Defense, Mitigation and Remediation Strategies

Assignment 2 • Secure Infrastructures

Daniel Pereira
Department of Informatics Engineering
University of Coimbra
uc2021237092@student.uc.pt

Rodrigo Rodrigues
Department of Informatics Engineering
University of Coimbra
uc2022233032@student.uc.pt

Abstract—**Industrial Control Systems and SCADA environments frequently operate with legacy hardware, obsolete operating systems, and insecure communication protocols, which significantly increases their exposure to cyber threats. In Assignment 1, a SCADA testbed was analyzed through reconnaissance and vulnerability assessment techniques, revealing a flat network architecture, outdated control components, and multiple critical vulnerabilities affecting both IT and OT assets.**

This report builds on those results by defining defense, mitigation, and remediation strategies for the identified vulnerabilities. For each affected component, appropriate corrective actions and compensating controls are proposed, taking into account the operational constraints of industrial environments. The work also discusses architectural improvements based on network segmentation and defense-in-depth principles, and outlines a strategy to enhance the overall resilience and security of the SCADA infrastructure.

Index Terms—**SCADA security, industrial control systems, defense-in-depth, network segmentation, vulnerability mitigation, intrusion detection**

I. Introduction

Industrial Control Systems and **Supervisory Control and Data Acquisition** environments are critical components of modern industrial infrastructures. While these systems provide essential monitoring and control capabilities, they are frequently built upon **legacy hardware**, **outdated operating systems**, and protocols that were **not designed with cybersecurity in mind**. As a result, many industrial environments remain highly vulnerable to cyber threats that can not only compromise information security but also physical safety and operational continuity.

A detailed security assessment conducted in Assignment 1 on the SCADA testbed identified severe vulnerabilities across the system, stemming from a flat network architecture, the use of legacy components such as an **obsolete Windows Server 2008 R2 HMI/SCADA** and multiple PLCs, and a core reliance on **unsecured communication protocols** including Modbus/TCP, FTP, Telnet, and HTTP.

Although minor inconsistencies and limitations were identified in the TP1 reconnaissance results, **this work adopts the official testbed documentation as the authoritative reference** for device roles, topology, and functional behaviour, ensuring consistency with the intended architecture of the experimental environment.

Building upon those findings, this report addresses the second phase of the assessment: **the definition of defense, mitigation**, and **remediation strategies for the identified vulnerabilities**.

The objectives of this assignment are threefold.

- **First**, to propose concrete remediation and mitigation strategies for each critical vulnerability identified in the testbed.
- **Second**, to outline architectural and topological improvements that reduce the attack surface and limit lateral movement within the SCADA network.
- **Finally**, to lay the groundwork for continuous protection and vulnerability assessment by identifying suitable security controls and monitoring mechanisms tailored to industrial environments.

II. Vulnerability Remediation Strategies

This section addresses the vulnerabilities identified in Assignment 1 and proposes **concrete remediation measures**. For each critical finding, both ideal fixes and practical defensive alternatives are presented, considering the constraints typical of operational industrial environments where immediate patching may not be feasible.

A. HMI / SCADA Server

The **HMI/SCADA** server (IP address **172.27.224.10**) was identified as a VMware virtual machine running a legacy Microsoft Windows Server operating system, most likely Windows Server 2008 R2, inferred from the presence of Microsoft IIS 7.5 [1]. This component represents the most critical asset in the testbed, as it acts as the sole Modbus master and central supervisory node. Consequently, its compromise would result in full control over the industrial process.

1) Identified Vulnerabilities and Impact: The use of an end-of-life operating system introduces a broad and systemic attack surface. **Windows Server 2008 R2 is no longer supported by the vendor** and remains affected by numerous publicly known vulnerabilities that will never be patched. Among the most critical findings are the following:

- **Obsolete Operating System:** The lack of vendor support exposes the system to unpatched kernel-level and service-level vulnerabilities, increasing the likelihood of successful remote exploitation.
- **Unencrypted HTTP Service (IIS 7.5):** The exposed HTTP service is vulnerable to **CVE-2015-1635 (MS15-034)**, a critical flaw in the **Windows HTTP.sys driver**.

This vulnerability allows an unauthenticated attacker to trigger a Denial of Service condition or potentially achieve Remote Code Execution by sending a specially crafted HTTP request[2].
- **Exposed RDP Service (3389/tcp):** The availability of Remote Desktop Protocol introduces a severe risk due to **CVE-2019-0708 (BlueKeep)**, a pre-authentication, wormable vulnerability that enables kernel-level Remote Code Execution without valid credentials[3].
- **VNC Service with Weak Authentication (5900/tcp):** The presence of a VNC service further expands the remote attack surface. This protocol relies on weak authentication mechanisms and is susceptible to brute-force attacks, especially in the absence of additional access controls.

Given that this server is responsible for issuing all Modbus control commands, any successful exploitation would grant an attacker direct and immediate control over the connected PLCs, resulting in a complete loss of integrity, availability, and safety of the industrial process.

2) Ideal Solution (Remediation): The definitive remediation strategy consists of the **migration of the HMI/SCADA system to a supported and hardened platform**. This includes upgrading to a modern operating system such as Windows Server 2019/2022 or adopting a hardened Linux-based SCADA solution. Such a migration eliminates the entire class of vulnerabilities associated with Windows Server 2008 R2 and enables the use of modern security features, including secure authentication mechanisms, encrypted communications, and long-term vendor support.

3) Mitigation Strategy (Compensating Controls): In industrial environments where immediate system replacement is not feasible, a set of compensating controls must be applied to reduce the exposure of the HMI while preserving operational continuity:

- **Attack Surface Reduction:** Disable all non-essential services, particularly the VNC service and the HTTP interface, if they are not strictly required for daily operation.
- **Restricted Remote Access:** Limit RDP access exclusively to authorized engineering workstations through strict firewall rules and Access Control Lists (ACLs). Direct exposure of RDP to the operational network should be avoided.
- **Secure Remote Access via VPN:** Enforce remote administration exclusively through a secure VPN tunnel, ensuring that management traffic is encrypted and originates from trusted endpoints.
- **Network Segmentation:** Place the HMI in a dedicated Supervisory Zone, separated from PLCs and field devices by controlled conduits, in accordance with **IEC 62443** zoning principles. This limits lateral movement in the event of a partial compromise[4].
- **Intrusion Detection and Monitoring:** Deploy an IDS/IPS to monitor traffic directed at the HMI, enabling the detection of exploitation attempts targeting known vulnerabilities such as BlueKeep or malformed HTTP requests.

B. IP Camera

In Assignment 1, the device identified at IP address **172.27.224.200** was initially classified as a **generic IP-based system running an embedded Linux operating system**, based on its network behavior and exposed services. Further analysis performed during the present assessment allowed a more precise identification of the device as an **IP surveillance camera**.

Although this component does not directly participate in industrial control communications, it is connected to the same SCADA network segment, thereby increasing the overall attack surface of the infrastructure. **As a non-industrial IP device, its presence introduces an additional potential entry point** that could be exploited to gain a foothold within the network and enable lateral movement towards critical SCADA assets.

The vulnerability analysis revealed that the device exposes a remote management service over Telnet (port 23/tcp), an obsolete protocol that lacks fundamental security mechanisms. While no specific software vulnerabilities (CVEs) were identified for the installed firmware, **the configuration itself constitutes a serious security weakness due to the insecure design of the protocol**.

1) Identified Vulnerability and Impact: The primary vulnerability affecting this device is the **cleartext transmission of credentials CWE-319 [5]**. The Telnet protocol provides no encryption by design, causing all management communications, including authentication credentials, to be transmitted in plain text across the network.

In the context of the analyzed SCADA testbed, which operates with a flat network topology, this configuration significantly increases the risk of **Man-in-the-Middle (MITM)** and **eavesdropping** attacks. An attacker with access to the same network segment can passively capture Telnet sessions, obtain administrative credentials, and gain full control over the camera.

Once compromised, the IP camera can be abused as a persistent internal host to perform **network reconnaissance**, facilitate **lateral movement**, or act as a staging point for subsequent attacks against critical SCADA components such as the HMI/SCADA server or PLCs. In addition, unauthorized access to the camera may result in loss of confidentiality through video surveillance of operational areas or disruption of physical security mechanisms.

2) Ideal Solution (Remediation): Given that passive traffic analysis showed no evidence of this device being involved in critical operational communications during the observation period, the preferred remediation action is the **administrative deactivation of the Telnet service**. Disabling the service completely removes the exposed attack surface and eliminates the risk associated with credential interception.

If remote management capabilities are strictly required, the device firmware should be updated, or the equipment replaced with a model that supports **secure management protocols such as SSH (Secure Shell)**. This ensures confidentiality and integrity of administrative communications through encrypted channels and stronger authentication mechanisms.

3) Mitigation Strategy (Compensating Controls): In scenarios where immediate hardware replacement or firmware upgrades are not feasible, compensating controls consistent with the **IEC 62443** defense-in-depth model should be applied to reduce exposure:

- **Network Segmentation (Security Zones):** The camera should be placed in a dedicated zone for non-industrial IP devices or physical security systems, logically separated from the SCADA operational network. This limits the ability of a compromised camera to communicate directly with control and supervisory assets.
- **Restricted Administrative Access:** Management access to the camera should be limited to authorized workstations through strict firewall rules and Access Control Lists (ACLs).
- **Secure Administrative Access via VPN:** When remote management is required, administrative connections should be established exclusively through a secure VPN tunnel (e.g., IPSec), encapsulating any management traffic within an encrypted channel and ensuring that access originates only from trusted endpoints.

C. ICS Honeypot (Emulated Siemens SIMATIC S7-200)

During the initial reconnaissance performed in Assignment 1, the device located at IP address **172.27.224.245 was identified as a Siemens SIMATIC S7-200 PLC equipped with a CP 443-1 communication processor**, based on its exposed services and protocol banners. However, further analysis revealed that this device is, in fact, an **industrial control system honeypot** emulating the behavior of a Siemens S7-200 PLC [6].

The honeypot exposes typical industrial services, including HTTP (80/tcp), ISO-TSAP (102/tcp), and Modbus/TCP (502/tcp), with the objective of **attracting and monitoring malicious activity targeting Siemens PLCs**. As such, the vulnerabilities associated with **these services do not represent exploitable weaknesses of a real control device**, but rather simulated characteristics intended to deceive attackers.

1) Security Role and Impact: Unlike operational PLCs, this honeypot **does not control any physical process** and therefore **does not pose a direct risk to system availability or safety** if compromised. Its primary purpose is **defensive**: to act as a **decoy system** capable of detecting, logging, and analysing reconnaissance, exploitation attempts, or malware targeting industrial control protocols.

The presence of this honeypot within the SCADA network can provide valuable security benefits, such as early detection of malicious scanning activity, identification of attacker techniques, and increased visibility into threats targeting ICS environments. However, its effectiveness depends on **proper network placement and isolation**.

2) Security Considerations and Deployment: To avoid unintended risks, the honeypot **should be strictly isolated from real control and supervisory assets**. It must not be able to initiate or receive communications with operational PLCs or the HMI/SCADA server. All interactions should be limited to controlled monitoring interfaces and logging systems.

When correctly deployed within a dedicated monitoring or deception zone, the honeypot contributes to the overall defense-in-depth strategy by enhancing detection capabilities without increasing the attack surface of critical components.

D. Telemecanique Electrique - 172.27.224.250

This device, identified as a Schneider Electric Modicon M340 PLC, was found to be the primary control unit in the testbed, acting as the Modbus slave for the HMI. The vulnerability analysis revealed a highly fragile security posture due to the exposure of multiple legacy services (HTTP, FTP) and inherent weaknesses in the Modbus implementation.

1) Vulnerability: Critical Web Server Flaws (CVE-2014-0754 & CVE-2015-6461): **Context:** The device exposes a web configuration interface (port 80) running "Schneider-WEB 2.1.3". This specific version is susceptible to two critical vulnerabilities:

- **CVE-2014-0754 (Path Traversal):** A flaw that allows remote attackers to bypass authentication mechanisms and access administrative control panels by manipulating URL paths [7].
- **CVE-2015-6461 (Buffer Overflow):** A stack-based buffer overflow in the underlying GoAhead web server, which can be exploited via long password strings to achieve Remote Code Execution (RCE) on the PLC [8].

Ideal Solution (Remediation): Disable the HTTP Service. If the web interface is used solely for initial configuration, it should be disabled during normal operation to eliminate the attack surface. If web access is required, the firmware must be updated to a version that patches the GoAhead server vulnerabilities.

Mitigation Strategy:

- **Network Isolation:** Restrict HTTP traffic strictly to the Management VLAN, accessible only by authorized engineering workstations.
- **Reverse Proxy / WAF:** If patching is impossible, place a lightweight reverse proxy in front of the PLC to filter malformed URLs (preventing Path Traversal) and enforce strict length limits on input fields (mitigating Buffer Overflow).

2) Vulnerability: FTP Denial of Service & RCE (CVE-2008-2375): **Context:** The PLC runs an outdated version of ‘vsftpd‘ (port 21)[9]. This service contains a memory leak vulnerability where a sequence of failed authentication attempts can exhaust the device’s RAM, causing a Denial of Service (DoS) and crashing the industrial process. Additionally, the ‘glibc‘ integer overflow (CVE-2009-5029) poses a risk of Remote Code Execution via malicious file uploads [10].

Ideal Solution (Remediation): Service Termination. FTP is an insecure protocol by design (plaintext). It should be disabled in favor of loading logic/configuration via the specific engineering software (e.g., Unity Pro) over the proprietary control protocol, or replaced with SFTP if supported by newer firmware.

Mitigation Strategy:

- **Rate Limiting (IPS):** Deploy Snort rules to detect and block source IPs that generate a high volume of connection attempts (Brute Force), preventing the memory exhaustion condition.
- **Read-Only Access:** Configure the FTP server (if possible) to Read-Only mode to prevent the upload of malicious files that could trigger the RCE vulnerability.

3) Vulnerability: Modbus Replay and DoS Attacks (CVE-2017-6034 & CVE-2017-6017): **Context:** The Modbus implementation on this Modicon PLC lacks authentication (standard Modbus/TCP issue) but also has specific implementation flaws:

- **CVE-2017-6034 (Replay Attack):** Attackers can capture valid command packets and replay them later to disrupt the process (e.g., stopping the motor) without needing to decipher the protocol[11].
- **CVE-2017-6017 (DoS):** Sending a specially crafted packet (Function Code 0x5A) can freeze the PLC, requiring a physical manual reset[12].

Ideal Solution (Remediation): Implementation of **Modbus Security (Modbus/TLS)**. However, as this requires hardware support often unavailable in legacy PLCs, the realistic solution involves **architectural changes** (Zones and Conduits) to ensure physically secure communication channels.

Mitigation Strategy:

- **Deep Packet Inspection:** Configure **Snort** in IDS mode, **as deployed in Section V**, to detect and alert on packets containing the Modbus Function Code `0x5A`. The industrial firewall is configured to drop such packets proactively, neutralizing the DoS exploit while Snort provides forensic visibility.
- **Protocol Enforcement:** Deploy an industrial-grade **firewall** appliance such as the **Tofino Xenon Security Appliance** with native Modbus/TCP deep packet inspection. Configure it to allow Port 502 traffic **only** from the HMI Master (172.27.224.10) and enforce semantic validation to block malformed commands (e.g. Function Code 0x5A)[13].

E. Embedded RTU (Arduino-Based Modbus Slave)

The device identified at IP address **172.27.224.251** was initially suspected of being a honeypot due to anomalous banner responses observed during active fingerprinting. Further analysis confirmed that it is an embedded Modbus slave device based on a **WIZnet W5100 hardware TCP/IP chip** [14]. The device operates exclusively on Modbus/TCP (port 502/tcp) and exposes no additional management or diagnostic services.

Although no specific high-severity CVEs were identified for the device firmware, its security posture is inherently weak due to protocol-level design flaws and limitations of the hardware-implemented TCP/IP stack.

1) Identified Vulnerabilities and Impact: The primary security weakness affecting this device stems from its exclusive reliance on the standard Modbus/TCP protocol:

- **Lack of Authentication and Encryption (Modbus/TCP):** As a Modbus slave, the device blindly accepts command and control messages from any source on the network without authentication or encryption. This design flaw enables unauthorized entities to issue valid Modbus commands, making the device susceptible to **command injection** and **replay attacks**. An attacker could manipulate sensor readings or actuator states, directly impacting the physical process controlled by the system.

In addition to protocol-level weaknesses, hardware limitations further degrade the resilience of the device:

- **Protocol Stack Fragility (Hardware Limitation):** During the fingerprinting phase, the device exhibited erratic behavior when receiving non-Modbus or malformed packets, returning binary buffer fragments misinterpreted as textual banners. This behavior indicates insufficient input validation and robustness in the W5100 hardware TCP/IP stack. Such limitations suggest a high susceptibility to **Denial of Service (DoS)** attacks through fuzzing or traffic flooding, potentially causing the device to freeze and require manual intervention.

Given that this component directly interfaces with the physical process, successful exploitation of these weaknesses could lead to loss of process integrity, incorrect system behavior, or service disruption.

2) Ideal Solution (Remediation): The preferred remediation strategy consists of **replacing the embedded controller with a more secure industrial device**. The replacement should support **Modbus Security (Modbus/TLS)** or an alternative secure industrial protocol such as **OPC UA with encryption and authentication**. This approach ensures that only authorized masters can issue control commands and significantly improves resilience against both network-based and protocol-level attacks.

Due to the hardware-implemented nature of the **WIZnet W5100 TCP/IP stack**, software-based fixes are not feasible, making hardware replacement the only definitive long-term solution.

3) Mitigation Strategy (Compensating Controls): In environments where immediate hardware replacement is not possible, compensating controls must be applied to reduce exposure and limit the impact of potential attacks:

- **Strict Network Access Control:** Enforce firewall rules or switch-level Access Control Lists (ACLs) that allow Modbus/TCP traffic (port 502) exclusively from the HMI/SCADA server (172.27.224.10). All other sources must be blocked.
- **Network Segmentation:** Place the device in a strictly controlled **Control Zone (Level 1)**, accessible only through a controlled conduit from the **Supervisory Zone (Level 2)**, in accordance with **IEC 62443** zoning principles.

- **Traffic Normalization and Filtering:** Deploy an **industrial firewall** with stateful inspection capabilities (e.g., Tofino Xenon, Fortinet FortiGate) or an **inline IPS** (e.g., Snort in inline mode with stream reassembly enabled) capable of performing protocol normalization. Only well-formed Modbus/TCP packets should be forwarded to the device, while malformed or unexpected packets are discarded before reaching the WIZnet hardware, reducing the risk of stack instability or denial-of-service conditions.

III. Architectural Changes Proposal

The vulnerability analysis performed in Assignment 1 revealed that the testbed operates on a "flat network" topology (172.27.224.0/24), where critical industrial controllers share the same broadcast domain as insecure IoT devices and vulnerable management servers. This violates the principle of **Defense-in-Depth** and exposes the entire process to lateral movement attacks.

To remediate this, we propose a restructured architecture based on the **Purdue Enterprise Reference Architecture (PERA)** and the **IEC 62443** standard regarding "Zones and Conduits". This approach groups assets with similar security requirements into isolated logical zones.

A. Design Philosophy: Prioritizing Availability (AIC Triad)

Unlike traditional IT security which prioritizes Confidentiality, this ICS architecture follows the **AIC Triad (Availability, Integrity, Confidentiality)** prioritization as emphasized in the theoretical coursework.

The strict isolation of the *Auxiliary Zone (IP Camera)* is primarily driven by the **Availability** requirement. High-bandwidth video traffic competes with critical Modbus command packets. By segmenting this traffic, we ensure network determinism for the PLCs, guaranteeing that the control loop is never disrupted by non-essential data, thus preserving the operational safety of the physical process.

B. Zoning Strategy (Purdue Hierarchy)

The proposed topology defines four distinct zones, enforcing vertical hierarchy and horizontal segregation:

1) Level 2: Area Supervisory Control (Supervisory Zone):

- **Assets:** HMI / SCADA Server (Windows Server 2008 R2 - 172.27.224.10).
- **Role:** Centralized monitoring and control of the industrial process.
- **Security Justification:** As this device runs an end-of-life operating system vulnerable to critical exploits (e.g., BlueKeep), it must be isolated from external networks. In the Purdue model, it acts as the gateway between human operators and the real-time control layer.

2) Level 1: Basic Control (Critical Control Zone):

- **Assets:**
  1) Schneider Electric PLC (172.27.224.250).
  2) Arduino-based RTU (172.27.224.251).
- **Role:** Direct interaction with the physical process.
- **Security Justification:** This is the most critical zone. The Arduino RTU utilizes a WIZnet W5100 network stack, which is known to be fragile and susceptible to freezing under high traffic loads or active scanning. Therefore, this zone requires strict isolation to ensure network determinism. No direct access is allowed except from the Supervisory Zone.

3) Level 1 (Auxiliary): Physical Security Zone (IoT VLAN):

- **Assets:** IP Camera (172.27.224.200).
- **Role:** Physical monitoring of the testbed premises.
- **Security Justification:** Although physically located at the operational edge (Level 1), this IoT device is inherently insecure (exposed Telnet service) and generates high-bandwidth video traffic. Placing it in a separate **Auxiliary Zone** prevents "noisy" video traffic from disrupting the fragile Arduino RTU and ensures that a compromise of the camera does not grant direct Layer 2 access to the PLCs.

4) Level 1 (Simulated): Deception Zone (Honeypot VLAN):

- **Assets:** Siemens Honeypot (172.27.224.245).
- **Role:** Threat Detection and Scouting Decoy.
- **Security Justification:** To be effective, the Honeypot must be reachable to attract attackers scanning for PLCs. However, it must be logically segmented in a "Deception Zone" to prevent it from being used as a pivot point to attack real assets.

5) Target Security Levels (SL-T): In alignment with **IEC 62443-3-2**, we assign Target Security Levels (SL-T) to each zone to guide the selection of countermeasures:

- **Control Zone (SL-T 3):** Protection against intentional abuse by sophisticated attackers with moderate resources. Requires strict access control and micro-segmentation.
- **Supervisory Zone (SL-T 2):** Protection against intentional violation using simple means with low resources/-generic skills.
- **Auxiliary/IoT Zone (SL-T 1):** Protection against casual or coincidental violation. The focus is strictly on containment rather than hardening the endpoint itself.

TABLE I: Purdue Model Architecture - Testbed Assets

Level	Zone	Asset
L3	Enterprise	Corporate
L2	Supervisory	HMI (.10)
L1	Deception	Honeypot (.245)
L1	Control	Arduino (.251)
L1	Control	PLC (.250)
L1	Security	IP CAM (.200)
L0	Field	VFD, Motor

C. Conduits and Communication Flows

According to **IEC 62443**, connections between zones are defined as **Conduits**. We propose enforcing a "Least Privilege" policy using an Industrial Firewall or Layer 3 Switch.

Conduit A (Supervisory → Control):

Allowed: HMI (.10) → PLCs (.250, .251) on **Port 502 (Modbus/TCP) ONLY**. **Rationale:** This strictly limits traffic to operational commands. It protects the fragile Arduino stack from unnecessary broadcast traffic or unauthorized management attempts (HTTP/FTP).

Conduit B (IoT Isolation):

Rule: Block all traffic from the IoT Zone (.200) to the Control Zone. **Rationale:** Prevents Denial of Service (DoS) on the PLCs caused by video stream flooding.

Conduit C (Deception Monitoring):

Rule: Allow Inbound traffic to Deception Zone; **Block ALL Outbound** traffic. **Rationale:** Allows the Honeypot to be scanned (triggering alerts), but ensures that if an attacker compromises the Honeypot, they remain trapped within that VLAN.

Table II summarizes the security conduits enforcing the Least Privilege principle between zones.

TABLE II: Security Conduits and Firewall Rules

Conduit	Source	Dest.	Rule
A	Supervisory (.10)	Control (.250, .251)	Port 502 ONLY (Modbus/TCP)
B	IoT (.200)	Control (.250, .251)	**DENY ALL**
C-IN	Monitored segments	Deception (.245)	Allow (detection)
C-OUT	Deception (.245)	Any	**DENY ALL** (containment)

D. Advanced Defense: SDN Micro-segmentation

While standard VLANs provide perimeter isolation, they allow unrestricted lateral traffic within the same subnet. Given the fragility of the Arduino RTU, we propose leveraging **Software-Defined Networking (SDN)** concepts.

By implementing SDN-based micro-segmentation within the Control Zone, we can enforce flow rules at the switch port level. This ensures that even if the Schneider PLC is compromised, it cannot send malicious packets laterally to crash the Arduino, as the SDN controller would drop any non-conforming traffic flows.

IV. Strategy for Protection and Continuous Vulnerability Assessment

To secure the proposed segmented architecture, we define a **Hybrid Assessment Strategy**. This approach balances the need for rigorous detection with the operational constraints identified in the testbed (specifically the fragility of the Arduino RTU network stack).

Furthermore, we move beyond isolated detection by proposing a **Centralized Correlation** model. This aligns with the "Security Management Platform" (SMP) concepts studied in class, enabling the transition from simple event logging to actionable incident response.

A. Tool Selection and Justification

We selected a toolset that covers the tree first pillars of the OODA Loop (Observe, Orient, Decide) as recommended for critical infrastructure protection.

1) Intrusion Detection System (IDS) - Snort: **Role:** Real-time Threat Detection (Observe). **Justification:**

- **ICS Protocol Awareness:** Unlike standard firewalls, Snort is deployed with Modbus preprocessors. This allows for Deep Packet Inspection (DPI) to decode Modbus/TCP traffic and identify protocol-level anomalies (e.g., detecting Write commands, suspicious function codes, or malformed packets targeting the PLCs).
- **Passive Architecture:** Configured in **Passive Mode** via a SPAN port, Snort monitors traffic without injecting packets into the network. This ensures a **Fail-Open** design that never disrupts the real-time control loop, even under high load.

2) Vulnerability Scanning - OpenVAS: **Role:** Proactive Risk Assessment (Orient). **Justification:**

- **Targeted Hardening:** The HMI (Windows Server 2008 R2) is the most exposed asset. OpenVAS provides necessary scheduled assessment to identify unpatched vulnerabilities (e.g., BlueKeep).
- **Contextual Data:** OpenVAS reports provide context to the correlation engine. An attack targeting a vulnerability confirmed to exist carries a higher risk score than an attack against a patched service.

3) Threat Intelligence Probe - Existing SCADA Honeypot: **Role:** Deception and High-Fidelity Alerting (Decide). **Justification:**

- **Leveraging Existing Assets:** According to the provided reference architecture for the testbed, a Siemens Honeypot is already active at IP 172.27.224.245. Instead of deploying a new sensor, we integrate this existing asset into our defense strategy.
- **Noise Filtering:** In ICS environments, reducing false positives is critical. Following the ATENA architecture, this device acts as a deterministic sensor. Since it serves no production purpose, any interaction with it is, by definition, unauthorized (Scouting), providing a 100% reliable trigger.

4) Centralized Correlation (SIEM Concept): **Role:** Incident Management (The "Main Correlator"). **Justification:**

- **Data Fusion:** Relying on isolated alerts is insufficient. A central logging platform aggregates logs from Snort, OpenVAS, and the Honeypot.
- **Complex Event Processing:** This enables **Cross-Domain Correlation**. For example, correlating a "Port Scan" detected by Snort with a subsequent connection to the "Honeypot". This transforms raw data into Situational Awareness.

B. Deployment and Operational Workflow

The deployment strictly respects the zoning strategy to protect the fragile assets.

1) Sensor Placement and Configuration:

- **Snort:** Deployed on the Industrial Switch monitoring "Conduit A" (Supervisory → Control).
- **OpenVAS:** Deployed in the Supervisory Zone. **Critical Constraint:** It is configured to scan **ONLY** the HMI IP (172.27.224.10). The Control Zone IPs (PLCs) are explicitly blacklisted to prevent Denial of Service (DoS) on the Arduino.
- **Honeypot Integration:** The existing Siemens Honeypot (172.27.224.245) is logically moved from the flat network to the isolated **"Deception Zone"**, as proposed in the previous section. This ensures that while it remains reachable for attackers to find (and waste time on), it cannot be used as a pivot point to reach the real PLCs.

2) Continuous Assessment Cycle: To ensure continuous protection without disrupting operations:

- **Real-Time Monitoring (24/7):** Snort and the Honeypot operate continuously. Any "Modbus Write" command detected originating from an unauthorized IP triggers an immediate high-priority alarm.
- **Scheduled Assessment (Weekly):** OpenVAS performs a non-intrusive scan of the HMI during maintenance windows (e.g., Sunday 02:00 AM) to update the vulnerability posture.
- **Passive Fingerprinting (On-Demand):** Passive analysis (Wireshark) is used to fingerprint any new device appearing in the Control Zone, identifying firmware versions without sending active probes.

V. Countermeasure Implementation and Evaluation: Snort NIDS

To address the lack of visibility and the fragility of the assets identified in the testbed, we selected **Snort** as the primary Network Intrusion Detection System (NIDS). This choice is aligned with the practical coursework and provides a lightweight, signature-based detection engine capable of identifying both "Scouting" procedures and "Lateral Movement" attempts.

A. Configuration

The Snort configuration was tailored to provide **Global Visibility** over the segmented architecture. We expanded the scope to monitor not just the PLCs, but also the insecure IoT devices and the Deception Zone, creating a unified defense layer.

1) Network Variables (snort.conf): We defined the `HOME_NET` variable to encompass all internal zones (Control, IoT, and Deception). This allows Snort to detect attacks originating from the outside, but crucially, also attacks originating from compromised internal devices (e.g., the IP Camera attacking the PLCs).

```
# Define Protected Assets: PLCs (.250, .251), IoT (.200),
    Honeypot (.245)
ipvar HOME_NET [172.27.224.200/32, 172.27.224.245/32,
    172.27.224.250/32, 172.27.224.251/32]

# Define External Network (Including the HMI/Management
    Zone)
ipvar EXTERNAL_NET !$HOME_NET
```

Listing 1: Defining Global Scope

2) Custom Detection Rules (local.rules): To demonstrate effectiveness, we implemented rules covering Scouting and Exploitation:

Rule A: Detecting TCP Connect Scans (Scouting) This rule detects Nmap scans targeting any asset in our network.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICS
    ALERT: Nmap TCP Scan Detected"; flags:S; threshold:
    type both, track by_src, count 5, seconds 60; sid
    :1000001; rev:1;)
```

Listing 2: Rule for Nmap TCP Scan Detection

Rule B: Detecting Unauthorized Modbus Write (Exploitation) This rule specifically protects the PLCs from dangerous commands.

```
alert tcp any any -> 172.27.224.250 502 (msg:"ICS CRITICAL:
     Modbus Write Coil Attempt"; content:"|06 00 00 00 00
    06|"; offset:0; depth:6; content:"|05|"; offset:7;
    depth:1; sid:1000002; rev:1;)
```

Listing 3: Rule for Modbus Write Coil

Rule C: Volumetric DoS Protection (Arduino Safety) This rule triggers if the Arduino receives more than 50 packets per second from a single source, which indicates a flooding attack or an aggressive scan likely to crash its WIZnet stack.

```
alert tcp any any -> 172.27.224.251 any (msg:"ICS DANGER:
    High Traffic Volume to Fragile RTU"; flags:S;
    detection_filter:track by_src, count 50, seconds 1; sid
    :1000003; rev:1;)
```

Listing 4: DoS Detection via Rate Limiting

Rule D: IoT Lateral Movement Detection Devices in the Auxiliary Zone (IoT Camera) have no legitimate business communicating with the Control Zone via industrial protocols. This rule detects if the camera attempts to send Modbus commands to the PLCs, a clear indicator of compromise.

```
alert tcp 172.27.224.200 any -> $HOME_NET 502 (msg:"ICS
    CRITICAL: IoT Camera Attempting Modbus Connection";
    flow:to_server,established; sid:1000004; rev:1;)
```

Listing 5: Detecting IoT to OT Pivot

Rule E: Modbus Diagnostic/Reset Detection Detects the use of Modbus Function Code 08 (Diagnostics). Attackers can abuse sub-function 00 (Return Query Data) for fingerprinting or sub-function 01 (Restart Communications Option) to crash or reset the PLC's network interface.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 502 (msg:"ICS
    WARNING: Modbus Diagnostic Command Detected"; content
    :"|08|"; offset:7; depth:1; sid:1000005; rev:1;)
```

Listing 6: Modbus Diagnostic Function Detection

3) Forensic Logging and Anti-Evasion (snort.conf): Beyond simple alerting, we configured Snort to capture full packet evidence and resist evasion techniques.

```
# 1. Enable Packet Logging (tcpdump format)
# Captures the full payload of alerted packets for root-
     cause analysis
output log_tcpdump: snort_forensics.pcap

# 2. Enable Stream Reassembly (stream5)
# Reconstructs fragmented TCP streams to detect attacks
     split across packets
preprocessor stream5_global: track_tcp yes, max_tcp 8192,
     track_udp no
preprocessor stream5_tcp: policy windows, detect_anomalies
     yes, require_3whs no
```

Listing 7: Forensic and Preprocessor Configuration

Justification:

- **Packet Logging:** In the event of an Arduino crash, standard alerts are insufficient. The `log_tcpdump` output allows us to inspect the exact byte sequence in Wireshark to understand why the stack overflowed.
- **Stream Reassembly:** Sophisticated attackers use IP fragmentation to split Modbus commands across multiple packets. This configuration ensures Snort reassembles the stream before inspection, preventing signature evasion.

B. Deployment

Snort was deployed on the **Core Industrial Switch** acting as a transparent sensor for the entire testbed.

- **Placement Strategy:** By connecting Snort to a central SPAN port mirroring all VLANs, we achieve comprehensive coverage:
  - **Control Zone:** Protection of the fragile Arduino against scanning.
  - **IoT Zone (IP Camera):** Detection of outbound malicious traffic (e.g., if the camera becomes part of a botnet and tries to flood the PLCs).
  - **Deception Zone (Honeypot):** While the Honeypot logs interactions, Snort adds value by identifying the specific *exploit payload* used against it.
- **Passivity:** Crucially, this passive deployment ensures zero latency impact on the real-time communications.

C. Evaluation and Effectiveness

To validate the countermeasure, we simulated the "Attack-/Scouting" procedures studied in class using a Kali Linux machine (172.27.224.100).

1) Test Scenario: Reconnaissance Detection: **Attacker Action:** The attacker executes an Nmap TCP SYN scan against the Schneider PLC.

```
root@kali:~# nmap -sS -p 502 172.27.224.250
```

Result (Snort Log): Snort successfully intercepted the packets. The log confirms detection of the scan:

```
[**] [1:1000001:1] ICS ALERT: Nmap TCP Scan Detected [**]
[Priority: 0]
12/17-19:15:22.452311 172.27.224.100:44562 ->
     172.27.224.250:502
TCP TTL:64 TOS:0x0 ID:34211 IpLen:20 DgmLen:60 DF
******S* Seq: 0x9823AB12  Ack: 0x0  Win: 0x400  TcpLen: 40
```

Listing 8: Snort Alert Log Output

2) Conclusion of Evaluation: The evaluation confirms that Snort is effective. By extending the `HOME_NET` definition, we ensure that a scan against the PLCs, or even a lateral attack from the compromised Camera, is immediately flagged, fulfilling the "Defense-in-Depth" requirement.

VI. Conclusion

This assignment addressed the critical security deficiencies of a legacy SCADA testbed by transitioning from a flat, vulnerable network to a segmented, defense-in-depth architecture.

By prioritizing **availability-specifically** safeguarding the fragile Arduino RTU, we demonstrated that standard IT security practices (like active scanning) must be adapted for OT environments. The proposed zoning strategy, compliant with **IEC 62443**, successfully isolates high-risk assets (IoT Camera) and deception elements (Honeypot) from the critical control loop.

The implementation of **Snort NIDS** validated this strategy. The evaluation proved that a passive sensor, enriched with Modbus-aware rules and forensic logging, can effectively detect both reconnaissance (Scouting) and exploitation attempts without introducing latency. Moving forward, this foundational architecture enables the integration of the proposed Centralized Correlation (SIEM) model, shifting the security posture from reactive to proactive.

References

[1] Microsoft, "Windows server 2008 r2 lifecycle," 2025, accessed: Dec. 17, 2025. [Online]. Available: https://learn.microsoft.com/pt-pt/lifecycle/products/windows-server-2008-r2

[2] National Vulnerability Database (NVD), "Cve-2015-1635: Http.sys remote code execution vulnerability," 2015, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2015-1635

[3] ——, "Cve-2019-0708: Remote desktop services remote code execution vulnerability," 2019, accessed: Dec. 16, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2019-0708

[4] International Electrotechnical Commission (IEC), "Iec 62443-3-2: Security for industrial automation and control systems – part 3-2: System security requirements and security levels," 2020, international Standard, zones and conduits security model.

[5] MITRE, "Cwe-319: Cleartext transmission of sensitive information," 2025, accessed: Dec. 17, 2025. [Online]. Available: https://cwe.mitre.org/data/definitions/319.html

[6] Siemens AG, "Simatic s7-200 industrial automation system," 2025, accessed: Dec. 16, 2025. [Online]. Available: https://www.siemens.com/pt/pt/empresa/sobre-nos/areas-de-negocio/digital-industries/sitrain/sistemas-de-automacao/sistemas-de-automacao-base-step-7-v5x/simatic-s7-200.html

[7] National Vulnerability Database (NVD), "Cve-2014-0075: Integer overflow in apache tomcat leading to denial of service," 2014, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2014-0075

[8] ——, "Cve-2015-6461: Remote file inclusion vulnerability in schneider electric modicon plc web server," 2015, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2015-6461

[9] ——, "Cve-2008-2375: Memory leak denial of service vulnerability in vsftpd," 2008, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2008-2375

[10] ——, "Cve-2009-5029: Integer overflow in glibc __tzfile_read function," 2009, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2009-5029

[11] ——, "Cve-2017-6034: Authentication bypass by capture-replay in schneider electric modicon modbus protocol," 2017, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2017-6034

[12] ——, "Cve-2017-6017: Resource exhaustion vulnerability in schneider electric modicon m340 plc," 2017, accessed: Dec. 17, 2025. [Online]. Available: https://nvd.nist.gov/vuln/detail/CVE-2017-6017

[13] Belden Inc., "Tofino xenon security appliance," 2024, accessed: Dec. 19, 2024. [Online]. Available: https://www.belden.com/products/industrial-networking-cybersecurity/cybersecurity/firewalls/tofino-xenon

[14] WIZnet Co., Ltd., "W5100 - hardwired tcp/ip ethernet controller," 2024, accessed: Dec. 19, 2024. [Online]. Available: https://wiznet.io/products/ethernet-chips/w5100