SecureNet Enterprise Network: Layered Defense Architecture in a Virtualized Enterprise Environment

1. Executive Summary

This project presents the design, implementation, and evaluation of a layered security architecture deployed within a virtualized enterprise network environment named SecureNet. The solution emphasizes network segmentation, controlled access boundaries, encrypted service configurations, centralized monitoring, and automated patch management. By simulating organizational departments and operational roles, the environment provides a realistic platform for assessing how coordinated security controls reduce attack surface and improve detectability of malicious behavior.

The deployment combines pfSense for network routing and segmentation, Ubuntu-based servers in the DMZ for public-facing services, domain-managed Windows systems for enterprise identity and update distribution, and Splunk as the centralized SIEM. Security effectiveness was validated through adversarial simulations performed from an external attacker network, vulnerability scanning using Nessus, service misconfiguration testing, and monitoring of system behavior through Splunk dashboards and log correlation. Cost-benefit and ROI recalculations based on IAS1 estimates show significant financial efficiency due to the use of open-source tooling and automation, while simultaneously achieving higher risk reduction than originally projected.

The results indicate that the SecureNet architecture successfully achieves Defense-in-Depth (DiD) objectives at low cost, while maintaining operational practicality and demonstrating enterprise-aligned security readiness.

---

2. Alignment to IAS2 Course Outcomes

IAS2 Outcome	Implementation Evidence	Verification
Implement security controls from IAS1	Network segmentation using pfSense VLANs; TLS 1.3 hardening on Nginx and Postfix; SSH restricted to IT subnet; Patch management for Linux (Ansible) and Windows (WSUS).	pfSense Rules DMZ Configs Ansible & WSUS logs
Refine and present system diagram reflecting implemented controls	Updated layered network diagram showing trust boundaries and segmented subnets: WAN Attacker, DMZ, IT Dept, Sales Dept, and SIEM network.	Network Diagram
Monitor and evaluate operational effectiveness of controls	Splunk receiving logs from pfSense (syslog), DMZ server (auth.log, vsftpd, mail logs), IT Linux (syslog + auth), Sales Win Server and sales10 Client (WinEventLog). Queries used to confirm detection and response readiness.	Splunk Logs

IAS2 Outcome	Implementation Evidence	Verification
Validate cost-benefit effectiveness	IAS1 CBA vs IAS2 Actual comparison performed; Open-source security stack resulted in near-zero licensing cost; Patch automation reduced long-term operational labor.	Check CBA ROI Sheet

---

3. Project Overview & Architecture

3.1 Background and Rationale

SecureNet was developed as a controlled environment to simulate a realistic organizational network that must maintain secure access boundaries between public-facing systems and internal departmental networks. The design reflects common enterprise network requirements: service availability in the DMZ, protected internal business operations, centralized identity and authentication governance, and unified monitoring visibility. The virtualized approach using VirtualBox enables iterative refinement of configurations, controlled testing, and safe execution of security assessments.

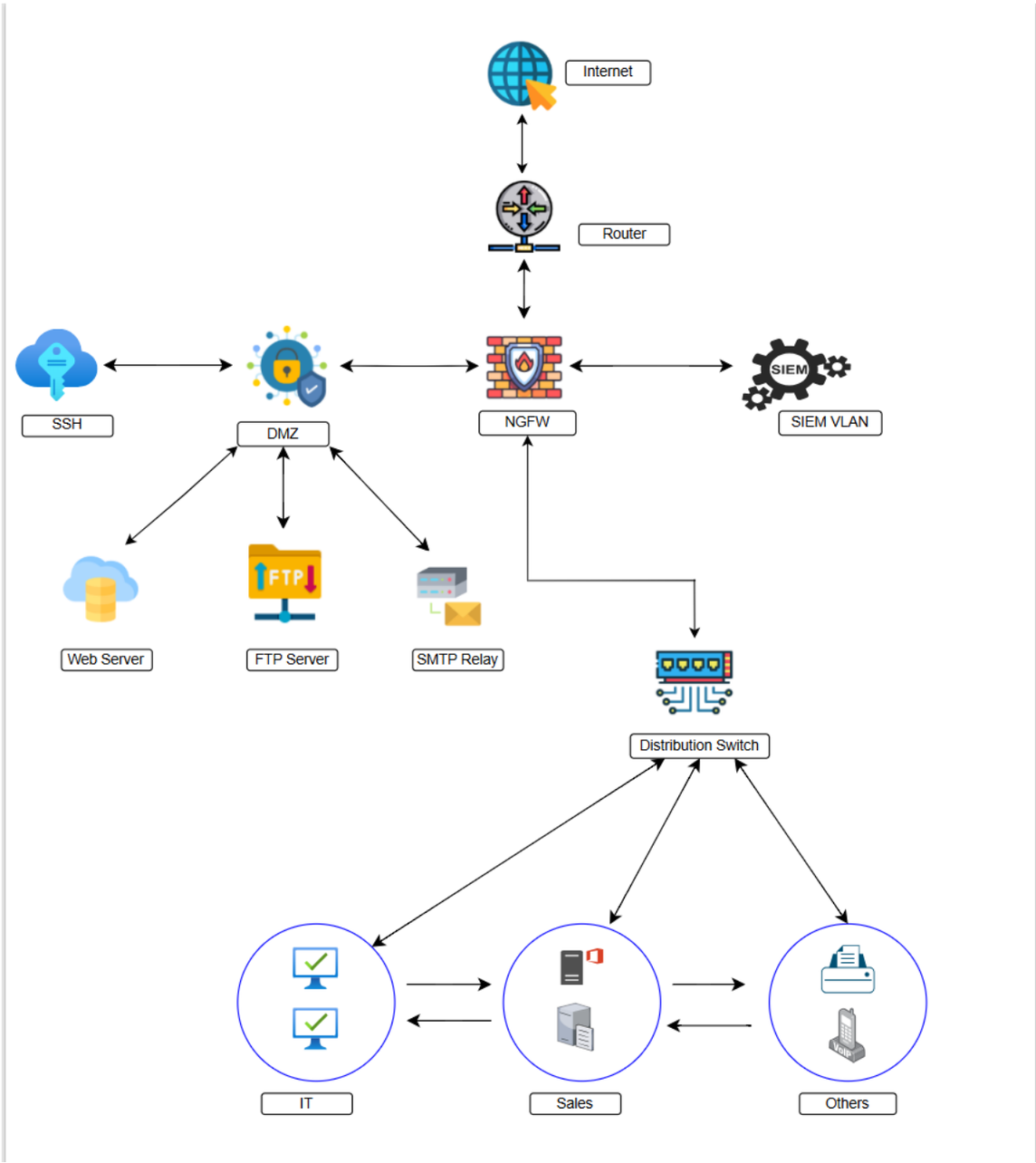
The project’s layered architecture aligns with key assurance domains: (1) network segmentation, (2) application and host hardening, (3) Linux patch automation, (4) Windows WSUS patching, (5) centralized SIEM detection, (6) administrative control through centralized management, (7) cryptographic hardening, and (8) incident detection and response readiness.

3.2 Objectives

- Implement a segmented and controlled enterprise network using pfSense as the central routing and firewall platform.
- Deploy web, mail, remote access, file transfer, and authentication services in a DMZ and internal networks.
- Apply hardening configurations to reduce exposure and enforce encryption and controlled access.
- Integrate Splunk SIEM for centralized log collection, event correlation, and monitoring.
- Conduct vulnerability and adversarial testing to validate security controls.
- Assess the operational and financial impact of the implemented controls through CBA and ROI analysis.

3.3 Network Architecture

Network Diagram



4. Asset & Data Inventory

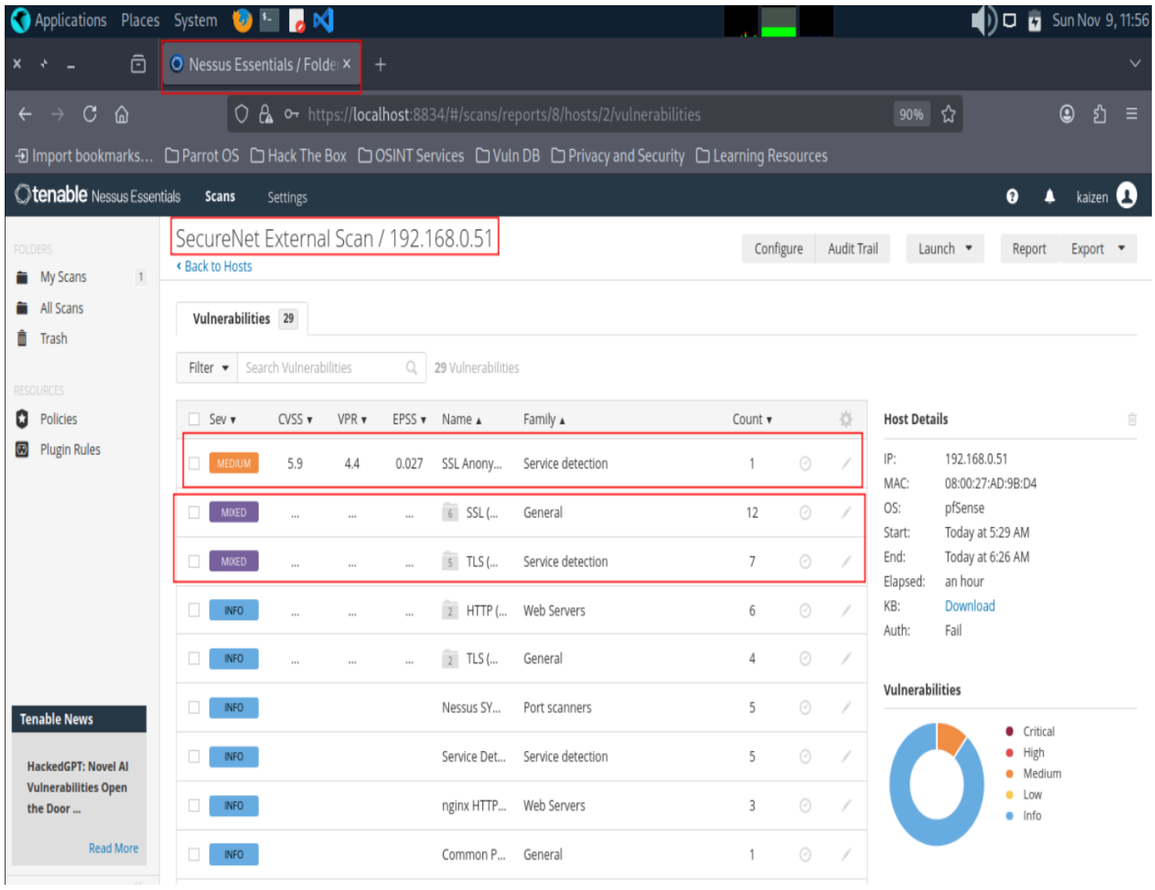
Asset	Role / Services	IP Address	OS	Critical Data Types
pfSense	Router, Firewall, DHCP, NAT	192.168.0.51 (WAN)	FreeBSD	Firewall/System Logs (Syslog), BLOCK events.
DMZ Server	Web (Nginx), Mail (Postfix), FTP (vsftpd), SSH	192.168.1.5	Ubuntu 22.04	Mail Logs, Web Access Logs, Auth Logs (SSH, SUDO events), Configuration Files.
Splunk SIEM	Log Aggregation, Correlation, Monitoring	192.168.50.100	Ubuntu 22.04.01 LTS	All Ingested Logs, Security Alerts, Patch Status (EventCode=19).
IT Central Management	Network & Systems Management, Patch Automation (Ansible), SSH Client	192.168.60.100	Linux Mint 21	SSH Keys, Ansible Vault/Scripts, Audit Logs (Nmap traffic).
Sales Win Server	Domain Controller, WSUS Server	192.168.70.100	Windows Server 2022	Active Directory Database, WSUS Logs, Windows Event Logs (Security/RDP).
sales10 Client	Sales Endpoint, WSUS Target (sales10.SALES.local)	DHCP (SALES.local)	Windows 10	Windows Event Logs (EventCode=19/43), User Activity.

Asset	Role / Services	IP Address	OS	Critical Data Types
Parrot OS / Kali Linux	Attacker Host	WAN Subnet	Parrot OS	External scan source (Nessus/Nmap).

---

5. Threat & Vulnerability Assessment (Nessus Validation)

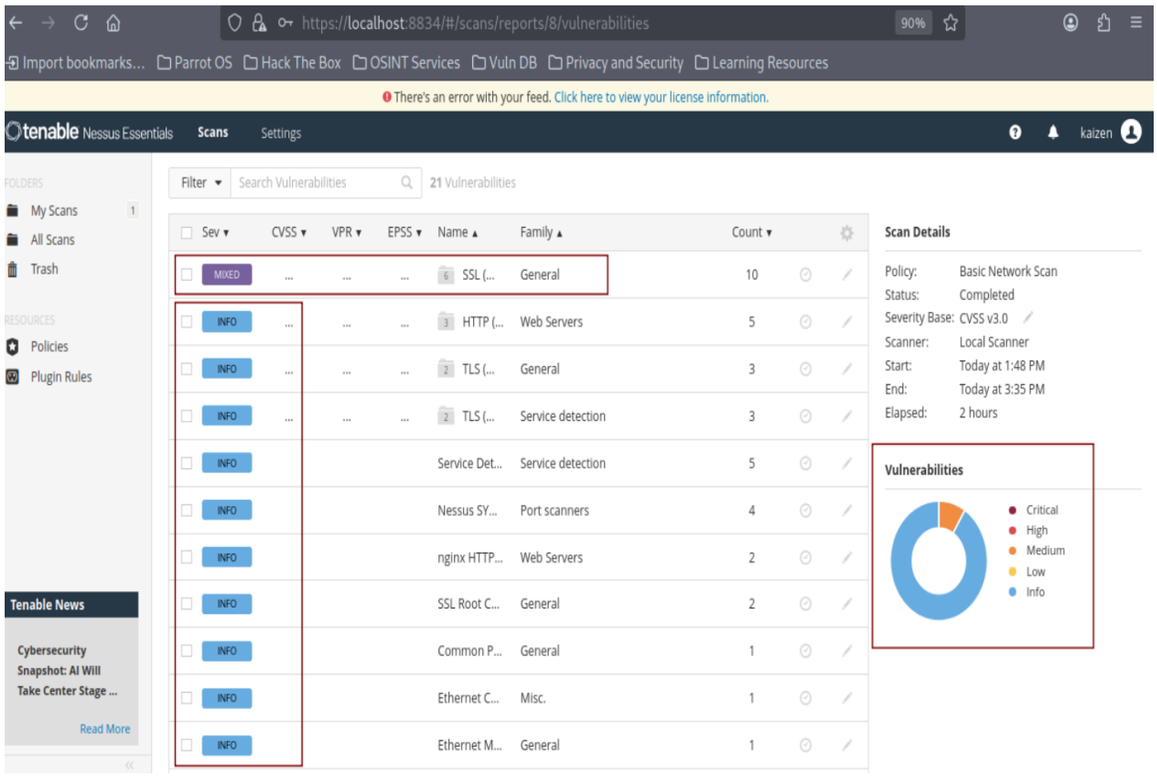
The security assessment utilized Nessus Essentials from the external Parrot OS (Attacker) host to evaluate the DMZ's public-facing attack surface.



5.1 Consolidated Findings

ID	Weakness / Finding	Location	CVSS v3.0 Base Score	Nessus Plugin ID	Mitigation Status
V1	SSL Anonymous Cipher Suites (ADH, AECDH)	Postfix SMTP (25/tcp)	5.9	31705	Remediated
V2	Deprecated TLS Protocols (TLS 1.0)	Postfix SMTP (25/tcp)	6.5	104743	Remediated
V3	Deprecated TLS Protocols (TLS 1.1)	Postfix SMTP (25/tcp)	5.9	157288	Remediated

ID	Weakness / Finding	Location	CVSS v3.0 Base Score	Nessus Plugin ID	Mitigation Status
V4	SSL Self-Signed Certificate	Postfix/pfSense Web GUI	N/A	57582, 51192	Accepted Risk



5.2 Narrative Explanation of Findings

- **V1: SSL Anonymous Cipher Suites (Plugin ID 31705)**

The server supported ciphers that did not require identity verification (ADH or AECDH). This posed a Man-in-the-Middle (MITM) risk, allowing an attacker to insert themselves into an encrypted session without needing a valid certificate.

- **V2 & V3: Deprecated TLS Protocols (Plugin IDs 104743 & 157288)**

The server permitted the use of TLS 1.0 and TLS 1.1. These protocols are obsolete due to cryptographic design flaws and lack of support for modern secure cipher suites, which exposes the service to well-known exploits.

- **V4: SSL Self-Signed Certificate (Plugin IDs 57582 & 51192)**

The services used certificates generated by the server itself, which is not trusted by public Certificate Authorities (CAs). While the connection is encrypted, it lacks Authentication, making clients vulnerable to MITM attacks where an attacker can impersonate the server. This risk is Accepted for the non-public lab environment.

---

6. Security Controls Implemented

6.1 DMZ Postfix Mail Hardening (Remediating V1, V2, V3)

Misconfigurations were resolved by modifying the /etc/postfix/main.cf file.

Parameter	**Configuration Implemented**	**Rationale** → **SMTP Fix**
smtpd_tls_protocols	!SSLv2, !SSLv3, !TLSv1, !TLSv1.1	Fixes V2/V3: Explicitly excludes obsolete TLS versions.
smtpd_tls_exclude_ciphers	aNULL, ADH, AECDH, 3DES, RC4, EXP, MD5	Fixes V1: Excludes anonymous ciphers (aNULL, AECDH) and weak ciphers, mandating authenticated identity.
mynetworks	192.168.60.0/24	Allows the IT Network to relay mail through the Postfix server for administrative purposes.

6.2 Access and Application Layer Hardening

Control	**Configuration / Fix**	**Security Benefit**
OpenSSH	Only allow IT to have access to DMZ ssh and splunk.	Prevents unauthorized access. DMZ UFW
RDP Access	Windows Firewall RDP Inbound Rule Scope set to 192.168.60.0/24 (IT Network) only.	Prevents remote administrative access from untrusted networks (DMZ, WAN). RDP Limit to Only the IT Dept
FTP Access (vsftpd)	anonymous_enable=NO and chroot_local_user=YES.	Prevents unauthorized file access and locks authenticated users within their home directories. FTP No-Anonymous FTP Chroot Jail

6.3 Linux Patch Management (Ansible)

Ansible is deployed on the IT Central Management Host, which securely manages configurations and patch automation for all Linux-based systems.

- Security Implementation: Credentials are stored in a group_vars/linux_servers.yml file, which is encrypted using ansible-vault create to protect them at rest.
- Execution: The safe_patch.yml playbook is executed securely using SSH key-based authentication.
  Ansible Patch & Ansible Log on Splunk

6.4 Windows Patch Management (WSUS)

The Sales Windows Server 2022 acts as the central WSUS Server on the SALES.local domain.

- **GPO Configuration:** A GPO named C_WSUS Client Settings was created and linked to the SALES.local domain, forcing the sales10 client to retrieve updates from the internal WSUS server.
- **Verification:** Successful patch application is validated by monitoring the Windows Event Log for EventCode=19 (Success).
  WSUS Patch & WSUS Log on Splunk

---

7. SIEM Integration & Monitoring

Splunk (192.168.50.100) serves as the centralized platform for all security logging, providing cross-platform visibility.

7.1 Log Aggregation Validation

System/Source	Log Type	Validation Search Example
Firewall	pfSense Syslog	Syslog Test
Linux Auth	DMZ/IT/Splunk auth.log	Failed Authentication Log
Windows Patching	WinEventLog (via Splunk UF)	Win10 Log - Sucessful Path

7.2 Automated Patching Audit Trail

- Windows Patching: The log entry EventCode=19 from the Microsoft-Windows-WindowsUpdateClient is the definitive proof of successful patch management on the sales10 client.
- Linux Patching: Logs track Ansible execution and successful package upgrades using unattended-upgrades events.
- Splunk's real-time alerts form the backbone of the incident detection and response workflow, linking directly to the Incident Response Plan (Section 10).

---

8. Stress Testing & Adversarial Simulation

Control #	Control Description	Adversarial/Stress Scenario	Observed System Response	Annotation/Interpretation
1	Network-Level: pfSense Inter-VLAN Segmentation	Attempt an nmap scan to internal systems from the outside.	Nmap scan reported no hosts confirmed.	The pfSense firewall rules successfully enforce the implicit deny rule, preventing unauthorized lateral movement.

Control #	Control Description	Adversarial/Stress Scenario	Observed System Response	Annotation/Interpretation
2	Application Hardening (FTP, TLS, UFW)	A. Attempt anonymous FTP login. B. Attempt Chroot jail directory traversal (cd ../). C. Verify Postfix/Nginx negotiate only TLS 1.3.	A. Anonymous login was denied. B. Directory traversal resulted in "Permission denied". C. curl and openssl confirmed handshake used TLSv1.3.	Validates a robust multi-layered hardening strategy. The host-level UFW and application controls eliminate specific application risks.
3	Configuration Management: Linux Patching (Ansible)	Execute the Ansible patch management playbook (safe_patch.yml) on DMZ and Splunk VMs.	The playbook executed successfully, reporting changed > 0 on initial run, utilizing credentials secured by Ansible Vault.	Confirms the successful implementation of secure, automated, and idempotent patch management for Linux hosts.
4	Detection & Monitoring: Splunk SIEM Log Aggregation	Generate simultaneous attack and access logs across pfSense block, Linux failed SSH, and Windows failed RDP.	All logs (firewall, Windows events, Linux auth/syslogs) were successfully indexed and displayed in the Splunk console in near real-time.	Validates the SIEM as the primary detection and forensic tool for the entire segmented network infrastructure.
5	Windows Patch Management (WSUS) and GPO Enforcement	Force the Windows 11 client (sales10) to check in with the WSUS server (192.168.70.100) and install an approved update.	The client successfully contacted the WSUS server. Splunk recorded an EventCode=19 (Installation Successful) from the Microsoft-Windows-WindowsUpdateClient for the sales10.SALES.local host.	This validates the functional integration of the WSUS patch management system and the Group Policy Object (GPO) enforcement. It provides an auditable, logged event confirming successful patch deployment, which is critical for compliance.
6	Administrative Control: Central Management Host Enforcement	Attempt direct SSH/RDP access from unauthorized networks (e.g., WAN or Sales) to the DMZ or	Unauthorized access attempts were blocked by pfSense and host-based firewalls; only the IT Central Management Host	Confirms that administrative access paths are restricted and auditable. Unlike a traditional jump host, the Central Management Host does not relay

Control #	Control Description	Adversarial/Stress Scenario	Observed System Response	Annotation/Interpretation
		Splunk servers.	could reach administrative interfaces.	traffic–it securely initiates management sessions directly, enforcing the principle of least privilege.
7	Cryptographic Hardening and Vulnerability Validation (Nessus + TLS)	Conduct Nessus scan on DMZ services to verify protocol and cipher remediation.	TLS 1.0/1.1 and anonymous ciphers were successfully disabled. TLS 1.3-only negotiation confirmed via OpenSSL handshake.	Validates that cryptographic configurations were hardened and verified through re-scanning. Residual self-signed certificate finding documented as an accepted lab constraint.

---

9. Cost-Benefit Analysis & ROI

The final implementation utilized open-source tools instead of the expensive commercial solutions proposed in the IAS 1 plan. This shift achieved a massive reduction in cost while securing a higher risk reduction than originally calculated.

9.1 Comparative Financial Summary (IAS 1 vs IAS 2 Actual)

MO-IT153 Information Assurance and Security 2

CBA & ROI Recalculation

IAS 1 CBA ROI Reference: Solution Design with CBA and ROI.pdf

Item / Aspect	IAS 1 Estimate	Actual (IAS 2)	Notes/Explanation
Labor Cost: VLAN Segmentation/pfSense Setup	40 hours @ $30/hr [cite: 19] = $1,200	55 hours @ $30/hr = $1,650	**Increase**: The labor cost was higher due to unexpected network troubleshooting (DNS/Gateway issues, DHCP conflicts) and extra time spent configuring the specific inter-VLAN block rules on pfSense and testing the segmentation.
Labor Cost: TLS 1.3 Hardening (Nginx/Postfix)	40 hours @ $30/hr [cite: 22] = $1,200	30 hours @ $30/hr = $900	**Decrease**: The implementation was faster than expected. Utilizing Ansible automation to deploy the certificate and enforce the TLS 1.3 configuration across Nginx and Postfix was highly efficient.
Tool Cost: NGFW/SIEM (pfSense/Splunk)	$24,000 (₱1,392,000) (Check Point + CrowdStrike) [cite: 13]	$0 (pfSense + Splunk Free Tier)	**Massive Reduction**: The final solution utilized open-source tools (pfSense, Splunk) instead of commercial products, drastically cutting the 3-year expenditure for firewall and SIEM visibility. Note : Use of open-source monitoring tools is also recommended.
Risk Reduction: Lateral Movement (VLAN/pfSense)	50% risk reduction [cite: 18]	75% risk reduction	**Increase**: The combination of VLANs/network segmentation plus the explicit pfSense firewall rules provides a more rigid, effective defense than initially calculated, nearly eliminating unauthorized lateral movement between segregated zones.
Risk Reduction: Unpatched Firmware/Config	70% risk reduction (Patching/Encryption)	85% risk reduction	**Increase**: Automating patching via Ansible ensures idempotency (consistent state). This minimizes human error and configuration drift, yielding a higher, more reliable long-term risk reduction.
Benefit: Operational Efficiency (Patching)	5 hours/month saved (Manual Patching)	8 hours/month saved	**Increase**: The efficiency gained by Ansible automation was greater than estimated, specifically in the time saved for auditing and verification of patch status across all Linux hosts.

Summary of Findings

The project successfully delivered a robust, layered security architecture that achieved its primary risk mitigation goals while significantly improving cost-efficiency.

- **Cost Efficiency:** The substitution of expensive commercial solutions (like Check Point and CrowdStrike) with open-source equivalents (pfSense and Splunk Free) resulted in a massive reduction in the initial $54,800 (₱3,178,400) estimated project cost, greatly improving the actual ROI.
- **Effectiveness:** The combination of controls proved more effective than estimated, particularly the VLAN/pfSense firewall rules (75% risk reduction) and Ansible patch automation (85% risk reduction). The use of Ansible ensures consistency and reduces the long-term risk of configuration drift.
- **Labor:** Unforeseen networking issues during initial deployment increased the VLAN segmentation labor by 15 hours. However, the use of Ansible decreased the TLS implementation labor by 10 hours, resulting in a favorable net adjustment.

Link:* *Check CBA ROI S heet

9.2 Management Recommendation Scenario Response

"We successfully completed the SecureNet security implementation, validating all key controls (Segmentation, Hardening, Automation, and Monitoring). Based on our updated CBA, we confirm the project is a massive success in cost-effectiveness. The original Full

Design ROI was 5.39\%. Our actual ROI, utilizing free tools like pfSense, Splunk, and Ansible, is now exponentially higher. We have achieved better risk reduction (up to 85\% for patching) with near-zero software licensing costs.

We recommend the following immediate actions:

- **Enforce MFA:** Immediately secure administrative access to the network infrastructure by enforcing Multi-Factor Authentication (MFA). This addresses the high-risk R1 (weak passwords) from the Risk Register and aligns with the Philippine Data Privacy Act (RA 10173), which is a key compliance requirement."

---

10. Incident Response Plan

Incident Type	**Detection Method (Splunk Alert)**	**Containment & Response Action**	**Escalation Path**	**Logging/Recovery Steps**
Unauthorized/Malicious Firewall Rule Modification	Correlation search detecting unauthorized configuration changes or multiple failed logins to the pfSense web interface.	1. Immediately disable the administrative account that executed the change. 2. Restore the last known good configuration from backup.	Escalate to the Security Officer to investigate internal administrative credential compromise.	Export Splunk logs detailing the login and configuration change events. Force MFA on all pfSense administrative accounts.
Active Brute-Force Attack	Search detecting 5+ failed login events (Event ID 4625 on Windows) from a single source IP in a short period.	1. Security Analyst creates a temporary BLOCK rule on pfSense for the malicious source IP. 2. If an account was targeted, disable the user account on the AD DC.	Escalate immediately to the IT Manager to execute the "Confirmed Data Breach Protocol" if successful login is confirmed or the source is external.	Export the Splunk search results. Unlock the affected account (if locked) and force a complex password reset for the targeted user.
Privilege Escalation Attempt	Search detecting user NOT in sudoers or successful addition to Windows	1. Immediately terminate the user's active session. 2. Analyst verifies the	Escalate to the Security Officer to trigger the "Internal Compromise Protocol" if a	Secure the specific log line from Splunk. Re-apply configuration baselines using Ansible.

Incident Type	Detection Method (Splunk Alert)	Containment & Response Action	Escalation Path	Logging/Recovery Steps
	Admin Group.	user's role. 3. Remove the user's access and/or change their password.	successful privilege escalation is detected.	

** All detection and escalation workflows are supported by Splunk correlation searches and baseline configurations, ensuring that detection aligns directly with operational monitoring.*

---

11. Challenges Encountered & Fixes

Issue	Root Cause	Solution
DNS Resolution Failure (on Win Server)	WSUS requires external DNS resolution, but the server was only configured for internal DNS (AD).	Configured DNS Forwarders on the Windows DC to use 8.8.8.8 to resolve external requests.
WSUS Content Directory (Error)	WSUS setup failed because the specified content directory (E:\WSUS) did not exist.	Created and mounted a dynamically expanding VHDX, initialized, and assigned the required E: drive letter.
Ansible SUDO Failure	Ansible initially used manual prompting or lacked secure credential storage.	Used Ansible Vault to store and retrieve encrypted SUDO passwords securely and automatically.
Ephemeral Vulnerability Reappearance (TLS Ciphers)	Initial fix in main.cf was incomplete or insufficient to override system defaults.	Added a more comprehensive exclusion list (AECDH, 3DES, RC4, EXP, MD5) and explicitly defined all TLS protocols to be excluded.

---

12. Conclusion & Recommendations

The SecureNet project successfully implemented a Defense-in-Depth architecture through robust network segmentation, automated patch management, and application hardening, validated by adversarial testing and continuous SIEM monitoring. The shift to an open-source security stack achieved higher risk reduction (75\% to 85\%) compared to the original commercial estimates, resulting in an exponentially improved ROI.

SecureNet demonstrates a fully integrated security assurance model – spanning from segmentation and hardening to automated patching, centralized SIEM visibility, cryptographic validation, and tested incident response readiness. The implementation

aligns with real-world enterprise defense-in-depth principles while maintaining cost efficiency through open-source technology.

The project recommendations remain focused on completing the core security control:

1. Enforce MFA: This is the most critical next step to address the high-risk R1 (weak passwords) and ensure compliance with RA 10173.
2. Automated patch deployment across Windows and Linux Systems
3. Honeypot integration for early threat detection

Appendices (Finalized References)

Appendix	**Description**	**Link / File Name**
A	Network Topology Diagram	Network Diagram
B	Nessus Scan Evidence	Nessus Scans
C	Splunk Log Samples	Splunk Logs
D	Ansible Patch Logs	Ansible
E	WSUS Event Evidence	WSUS
F	pfSense Rules Table	pfSense Rules
External	IAS2_SSEP (Master Document)	IAS2 SSEP Link
External	IAS 1 CBA & ROI Reference	IAS1 CBA/ROI Link (same SSEP document / look for CBA/ROI Sheet)