Network Security

Assignments Project List

Manuel E. Correia

MSI – 2025/2026

1 - OWASP top 10 vulnerabilities (max 4 groups)

OWASP is a non-profit organization that captures and catalogues the most common system vulnerabilities. Its goal is to raise awareness about pervasive issues in IT, as well as to lead open-source software projects focused on ensuring web security.

The goal if this work is to analyse and describe protections against the top 10 list of OWASP vulnerabilities. Each group must also select 1/2 vulnerabilities on the list, describe how these can be exploited as attacks, give examples, and explore how one can devise effective countermeasures.

Refs: owasp-top10, owasp-top10-old

Note: no two groups can select the same vulnerabilities.

Assignment #2: The goal is to design a proof-of-concept target application and attack, implement countermeasures, and evaluate their efficacy.

2 - Anonymity and Onion Routing (max 2 groups)

Tor and the Invisible Internet Project (I2P) are two anonymous networks with the goal of ensuring user anonymity. These networks rely on unidirectional encryption tunnels to hinder eavesdroppers from understanding where traffic is originating, as well as its destination.

The goal of this work is to describe and analyse the mechanisms underlying these networks. Students must select a system for anonymous networks, describe how it can be installed and configured, and analyse how the underlying mechanisms ensure anonymity of users and servers.

Refs: Tor, I2P

Note: no two groups can select the same system

Assignment #2: The goal is to design a proof-of-concept service for this anonymous network, configure it, collect traffic data, and evaluate the anonymity mechanisms used.

3 - (non) Anonymity in Tor (max 2 groups)

As the most popular anonymous network, Tor has been the target of many attacks that attempt to breach its anonymity guarantees. This is a crucial security target, as its main goal is to ensure censor-free speech and anonymity even in the presence of malicious governments.

Students must select a peer-reviewed paper describing a threat to anonymity networks, which will be the focus for the assignment. The goal is to analyse the techniques used to breach anonymity of the tor network, and describe potential countermeasures that can be used to strengthen the system against such threats.

Refs: List of papers

Note: no two groups can select the same paper; papers can be no older than 2010

Assignment #2: The goal is to demonstrate how one can capture communication patterns allowing for the de-anonymization of Tor services, and to explore how one can devise effective countermeasures for speecific attack strategies.

4 - Certificate Revocation Issues (max 2 groups)

As systems will inevitably become vulnerable, revocation of permissions is an integral part of PKIs. x.509 certificates allow for revocation mechanisms, which are designed to prevent system breaches from provoking critical damage. However, adversaries have a wide array of attack opportunities to disrupt and delay the revocation process.

Students should explore the state-of-the-art for certificate revocation mechanisms and their vulnerabilities. The goal is to capture the standard mechanisms used in standard PKIs, as well as propose the current challenges for these systems.

Refs: Certificate revocation challenges, Short-lived Certificates

Note: If two groups select this topic, two methodologies for certificate revocation must be chosen (or a certificateless alternative).

Assignment #2 The goal is to develop a proof-of-concept PKI, demonstrate an attack on the explored revocation mechanisms, and implement the adequate countermeasures.

Refs: Running your own Certificate Authority with ACME support

5 - Ransomware (max 2 groups)

Currently, ransomware is a common term in IT security. It is a malware that blocks access to files of a machine (via encryption), demanding a ransom for data recovery. The maturity of modern cryptographic algorithms ensures that, after encryption, the chance of data recovery without access to the secret key used for encryption is negligible.

Students should explore the methodologies used for ransomware to be installed in machines. Specifically, what are the common network vulnerabilities that allow for this malware to damage modern systems.

Refs: Protecting against ransomware, Ransomware awareness and response

Note If two groups select this topic, two different network vulnerabilities leading to the ransomware attack must be selected.

Assignment #2 The goal is to exemplify how a common network vulnerability can be explored to install a ransomware attack, and implement countermeasures to block the attack.

6 - Intrusion Detection Systems (max 2 groups)

Intrusion detection systems (IDSs) typically monitor incoming and outgoing network packets to detect possible malicious actors attempts to gain control of a system. However, as malicious actors' knowledge and skill increases so must ID systems.

Students should explore the state-of-the-art for intrusion detection systems, focusing on the different approaches and identifying tradeoffs.

Note If two groups select this topic, different improvements must be selected (e.g., ML, eBPF, etc.).

Refs: IDS, SNORT

Assignment #2 The goal is to propose improvements on existing IDS, for example, by adding support to deal with 'zero day' exploits.

7 - Intrusion Prevention Systems (max 2 groups)

Intrusion prevention systems (IPSs) typically work along side intrusion detection systems (IDSs) and manage incoming and outgoing network packets to prevent possible malicious actors attempts to gain control of a system.

Students should explore the state-of-the-art for intrusion prevention systems, focusing on the different approaches and identifying tradeoffs.

Note If two groups select this topic, different improvements must be selected (e.g., ML, eBPF, etc.).

Refs: IPS, SNORT

Assignment #2 The goal is to propose improvements on existing IPS, for example, by adding support to deal with 'zero day' exploits.

8 - HoneyPots & HoneyNets for detecting intrusions (max 2 groups)

HoneyPots & HoneyNets are typically used to detect and discover intrusion techniques by attracting malicious actors.

In this project, students are expected to explore the state-of-the-art on this topic, focusing on how these approaches can be used to detect intrusion attempts, gather information and increase the knowledge base used by other systems (e.g., IDSs).

Note If two groups select this topic, different HoneyNet projects must be selected.

Assignment #2 The goal of this project is to design and develop a HoneyPot or HoneyNet project for gathering information for increasing knowledge base of other systems.

Refs: HoneyNet.org, HoneyNet projects, Awesome HoneyPot projects

9 - Increasing network security with eBPF (max 2 groups)

eBPF (enhanced Berkeley Packet Filtering) is a technology that allows developers to run programs in the operating system kernel without requiring changes kernel source code or load kernel modules.

In this project, students are expected to study the eBPF technology, and explore how it can be used to increase network security (for example, for monitoring and managing incoming and outgoing connections such as a traditional firewall).

Note If two groups select this topic, different eBPF projects must be selected.

Assignment #2 The goal of this project is to design and develop an eBPF application increasing network security of a system.

Refs: eBPF eBPF Explained

10 - The European Digital Identity Wallet Architecture and Reference Framework (ARF) (max 2 groups)

The European Digital Identity Wallet (EUDI Wallet) is a secure digital tool introduced under the eIDAS 2.0 regulation, designed to give EU citizens and businesses a reliable way to verify their identities and share personal information across the European Union. It functions as a digital wallet stored on a personal device, enabling users to authenticate themselves online for various services, from accessing eGovernment portals to conducting secure financial transactions. Its key feature is ensuring a unified digital identity system that works across all EU member states, promoting cross-border access to services in line with the EU's digital market goals.

A major advantage of the EUDI Wallet is that it provides users with full control over their personal data. In compliance with GDPR, individuals can choose what information to share and with whom, ensuring privacy and security. The wallet is built on a trust framework, involving certified identity providers that guarantee the authenticity of the digital identities, along with cryptographic safeguards to ensure the integrity and confidentiality of the data shared.

The European Digital Identity Wallet (EUDI Wallet) is built around a comprehensive **Architecture and Reference Framework (ARF)**, outlined under the **eIDAS 2.0** regulation. The ARF provides the technical blueprint for secure, interoperable, and privacy-respecting digital identity systems across the European Union.

This project focuses on conducting an in-depth analysis of the ARF, with the goal of understanding how its components contribute to the wallet's functionality.

Project Goals

1. **Architectural Analysis**:
   - Perform a detailed analysis of the EUDI Wallet's Architecture and Reference Framework (ARF), focusing on its core components (e.g., identity management, trust services, and data exchange protocols).
   - Evaluate how **Self-Sovereign Identity (SSI)** principles are integrated into the EUDI Wallet, allowing users to control their personal data and selectively share identity attributes while maintaining privacy and GDPR compliance.
   - Examine the roles and responsibilities of different entities within the ARF, such as identity providers, service providers, and trust authorities.
   - Assess how the ARF ensures compliance with privacy, security, and interoperability standards, specifically addressing cross-border scenarios.

Assignment #2: The Goal of this project is to investigate the specific components within the ARF, that support digital identity transactions.

Start by conduction a more rigourous:

- **Component Evaluation**:
  - **Identity Verification and Authentication**: Analyze how different levels of assurance are managed within the architecture.
  - **Data Minimization and Consent Management**: Explore the privacy safeguards in place to ensure GDPR compliance.
  - **Interoperability Mechanisms**: Evaluate how the ARF ensures smooth interaction between national identity systems and cross-sector services.

Followed by a minimally functional:

- **Pilot Design**:
  - Based on your analysis of the ARF, design a high-level pilot implementation for a specific use case. The pilot should demonstrate:
    - How identity attributes are securely managed and exchanged between stakeholders.
    - The flow of trust and data verification in a practical application.
  - Focus on conceptualizing how the ARF's principles are applied in real-world scenarios rather than building a complete working system.

References

1. **European Digital Identity Wallet - Architecture and Reference Framework (ARF)**:
   The technical and architectural foundation for the EUDI Wallet under the eIDAS 2.0 framework.
   https://github.com/eu-digital-identity-wallet

2. **EUDI Wallet Reference Implementation** The EUDI Wallet Reference Implementation is built based on the Architecture Reference Framework and aims at showcasing a robust and interoperable platform for digital identification, authentication and electronic signatures based on common standards across the European Union.
   https://github.com/eu-digital-identity-wallet/.github/blob/main/profile/reference-implementation.md

3. **eIDAS 2.0 Regulation**:
   The regulatory framework governing digital identity in the European Union, outlining legal requirements for the EUDI Wallet.
   https://eur-lex.europa.eu/eli/reg/2024/1183/oj

4. **SSI Technology in the Context of eIDAS 2.0**:
   A Master thesis exploring the relationship between Self-Sovereign Identity (SSI) concepts and the the new European digital identity landscape outlined by eIDAS 2.0. It focuses on how SSI offers a decentralized identity management solution, where users retain control over their personal data and credentials, in contrast to traditional siloed systems.
   https://repositorio-aberto.up.pt/bitstream/10216/156528/2/655820.pdf

11 - Laboratory: Man-in-the-Middle (MITM) with Ettercap and complementary Metasploit demonstration (max 3 groups)

Overall objective

Understand, *reproduce* in a controlled lab and critically *evaluate* MITM techniques applied to Windows and Linux services.

Students must build a laboratory-safe proof-of-concept that (i) demonstrates classic LAN MITM attacks using **Ettercap** (ARP poisoning, traffic capture and manipulation, use of packet filters), and (ii) presents a complementary demonstration with **Metasploit** that shows how credentials or sessions obtained via MITM can be leveraged for further exploitation/pivoting. All work must be performed only on virtual machines under the students' control.

Ettercap is an open-source suite for LAN Man-in-the-Middle tasks (ARP poisoning, live sniffing, on-the-fly packet filtering and injection); see the project homepage and source repository for details (ettercap-project.org)

For this assignment expect **Ettercap** to be the MITM positioning and capture tool — run ARP-poison scenarios, export sanitized pcap artefacts and (optionally) apply simple filters/injections to demonstrate traffic manipulation; use the captures as evidence for the Metasploit follow-up.

Kali Linux bundles Ettercap and related MITM tooling, reducing setup friction and improving reproducibility in isolated VM labs — which is why it is the recommended test platform here (always use host-only/internal networks and obtain authorization)

Metasploit is a comprehensive, modular penetration-testing framework that provides exploit modules, payloads (e.g., Meterpreter), scanners and auxiliary tools to validate vulnerabilities and emulate post-compromise activity in a repeatable, instrumented way.

For this assignment, expect Metasploit to be the controlled post-capture platform: correlate reconnaissance (e.g., **Nmap**) and captured artefacts, select appropriate auxiliary/exploit modules to demonstrate the consequence of compromised credentials or sessions in the lab (e.g., controlled Meterpreter session, credential validation, lateral-movement simulation), and always confine actions to authorized VMs while sanitizing evidence for the report.

Kali Linux ships Metasploit preinstalled and integrates its dependencies, reducing setup overhead.

Assignment #1

1. Context and preparation

Build an isolated lab (virtual machines in VirtualBox, VMware or similar). Use an internal/host-only network (or a bridged network limited to the lab VMs) so no experiment touches real/external networks. Minimum topology: one attacker VM (Kali Linux), one Windows target VM (recent version), and one Linux target VM (web server and/or SSH server). Document the network topology clearly.

Mandatory rule: no attacks or tests against infrastructure, hosts or networks outside the laboratory VMs. Provide written authorization if any testing outside strictly controlled lab VMs is proposed (this will normally be refused).

2. Ettercap

Specific objectives

- Demonstrate ARP poisoning / MITM on a LAN and capture credentials/cleartext traffic (HTTP, FTP, Telnet, etc.) using Ettercap. Explore Ettercap features such as packet filtering, character injection, and automatic password collection.
- Explain the concept and limitations of HTTPS downgrade attacks (e.g., SSL-strip integration) at a conceptual/experimental level; if SSL-strip is used it must be confined to the lab and documented with clear ethical/technical comments.

- **Deliverables**
  - A technical report (max 8 pages) including: lab setup and topology, step-by-step conceptual description of the experiment (capture and manipulation flow), sanitized logs/screenshots, analysis of captured packets (pcap excerpts with sensitive data redacted), and an evaluation of preconditions for success (same subnet, switch behaviour, absence of protections such as HSTS/HPKP, TLS versions, certificate validation).
  - Discuss why many MITM attacks only succeed against plaintext traffic and how modern web security mechanisms (HSTS, certificate pinning, TLS 1.3, proper certificate validation) mitigate these attacks.

Assignment #2

Metasploit

1. **Objectives**
    - Use **Metasploit** in a controlled scenario to illustrate *how captured credentials or intercepted session tokens* could be used for post-compromise activities or lateral movement (e.g., SMB lateral access on Windows, HTTP service exploitation where appropriate). The emphasis is on demonstrating the attack flow: *capture → analysis → validate exploitation vector → containment/mitigation*, not on causing real harm.
    - Demonstrate selected (safe) post-exploitation modules in test VMs to illustrate impact (credential extraction from the compromised VM, simulated privilege escalation attempts) while ensuring all actions remain confined to the lab.

- **Deliverables**
    - An extended technical report that includes attack maps, linking the evidence and security artifacts collected in **Assignment #1** to the controlled attack experiments conducted with Metaploit with a technical explanation of why they work.
- **Ethical/legal note**
    - Include an explicit section on legal and ethical constraints. All exploitation must be executed only on VMs that belong to the lab and only after the required ethics/compliance briefing.

Defences and evaluation

- Implement and evaluate at least **three** practical countermeasures across client, network and server layers. Examples: **ARP anomaly detection and alerting**, **static ARP entries in controlled scenarios**, **browser HSTS and certificate validation**, **deployment of an IDS/IPS to detect ARP poisoning**, **mutual TLS**, use of strong authentication and least privilege on services.
- Provide reproducible measurements showing the efficacy of each countermeasure (e.g., can the MITM still capture credentials? are alerts generated? false positive/negative observations). Present before/after results and an interpretation of the metrics.

Rules for project submission and evaluation

- Projects are to be completed in **groups of 3 students**.
  - Exceptions to this rule must be previously approved by the teacher.

- Topics must be chosen by all groups until **12/10/2025**.

- **Assignment #1** consists of a report describing and discussing state-of-the-art techniques on a specialized network security topic chosen by the group.
  - Report submitted by **02/11/2025**.
  - Firt Report will be defended in a short presentation (**15m**) on **06/11/2025** and **07/11/2025**, during the assigned **TP** class.

- **Assignment #2** is a more detailed exploration of the chosen topic:
  - Report submitted by **08/12/2024**
  - Project will be defended in a short presentation (**15m**) on **12/12/2024** and **13/12/2024**, during groups assigned **TP** class.

- Submissions are made via Moodle. Only a student in a group should submit the archive with the submissions files.

- The structure for project submission should be as follows. Let X be your assigned **TP** class (1 or 2) and **Y** your group number:
  - The submission itself must be a zip file named: TPX_groupY.zip
  - The zip file will contain all the relevant code, in a folder called "**code**", alongside a **README** file with clear instructions for compilation and the project proof of concept execution if applicable.
  - The reports should be in a **pdf** file named **TPX_groupY_report_1.pdf** for the first assignment and **TPX_groupY_report_2.pdf** for the second.
- Submissions that do not follow these specifications risk being discarded from evaluation.