere Hinweise finden Sie im Netz hier und hier oder hier.

Zitierweise dieses Dokuments

Wenn Sie Inhalte aus diesem Werk nutzen oder darauf verweisen wollen,
zitieren Sie es bitte wie folgt:

Clemens H. Cap: Penetration Testing. Electronic document. https://iuk.one/1033-1093 28. 6. 2026.

Bibtex Information: https://iuk.one/1033-1093.bib

```
@misc{doc:1033-1093,
    author        = {Clemens H. Cap},
    title         = {Penetration Testing},
    year          = {2026},
    month         = {6},
    howpublished  = {Electronic document},
    url           = {https://iuk.one/1033-1093}
}
```

Typographic Information:

Typeset on 2026-06-28 at 19:00
This is pdfTeX, Version 3.141592653-2.6-1.40.29 (TeX Live 2026) kpathsea version 6.4.2
This is pgf in version 3.1.11a
This is preamble-slides.tex ©C.H.Cap
Jobtype flag=foreground
Active modes (if any): biblioKeywords: keywords
Subject: subject

Abkürzungsverzeichnis

Verzeichnis aller Folien

Legende:
Fortsetzungsseite
Seite ohne Überschrift
Bildseite