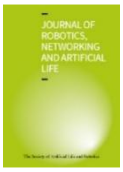
Journal of Robotics, Networking and Artificial Life
Vol. 11(3); March (2026), pp. 210-213
ISSN (Online): 2352-6386; ISSN (Print): 2405-9021
https://alife-robotics.org/jrnal.html



Research Article

Industrial Intrusion Detection System for Schneider Twido Systems

Nai-Yu Chen[1], Jung-Shian Li[2], Hui-Chun Pan[3], I-Hsien Liu[1*]

[1]*Department of Electrical Engineering / M.S. Degree Program on Cyber-Security Intelligence, National Cheng Kung University, No.1, University Rd., East Dist., Tainan City 701401, Taiwan*

[2]*Department of Electrical Engineering / Institute of Computer and Communication Engineering, National Cheng Kung University, No.1, University Rd., East Dist., Tainan City 701401, Taiwan*

[3]*Department of Information Management, Southern Taiwan University of Science and Technology, No. 1, Nan-Tai St., Yongkang Dist., Tainan City 710301, Taiwan*

*Email: nychen@cysec.ee.ncku.edu.tw, jsli@cans.ee.ncku.edu.tw, jopan@stust.edu.tw, ihliu@cysec.ee.ncku.edu.tw**

*Corresponding Author

ARTICLE INFO

Article History

Received 07 November 2024
Accepted 18 January 2026

Keywords

Intrusion Detection System (IDS)
Unified Messaging Application Services (UMAS) Protocol
Programmable Logic Controller (PLC)
Signature-based Detection

ABSTRACT

The Schneider Electric Twido PLCs are analyzed, focusing on the UMAS protocol vulnerabilities, which lack essential security features such as encryption, authentication, and replay attack protection. Key design flaws, including unencrypted communications and unrestricted memory access, are identified and real-world attacks are replicated using publicly disclosed CVEs. A Snort-based Intrusion Detection System (IDS) is developed to address these vulnerabilities, incorporating custom rules to detect abnormal traffic patterns and high-risk function codes within the UMAS protocol. Simulated attack scenarios confirm the IDS's ability to identify unauthorized operations. This solution is lightweight, scalable, and offers practical security improvements for industrial control systems relying on proprietary protocols like UMAS.

1. Introduction

The emergence of Industry 4.0 and the development of the Industrial Internet of Things (IIoT) have led industrial control systems (ICS) to become increasingly networked and intelligent. This transformation, while enhancing efficiency, also exposes key components like Programmable Logic Controllers (PLCs) [1] to new cybersecurity risks. Legacy industrial protocols, including Modbus TCP [2], FINS, and the Unified Messaging Application Services (UMAS) protocol, are still widely used in many PLCs. Since these protocols were not designed with modern cybersecurity measures in mind, they often omit critical protections such as encryption, authentication, and replay defense, leaving systems exposed to potential cyber threats.

In particular, Schneider Electric's Twido PLCs use the proprietary UMAS protocol, which contains several critical vulnerabilities. Attackers may exploit these flaws to bypass authentication, manipulate memory directly, or replay captured commands [3]. Combined with the limited security infrastructure in many ICS environments - such as the absence of gateways or real-time monitoring - these risks become more severe [4].

A lightweight, signature-based intrusion detection system (IDS) utilizing Snort is proposed in this study to address the identified challenge. The system is specifically designed to detect abnormal UMAS traffic by identifying suspicious function codes and packet behavior. It can be integrated into existing ICS network layers to enhance security monitoring without adding operational burden, thereby strengthening the resilience of PLCs in IIoT settings.

2. Background

2.1. UMAS Protocol

UMAS is a Schneider Electric protocol extending Modbus TCP for advanced control and memory operations between Twido PLCs and TwidoSuite [5]. Each packet starts with standard TCP/IP and Modbus headers, followed by a 0x5A function code identifying UMAS operations. Requests include a session ID, UMAS function code, and parameters like memory addresses or variable values. The PLC responds with a status code (0xFE for success, 0xFD for failure) and relevant data (Figure 1).

Privileged actions, such as memory writes or task control, require a session ID obtained via reservation (function 0x10). However, UMAS lacks authentication, encryption, access control, and replay protection, making it vulnerable to unauthorized commands in unsecured ICS networks [6]. A lightweight IDS tailored to UMAS traffic can detect anomalies and mitigate threats.

This section also examines protocol-level vulnerabilities in Twido PLCs, focusing on UMAS. Critical issues identified include the lack of authentication, unencrypted communication, and unrestricted memory access, based on Schneider Electric's Vulnerability Disclosure (SEVD) and related CVEs [7]. Five CVEs related to UMAS were analyzed: (1) CVE-2020-28212: Brute-force attack on PLC reservation; no login attempt limit, (2) CVE-2020-7537: DoS through malformed UMAS function code, (3) CVE-2021-22779: Full read/write access without authentication, (4) CVE-2024-8938: Arbitrary code execution via function pointer injection, (5) CVE-2024-11737: DoS via unauthenticated Modbus/UMAS request. These CVEs were used to model attack scenarios, providing a basis for evaluating intrusion detection rule effectiveness in real ICS environments.

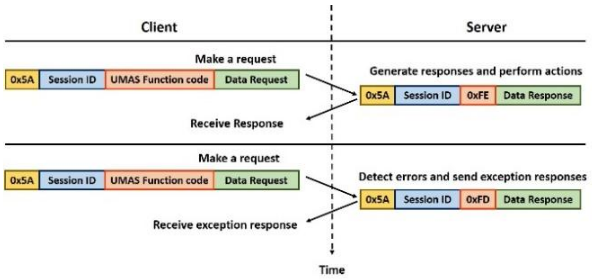


Figure 1 UMAS Communication Transaction Flow

2.2. *Intrusion Detection System*

This study utilizes Snort as the core intrusion detection engine due to its flexible rule definition and strong community support. In ICS environments, Snort operates in NIDS mode, analyzing traffic based on the configuration file (snort.conf) to identify suspicious patterns and generate alerts [8].

Snort's detection process includes four key stages:

(1) capturing and decoding packets via Npcap,

(2) preprocessing fragmented or out-of-order traffic,

(3) matching packets against user-defined rules, and

(4) executing actions such as alerting or logging when a rule is triggered [9].

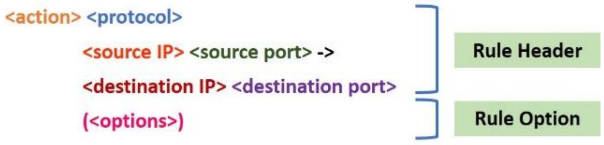

<action> <protocol>
<source IP> <source port> ->
<destination IP> <destination port>
(<options>)
Rule Header
Rule Option

Figure 2 Snort Rule Format

Each Snort rule consists of a header and an option section (Figure 2). By customizing rules to match UMAS protocol behaviors and known exploit patterns, we can detect abnormal activity targeting Twido PLCs.

3. UMAS-Based Attacks Detection Design

In order to effectively detect and mitigate attacks exploiting vulnerabilities in the UMAS protocol, it is crucial to first analyze specific weaknesses that attackers can leverage. For example, CVE-2024-8938 is a vulnerability in Schneider Electric PLCs that arises when the PLC fails to properly verify the legitimacy of the target address and data during the processing of the Write Physical Address function code (0x29) in the UMAS protocol. This flaw allows attackers to perform arbitrary memory writes, overwriting function pointers, stack contents, or other control structures, enabling the execution of injected malicious commands (Figure 3).

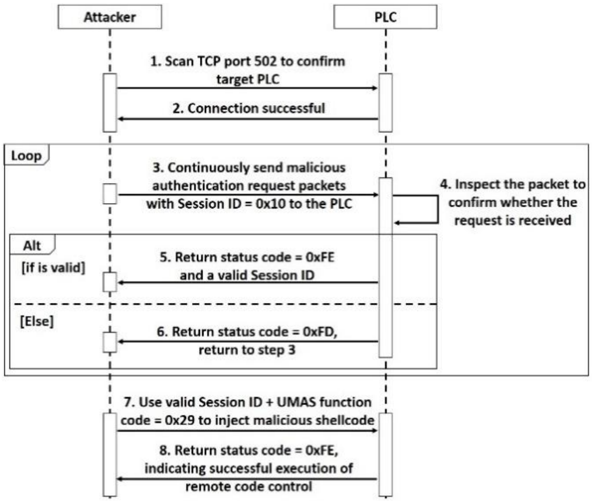
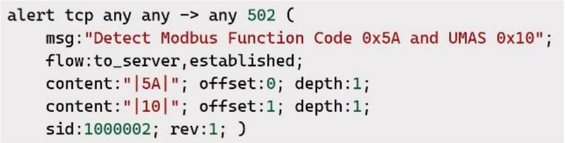

Attacker
PLC
1. Scan TCP port 502 to confirm target PLC
2. Connection successful
Loop
3. Continuously send malicious authentication request packets with Session ID = 0x10 to the PLC
4. Inspect the packet to confirm whether the request is received
Alt
[if is valid]
5. Return status code = 0xFE and a valid Session ID
[Else]
6. Return status code = 0xFD, return to step 3
7. Use valid Session ID + UMAS function code = 0x29 to inject malicious shellcode
8. Return status code = 0xFE, indicating successful execution of remote code control

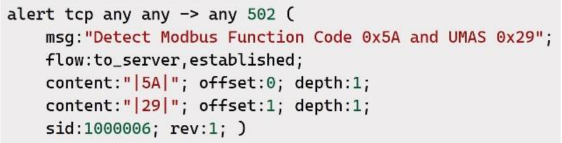
Figure 3 CVE-2024-8938 Attack Timing Diagram

To detect specific UMAS-related attack patterns, Snort-related detection rules were developed. These rules include detecting Modbus Privileged actions (Figure 4), as mentioned in session 2.1 (function 0x10), and the Address & Data Validity Lack (Figure 5) that UMAS seeks to access, as presented in step 7 of Figure 3 (function 0x29).



```
alert tcp any any -> any 502 (
    msg:"Detect Modbus Function Code 0x5A and UMAS 0x10";
    flow:to_server,established;
    content:"|5A|"; offset:0; depth:1;
    content:"|10|"; offset:1; depth:1;
    sid:1000002; rev:1; )
```

Figure 4 Detection of UMAS Privileged Actions



```
alert tcp any any -> any 502 (
    msg:"Detect Modbus Function Code 0x5A and UMAS 0x29";
    flow:to_server,established;
    content:"|5A|"; offset:0; depth:1;
    content:"|29|"; offset:1; depth:1;
    sid:1000006; rev:1; )
```

Figure 5 Detection of Address & Data Validity Lack

4. Experiment

The attack simulation process demonstrates how the attacker initiates the attack and how the intrusion detection system detects and alerts abnormal behaviors (Figure 6). It also illustrates the detailed workflow of the IDS, from establishing a connection with the PLC to applying detection engine rules, recording abnormal packets, and triggering alarms (Figure 7) [10].

The simulation results (Figure 8) demonstrate the behavior of the detection rules during repeated attacks occurring every 10 seconds, along with the corresponding PLC responses and IDS detection results. Two response states were observed: State A indicates the detection of a 0xFE response, while State B indicates the detection of a 0xFD response. The horizontal axis represents the time of each attack attempt.

The results indicate that, regardless of whether the PLC accepts the session request, the NIDS consistently detects the attack. When the attack interval exceeds 50 seconds, the PLC grants access, reflecting its timeout logic. Although the PLC blocks frequent requests, it lacks protection against delayed attacks that exceed the timeout window.

Under high-frequency traffic (every 10 seconds), no packet loss or delay was observed from the PLC, demonstrating the stability of the system. The proposed NIDS was able to continuously detect the attack, confirming its practical applicability in real-world ICS deployments.

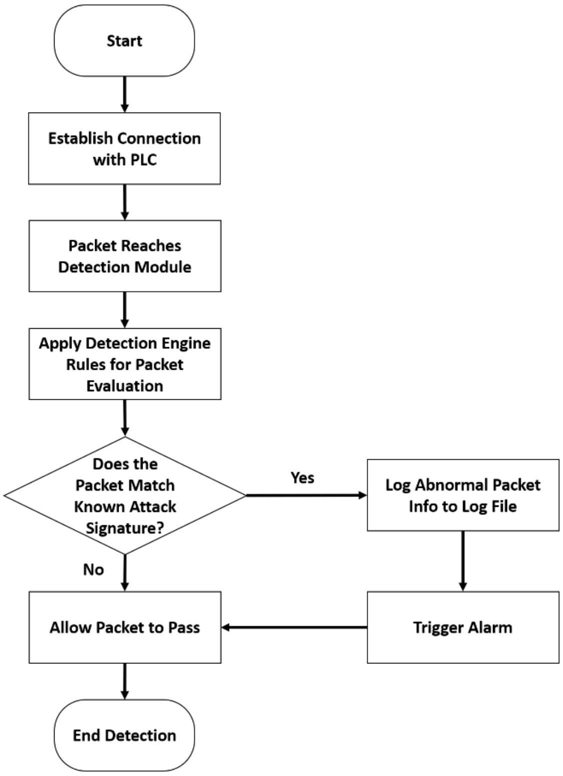

Start
Establish Connection with PLC
Packet Reaches Detection Module
Apply Detection Engine Rules for Packet Evaluation
Does the Packet Match Known Attack Signature?
Yes
No
Log Abnormal Packet Info to Log File
Trigger Alarm
Allow Packet to Pass
End Detection

Figure 6 Intrusion Detection System Interface Monitoring Flowchart

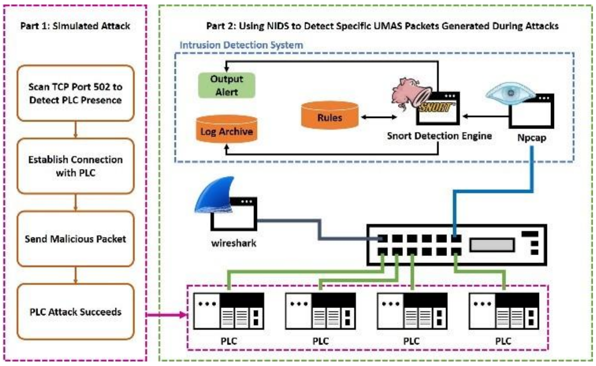

Part 1: Simulated Attack
Scan TCP Port 502 to Detect PLC Presence
Establish Connection with PLC
Send Malicious Packet
PLC Attack Succeeds
Part 2: Using NIDS to Detect Specific UMAS Packets Generated During Attacks
Intrusion Detection System
Output Alert
Log Archive
Rules
SNORT
Snort Detection Engine
Npcap
wireshark
PLC
PLC
PLC
PLC

Figure 7 Attack Script Workflow and Experimental Environment Architecture

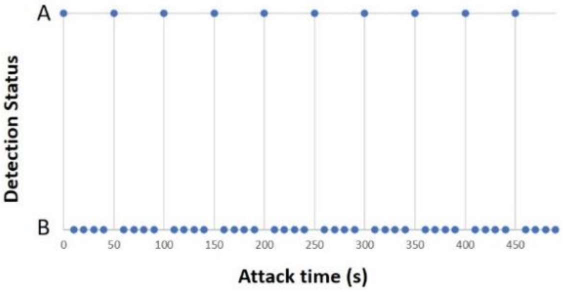


Figure 8 Detection Results of Rules Under Fixed 10-Second Interval Attack Scenario

5. Conclusion

This study proposed a Snort-based IDS for Schneider Twido PLCs, focusing on UMAS protocol vulnerabilities such as the lack of encryption, authentication, and replay protection. Although UMAS has existed for years, it still holds a significant market share, which makes its security weaknesses highly relevant in real-world industrial environments. These issues pose considerable cybersecurity challenges that cannot be overlooked. The proposed system detects abnormal traffic through rule-based analysis and passive monitoring, achieving effective protection without altering existing ICS infrastructure. By offering a lightweight and practical solution, our approach helps mitigate potential risks associated with UMAS-based communications and enhances the overall security posture of legacy systems. Future work will explore integrating machine learning techniques, extending multi-protocol support, and deploying the framework across diverse PLC networks, aiming to further strengthen industrial cybersecurity and contribute to advancing both defensive practices and educational applications.

Acknowledgements

This work was supported by the National Science and Technology Council (NSTC) in Taiwan under contract numbers 114-2222-E-006-005- and 114-2634-F-006-001-MBK.

References

1. K. O. Akpinar and I. Ozcelik, "Methodology to Determine the Device-Level Periodicity for Anomaly Detection in EtherCAT-Based Industrial Control Network," IEEE Transactions on Network and Service Management, 18(2), pp. 2308-2319, 2021.
2. P.-H. Wang, I-E. Liao, K.-F. Kao and J.-Y. Huang, "An intrusion detection method based on log sequence clustering of honeypot for Modbus TCP protocol," IEEE ICASI 2018, Chiba, Japan, 13-17 Apr., 2018.
3. N.-Y. Chen, C.-Y. He, J.-S. Li, C.-S. Yang and I-H. Liu, "Obstructing PLC Operations through Modbus Command Manipulation", ICAROB 2025, Oita, Japan, Feb. 13-16, 2025.
4. Y.-C. Lai, C.-L. Yu, M.-L. Liao, Y.-S. Lin, Y.-C. Chang and J.-L. Chen, "An Intelligence Defense System with SNORT Rules," ICACT 2023, Pyeongchang, South Korea, 19-22 Feb., 2023.
5. F. Katulić, D. Sumina, I. Erceg and S. Groš, "Enhancing Modbus/TCP-Based Industrial Automation and Control Systems Cybersecurity Using a Misuse-Based Intrusion Detection System," SPEEDAM 2022, Sorrento, Italy, 22-24 Jun., 2022.
6. M. M. Aslam, A. Tufail, R. A. A. H. M. Apong, L. C. De Silva and M. T. Raza, "Scrutinizing Security in Industrial Control Systems: An Architectural Vulnerabilities and Communication Network Perspective," IEEE Access, 12, pp. 67537-67573, 2024.
7. Schneider Electric, "Security notifications," 2025, [Online]. Available: https://www.se.com/ww/en/work/support/cybersecurity/security-notifications.jsp. [Accessed 30 Apr. 2025].
8. Cisco, "Snort Documents," 2025, [Online]. Available: https://www.snort.org/documents#preprocessor_documentation. [Accessed 2 Jun. 2025].
9. N. Naik, R. Diao and Q. Shen, "Dynamic Fuzzy Rule Interpolation and Its Application to Intrusion Detection," IEEE Transactions on Fuzzy Systems, 26(4), pp. 1878-1892, 2018.
10. Y. Geng, Y. Chen, R. Ma, Q. Wei, J. Pan, J. Wang, P. Cheng and Q. Wang, "Defending Cyber-Physical Systems Through Reverse-Engineering-Based Memory Sanity Check," IEEE Internet of Things Journal, 10(10), pp. 8331-8347, 2023.

Authors Introduction

Ms. Nai-Yu Chen

She earned her M.S. degree in the M.S. program on Cyber-Security Intelligence from National Cheng Kung University, Taiwan in 2025. Her research focuses on ICS security and network-based intrusion detection.

Prof. Jung-Shian Li

He is a full professor at the Department of Electrical Engineering, National Cheng Kung University, Taiwan. He earned his B.S. and M.S. degrees in Electrical Engineering from National Taiwan University in 1990 and 1992 and obtained his Ph.D. in Computer Science from the Technical University of Berlin, Germany, in 1999. His academic responsibilities include teaching courses on communication systems. His current research interests involve cloud computing, cybersecurity and network management. In addition, he leads funded research projects on cybersecurity and critical infrastructure protection.

Prof. Hui-Chun Pan

She is an assistant professor in the Department of Information Management at Southern Taiwan University of Science and Technology, Taiwan. She received her Ph.D. in Management Information Systems from National Kaohsiung University of Science and Technology. Her research interests encompass managerial psychology, knowledge management, IT usage behavior, e-commerce, and STEM education.

Prof. I-Hsien Liu

He is an assistant professor at the Department of Electrical Engineering, National Cheng Kung University (NCKU), Taiwan. In 2015, he earned his Ph.D. in Computer and Communication Engineering from NCKU. His research focuses on OT security, cybersecurity, and communication systems.