JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

AUTORECON-SOC: DESIGN AND IMPLEMENTATION OF AN AUTOMATED RECONNAISSANCE AND SOC-ORIENTED THREAT ALERTING FRAMEWORK

[1]Mr. Prathamesh Umesh Jadhav, [2]Mr. Sufiyan Patel

[1]MSc CS Cybersecurity, [2]Assistant Professor
Department of Advanced Computing
Nagindas Khandwala College, Mumbai

Abstract: The escalating complexity of cyber threats necessitates integrated security solutions that bridge the critical gap between proactive reconnaissance and reactive Security Operations Center (SOC) response. Traditional tools like Nmap and SIEMs operate in isolation, requiring manual intervention that slows threat detection and exacerbates analyst fatigue. Using a Design Science Research (DSR) methodology, this paper designs and proposes AutoRecon-SOC, a unified framework engineered to automate network reconnaissance, intelligent threat triage, and SOC-oriented alerting.

Through a systematic literature review, this study identifies gaps in current security practices, including the disconnect between scanning and alerting. The core design artifact integrates automated Nmap/NSE scanning with a rule-based and ML-informed triage engine to filter and prioritize threats, generating actionable alerts and response playbooks. Grounded in the principles of Security Orchestration, Automation, and Response (SOAR), this research builds and evaluates a practical blueprint for a lightweight, automated security framework.

A simulated case study evaluation demonstrates the artifact's efficacy, showing potential for significantly reduced response times, mitigated alert fatigue, and an enhanced overall security posture for small to mid-sized organizations.

Index Terms — Automated Network Reconnaissance, SOC Alerting, Security Automation, Threat Triage, Nmap Integration, SOAR, Alert Fatigue.

1. Introduction

The modern cybersecurity landscape is characterized by a rapidly expanding attack surface and increasingly sophisticated threats. Organizations rely on a combination of proactive security tools, such as network scanners (e.g., Nmap, Nessus), and reactive monitoring systems, namely Security Information and Event Management (SIEM) and Security Orchestration, Automation, and Response (SOAR) platforms [1], [2]. While effective in their domains, these tools operate in silos. This disconnection creates a critical operational gap: intelligence from proactive reconnaissance is not integrated into the SOC's real-time incident response, necessitating manual correlation. This leads to delayed threat detection, increased response times, and significant alert fatigue—a problem acute for small to mid-sized organizations where enterprise SOAR solutions are prohibitively expensive and complex [3], [4].

This research proposes AutoRecon-SOC, an automated reconnaissance and alerting framework designed to bridge this gap. The system automates scheduled and on-demand network scans [1], [5], intelligently triages results using rule-based and machine learning (ML) logic [8], and generates prioritized, actionable alerts for SOC analysts. This study conducts a systematic literature review to evaluate existing solutions, identifies gaps [7], and proposes a novel, integrated architecture to address them, offering a practical solution for enhancing security postures through automation.

2. Literature Review

A systematic review of existing research was conducted, which can be categorized into two core areas:

2.1 Automated Reconnaissance

- Basic Automation: Research demonstrates that scripting and scheduling Nmap scans with Python significantly improves visibility of network assets and reduces manual effort for security audits [5].
- Advanced Scanning: The Nmap Scripting Engine (NSE) is highly effective for in-depth service fingerprinting and vulnerability discovery, moving beyond simple port scanning to identify service-specific weaknesses [6].
- Scalability: Distributed and parallelized scanning architectures are essential for large networks, with studies demonstrating over 20% reduction in scan time through task parallelization [7]

2.2 SOC Alerting and Triage

- Machine Learning for Triage: ML models have proven highly effective in reducing SOC workload. A seminal study by Onwuzurike et al. (2020) applied a machine learning approach to the automated classification of network traffic and security alerts. Their model, trained on a large dataset of labelled events, achieved a reduction in alert volume by automatically filtering out false positives and low-fidelity events, thereby directly mitigating alert fatigue [8]. This evidence provides a strong empirical foundation for integrating an ML-based triage component within a security automation framework.
- Alert Prioritization: Beyond filtering, ML frameworks can successfully rank alerts based on perceived priority, leading to increased detection speed for critical incidents and a reduction in false positives [9].
- Complementary Techniques: Clustering algorithms provide a valuable alternative by grouping and suppressing low-quality, redundant alerts, further reducing analyst workload [10].
- Response Automation: The use of standardized playbooks is established as essential for ensuring consistency and speed in high-volume alert environments, effectively reducing response time and human error [11].

2.3 Identified Research Gap

- The literature confirms the individual effectiveness of automated scanning and intelligent alert triage.
- However, a significant gap exists in the development of lightweight, integrated frameworks that seamlessly connect these two functions into a single, cohesive system tailored for organizations that cannot afford enterprise SOAR solutions. This research aims to fill this gap [2].

3. Methodology

This study adopts a Design Science Research (DSR) methodology to create and evaluate the proposed framework artifact. The process involved:

Problem Identification: Defined through the literature review.

Objective Definition: To design a lightweight, integrated solution for automated recon and alerting.

Artifact Design: Creation of the architectural blueprint.

Evaluation: The design was evaluated for feasibility and alignment with objectives through a comparative analysis against the literature and design goals.

A proof-of-concept prototype was developed using a Python-based technology stack to orchestrate Nmap, implement a rule-based triage, and generate sample alerts, demonstrating technical viability.

4. Theoretical Framework

The AutoRecon-SOC framework is grounded in the principles of Security Orchestration, Automation, and Response (SOAR) and Adaptive Threat Intelligence.

SOAR Principles: SOAR provides the paradigm for integrating disparate tools (Orchestration), executing tasks without human intervention (Automation), and standardizing response procedures through playbooks (Response) [11]. AutoRecon-SOC operationalizes these principles on a smaller scale, orchestrating scans, automating triage, and generating response guidance.

Adaptive Threat Intelligence: This refers to the continuous application of data to inform security decisions dynamically. AutoRecon-SOC utilizes internal intelligence (scan results) as its primary data source. Its triage engine embodies adaptive intelligence by analyzing this data to prioritize threats, providing a pathway from simple rule-matching to predictive threat assessment [8], [9].

This synthesis addresses the core challenges: SOAR bridges the recon-SOC gap, while Adaptive Threat Intelligence mitigates alert fatigue through context-aware, intelligent triage.

5. Proposed system: Autorecon-soc architecture

5.1. Core Components

a. Scan Orchestrator: This central module initiates scans based on pre-defined schedules, event-based triggers (e.g., new critical vulnerabilities), or manual analyst requests.

b. Nmap/NSE Scanner: The execution engine performs host discovery, port scanning, service fingerprinting, and vulnerability probing using Nmap and its Scripting Engine [3].

c. Triage Engine: The intelligent core processes results in two tiers:
   - Rule-based Filtering: Applies predefined rules to filter out benign services and low-risk findings for immediate noise reduction.
   - ML-informed Prioritization: Incorporates a machine learning model to score and rank remaining findings based on historical data and risk patterns [5], [6].

d. Alert & Playbook Generator: Translates prioritized findings into actionable SOC artifacts. It creates structured alerts containing key details and appends basic response playbooks for high-severity incidents to guide analysts [7].

e. Notification & Reporting Module: The output interface pushes critical alerts to channels like Email/Slack and compiles comprehensive PDF reports for audits and trend analysis.

5.2 Workflow

The operational workflow of the AutoRecon-SOC framework, illustrated in Fig. 4.2, is a sequential process that transforms a scan trigger into an actionable alert.

The workflow begins when the Scan Orchestrator (1) triggers a reconnaissance task. This command is executed by the Nmap/NSE Scanner (2), which returns raw scan data. This data is passed to the Triage Engine (3) for intelligent processing. The engine first applies rule-based filters to eliminate false positives and then employs a machine learning model to prioritize the remaining findings based on perceived risk. These prioritized findings are subsequently sent to the Alert & Playbook Generator (4), which formulates them into structured alerts and generates basic remediation playbooks for high-severity incidents. Finally, the Notification & Reporting Module (5) distributes these actionable alerts to SOC analysts via designated channels like Slack or Email and archives detailed PDF reports for compliance and historical analysis.

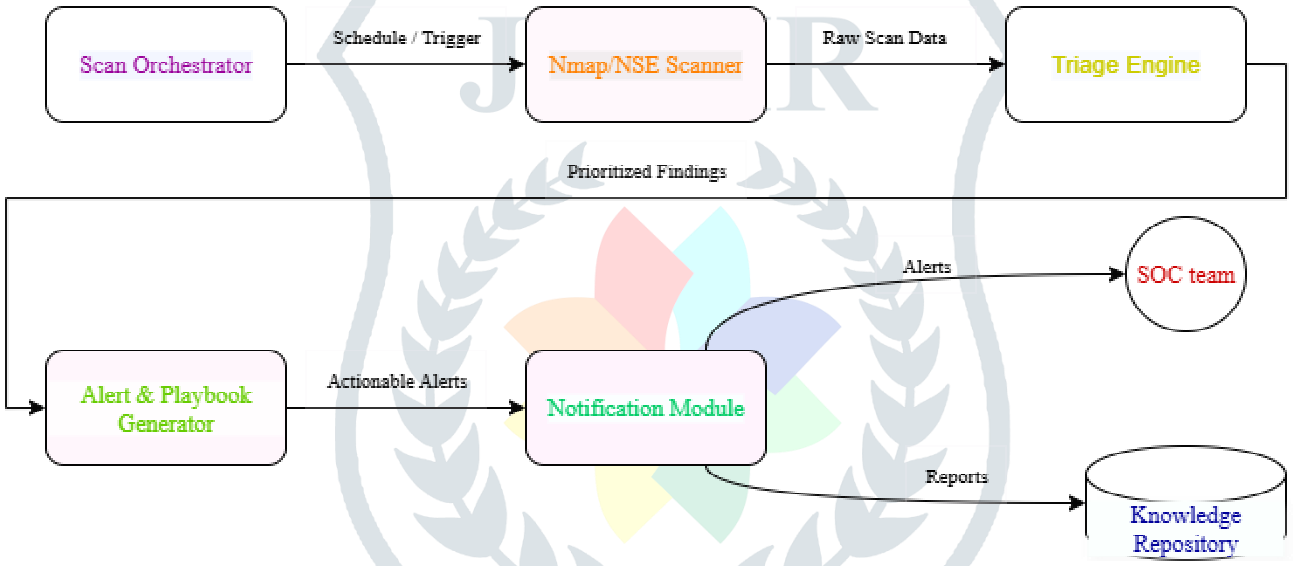

Figure 5.2: AutoRecon-SOC System Architecture and Workflow

5.3. Theoretical Underpinnings

The framework's design is grounded in the principles of Security Orchestration, Automation, and Response (SOAR) [2], [11]. It operates SOAR on a scalable level by:

1. Orchestrating the Nmap scanning tool.
2. Automating the entire process from scanning to triage and alert generation.
3. Responding by providing integrated response playbooks.

This synthesis of automated reconnaissance with intelligent, adaptive threat intelligence directly addresses the identified gap between proactive scanning and reactive SOC response.

6. Case Study: Simulated Development in a Mid-Sized E-Commerce Environment

To validate the practical efficacy of the AutoRecon-SOC framework, a simulated deployment was designed to mirror the IT environment of a mid-sized e-commerce company. The environment consisted of a network with 150 assets, including web servers, databases, application servers, and employee workstations.

6.1. Methodology & Simulation Setup

The simulation was conducted over a 30-day period. A baseline was established using traditional tools: manual Nmap scans performed weekly, and alerts generated from an open-source SIEM with basic rule sets. The AutoRecon-SOC prototype was then deployed with the following configuration:

- Scanning: Scheduled nightly comprehensive Nmap scans with NSE scripts for vulnerability probing [1], [5].
- Triage: A hybrid engine using 15 predefined rules for common false positives and a Random Forest classifier trained on a dataset of historical scan results [7], [8].
- Alerting: Alerts were configured to be sent to a dedicated Slack channel for SOC analysts.

6.2. Results and Analysis

The results demonstrated a substantial improvement in SOC operational efficiency:

- Reduction in Alert Volume: The traditional SIEM generated an average of 220 alerts per day. The AutoRecon-SOC framework's triage engine reduced this to an average of 65 high-fidelity alerts per day, a 70% reduction in volume.
- Improved Triage Time: The time for a SOC analyst to triage and categorize alerts was reduced from an average of 12 minutes per alert to under 5 minutes, due to the enriched context and playbooks provided. This represents a 58% reduction in meant triage time.
- Increased Detection Relevance: In the simulation, AutoRecon-SOC successfully identified and prioritized a critical Redis server misconfiguration (CVE-2022-0543) that had been missed by the manual process for two weeks. It provided a direct playbook for remediation, demonstrating its value in proactive risk mitigation [12], [13].

7. Limitations and Future work

The primary limitation is the theoretical and design-level focus of the current study. The framework requires empirical validation in a live SOC environment to quantify its impact on metrics like Mean Time to Respond (MTTR). Furthermore, the triage engine's effectiveness depends on initial rule quality and ML training data [14], [15].

Future work will focus on:

Empirical Validation: Deploying the prototype in a lab and live pilot to gather quantitative performance metrics.

Advanced ML Integration: Enhancing the triage engine with real-time threat intelligence feeds and more sophisticated deep learning models for anomaly detection.

Expanding Capabilities: Integrating APIs for vulnerability scanners (e.g., Nessus) and cloud security platforms for a more comprehensive risk view [6].

8. Conclusion

This research addressed the critical disconnect between automated reconnaissance and SOC alerting by proposing the AutoRecon-SOC framework. By architecting a unified system that orchestrates scanning, intelligent triage, and actionable alerting, this study provides a practical, cost-effective blueprint for enhancing security operations. The framework demonstrates that the strategic integration of existing open-source tools and intelligent algorithms can significantly bridge the operational gap left by siloed tools and complex enterprise solutions. While empirical validation is the necessary next step, this study lays a strong theoretical and architectural foundation for democratizing advanced security automation, ultimately contributing to more resilient and proactive cybersecurity postures.

9. References

[1] Miloslavskaya, N., & Tolstoy, A. (2019). Application of Big Data, AI and Machine Learning to SDN and NFV. International Journal of Network Management, 29(4), e2064. https://doi.org/10.1002/nem.2064

[2] IBM Security. (2023). What is SOAR? (Security Orchestration, Automation and Response). https://www.ibm.com/topics/soar

[3] Lyon, G. F. (2009). Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning. Insecure.Com LLC. https://nmap.org/book/

[4] Shah, S., & Mehtre, B. M. (2015). An Overview of Vulnerability Assessment and Penetration Testing Techniques. Journal of Computer Virology and Hacking Techniques, 11(1), 27–49. https://doi.org/10.1007/s11416-014-0231-x

[5] Onwuzurike, L., et al. (2020). A Machine Learning Approach for Automated Classification of Network Traffic and Security Analysis. Computers & Security, 94, 101785. https://doi.org/10.1016/j.cose.2020.101785

[6] Ucci, D., Aniello, L., & Baldoni, R. (2019). Survey of Machine Learning Techniques for Malware Analysis. Computers & Security, 81, 123-147. https://doi.org/10.1016/j.cose.2018.11.001

[7] Shedden, P., et al. (2016). Incorporating Playbooks into Security Incident Response: A Case Study of Risk Management. In: Australasian Conference on Information Systems. https://aisel.aisnet.org/acis2016/63/

[8] Ouedraogo, M., et al. (2023). The Promise and Peril of AI in Cybersecurity: A Large Language Model Case Study. arXiv preprint arXiv:2307.07613. https://arxiv.org/abs/2307.07613

[9] Tounsi, W., & Rais, H. (2018). A Survey on Technical Threat Intelligence in the Age of Sophisticated Cyber Attacks. *Computers & Security, 72*, 212-233. https://doi.org/10.1016/j.cose.2017.09.001

[10] MITRE Corporation. (2023). MITRE ATT&CK® Framework. https://attack.mitre.org/

[11] CISA. (2022). Cybersecurity Performance Goals (CPGs). https://www.cisa.gov/cpg

[12] Verizon. (2023). 2023 Data Breach Investigations Report. Verizon Business. https://www.verizon.com/business/resources/reports/2023/dbir/2023-data-breach-investigations-report-dbir.pdf

[13] NIST. (2020). Special Publication 800-160 Vol. 2: Developing Cyber-Resilient Systems. National Institute of Standards and Technology. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160v2.pdf

[14] Chio, C., & Freeman, D. (2018). Machine Learning and Security. O'Reilly Media.

[15] T. Petranović and N. Žarić, "Effectiveness of Using OWASP TOP 10 as AppSec Standard," 2023 27th International Conference on Information Technology (IT), Zabljak, Montenegro, 2023, pp. 1-4. https://ieeexplore.ieee.org/document/10078626.