SECURE ICS TESTBED: ZERO TRUST & ENCRYPTED TRAFFIC

Ahmed Al-Senaidi, Al-Salt Al-Naabi, Ahmed Al-Khanbashi, Mohamed Al-Harrasi
Supervisor: Dr. Shdha Al-Amri
Department of Computer Science, College of Science, Sultan Qaboos University

Abstract

Industrial Control Systems (ICS) form the foundation of critical sectors like energy, water, and manufacturing but face increasing cybersecurity risks due to legacy protocols and IT/OT convergence.

This project developed a fully virtualized ICS testbed that simulates industrial operations and integrates security measures including network segmentation, TLS encryption for Modbus TCP, centralized monitoring with Wazuh SIEM, and threat detection using Suricata IDS.

Attack simulations using Metasploit tested system resilience against unauthorized access and command injection. Results confirmed the effectiveness of segmentation, encryption, and real-time alerting in defending against ICS cyber threats.

Introduction

Industrial Control Systems (ICS) are essential for critical infrastructure sectors such as energy, water, and manufacturing. Historically isolated, ICS environments are now exposed to cybersecurity threats due to increasing IT/OT integration.

Vulnerabilities such as unencrypted protocols, flat network architectures, and weak access control have made ICS environments attractive targets. High-profile attacks like Stuxnet and Triton illustrate the risks.

This project addresses these challenges by designing a secure, virtual ICS testbed based on Zero Trust principles, network segmentation, and real-time encrypted communications.

Objectives

- Build a realistic ICS environment using GRFICSv2 to simulate SCADA, PLCs, and HMI systems.
- Apply Zero Trust Architecture principles to enforce strict access control.
- Segment the network into IT, DMZ, and OT zones using pfSense.
- Secure Modbus TCP traffic using TLS encryption via stunnel.
- Deploy Suricata IDS for network traffic analysis and Wazuh SIEM for centralized log management.
- Simulate real-world attack scenarios using Metasploit to evaluate system resilience.

Methods and Materials

The ICS testbed was built using VirtualBox to host multiple virtual machines replicating industrial components. The architecture followed the Purdue Model, dividing the environment into IT, DMZ, and OT zones.

- **GRFICSv2**: Provided virtual PLCs, SCADA servers, and plant simulations.
- **pfSense**: Acted as a firewall and router enforcing strict network segmentation policies.
- **Jumpserver**: Provided controlled, authenticated access to OT systems.
- **stunnel**: Enabled TLS encryption to protect Modbus TCP communications.
- **Suricata IDS**: Monitored network traffic for unauthorized activities.
- **Wazuh SIEM**: Aggregated and correlated security event logs for centralized monitoring.

Cyberattack simulations were conducted using Metasploit to validate security mechanisms.

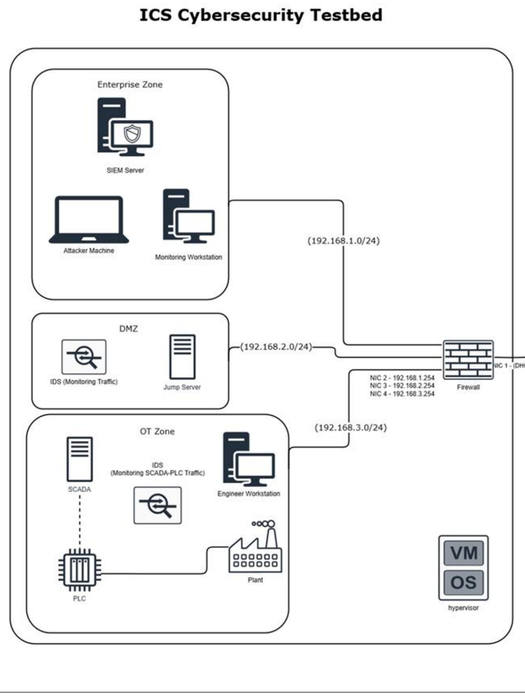

Figure 1. ICS Testbed Architecture Based on Purdue Model — showing segmentation into IT, DMZ, and OT zones.

Testing Scenarios

We executed 8 targeted test cases to validate the system's security measures:

Table 1. Attack Simulation Results Summary.

#	Scenario	Outcome
1	Direct IT to OT network access attempt	Blocked by pfSense firewall rules
2	Authorized OT access via Jumpserver	Successful access with authentication logs generated
3	Sniffing unencrypted Modbus TCP traffic	Modbus requests visible in clear text (Wireshark capture)
4	Malicious Modbus command injection	Detected and alerted by Suricata IDS
5	Emergency stop attack (Simulated shutdown)	Attack executed; event logged by Wazuh and visible on SCADA
6	TLS encryption enabled on Modbus communications	Traffic secured; unauthorized command injection prevented
7	Replay attack over TLS-encrypted Modbus	Replay attack failed due to encrypted session protection
8	Event correlation across Suricata, pfSense, Wazuh	Correlated alerts generated, showing full attack chain

Results

The pfSense firewall successfully enforced network segmentation, blocking unauthorized access between zones. Jumpserver authenticated access to the OT environment and generated audit logs.

Suricata IDS detected unauthorized Modbus TCP activities, while Wazuh SIEM correlated alerts from firewall, IDS, and Jumpserver logs.

Before encryption, Modbus traffic was captured in clear-text; after stunnel configuration, Wireshark confirmed encrypted communications, successfully preventing injection and replay attacks.

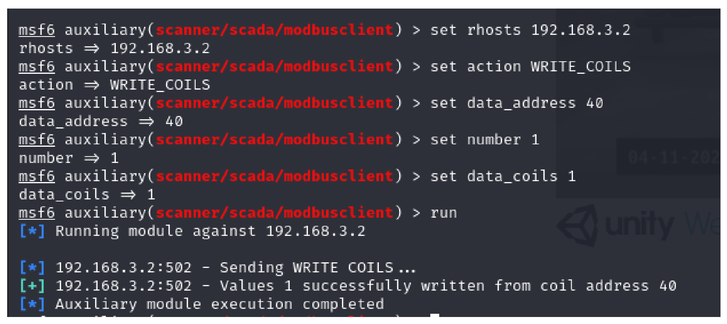

Figure 3. Modbus Command Injection via Metasploit — simulating unauthorized function code to target PLC.

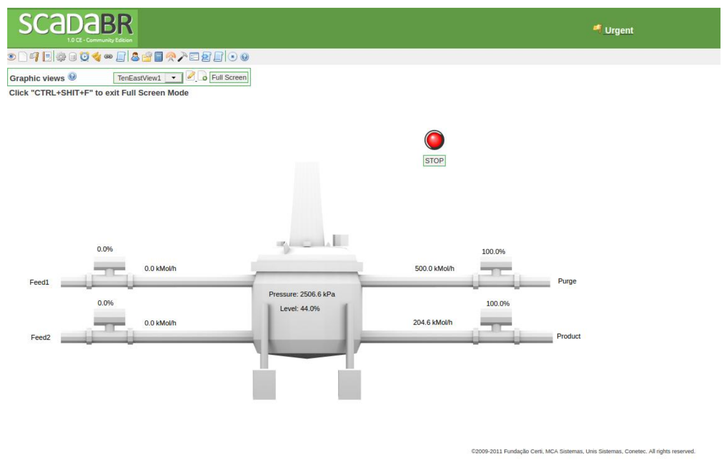

Figure 4. SCADA Interface Showing Emergency Stop — triggered by simulated attack to test system response.

Discussion

The virtual ICS testbed showed that Zero Trust, segmentation, encryption, and real-time detection can be effectively implemented using open-source tools.

Suricata and Wazuh detected coordinated attacks, while pfSense and Jumpserver enforced access control. TLS protected Modbus TCP from tampering.

The testbed is modular and scalable, offering a strong foundation for future ICS security research.

Conclusions

This project built a secure ICS testbed using open-source tools and a virtual Purdue Model layout.It applied Zero Trust, segmented networks, encrypted Modbus TCP, and detected threats in real time using Suricata and Wazuh.

Simulated attacks confirmed the system's ability to stop unauthorized access and replay attacks.

Limitations included no physical devices, limited IDS rules, and partial TLS use. Future work will expand encryption, add hardware, and improve protocol-specific detection.

References

Demo

GitHub repository