DEFENSE IN DEPTH FOR UNIVERSITY CAMPUS NETWORKS: COMPARATIVE ANALYSIS OF SECURITY ARCHITECTURES AND PRACTICAL IMPLEMENTATION.

Muad Errafik and Haris Islamovic
V. V. Popovskyy Department of Infocommunication Engineering,
Kharkiv National University of Radio Electronics,
Ukraine
Department of Engineering,
International University of Sarajevo,
Bosnia and Herzegovina

E-mail: muad.errafik@nure.ua ,

harisislamovic@student.ius.edu.ba

Abstract

This study presents a comparative analysis of several security architectures that are relevant to university environments, including Defense in Depth (DiD), Zero Trust Architecture (ZTA), Secure Access Service Edge (SASE), and Security Information and Event Management (SIEM). Building on this analytical foundation, the study develops and evaluates a secure university campus network model using the GNS3 simulation environment. The network is designed with pfSense security appliances and Cisco IOU switching and is organised into several functional Virtual Local Area Networks (VLANs). Redundancy is provided through the Link Aggregation Control Protocol (LACP) and the Rapid Per-VLAN Spanning Tree Protocol (Rapid PVST). A baseline assessment is performed using Nmap, Nikto, Hydra, and hping3 to identify weaknesses in the initial configuration. The results show that the network is vulnerable to reconnaissance, Secure Shell (SSH) brute-force attacks, and flood-based Denial of Service (DoS) conditions. These issues are addressed through a DiD approach that includes inter-VLAN Access Control Lists (ACLs), Layer 2 security controls, and a Suricata-based Intrusion Detection and Prevention System (IDS/IPS) with custom signatures. Follow-up testing confirms that all previously successful attacks are blocked. Performance measurements using iperf3 show only a small difference of approximately 4.5 Mb/s when Deep Packet Inspection (DPI) is active. The findings indicate that a carefully designed DiD architecture can offer effective protection for university campus networks while maintaining acceptable performance.

Keywords: Campus Network Design, Network Security Architecture, Vulnerability Assessment, Défense in Depth, Intrusion Detection System, GNS3 Simulation.

University campus networks support teaching, research, administration, student services, and public online resources. They must also accommodate a wide range of unmanaged devices, including personal laptops, phones, tablets, laboratory computers, and Internet of Things (IoT) systems, often under Bring Your Own Device (BYOD) policies. This level of openness increases exposure to cybersecurity threats and makes network protection more challenging than in tightly controlled corporate environments. Prior work has shown that universities are frequent targets of attacks such as phishing, data breaches, and unauthorized access, often due to weak segmentation or misconfigured services [1-3].

The literature highlights the importance of combining governance frameworks with technical measures such as segmentation, strong authentication, and monitoring [4, 5]. However, universities face additional barriers, including decentralised decision making and varied technical expertise across departments [2, 3]. Considering these challenges, the paper pursues two primary objectives. The first is to examine and compare several widely adopted security architectures, assessing their applicability and effectiveness within university

campus environments. The second is to design and evaluate a DiD-based network architecture within a simulated setting that replicates realistic campus conditions.

Several security models are commonly referenced in the cybersecurity domain. DiD organises security controls into several layers that work together. Typical layers include perimeter firewalls, VLAN segmentation, Layer 2 protections, host-based firewalls, and network intrusion detection. The idea is that if one layer is bypassed, others may still prevent or reduce the impact of an attack. The main challenge lies in coordinating and maintaining all layers over time, especially in large networks [4–6].

ZTA removes the assumption that users or devices inside the network can be trusted by default. Instead, access is granted only after verification and is limited to what is strictly necessary. ZTA often uses microsegmentation and detailed access policies to reduce lateral movement. This architectural model is formally defined in NIST SP 800-207, which emphasises continuous verification, least privilege, and strong identity-based controls [7, 8].

SASE combines wide area networking with cloud-based security services, such as secure web gateways and cloud firewalls. It is designed to support distributed users and heavy use of cloud applications. Recent studies highlight that SASE offers scalability and unified policy enforcement but also requires stable cloud connectivity and strong identity integration [9, 10].

SIEM systems focus on monitoring. They collect and correlate event logs from firewalls, servers, endpoints, and applications to detect suspicious activity and support forensic analysis and compliance work. Modern SIEM platforms rely heavily on big-data analytics, centralised log ingestion, and automated rule correlation to detect anomalies in large-scale environments [11].

Each of these approaches has strengths and limitations. DiD is relatively straightforward to apply to existing campus topologies and can be implemented using open-source tools and standard equipment. Additional research on DiD shows that its layered structure remains highly suitable for segmented or legacy campus deployments because it improves resilience without requiring full architectural redesign [6, 12]. ZTA and SASE promise finer-grained control but require strong identity management and, in the case of SASE, dependence on cloud infrastructure. SIEM improves visibility but requires expertise and operational resources. Network segmentation and Layer 2 hardening, as recommended in enterprise and campus network guidelines, play a critical role in reducing lateral movement and attack propagation [4, 5]. Table 1 summarises the main characteristics of DiD, ZTA, SASE, and SIEM.

Table 1. Comparative Analysis of Security Frameworks

Feature	Defense in Depth	Zero Trust	SASE	SIEM
Primary Focus	Layered Redundancy	Identity and Verification	Cloud Convergence	Visibility and Logging
Trust Model	Trust Inside Perimeter	Never Trust, Always Verify	Trust via Context	N/A (Monitoring)
Scalability	Low (Hardware heavy)	High (Software defined)	Very High (Cloud Native)	Medium (Data Volume)
Implementation	Moderate Complexity	High Complexity	Moderate Complexity	High Complexity
Best For	Legacy Infrastructures	Hybrid/Remote Workforces	Distributed Enterprises	Compliance/SOCs

From Table 1, it can be observed that DiD aligns naturally with environments that already rely on traditional VLAN-based segmentation and hardware firewalls, such as many university campuses [4–6]. ZTA and SASE offer stronger identity-centric and cloud-integrated security postures, but their full adoption typically requires substantial changes in identity management, policy orchestration, and network architecture [7–10]. For universities with heterogeneous legacy systems, limited budgets, and decentralised administration, a staged approach is often more realistic: starting from DiD and improved segmentation, adding SIEM-style monitoring for visibility, and only then progressively incorporating ZTA or SASE concepts in high-risk areas [1–3], [11]. This layered view supports the choice made in this work to implement DiD in practice, while still situating it within a broader landscape of more advanced architectures.

To evaluate the applicability of the previously discussed security architectures within a realistic environment, a secure university campus network based on DiD approach was designed and implemented in GNS3. The design follows a hierarchical campus topology.

A pfSense firewall is positioned at the boundary between the external and internal networks. Inside the campus, the network is segmented into several VLANs that correspond to typical functional areas:

- Servers VLAN (DMZ, VLAN 10) for public-facing services.
- Labs VLAN (VLAN 100) for student laboratory devices.
- Library VLAN (VLAN 120) for library systems.
- Offices VLAN (VLAN 140) for academic and administrative staff.
- IT VLAN (VLAN 160) for IT personnel and network management.
- Additional management VLANs for device administration.

The switching infrastructure uses Cisco IOU Layer 2 devices arranged into core, distribution, and access layers. Redundant links between switches are provided through LACP, while Rapid PVST is configured to prevent loops and enable fast convergence in the event of link or device failures. The resulting VLAN-based segmentation and controlled inter-VLAN routing align with recommended campus network security practices.

The testbed includes two Ubuntu virtual machines. One hosts a web server (for example, nginx) running on HTTP and HTTPS, and the other hosts an SMTP mail server. A Kali Linux virtual machine is connected to the external interface of the pfSense firewall and is used to emulate an external attacker. End-user hosts located within different VLANs are represented by lightweight virtual PCs in GNS3.

A baseline vulnerability assessment is conducted to determine how the initial configuration responds to common reconnaissance, exploitation, and denial-of-service activities before any security hardening is applied. The tools used during this baseline assessment, along with their explanations and intended purpose, are summarised in Table 2.

Table 2. Vulnerability Assessment Tools

Tool	Explanation	Use
Nmap	A network scanning tool that discovers hosts, open ports, services, versions, and OS details using various scan types.	Reconnaissance and vulnerability assessment; mapping networks, identifying open ports/services for potential exploitation.
Nikto	A web server scanner that tests for vulnerabilities, misconfigurations and outdated software.	Web application security testing; detecting issues like server headers, default pages, or known exploits on web servers.
Hydra	A parallelized brute-force tool that cracks passwords/logins by trying combinations on many protocols.	Credential cracking and unauthorized access testing; simulating brute-force attacks to test password strength.
hping3	A packet crafting tool that generates custom TCP/UDP/ICMP packets, including floods, with control over flags, size, and rate.	Firewall testing, DoS simulation, and packet manipulation; creating floods or spoofed traffic to test network resilience.

The following tools are applied:

- Nmap to identify live hosts, open ports, and running services.
- Nikto to assess the web server for misconfigurations and known vulnerabilities.
- Hydra to perform SSH brute-force attacks using a dictionary.
- hping3 to generate a SYN flood and evaluate service resilience under DoS conditions.

These tools are widely used in network security experiments and provide a realistic representation of attacker behaviour. The baseline results show that the network is vulnerable in several ways. Nmap reveals exposed services, including HTTP on the DMZ server.

Nikto identifies configuration weaknesses on the web server. Hydra successfully obtains valid SSH credentials through brute forcing. The SYN flood generated by hping3 causes the web service to become unreachable for legitimate users.

To strengthen detection and prevention capabilities, custom Suricata rules were created for the two attack categories not detected by default rule sets: brute-force SSH attempts and SYN flood behaviour. An example of the implemented rules is shown below:

```
SSH brute-force detection
drop tcp any any -> $HOME_NET 22 \
(msg:"SSH Brute Force Attempt"; \
threshold:type threshold, track by_src, count 3, seconds 1; \
sid:1000001; rev:2;)

DoS SYN flood detection
drop tcp any any -> $HOME_NET [80,443] \
(msg:"DoS Flood on Web Ports"; flags:S; \
threshold:type threshold, track by_src, count 10, seconds 1; \
sid:1000005; rev:1;)
```

Table 3 summarises the initial outcomes and the improvements achieved following the deployment of the DiD security controls.

Table 3. Security Effectiveness Results

Attack Vector	Tool Used	Initial Outcome	Post-Implementation Outcome	Defense Mechanism
Reconnaissance	Nmap	Open ports exposed	Blocked (Scan Detected)	Suricata (ET Open Rules)
Web Vulnerability	Nikto	Misconfig found	Blocked (Scan Detected)	Suricata (ET Open Rules)
Credential Attack	Hydra	Password Cracked	Blocked (Connection Dropped)	Suricata (Custom Rule)
DoS Flood	hping3	Service Offline	Mitigated (Service Online)	Suricata (Custom Rule)

Finally, to verify that the implemented security measures did not introduce unacceptable performance penalties, quantitative bandwidth testing was carried out using iperf3. Baseline throughput was compared against throughput under active DPI with Suricata operating in IDS/IPS mode. The results showed an approximate reduction of only 4.5 Mb/s, indicating a marginal processing overhead. This confirms that the proposed architecture achieves an effective balance between robust security hardening and the performance requirements typically expected in university campus network environments.

The results obtained in this study demonstrate that a DiD-oriented design, when combined with VLAN-based segmentation, Layer 2 protections, and an inline IDPS, can significantly reduce the success of common attack vectors in university environments. The experimental outcomes are consistent with prior work on campus network security, which emphasises the importance of segmentation, host hardening, and continuous monitoring to contain breaches and limit their impact [1–3], [5, 6, 11].

In particular, the effective blocking of reconnaissance, web misconfiguration exploitation, credential brute-force attacks, and SYN-based DoS traffic illustrates how multiple defensive layers complement each other rather than acting in isolation.

At the same time, several limitations should be acknowledged. First, the evaluation was performed in a virtual GNS3 environment with a limited number of hosts and controlled traffic patterns. Real university networks are far larger, more heterogeneous, and subject to highly variable workloads. As a result, performance overheads and operational complexity may grow when scaling this approach to production deployments.

Second, the attacks considered were representative but not exhaustive; more advanced threats, such as encrypted command-and-control channels, application-layer evasion techniques, or insider attacks, would require additional detection mechanisms, including more sophisticated rule sets, anomaly-based IDS techniques, or integration with SIEM platforms [11, 12].

Third, while ZTA and SASE were analysed conceptually, they were not implemented in the testbed. Future work could investigate hybrid models in which DiD-style segmentation is combined with ZTA policies and cloud-delivered SASE services to support remote users and multi-cloud workloads [7–10].

Overall, the study provides a structured and empirically grounded reference design that can be adapted by universities seeking to strengthen their network security posture incrementally. The proposed GNS3-based testbed also offers a reusable platform for teaching and further research on campus network security, enabling students and practitioners to experiment with different architectures, tools, and attack scenarios before making changes in production environments.

References

1. Ulven, J. B., Wangen, G. (2021). A Systematic Review of Cybersecurity Risks in Higher Education. Future Internet, Vol. 13, No. 2, Article 39, pp. 1-20. DOI: https://doi.org/10.3390/fi13020039

2. Zheng, R., Ma, H., Wang, Q., Fu, J., & Jiang, Z. (2021). Assessing the Security of Campus Networks: The Case of Seven Universities. Sensors, 21(1), 306. DOI: https://doi.org/10.3390/s21010306

3. Li, J., Xiao, W., & Zhang, C. (2023). Data security crisis in universities: Identification of key factors affecting data breach incidents. Humanities and Social Sciences Communications, 10(1), pp. 1-18. DOI: https://doi.org/10.1057/s41599-023-01757-0

4. Álvarez, D., Nuño, P., González, C. T., Bulnes, F. G., Granda, J. C., García-Carrillo, D. (2023). Performance Analysis of Software Defined Networks to Mitigate Private VLAN Attacks. Sensors, Vol. 23, No. 4, Article 1747. DOI: https://doi.org/10.3390/s23041747

5. Pilamunga, N., Mantilla, C., Arellano, A. et al. (2018). Security Policies to Mitigate Attacks on VLAN Hopping in the Data Link Layer of Local Area Networks. KnE Engineering, Vol. 3, No. 9, pp. 111-119. DOI: https://doi.org/10.18502/keg.v3i9.3649

6. Alsaqour, R., Majrashi, A., Alreedi, M., Alomar, K., & Abdelhaq, M. (2021). Defense in Depth: Multilayer of security. International Journal of Communication Networks and Information Security, 13(2), pp. 242-248. DOI: https://doi.org/10.17762/ijcnis.v13i2.4951

7. Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero Trust Architecture. NIST SP 800-207. DOI: https://doi.org/10.6028/NIST.SP.800-207

8. Ghasemshirazi, S., Shirvani, G., & Alipour, M. A. (2023). Zero trust: Applications, challenges, and opportunities. DOI: https://doi.org/10.48550/arXiv.2309.03582

9. Tanguturi, R., & Bhimini, S. (2023). The Future of Networking is Here: SASE for a Stronger, More Secure Network. Journal of Network & Information Security, 11(1). URL: http://www.publishingindia.com/jnis

10.Bashi, Z. S. M. A., Basri, A. B., & Senan, S. (2025). Unified Secure Access Service Edge (SASE): Transforming Security for Hybrid Workforce and Multi-Cloud Environments. International Journal on Perceptive and Cognitive Computing, 11(2), pp. 1-7. DOI: https://doi.org/10.31436/ijpcc.v11i2.528

11.López Velásquez, J. M., Martínez Monterrubio, S. M., Sánchez Crespo, L. E., & Garcia Rosado, D. (2023). Systematic review of SIEM technology: SIEM-SC birth. International Journal of Information Security, 22(3), pp. 691-711. DOI: https://doi.org/10.1007/s10207-022-00657-9

12. Abdelghani, T. (2019). Implementation of defense in depth strategy to secure industrial control system in critical infrastructures. American Journal of Artificial Intelligence, 3(2), pp. 17-22. DOI: https://doi.org/10.11648/j.ajai.20190302.11