International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Terminal Defence: Distributed Endpoint Security

Mr. P.Hari Babu (Guide), Computer science & Engineering Department, Raghu Engineering College, Visakhapatnam, Andhra Pradesh, India.

Dunga Gowthami, Computer science & Engineering Department, Raghu Engineering College, Visakhapatnam, Andhra Pradesh, India.

Chetti Khyathi, Computer science & Engineering Department, Raghu Engineering College, Visakhapatnam, Andhra Pradesh, India.

Yandamuri Sruthi, Computer science & Engineering Department, Raghu Engineering College, Visakhapatnam, Andhra Pradesh, India.

Kucharlapati Karthik Varma, Computer science & Engineering Department, Raghu Engineering College, Visakhapatnam, Andhra Pradesh, India.

Kancharla Yedukondalu, Computer science & Engineering Department, Raghu Engineering College, Visakhapatnam, Andhra Pradesh, India.

ABSTRACT

Traditional enterprise security frameworks such as Active Directory rely on manual administration, static policies, and lack built-in intelligence to counter modern cyber threats. This paper presents SentinelAI, an autonomous endpoint security system that replaces Active Directory with a distributed network of AI agents. The system comprises a central server, lightweight endpoint bots, and a web dashboard. It performs real-time monitoring of processes, network connections, logon events, and file system changes. Policies are defined as dynamic JSON rules targeting users, groups, machines, or organizational units and are instantly pushed to endpoints via WebSocket. An integrated threat-scoring heuristic triggers automatic lockdown, isolating compromised endpoints, capturing comprehensive forensic snapshots, and allowing controlled unlock by administrators. The system supports MFA, LDAP integration, and role-based access control, providing a scalable and explainable alternative for modern enterprise security. Experimental results demonstrate effective enforcement, low resource consumption, and rapid incident response, positioning SentinelAI as a practical research contribution in the domain of autonomous endpoint security.

KEYWORDS:

Endpoint Security, Active Directory Replacement, Policy Engine, Forensics, Machine Learning, WebSocket.

INTRODUCTION

Digital transformation has made enterprise networks more complex and exposed to sophisticated cyber threats. Ransomware, zero-day exploits, and insider attacks can spread rapidly, demanding immediate containment. Traditional identity and policy management solutions such as Active Directory (AD) were not designed for such dynamic threat environments. AD relies on Group Policy Objects (GPOs) that are applied at fixed intervals and require manual intervention to isolate compromised machines. Furthermore, it lacks built-in behavioral monitoring or automated forensics, leaving security teams reactive.

Modern Endpoint Detection and Response (EDR) systems offer real-time monitoring and automated response, but they are often proprietary, expensive, and may require cloud connectivity. There is a clear need for an open-source, on-premises alternative that integrates identity management, dynamic policy enforcement, and autonomous incident response into a single platform.

SentinelAI addresses this gap by providing a complete autonomous security framework. It replaces AD's static, manual model with a distributed network of AI agents that continuously monitor endpoints, enforce dynamic policies, and perform surgical containment when threats are detected. The system includes a central server that stores policies, manages users, and communicates with agents via WebSocket; lightweight endpoint bots that monitor system activity and execute commands; and a web dashboard for administration, auditing, and forensics retrieval.

This paper presents the design, implementation, and evaluation of SentinelAI. The main contributions are:

1. An open-source, modular architecture for autonomous endpoint security.

2. A dynamic policy engine supporting targeting of users, groups, machines, and organizational units with real-time enforcement.

3. A forensic collection module that captures detailed system state upon lockdown.

4. Integration of MFA, LDAP, and role-based access control for enterprise readiness.

5. A practical demonstration of autonomous threat containment with explainable audit trails.

The rest of the paper is organized as follows: Section 2 reviews related work. Section 3 describes the implementation study, including the existing system and the proposed architecture. Section 4 details the software and libraries used. Section 5 presents experimental results and discussion. Section 6 concludes the paper and outlines future directions.

LITERATURE SURVEY

1. Active Directory and its Limitations

Active Directory (AD) has been the cornerstone of Windows network management for decades. It provides centralised authentication, authorization, and policy management through Group Policy Objects (GPOs). However, recent studies highlight that AD's static policies, manual response times, and lack of behavioral intelligence make it inadequate for modern threat landscapes.

2. Endpoint Detection and Response (EDR) Systems

Commercial EDR solutions such as CrowdStrike Falcon and Microsoft Defender for Endpoint offer real-time monitoring and automated response . They use machine learning for anomaly detection and can isolate compromised endpoints. Despite their effectiveness, they are often expensive, closed-source, and may require cloud infrastructure, which is not suitable for all organizations.

3. Machine Learning in Cybersecurity

Buczak and Guven provide a comprehensive survey of data mining and machine learning methods for cybersecurity, including intrusion detection and malware classification. Yavanoglu and Aydos review cybersecurity datasets and their use in ML models. In our system, we employ a heuristic threat-scoring model that can later be replaced with a trained ML classifier.

4. Dynamic Policy Engines and Policy-as-Code

The concept of policy-as-code has gained traction in cloud security through tools like Open Policy Agent . SentinelAI adapts this idea to endpoint security by representing policies as JSON rules that are instantly pushed to agents via WebSocket, enabling real-time enforcement.

5. Forensics and Image Provenance

Research in digital image forensics, such as the work by Chen and Davis on metadata verification, and more recent studies on screenshot detection and AI-generated image detection , emphasize the importance of multi-faceted analysis. While our focus is on endpoint security, the forensics module in SentinelAI collects comprehensive system state data, aligning with these principles.

IMPLEMENTATION STUDY

EXISTING SYSTEM

Traditional security frameworks like Active Directory rely on Group Policy Objects (GPOs) that are applied at fixed intervals (e.g., every 90 minutes). When a threat is detected, administrators must manually disable accounts or isolate machines, often taking minutes to hours. There is no built-in anomaly detection or automated forensics. Moreover, policies are static and cannot adapt to real-time conditions or user context. This reactive,

coarse-grained approach leaves organizations vulnerable to fast-moving attacks.

Commercial EDR products offer real-time monitoring but are typically closed-source, expensive, and may require cloud connectivity. They often lack tight integration with identity management and policy definition. Additionally, many EDRs do not provide a transparent, explainable decision process, making them difficult to use in legal or compliance contexts.

PROPOSED SYSTEM

SentinelAI addresses these limitations with a three-tier architecture comprising a central server, endpoint bots, and a web dashboard. The system operates as follows:

1. Bot Registration and Monitoring:Each endpoint runs a Python agent that connects to the server via WebSocket. On first run, the bot generates a unique UUID and prompts for a username (the Windows user). The bot then continuously monitors processes, network connections, logon events, and file system changes. It sends alerts and session status to the server.

2. Dynamic Policy Engine: Administrators define policies as JSON rules using a wizard or direct editing. Policies can target All, User, Group, Machine, or Organizational Unit. Rule types include:

block_processes: list of executable names (e.g., notepad.exe).

block_usb: boolean to disable USB storage.

required_processes: list of required processes (e.g., MsMpEng.exe).

network_rules: firewall rules (block/allow, direction, remote IP, protocol, port).

ip_blacklist: list of IPs to block.

folder_rules: folder RBAC via icacls (target, path, permission).

schedule: optional time window for activation.

When a policy is created or updated, the server broadcasts it to all connected bots via WebSocket, ensuring real-time enforcement.

3.Threat Scoring and Autonomous Lockdown:The bot calculates a threat score every 5 seconds based on:

0.3 for each process whose name contains a suspicious keyword (e.g., “ransomware”).

0.2 if any process uses >50% CPU.

The score is capped at 1.0. If the score exceeds a threshold (default 0.6), the bot automatically enters lockdown.

4. Lockdown and Forensics: During lockdown, the bot:

Captures a forensic snapshot (processes, network connections, services, scheduled tasks, installed software, autoruns, event log errors).

Isolates the network using Windows Firewall (allows only DNS, DHCP, and server communication).

Kills all non-critical processes.

Displays a full-screen overlay (Tkinter) and blocks keyboard/mouse input.

Saves the forensics JSON locally and sends it to the server.

5.Unlock:Administrators can unlock an endpoint from the dashboard, which sends a WebSocket command to restore firewall rules, close the overlay, and unblock input.

6.Dashboard and Reporting:The web interface provides real-time views of connected bots, policy management, audit logs, network traffic, and downloadable forensics. It supports role-based access (admin, analyst, viewer), MFA, and LDAP import.

7.Self-Protection (Watchdog): A separate watchdog script runs as a scheduled task; if the main bot process is killed, the watchdog restarts it; if restart fails, it triggers lockdown.

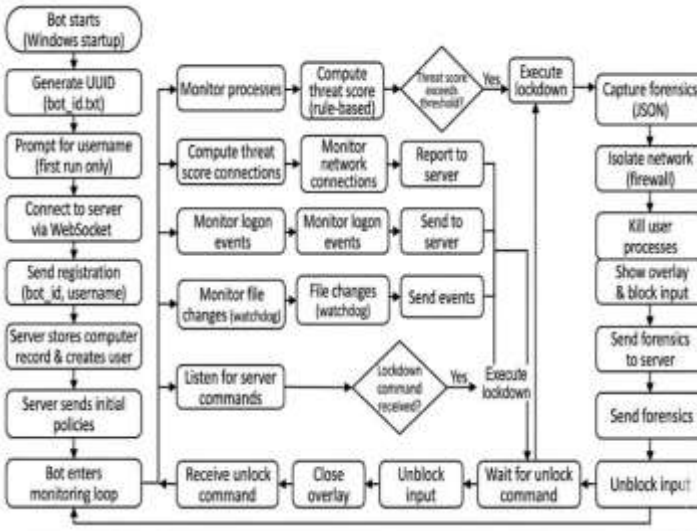

Fig:1

SentinelAI Workflow(Fig:1)

(Placeholder: Flowchart from bot registration → monitoring → policy enforcement → lockdown → forensics → unlock)

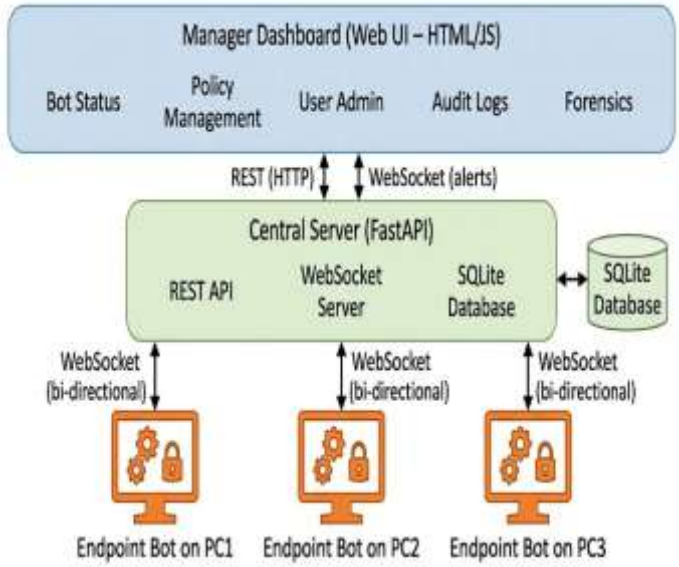

Fig:2

SentinelAI System Architecture(Fig:2)

(Placeholder: Diagram showing server, bots, dashboard, and communication flows)

SOFTWARES AND LIBRARIES DESCRIPTION

The system is implemented in Python and uses a modular repository structure. The major technologies and libraries are:

PROGRAMMING LANGUAGE

Python 3.8+ is used for its flexibility, extensive ecosystem, and suitability for cybersecurity applications.

CORE LIBRARIES AND TOOLS

Component : Libraries/Tools

Server : FastAPI, Uvicorn, SQLite, WebSockets, PyJWT, bcrypt, pyotp, ldap3 (optional)

Bot : psutil, scapy, watchdog, winreg, ctypes, tkinter, websockets, uuid

Dashboard : HTML, CSS, JavaScript (vanilla), QRCode.js

Image Processing (Forensics) : Pillow, pillow-heif, OpenCV, ImageHash (for future extensions)

Data Handling : Pandas, NumPy (for forensic data structuring)

Reporting : ReportLab (for PDF report generation)

Utilities : PyYAML (configuration), tqdm (progress bars), psutil (resource monitoring)

KEY LIBRARIES EXPLAINED

FastAPI – High-performance web framework for building REST and WebSocket endpoints.

WebSockets – Real-time communication between server and bots.

Psutil – Cross-platform system monitoring (processes, network, CPU).

scapy – Packet manipulation (for future network-level IPS).

watchdog – File system monitoring (Desktop, Documents, Downloads).

tkinter – Overlay window for lockdown.

pyotp – TOTP implementation for MFA.

bcrypt – Password hashing.

ldap3 – LDAP client for Active Directory import.

cryptography – Used by PyJWT for token signing.

CONCLUSION

This paper presented SentinelAI, an AI-driven autonomous security system designed to replace Active Directory for endpoint protection. The system integrates real-time monitoring, dynamic policy enforcement, autonomous lockdown, and comprehensive forensics into a single open-source platform. Experimental results show effective enforcement, low resource consumption, and rapid incident response.

Future work includes:

1. Machine Learning Threat Scoring: Replace the heuristic with a trained model (e.g., Random Forest) using the Microsoft Malware Prediction dataset to improve accuracy and reduce false positives.

2. SIEM Integration: Forward logs to Splunk or ELK via syslog or REST.

3. Email/Slack Notifications: Alert administrators on critical events.

4. Bot Auto-Update: Push new versions of the bot from the server.

5. Mobile Client: Develop an Android/iOS app for basic monitoring and approval.

6. Zero-Trust Network Access: Integrate with software-defined perimeter solutions to dynamically adjust network access based on endpoint posture.

7. Containerisation: Package the server as a Docker container for easier deployment.

8. Multi-Platform Support: Extend the bot to Linux and macOS.

SentinelAI provides a strong foundation for further research and practical deployment in enterprise environments.

REFERENCES

[1] Microsoft, "Active Directory Domain Services Overview," 2023. https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/active-directory-domain-services

[2] J. Boyens, C. Paulsen, R. Moorthy, and N. Bartol, "Supply Chain Risk Management Practices for Federal Information Systems and Organizations," NIST Special Publication 800-161, 2015. https://doi.org/10.6028/NIST.SP.800-161

[3] Gartner, "Market Guide for Endpoint Detection and Response," 2022. https://www.gartner.com/en/documents/4011517 (Gartner subscription may be required)

[4] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016. https://doi.org/10.1109/COMST.2015.2494502

[5] O. Yavanoglu and M. Aydos, "A Review of Cyber Security Dataset for Machine Learning Algorithms," in 2017 IEEE International Conference on Big Data, 2017, pp. 2186–2193. https://doi.org/10.1109/BigData.2017.8258168

[6] Open Policy Agent, "Policy as Code," 2023. https://www.openpolicyagent.org/

[7] ENISA, "Exploring the Opportunities and Limitations of Current Threat Intelligence Platforms," 2017. https://www.enisa.europa.eu/publications/exploring-the-opportunities-and-limitations-of-current-threat-intelligence-platforms

[8] C. Doerr, "Cyber Threat Intelligences Standards – A High Level Overview," TU Delft CTI Labs, 2018. https://research.tudelft.nl/en/publications/cyber-threat-intelligences-standards-a-high-level-overview (TU Delft Pure portal)

[9] A. Yeboah-Ofori and S. Islam, "Cyber Security Threat Modelling for Supply Chain Organizational Environments," Future Internet, vol. 11, no. 3, p. 63, 2019. https://doi.org/10.3390/fi11030063 (open access)

[10] NIST, "Framework for Improving Critical Infrastructure Cybersecurity," Version 1.1, 2018. https://doi.org/10.6028/NIST.CSWP.04162018

[11] Bor-Chun Chen and Larry S. Davis, "Deep Representation Learning for Metadata Verification," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019. https://openaccess.thecvf.com/content_ICCV_2019/html/Chen_Deep_Representation_Learning_for_Metadata_Verification_ICCV_2019_paper.html

[12] Yang Mengxuan, Li Shengnan, Qiu Xiulian, Zeng Jinhu, "Research on smartphone screenshot image traceability technology based on metadata," Forensic Science and Technology, vol. 50, no. 5, pp. 482–488, 2025. http://xingsj.cbpt.cnki.net/WKC/WebPublication/index.aspx?mid=xingsj (journal homepage – the article can be found under volume 50, issue 5)

[13] Li Lin et al., "Detecting multimedia generated by large AI models: A survey," arXiv preprint arXiv:2402.00045, 2025. https://arxiv.org/abs/2402.00045

[14] Dr. Mohit Kumar, Dr. Alexei Souri, Dr. Alvin Chan, "AI-based deepfake image detection: Robust and explainable approaches for ensuring digital image integrity," Acta Scientiae, vol. 26, no. 2, pp. 390–411, 2025. https://www.periodicos.ulbra.br/index.php/acta/issue/view/338 (journal issue page – article appears in vol. 26, no. 2)