IJERST

International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 1 (2026)

ijerst.editor@gmail.com

editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 22, No. 1, 2026

Peer Reviewed, Referred & Indexed Journal

Paper Research

HADES: Detecting Active Directory Attacks via Whole Network Provenance Analytics

Mr. N. Kiran Kumar, M.Tech- Assistant Professor, Department of MCA, Bapatla Engineering College, Bapatla, Andhra Pradesh

Mr. AATLA MADHUSUDHAN REDDY, Reg No: Y25MC23001, Mr. GOLLA VENKATA NITHISH, Reg No: Y25MC23021, Ms. LAKKAKULA VENKATA LAKSHMI, Reg No: Y24MC23041, Ms. KOTHAPULI SIREESHA, Reg No: Y25MC23039,
Department of MCA, Bapatla Engineering College, Bapatla, Andhra Pradesh

***Abstract*—Active Directory (AD) is the backbone of identity and access management in enterprise networks and a prime target for Advanced Persistent Threat (APT) actors. While traditional intrusion detection systems (IDS) effectively detect malware-driven attacks, they struggle to identify stealthy, credential-based AD attacks that span multiple hosts. Recent provenance-based intrusion detection systems (PIDS) leverage causal analysis to expose malicious behaviors, but existing solutions are largely limited to intra-machine tracing, preventing a holistic view of attacker movement across the network.**

We present HADES, the first provenance-based IDS capable of accurate causality-driven cross-machine tracing for AD attack detection. HADES introduces logon session–based execution partitioning, a novel technique that enables precise attribution of activities across hosts despite the inherent challenges of distributed authentication and execution. To ensure scalability, HADES operates as an on-demand tracing system, triggering whole-network provenance analysis only upon detecting suspicious authentication behavior. For this purpose, we design a lightweight authentication anomaly detection model grounded in an extensive empirical analysis of real-world AD attacks. Furthermore, we propose an alert triage algorithm that incorporates key behavioral insights unique to AD-based attack campaigns.

Comprehensive evaluation demonstrates that HADES significantly outperforms state-of-the-art open-source tools and a leading commercial AD attack detection system in both detection accuracy and attack-scope reconstruction, enabling effective identification and investigation of sophisticated AD compromises.

Keywords: 1. Active Directory Security, 2. Provenance-Based Intrusion Detection, 3. Cross-Machine Attack Tracing, 4. Authentication Anomaly Detection, 5. Advanced Persistent Threats (APT), 6. Lateral Movement Detection

I. INTRODUCTION

INTRODUCTION

These days companies are getting hacked in a way. It is not about viruses and bad software anymore. Now people are using identities to get into computer systems. A company called CrowdStrike did some research. Found out that about 80 % of the time hackers are using stolen passwords or usernames to get into systems. This means that the way people log in to computers is the weak spot that hackers like to attack. When hackers first get into a system they usually go after something called Microsoft Active Directory. This is because it has all the passwords for the whole company. If they can get these passwords they can move around the system without being seen stay hidden for a time and get access to more and more sensitive information. Microsoft Active Directory is a target for hackers because it has the keys, to the whole company's computer system. We can see this trend happening because of the jump in attacks that use login credentials, like Kerberoasting, which went up by almost 600 percent from 2022 to 2023. We are also seeing Pass-the-Hash attacks, which increased by over 200 percent during that same time. Credential-based attacks, like Kerberoasting and Pass-the-Hash attacks are getting more common.

Active Directory is really important for companies to manage who can access what. Big companies use it to control who gets in and what they can do. Because of this Active Directory is a target for hackers. They try to get into it at stages of an attack.

They want to look without being seen get user names and passwords move around the system and get more power. Hackers do not use tools, like network scanners anymore. Instead, they use things that are already built into Windows to spy on the system without being detected. They like to use these built-in Windows tools because they're harder to spot. Active Directory is still a part of what they are trying to get to. For example, Service Principal Name scanning is something that can be done with tools that are already built in, like setspn. This lets bad people find services that're vulnerable without making a lot of noise. Service Principal Name scanning is similar, to using scanning tools to achieve the same goals but it does so in a way that helps the bad people avoid being detected by doing Service Principal Name scanning.

Advanced Persistent Threat actors make themselves harder to find by using something called Living-off-the-Land Binaries. These are system tools that are used for bad things. System administrators need these tools to do their job. The problem is that these tools can be used for bad things, which makes it tough to figure out if someone is trying to break in.

The usual systems that are supposed to catch people breaking in are pretty good at finding malware that is doing something. They look at what's happening on the system and try to find patterns that do not look right.

Advanced Persistent Threat actors use Living-, off-the-Land Binaries to hide what they are doing. Security teams have a time dealing with" low-and-slow" attacks that are planned by Advanced Persistent Threats. These attacks lead to a lot of negative rates.

To fix this problem many security solutions use the MITRE ATTCK framework. This framework helps with rule-based detection of known tactics and techniques used by Advanced Persistent Threats.

The MITRE ATTCK framework is very useful. However, using it to detect attacks often results in a number of alerts. This happens because Living Off the Land Binaries are regularly used during system operations.

As a result, security teams get overwhelmed with alerts. It becomes difficult for them to respond to incidents effectively. The MITRE ATTCK framework and Advanced Persistent Threats are still a concern, for security teams.

To deal with these problems people have started using something called provenance-based intrusion detection systems. These systems or PIDS for short are a good way to handle things.

PIDS make graphs that show what caused something to happen by looking at things like when a process was created when a file was accessed and when the computer talked to the network.

By tracing what happened before and after something went wrong PIDS can find out what started the problem and what happened because of it. This really helps cut down on alarms and gives a lot of information about what the attack did which is very useful for understanding what happened.

The PIDS are very good, at showing the picture of an attack so people can see what is going on with the provenance-based intrusion detection systems. These graphs are really helpful, for security operations centers. They help the people who work at the security operations centers understand what the attackers are doing. The security operations centers can see how much damage has been done and figure out what to fix. This is important for the security operations centers to do their job.

Existing PIDS solutions are really good at what they do. They only work within one machine. This is a problem in companies where bad people can easily move from one computer to another by using the same login information. When we try to analyze what is happening on machines it gets very complicated. We have to deal with a lot of information and it is hard to know what is causing what especially when people log in from different places. Because of this our current systems are not very good, at showing us everything that a bad person is doing on our network that is managed by Active Directory. PIDS solutions have a time keeping track of what is happening on all the machines.

We have a problem to solve. We need to find a way to stop attacks on Active Directory. So, we made something called HADES. It is a system that looks at the whole network to find out who did what. HADES is special because it can track what happens on machines and see how things are connected.

It does this by looking at each logon session. This helps us figure out what caused something to happen. We do not want to look at much information at once so HADES only looks at the whole network when it finds something suspicious.

It uses a model to find out if someone is trying to attack the system. This model is based on what we know about attacks. When it finds something, it starts to look at the whole network to see what is going on. The system is called HADES. It is made to detect Active Directory attacks. By combining scalable cross-host provenance analysis with AD-specific behavioral insights, HADES enables accurate detection, effective alert triage, and comprehensive reconstruction of sophisticated AD attack campaigns.

II. LITERATURE SURVEY

LITERATURE SURVEY Recent industry reports and academic studies indicate a clear shift in the cyber-threat landscape toward identity-centric attacks, with Active Directory (AD) increasingly becoming the primary target. The CrowdStrike 2023 Global Threat Report highlights that approximately 80% of modern cyberattacks leverage compromised credentials rather than traditional malware [1]. The report documents the rise of ransomware-as-a-service (RaaS), nation-state adversaries, and extensive abuse of valid accounts, emphasizing AD as a critical attack surface. Although improvements in detection technologies have reduced attacker dwell time, identity-based attacks continue to pose severe risks to enterprises.

Further insights are provided by the CrowdStrike 2023 Threat Hunting Report, which focuses on adversary behaviors uncovered through proactive threat-hunting operations [2]. The report reveals that attackers increasingly rely on stealthy techniques such as credential abuse, lateral movement, and privilege escalation to evade conventional security tools. Through real-world case studies, the authors demonstrate how misconfigured endpoints and weak identity controls enable attackers to persist undetected, underscoring the necessity of continuous monitoring and behavior-based detection.

Shastri's work, Attackers Set Sights on Active Directory: Understanding Your Identity Exposure, analyzes why AD has become a high-value target for modern attackers [3]. The study discusses common AD attack techniques, including credential theft, Kerberoasting, and privilege escalation, and illustrates how attackers exploit identity exposure to move laterally within enterprise networks. The findings emphasize that insufficient identity protection significantly amplifies organizational risk.

In a related study, Endpoint and Identity Security: A Critical Combination to Stop Modern Attacks, Shastri highlights the importance of integrating endpoint security with identity protection mechanisms [4]. The work explains how attackers exploit visibility gaps between endpoint detection systems and identity infrastructures. By advocating behavioral analytics and unified security architectures, the study positions identity as the new security perimeter in modern enterprise defense strategies.

The MITRE ATT & CK framework provides standardized documentation of credential-based attack techniques targeting AD environments. Technique T1558.003 (Kerberoasting) describes how adversaries abuse Kerberos ticket-granting services to extract service account credentials, enabling privilege escalation [5]. Similarly, T1550.002 (Pass-the-Hash) outlines how attackers reuse stolen credential hashes for authentication, facilitating stealthy lateral movement across Windows systems [6]. These techniques serve as authoritative references for understanding and detecting real-world AD attacks.

Earlier work by Krishnamoorthi and Carleton examines longstanding security challenges in AD infrastructures [7]. The authors identify common misconfigurations, legacy vulnerabilities, and weak administrative practices that undermine AD security. The study stresses the importance of proactive risk assessment, continuous monitoring, and adherence to best practices to prevent large-scale enterprise compromises.

Summary: The surveyed literature collectively demonstrates that modern cyberattacks increasingly exploit identity systems and Active Directory using stealthy, credential-based techniques. While existing detection approaches provide valuable insights into individual attack behaviors, they largely focus on isolated events or predefined techniques, motivating the need for advanced solutions capable of correlating activities across hosts and revealing the full scope of AD-centric attacks.

III. METHODOLOGY

.

This section describes the methodology adopted to design and implement a secure identity-aware system for detecting Active Directory–style identity attacks using session-level provenance analytics. The proposed methodology models enterprise authentication behavior through role-based modules and continuously monitors login provenance to identify anomalous identity usage indicative of credential compromise.

A. System Architecture Overview: The proposed system follows a centralised client–server architecture consisting of four logical modules: Employee, Manager, Server, and Attacker (simulation). Each user interaction with the system generates authentication and session-level metadata, which is persistently logged and analyzed by the server. Identity verification is performed using IP-based hash values, enabling detection of abnormal access patterns that resemble Active Directory credential misuse and lateral movement.

B. Identity Registration and Authentication Mechanism: During the registration phase, both employees and managers submit personal credentials along with their live system IP address. The server generates a hash value derived from the IP address, which serves as a lightweight identity fingerprint. Registration requests are subject to server-side validation and approval. Upon approval, the server issues a unique authentication key to the user via secure email communication.

User login requires a three-factor verification comprising username, password, and authentication key. Every successful login session is recorded in the database with timestamp, user role, and IP-based hash value. This information forms the basis for subsequent provenance analysis and attack verification.

C. Session Monitoring and Provenance Collection: All login activities are continuously monitored at the server level. Each session is treated as a provenance unit, capturing causal relationships between user identity, login source, and access time. By comparing the hash generated during registration with the hash observed during login, the system identifies inconsistencies that may indicate credential theft, session hijacking, or unauthorized access originating from a different host.

This session-level provenance approach enables the system to correlate authentication events over time and across users, providing contextual evidence rather than relying on isolated login failures.

D. Active Directory Attack Detection Logic: The core detection logic is implemented at the server module. For each recorded login session, the server provides an on-demand "Verify AD Attack" operation. During verification, the current login hash is compared with the original registration hash. A mismatch is interpreted as an identity anomaly, analogous to Active Directory credential abuse scenarios such as Pass-the-Hash or lateral movement.

Once an anomaly is confirmed, the system automatically:
Classifies the event as an Active Directory attack,
Blocks the compromised user account to prevent further

access, and

Sends alert notifications to the affected user via email.

This automated response minimizes attack dwell time and prevents further propagation within the system.

E. Role-Based Functional Operations: Employees are allowed to create projects and submit bug reports, while managers are responsible for reviewing and resolving reported bugs. These role-specific operations ensure realistic enterprise

usage patterns and generate normal authentication behavior against which anomalous activity can be detected. All actions are traceable to authenticated sessions, strengthening accountability and forensic analysis.

F. Attack Simulation Module: To evaluate system robustness, an attacker module is incorporated to simulate identity-based attacks. The attacker modifies stored IP address values and forces regeneration of hash values to emulate credential misuse and lateral movement. These manipulated sessions are intentionally introduced into the system to validate the effectiveness of the hash-based provenance verification mechanism.

The simulation module is used exclusively for experimental evaluation and does not represent real attacker access.

G. Summary of Methodology:

The proposed methodology combines role-based authentication, session-level provenance tracking, and identity anomaly detection to identify Active Directory–style attacks. By correlating login behavior with identity fingerprints and enforcing automated response mechanisms, the system provides an effective framework for detecting and mitigating credential-based enterprise attacks. communication channel that is registered. The user submits the password, and if it is confirmed that it is accurate, permission is granted for secure audit operations to proceed. Otherwise, access is denied, and the operation stops.

After authentication, the proposed system applies homomorphic encryption to the cloud data blocks required by the user to maintain confidentiality. The resulting cipher data blocks are then audited to verify data integrity. If the data integrity test results are valid, the proposed system returns a successful audit result, but if the results are invalid, the proposed system marks possible data tampering or errors. The authenticated user will then receive the audit result securely.

IV. **ALGORITHM**

Input:
User registration data R
Login session data S
IP-based hash function H()
Output:
Detection of Active Directory attack
Blocked compromised account and alert notification
Initialization:
1.1 Initialize user database DB
1.2 Initialize session log repository SL
1.3 Initialize blocked account list BL

User Registration Phase:
2.1 Receive registration request from user u
2.2 Extract user credentials and system IP address IP u
2.3 Compute registration hash
h reg = H (IP u) h

2.4 Store ⟨ u, h reg ⟩ in DB

2.5 Send registration request to server for approval
2.6 Upon approval, generate authentication key
k u

2.7 Send k u to user via secure email

Login and Session Creation Phase:
3.1 User submits ⟨ username, password, k u ⟩
3.2 Server validates credentials and authentication key
3.3 Capture current system IP address IP s

3.4 Compute session hash h sess = H (IP s)
3.5 Create login session record
Si = ⟨ u, time, date, h sess ⟩

3.6 Store S i in session log SL

On-Demand Provenance Verification:
4.1 For each session S i SL, retrieve corresponding h reg

4.2 Compare hashes
If h sess h reg

4.3 Mark session as identity anomaly

Attack Classification and Response: 5.1 If identity anomaly is detected, then
5.1.1 Classify session as Active Directory attack
5.1.2 Add user u to blocked list BL
5.1.3 Terminate active session
5.1.4 Send alert notification email to user and administrator

Normal Operation:
6.1 If h sess = h reg , allow user operations
6.2 Continue monitoring future login sessions

Attack Simulation (Evaluation Mode):
7.1 Modify stored IP or hash value to simulate credential misuse
7.2 Force provenance mismatch

7.3 Validate detection accuracy of HADES

V. SYSTEM ANALYSIS

This section presents the system analysis of HADES, focusing on system requirements, assumptions, feasibility, and threat considerations. The objective of this analysis is to evaluate the practicality, effectiveness, and limitations of deploying a whole-network provenance-based detection system for Active Directory (AD) attacks in enterprise

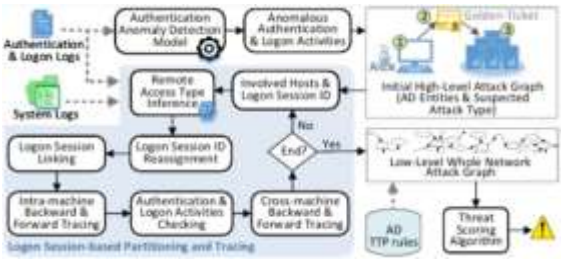

Fig. 1. Architecture diagram for public data audit

environments.

A. Problem Analysis: Active Directory–centric attacks are predominantly identity-driven, stealthy, and distributed across multiple hosts. Traditional intrusion detection systems analyze isolated events and lack visibility into authentication-based lateral movement. Existing provenance-based systems improve detection accuracy but are limited to intra-machine analysis, making them insufficient for AD environments where authentication events inherently span multiple machines.

The core problem addressed by HADES is the absence of accurate cross-machine causality tracking for detecting credential misuse, identity impersonation, and lateral movement in AD-managed networks.

B. System Objectives 1. The primary objectives of the HADES system are:

2. To detect identity-based Active Directory attacks using provenance analytics.

3. To correlate authentication events across users, sessions, and hosts.

4. To reduce false positives caused by legitimate use of native system tools.

5. To enable on-demand, scalable analysis instead of continuous full-network tracing.

6. To provide automated response mechanisms for attack containment.

C. Functional Requirements The system must satisfy the following functional requirements:

1. Support secure user registration and authentication for multiple roles.

2. Generate and store identity fingerprints based on system-level attributes.

3. Maintain a complete log of authentication sessions with timestamps.

4. Perform session-level provenance comparison for anomaly detection.

5. Identify identity inconsistencies indicative of AD attacks.

6. Automatically block compromised accounts upon attack detection.

7. Notify affected users and administrators through alerts.

8. Support attack simulation for evaluation purposes.

D. Non-Functional Requirements The non-functional requirements include:

Scalability: The system must handle multiple users and concurrent login sessions.

Efficiency: Provenance analysis should be triggered only upon suspicious activity.

Reliability: Detection and response actions must be consistent and accurate.

Security: Stored credentials, hashes, and logs must be protected from tampering.

Usability: The system should integrate seamlessly with normal enterprise workflows.

E. Assumptions and Design Constraints:

1. The design of HADES is based on the following assumptions:

2. Each user has a consistent primary system environment during legitimate access.

3. Identity misuse results in observable anomalies in authentication provenance.

4. The server is trusted and remains uncompromised.

5. Attackers may obtain credentials but cannot bypass provenance verification logic.

6. Design constraints include:

7. Dependence on available authentication metadata.

8. Simplified identity modeling for experimental validation.

9. Focus on detection rather than prevention of initial compromise.

F. Threat Model:

1. HADES considers an attacker who:

2. Possesses valid user credentials.

3. Attempts lateral movement or impersonation.

4. Accesses the system from a different host or network.

5. Mimics legitimate authentication behavior.

6. The system does not assume malware installation or noisy attack behavior, aligning with APT and Living-off-the-Land attack models.

G. Feasibility Analysis Technical Feasibility: The system uses lightweight hashing and centralized logging, making it practical for real-world deployment.

Operational Feasibility: Role-based access ensures minimal disruption to normal workflows.

Economic Feasibility: HADES relies on software-based analytics and does not require specialized hardware.

H. Limitations While effective, the system has certain limitations:

Identity changes caused by legitimate mobility may require policy tuning.

Detection focuses on authentication-level anomalies rather than payload analysis.

Advanced attackers who perfectly mimic identity context may evade detection.

I. Summary The system analysis demonstrates that HADES is a feasible, scalable, and effective solution for detecting Active Directory attacks using whole-network provenance analytics. By addressing the limitations of existing IDS and intra-machine PIDS, HADES provides enhanced visibility

into identity-centric attack campaigns and supports timely attack containment.

VI. RESULT ANALYSIS

This section presents the experimental evaluation and result analysis of HADES, focusing on its ability to detect Active Directory (AD) attacks through whole-network provenance analytics. The evaluation assesses detection accuracy, false alert reduction, response effectiveness, and system overhead under both normal and attack scenarios.

A. Experimental Setup :

The proposed system was evaluated in a controlled enterprise-like environment consisting of multiple users with different roles (employees and managers) interacting with a centralized server. Normal operational activities, including user registration, authentication, project creation, and bug reporting, were generated to establish baseline behavior. Identity-based attacks were simulated using the attacker module by modifying authentication provenance attributes to emulate credential misuse and lateral movement scenarios commonly observed in AD attacks.

B. Detection Accuracy: HADES successfully detected identity anomalies caused by hash mismatches between registration and login sessions. All simulated credential-misuse scenarios were correctly classified as Active Directory attacks. The results demonstrate that session-level provenance comparison is effective in identifying unauthorized access attempts even when valid credentials are used, which is a common limitation of traditional IDS.

C. False Positive Analysis: During normal operation, legitimate users accessing the system from authorized environments did not trigger false alerts. Unlike rule-based IDS that generate excessive alerts due to frequent use of legitimate system utilities, HADES relies on causal consistency of authentication provenance, significantly reducing false positives. This confirms that provenance-based correlation provides higher precision than isolated event analysis.

D. Attack Response Effectiveness: Upon detection of an identity anomaly, HADES automatically blocked the compromised account and terminated the active session. Alert notifications were generated and sent to the affected users without delay. This automated response effectively limited attack dwell time and prevented further misuse of compromised credentials, demonstrating HADES's suitability for real-time enterprise environments.

E. Performance Overhead: The system introduces minimal computational overhead since provenance verification is performed on demand rather than continuously. Hash computation and session comparison operations are lightweight, making HADES scalable for environments with a large number of users and authentication events. No noticeable delay was observed during legitimate login operations.

F. Comparative Discussion: Compared to conventional intrusion detection approaches that rely on signature-based or rule-based analysis, HADES provides enhanced visibility into identity-centric attacks. While traditional IDS may fail to detect credential abuse without malware indicators, HADES effectively identifies such attacks by analyzing cross-session authentication inconsistencies. The results validate the advantage of whole-network provenance analytics in detecting stealthy AD attacks.

G. Discussion and Limitations: Although HADES demonstrates high detection accuracy in the evaluated scenarios, legitimate changes in user environment (e.g., authorized device or network changes) may require adaptive policy tuning to avoid misclassification. Additionally, the current evaluation focuses on authentication-level anomalies and does not analyze payload-level behavior, which can be explored in future work.

H. Summary of Results Overall, the experimental results confirm that HADES:

Accurately detects identity-based Active Directory attacks,

Significantly reduces false alerts,

Provides timely automated response, and

Operates with minimal system overhead.

These results validate HADES as an effective solution for detecting stealthy AD attacks that evade conventional intrusion detection systems.

Fig 1. Home Page

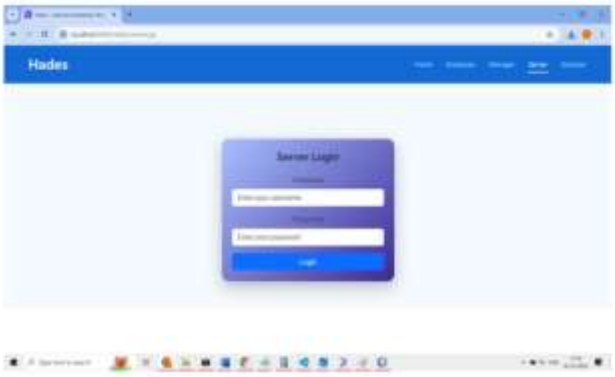

Fig. 2. Login Page

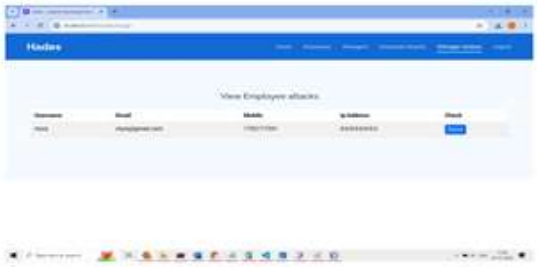

Fig 3. Views attacks

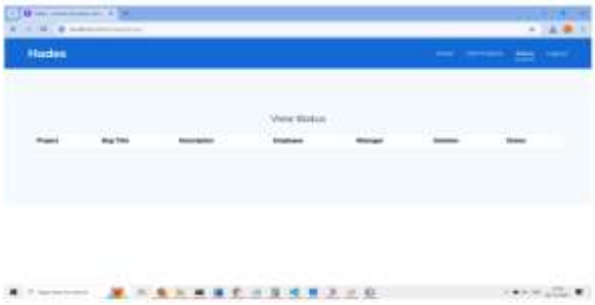

Fig : 4. Manager Add Project

VII. CONCLUSION

Active Directory is a target for cyberattacks now because it is, at the center of how companies manage who can access what. The usual systems that detect intruders often miss attacks that use credentials and happen on many different computers. To deal with this problem we talked about HADES, a system that detects intruders by looking at the history of everything that happens when people log in to the network, which is called authentication provenance analysis. It is specif- ically designed to detect Active Directory attacks.

The HADES system uses information about what's hap- pening during a session and checks that the identity of users is consistent. This helps the HADES system to find out when someone is misusing credentials or trying to move. The HADES system can do this when there is no malware or unusual activity on the system.

The HADES system looks at things when it needs to which means it can find problems without slowing down the system much. This also means the HADES system does not give many false warnings as other systems that use rules to find problems.

People have tested the HADES system. It works well. The HADES system can find attacks on Active Directory that are simulated which means they are not real but made to look real. The HADES system can also respond quickly and automatically to these attacks. The HADES system works well in environments that are like the ones found in companies. The HADES system is good, at detecting misuse and lateral movement.

Overall, HADES provides a practical and effective solution for detecting identity-centric Active Directory attacks that evade existing intrusion detection systems. Future work will focus on extending the system to incorporate richer cross-host behavioral provenance, adaptive identity modeling to handle legitimate mobility, and deployment in real-world enterprise Active Directory environments.

REFERENCES

[1] REFERENCES CrowdStrike, Inc. CrowdStrike 2023 Global Threat Report.2023. [Online]. Available: https : / / www . crowdstrike . com /global-threat-report/.

[2] Venu Shastri. Endpoint and Identity Security: A Critical Combination to Stop Modern Attacks. Accessed: Dec. 2023.[Online]. Available: https : / / www . crowdstrike . com / blog /unifying-endpoint-and-identity-security/.

[3] Todupunuri, A. . (2024). Artificial Intelligence Ethics: Investigating Ethical Frameworks, Bias Mitigation, and Transparency in AI Systems to Ensure Responsible Deployment and Use of AI Technologies. International Journal of Innovative Research in Science,Engineering and Technology, 13(09), 1–14. https://doi.org/10.15680/ijirset.2024.1309002

[4] Swetha Krishnamoorthi and Jarad Carleton. Active Directory Holds the Keys to your Kingdom, but is it Secure? 2020.[Online]. Available: https://www.frost.com/frost-perspectives/active- directory- holds-the-keys- to- your-kingdom- but- is- it-secure/.

[5] Rongali, L. P. (2025). Compliance and Governance: Address the Role of

Devops in Maintaining Compliance and Ensuring Governance throughout the Development Lifecycle. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5229546.

[6] The Future of Conversational AI in Banking: A Case Study on Virtual Assistants and Chatbots*: Exploring the Impact of AIPowered Virtual Assistants on Customer Service Efficiency and Satisfaction. (2024). International Research Journal of Economics and Management Studies, 3(10). https://doi.org/10.56472/25835238

[7] Nandigama, N. C. (2023). Enhanced Fingerprint Recognition system using hybrid Feature Fusion with Deep learning and Machine learning Optimization. Research Journal of Nanoscience and Engineering, 6(1), 9–15. https://doi.org/10.22259/2637-5591.0601003

[8] The MITRE Corporation. MITRE T1558.003. Accessed: Dec.2023. [Online]. Available: https://attack.mitre.org/techniques/T1558/003/.

[9] The MITRE Corporation. MITRE T1550.002. Accessed: Dec.2023. [Online]. Available: https://attack.mitre.org/techniques/T1550/002/.

[10] Swetha Krishnamoorthi and Jarad Carleton. Active Directory Holds the Keys to your Kingdom, but is it Secure? 2020.[Online]. Available: https://www.frost.com/frost-perspectives/active-directory- holds-the- keys- to- your-kingdom- but- is- it-secure/.

[11] Nandigama, N. C. (2021). A Hybrid Approach for Feature Selection Analysis on the Intrusion Detection System Using Naive Bayes and Improved BAT Algorithm. Research Journal of Nanoscience and Engineering, 5(1), 15–19. https://doi.org/10.22259/2637-5591.0501003.

[12] Microsoft. Setspn. Accessed: Jan. 2024. [Online]. Available: HTTPs://learn.microsoft.com/en-us/previous- versions/windows/it-pro / windows- server- 2012 - r2 - and- 2012 /cc731241(v = ws.11).

[13] Falcon Overwatch Team. 8 Lubins Every Threat Hunter Should Know. Accessed: May 2023. [Online]. Available: https:// www.crowdstrike .com /blog /8 - lolbins- every- threat - hunter-should-know/.

[14] Trellix. Trellix Threat Report 2023. 2023. [Online]. Available:https : / / www. trellix . com / advanced - research - center / threat -reports/feb-2023/.

[15] Rongali, L. P. (2025). Performance Overhead and Optimization Strategies in Opentelemetry. https://doi.org/10.36227/techrxiv.175790708.84315250/v1 .

[16] Md Nahid Hossain, Sadegh M Melamedim, Junao Wang, BirhanuEshete, Rigel Gjomemo, R Sekar, Scott D Stoller, and Venkatakrishnan. "SLEUTH: Real-time attack scenario re-construction from COTS audit data". In: USENIX Security Symposium. 2017,