Softwarica COLLEGE OF IT & E-COMMERCE | in collaboration with Coventry University

PROJECT: STRENGTHENING CYBER RESILIENCE

PROJECT DETAILS:
Module: ST6047CEM
Prepared By: Suren R. Tuladhar (ID: 250182, 16546891)
Prepared For: Manoj Shrestha
Institution: Softwarica College & Coventry University

TABLE OF CONTENTS

LIST OF FIGURES

Enhancing Cyber Resilience in the Nepalese Banking Sector A Cost-Effective Implementation of Wazuh SIEM

INTRODUCTION

Cybersecurity has become one of the biggest problems that banks and other financial institutions in Kathmandu have to deal with. As Nepalese banks quickly switch to digital platforms, mobile banking (like Fonepay and ConnectIPS), and cloud computing, the attack surfaces are more complicated than ever. Multinational companies may have a lot of money, but banks in developing countries like Nepal face two problems: the rapid rise of cyber threats and tight budgets that make it hard to buy high-quality enterprise security tools. Even though companies spend money on firewalls and basic antivirus software, their security is still very risky and uncertain. Many banking security teams in Kathmandu use reactive methods to look into logs only after a customer reports fraud or a system crash. They do not use proactive, data-driven methods to find weaknesses. Because of this, vulnerability management turns into a way to deal with breaches instead of a way to stop them from happening in the first place.

Figure 1 SIEM Protection

CONCEPTUAL DESIGN

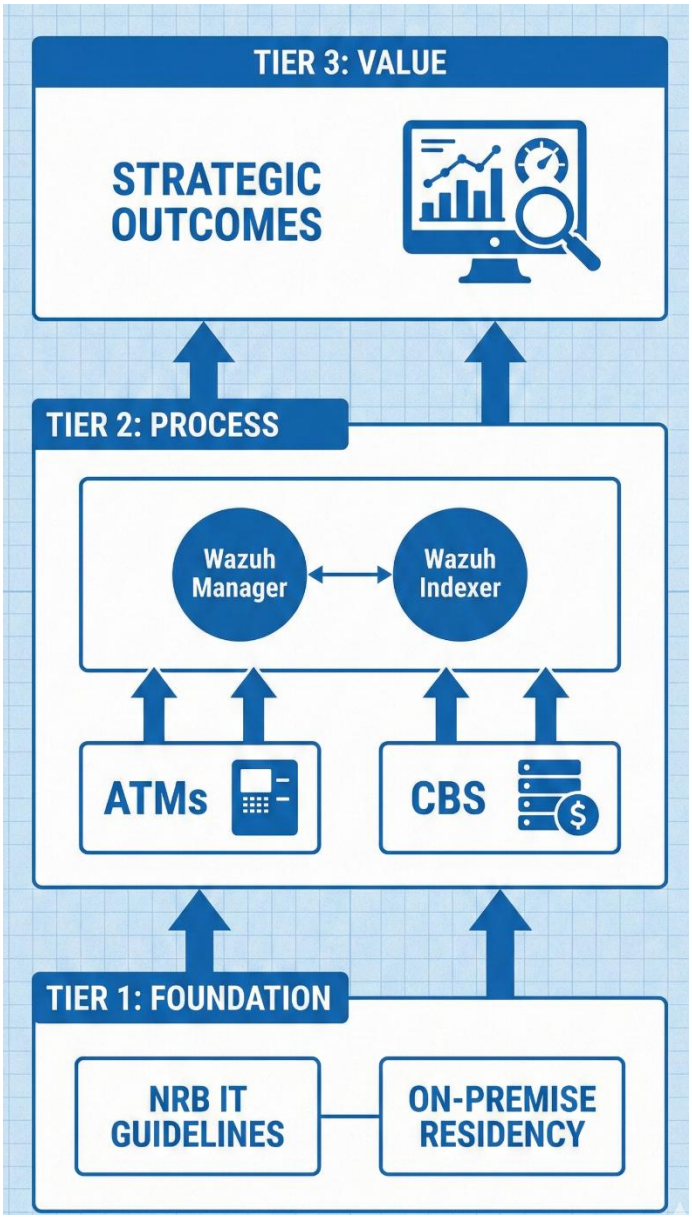

Figure 2 Conceptual Diagram

This thesis proposes the design and implementation of a dependable, cost-efficient Security Information and Event Management (SIEM) framework utilizing Wazuh, to meet the specific compliance requirements of the Nepal Rastra Bank (NRB) and to circumvent the resource limitations faced by human analysts in the Nepalese banking sector. The system will systematically find and rank security incidents by using a lot of log data from Core Banking Systems (CBS), network perimeters, and endpoints.

.

PROBLEM CONTEXT AND MOTIVATION

Most of the ways that organizations in Nepal manage their vulnerabilities do not use systematic data analysis or comprehensive real-time monitoring. Many banks do not process or understand raw logs from firewalls and servers well enough to protect themselves ahead of time, even though they have access to them.

This research seeks to rectify the deficiency of cost-effective, organized, evidence-based threat detection instruments accessible to Nepalese enterprise security teams. This project aims to offer a logical, data-driven alternative to security management based on gut feelings by using an open-source SIEM (Wazuh) that works with Threat Intelligence.

Functional Requirements

- The SIEM should be able to collect data from log sources in real time and
- parse those collected logs.
- The system must support custom rules to detect specific threats and
- generate alerts based on those predefined rules.
- The alerts generated in the SIEM should be automatically created.

Non-Functional Requirements:

- The system should be easy to use with a user-friendly interface.
- The system must be scalable to accommodate future growth.
- The system must be highly available and reliable.
- The entire system should be designed with security in mind, to protect the
- CIA of the data.

Figure 3 SOC Model

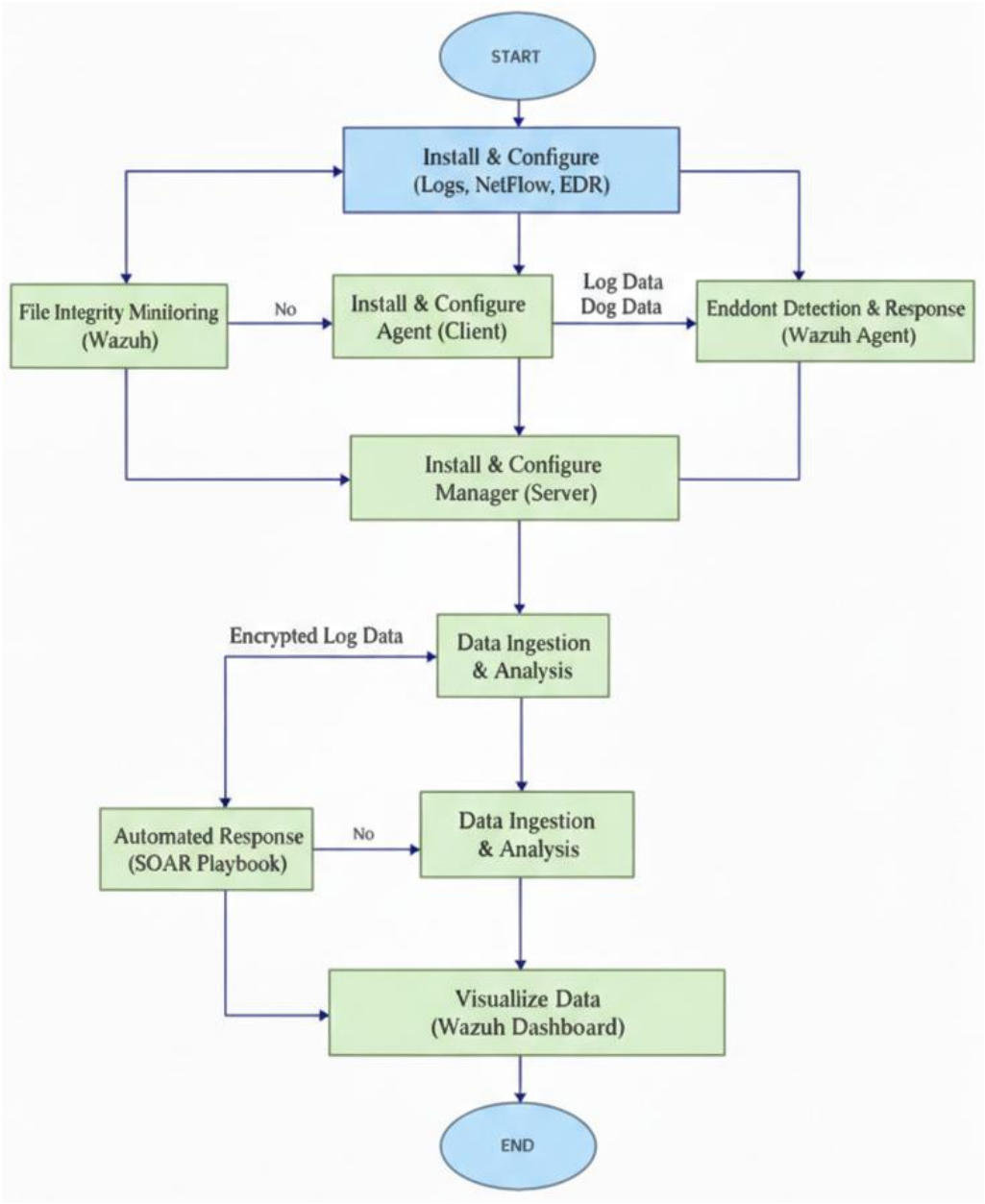

Figure 4 System Workflow

CYBERSECURITY THEORIES AND BEHAVIORAL FACTORS IN SECURITY DECISION-MAKING

Enterprise cybersecurity is a complicated way for organizations to make decisions when they are under attack. This research utilizes the present market conditions and professional experience as its theoretical framework. This model posits that robust banking security necessitates complete visibility throughout all phases of potential attacks, from initial reconnaissance of the bank's public web servers to the ultimate goals of data exfiltration or fraudulent fund transfer.

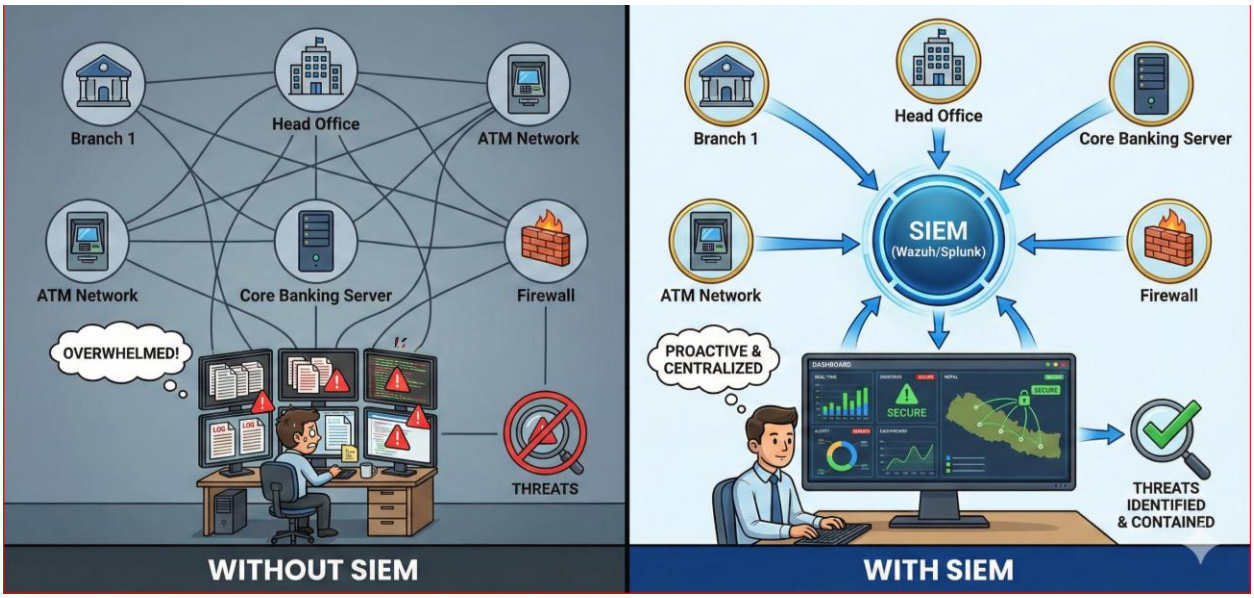

Figure 5 With & Without SIEM

In practice, banking security teams often focus on fixing individual problems, like patching Windows servers, instead of looking at the bigger picture of how all the problems fit together. This leaves banks open to multi-stage threats. This is often because of Prospect Theory, which says that the fear of loss leads to bad resource management. For example, banks often treat security like a luxury item, spending millions on "prestigious" foreign firewalls for peace of mind instead of investing in the high problem-solving value of internal log monitoring. An intelligent SIEM framework fixes this by getting rid of the main problem of people not being able to think clearly and getting tired of alerts. It does this by using the same analytical methods on huge datasets to make sure that NRB guidelines are followed while also providing a much higher "value-to-price" ratio than traditional, separate security tools.

INTEGRATION OF OPEN-SOURCE TOOLS AND THREAT INTELLIGENCE

Many businesses have used data-driven decision-making to make their operations more efficient. Wazuh (as the Manager), Elasticsearch (for Indexing), and Kibana (for Visualization) work together in this project to fix the problem of security data being spread out. This integrated framework combines

1. **Log Analysis:** Collecting data from the CBS and Active Directory i.e pre-existing database.

2. **File Integrity Monitoring (FIM):** ensuring no unauthorized changes occur in critical banking configuration files.

3. **Threat Intelligence:** Integrating feeds to detect known malicious IPs targeting the financial sector.

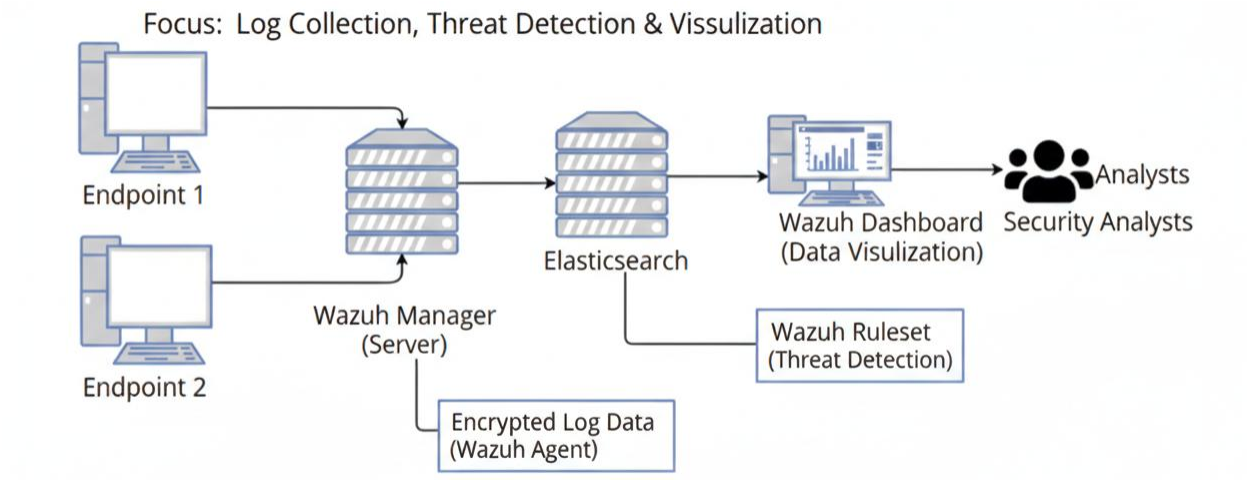

Figure 6 Workflow of the SIEM

Use Case Diagram

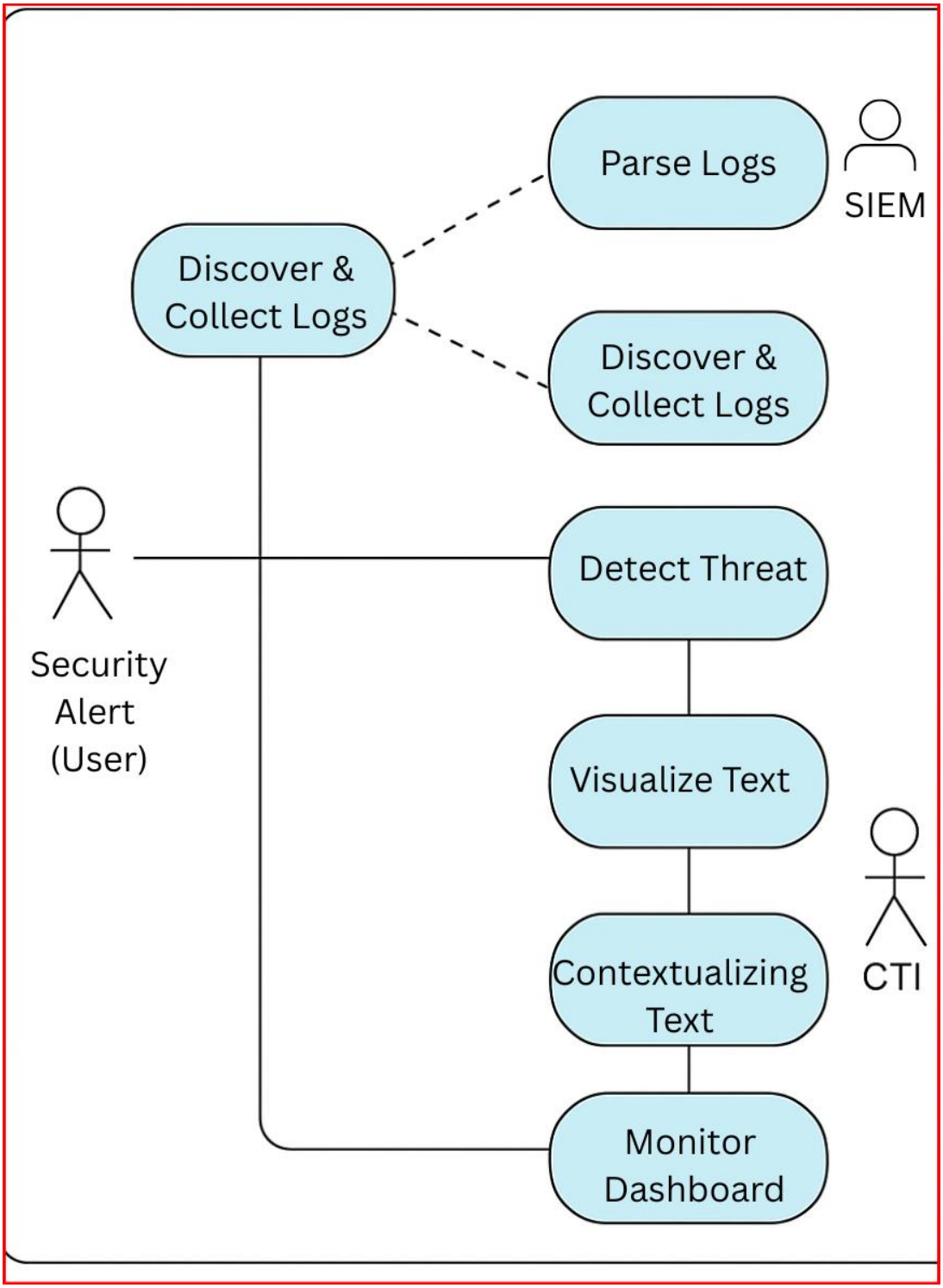

Figure 7 Use Case Diagram For SIEM

This case diagram outlines the steps involved in the SIEM (Security Information and Event Management) process:

1. **Discover Logs** : Initial step where security logs are identified and collected from various sources.

2. **Parse Logs** : The collected logs are parsed to extract meaningful information and prepare them for further analysis.

3. **Detect Threat:** The parsed logs are analyzed to detect potential security threats. When an anomaly or threat is identified, an alert is generated.

4. **Create Case:** Upon detection of a potential attack, a case is created. This case includes detailed information about the alert for further investigation and response.

5. **Contextualize Alerts:** The information obtained from threat intelligence is used to add context to the alerts. This step provides a comprehensive understanding of the alert, aiding in effective decision-making.

6. **Close Case :** Once the alert has been fully investigated and contextualized, the case is closed concluding the response process. This sequence ensures that logs are systematically collected, analyzed, and investigated to identify and respond to security threats effectively.

RESEARCH AIM

The main goal of this thesis is to set up and test a custom Security Information and Event Management (SIEM) framework for a commercial bank in Kathmandu using Wazuh. The system's goal is to improve the security of business networks by automatically analyzing logs, prioritizing risks in real time, and giving actionable remediation advice that follows the rules set by the Nepal Rastra Bank.

- Collecting and integrating comprehensive security data from network monitoring systems and banking applications.
- Implementing custom rules and decoders to identify vulnerability patterns specific to the banking environment.
- Developing contextual risk assessment capabilities that consider the criticality of banking assets.

Research Objectives

1. To analyze current enterprise vulnerability management practices in the Nepalese banking sector, identifying common failure modes, resource constraints, and gaps in NRB compliance.

2. To investigate and evaluate open-source SIEM capabilities (specifically Wazuh) that have demonstrated effectiveness in financial cybersecurity applications, focusing on log retention, integrity monitoring, and incident response.

3. To design and deploy a High-Availability (HA) Wazuh cluster architecture that integrates with existing banking infrastructure (Active Directory, Firewalls, CBS) to identify capability gaps and opportunities for automation.

4. To develop custom correlation rules and dashboards that specifically target financial fraud indicators and compliance requirements (NRB IT Guidelines Directive No. 4).

5. To validate the framework through controlled testing (Red Teaming/Attack Simulation), gather feedback from SOC analysts, and document the findings to support a cost-effective security model for developing nations.

CONTRIBUTION AND SIGNIFICANCE

This research is important for many reasons. From a technical standpoint, it enhances the domain of security engineering by illustrating the optimization of open-source tools for high-security financial settings and innovating methods for log correlation within limited infrastructure.

From a practical point of view, the research looks at important problems that security teams in Kathmandu face on a daily basis, such as alert fatigue, not having enough money for commercial SIEMs (like Splunk), and not having enough skilled workers. The framework gives businesses useful tools to make their security better while still following local laws.

.

JUSTIFICATION OF THE STUDY

The world of cybersecurity has changed, and in 2024, the average cost of a data breach will be more than $4.88 million. For a bank in Nepal, even a small part of this cost could cause it to go bankrupt or hurt its reputation badly. The lack of qualified cybersecurity professionals around the world (about 3.5 million open positions) is especially bad in Nepal because many skilled IT workers have left the country.

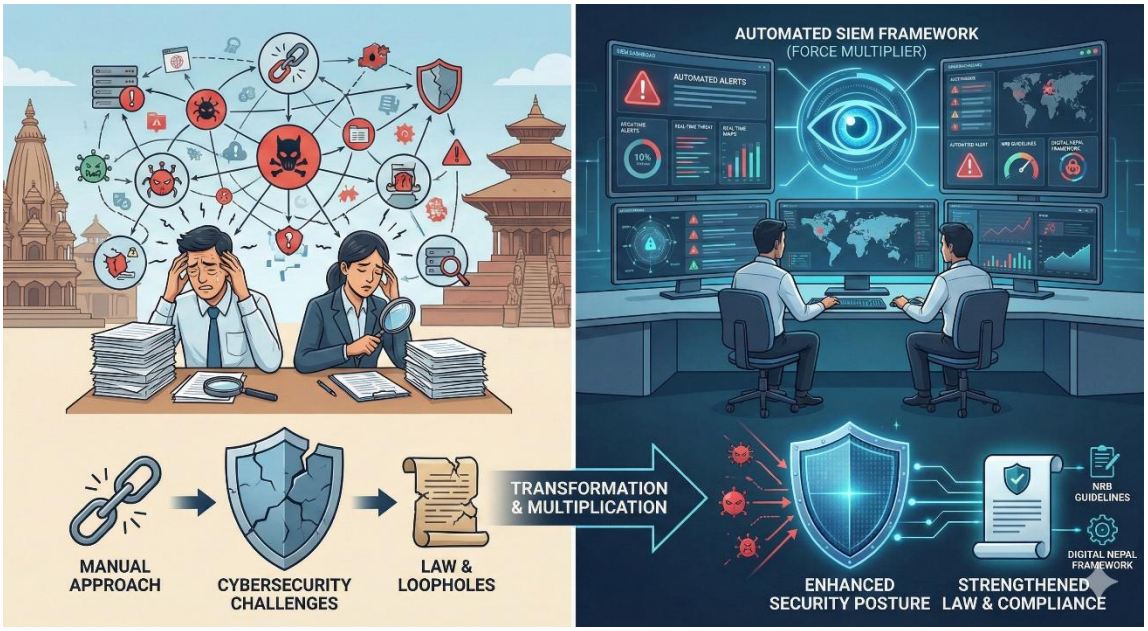

Figure 8 Transformation

A lot of businesses in Nepal do not have enough people to do thorough vulnerability assessments using traditional methods. An automated SIEM framework is a big step forward because it lets a small number of analysts keep an eye on a lot of endpoints quickly and easily. Security professionals can spend more time on strategic tasks like threat hunting by automating routine identification tasks.

When organizations utilize commercial SIEM solutions, it usually have to send data to cloud servers that are not in Nepal. This makes me think a lot about Data Sovereignty. This study validates the deployment of a self-hosted solution (Wazuh) that ensures the safeguarding of all sensitive banking logs within the organization's on-premise infrastructure, adhering to rigorous data privacy regulations.

Research Questions

1. **RQ1:** How can an open-source SIEM framework (Wazuh) be architected and customized to meet the specific regulatory compliance (NRB) and threat detection requirements of a commercial bank in Kathmandu, while minimizing total cost of ownership?

2. **RQ2:** To what extent does the integration of File Integrity Monitoring (FIM) and automated log correlation improve the Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR) for security incidents compared to traditional manual log review processes?

Research Hypotheses

Hypothesis 1: By using a customized Wazuh SIEM framework instead of relying on the "luxury" of expensive, ready-made software, and by adding specific banking correlation rules and live threat intelligence, we can do a lot more than just feel safe. This method directly fixes the problem of detection lag and makes both how accurately we find vulnerabilities and how quickly we catch incidents statistically significant better. This framework has a better "problem-to-price" ratio than traditional signature-based antivirus, which is expensive and does not work very well, or manual log audits, which are prone to human error. This means that banking resilience in Nepal is based on real data visibility instead of just a brand-name security spend.

Hypothesis 2: It may be tempting to think of an automated SIEM as a "silver bullet" for defense, but using it without a High-Availability (HA) architecture or strict rule-tuning is a huge risk for the business. Without that backbone, The organization not really fixing the security problem; Our team just replacing one problem with another: a lot of false positives. This "Alert Fatigue" does not just bother analysts; it also hides the most important, high-stakes threats. If the system is not set up to tell the difference between normal traffic and an actual breach in our local banking environment, the investment becomes a "luxury" expense that gives a false sense of security while the "problem-to-price" ratio stays negative.

.

IMPLEMENTATION DETAILS

The goal of this project is to make SIEM software that is cheap and works well for businesses in Nepal. By combining open-source platforms like Wazuh, Elasticsearch, Filebeat, Kibana, and threat intelligence sources, we hope to provide extensive security monitoring and threat mitigation capabilities. This solution gives Nepalese businesses a cheap way to protect their digital assets.

CONFIGURATION OF WAZUH-AGENT

The Wazuh-Agent is configured on the endpoints we aim to monitor, collecting logs from various directories, which are then sent to the Wazuh manager. Configuring the directories we wish to monitor enhances our understanding of our security infrastructure and identifies vulnerable points on the bank`s endpoints. For the purposes of this project, our team have configured wazuh-agents in the two designated two endpoints

Installation of Wazuh-Agent in Windows Operating System

The first step to install the Wazuh agent on a Windows machine is to download the Windows installer from the packages list.

Using the GUI

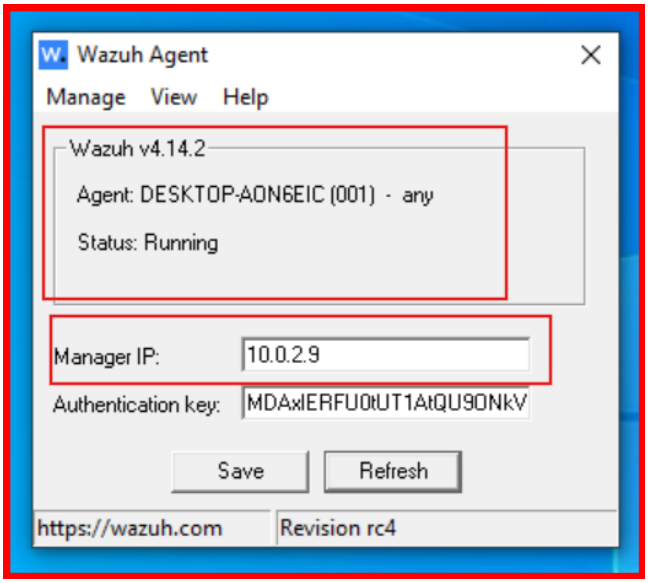

Figure 9 GUI installation verification of Wazuh- agent In Windows

After installing, the above GUI opens. Here, we need to add the manager IP and the authentication key to connect the windows agent with the wazuh manager.

Verification of Wazuh Agent

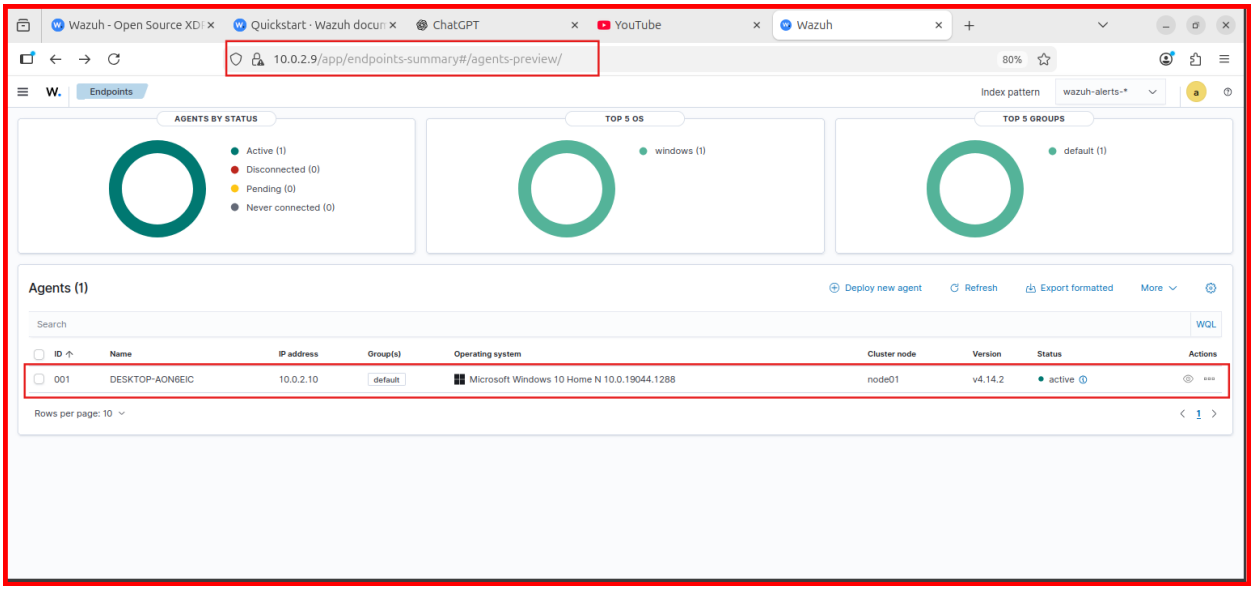

Figure 10 Agent Is Online Seen From Wazuh Manager

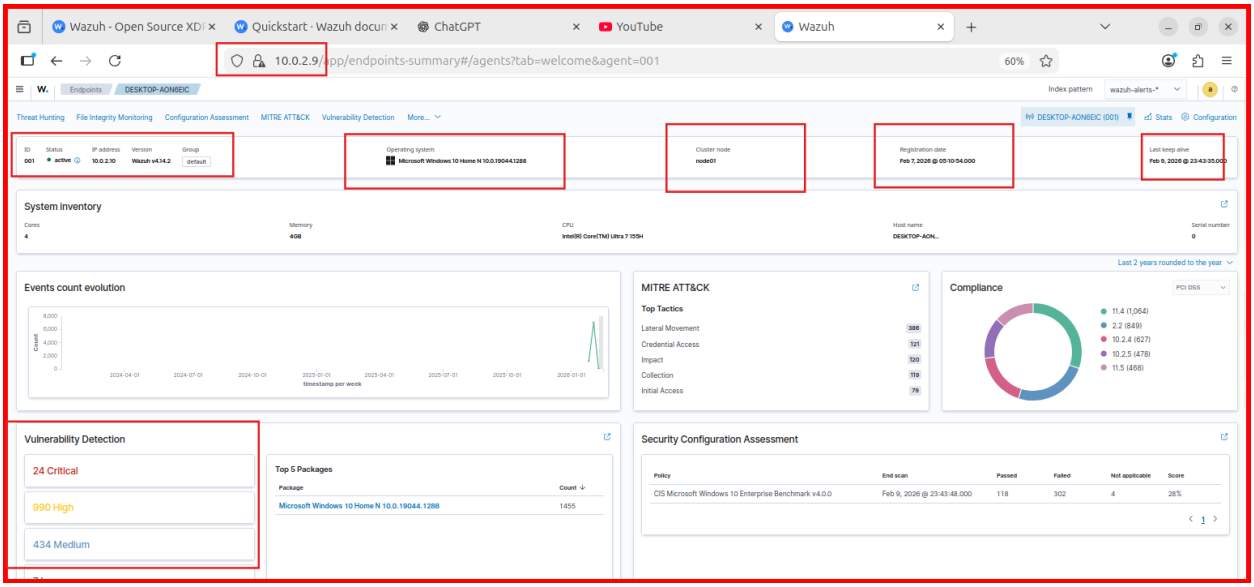

Figure 11 Detailed overview of the Agent

CONFIGURATION OF WAZUH-MANAGER

The Wazuh manager takes charge of coordinating the agents and processing their data. It gathers information from all connected agents, standardizes it into a common format, and analyses it to detect potential security incidents. Additionally, it oversees rulesets, configurations, and policies that dictate how the system responds to security threats

Installation Wazuh

Figure 12 Installing Wazuh Manger i.e Already Install

Verification of Wazuh

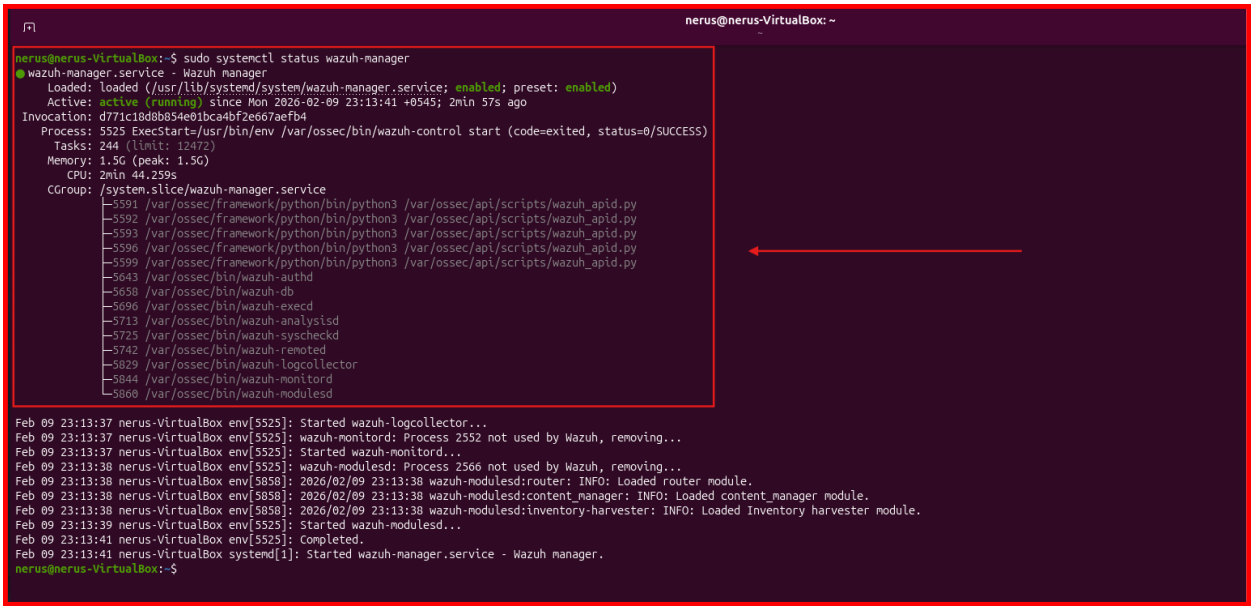

Figure 13 Status Check

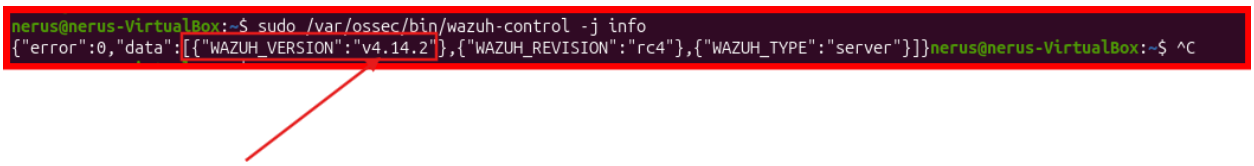

Figure 14 Wazuh Version 4.14.2

Verification Of Wazuh Localhost

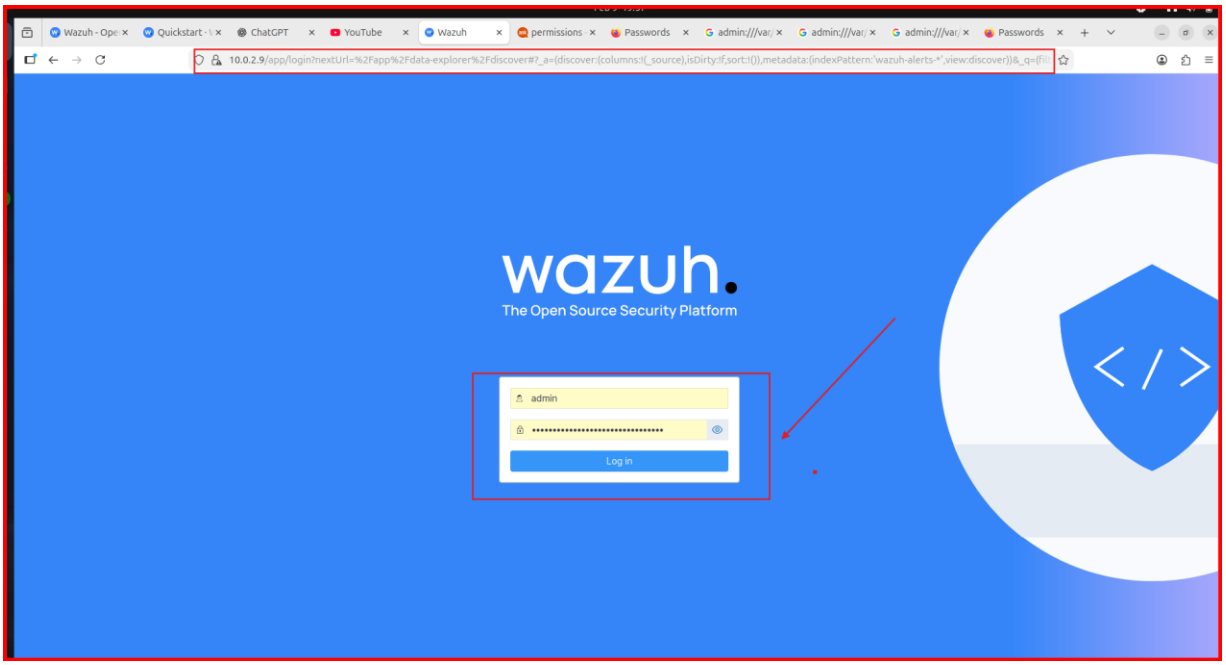

Figure 15 Wazuh Login Dashboard

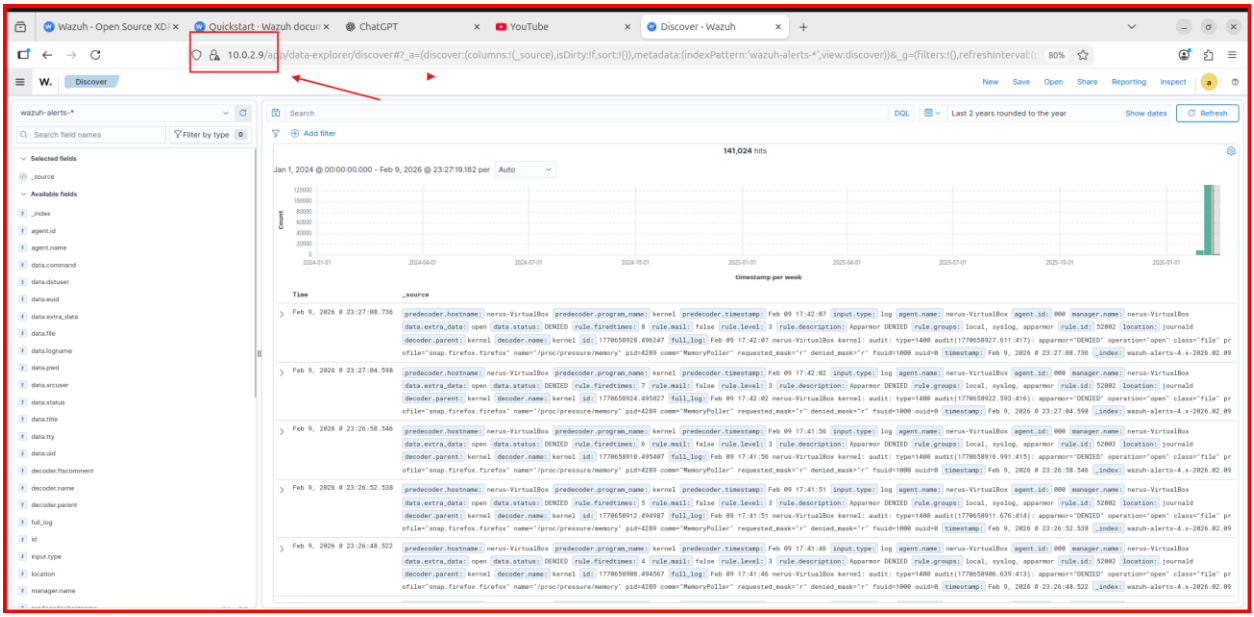

Figure 16 Wazuh UI Local Host

Verification Of File Beat

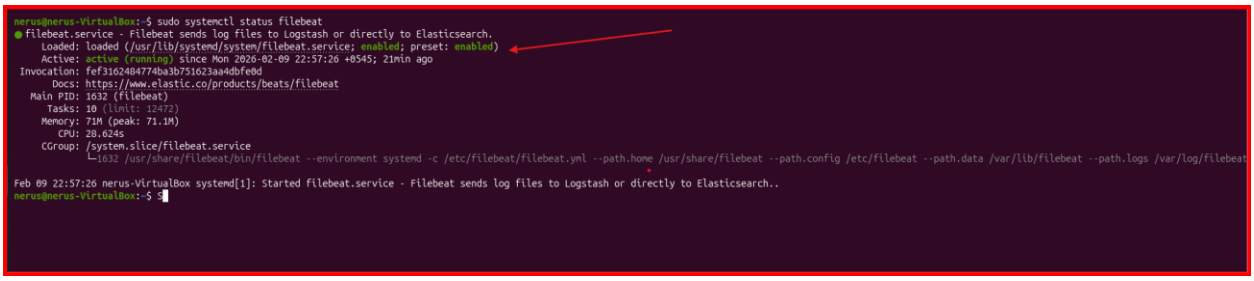

Figure 17 File Beat Installed

Conclusion

Nepal's banking sector can greatly improve its online security by using free and open-source SiEM (Security Information and Event Management) frameworks. The Wazuh platform is a good example of this idea. The new ELK stack, which includes Wazuh Indexer and Wazuh Dashboard, works better and is easier to manage than the old one, which includes Elasticsearch, Logstash, and Kibana.

The first deployment had trouble with memory allocation for Java-based indexing because there were not enough resources. By improving the infrastructure, a safe space is created that makes it possible to quickly find threats and analyze logs. This architecture meets the technical needs for keeping an eye on bank assets while also following the rules set by the Nepal Rastra Bank (NRB). It gives a basic framework that can build on to deal with incidents and make sure follow the rules.

.

RESEARCH METHODOLOGY

RESEARCH DESIGN

- **Design Science Research (DSR) Approach** This research employs a comprehensive Design Science Research (DSR) methodology combined with controlled experimental validation.

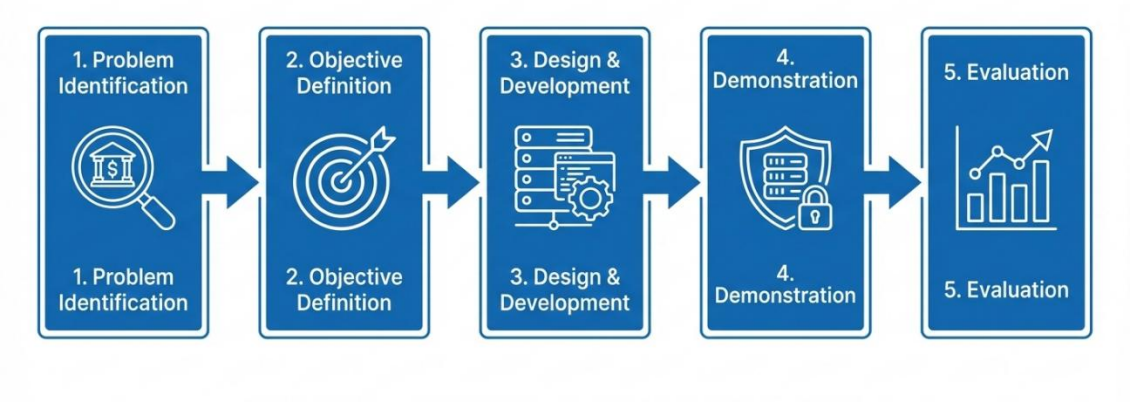

Figure 18 Research Methodology

- Design Science is a great way to study cybersecurity because it focuses on making and testing IT artifacts, like the Open-Source SIEM Framework (Wazuh), to solve problems that have been found in an organization. This method lets the system be tested very thoroughly against real-world banking threats, and it also makes sure that the solution can be used in the resource-limited setting of Nepalese banks.

- **Iterative Development and Prototyping** Our Team methodology departs from rigid academic models by pulling in Agile practices specifically to handle the "data tax" issues facing smaller Nepali banks. The framework evolved through a "break-and-fix" cycle rather than a linear plan. Our team The initial deployment consisted of. a bare-bones Wazuh architecture essentially a baseline and immediately tested it against common banking threat vectors to find the gaps. Instead of sticking with "luxury" default rules that generate noise, Our team spent the bulk of the process manually tuning correlation rules to meet the specific log retention needs of NRB Directive No. 4. This iterative loop was vital; it ensured the SIEM actually solves the problem of alert fatigue for understaffed security teams in Kathmandu, moving the project beyond a theoretical exercise and into a functional tool for real-world resilience.

DATA COLLECTION AND DATASET GENERATION

- **Utilization of Synthetic Banking Data** Because Nepal has strict banking privacy laws, Our team couldn't handle live customer data due to strict privacy regulations. Instead, Our team built a high-fidelity synthetic dataset called **banking_live** to avoid that ethical minefield. This is not just random data; it's a simulation of a full month of operations for a Class "A" commercial bank right here in Kathmandu, with 500,000 records. The research successfully recreated the real statistical chaos of a banking network without touching any sensitive customer information by focusing on "problem-solving" instead of "luxury" real-world data access, which is almost impossible to get. This lets stress-test the Wazuh framework against real-world traffic patterns while still following all local privacy laws.

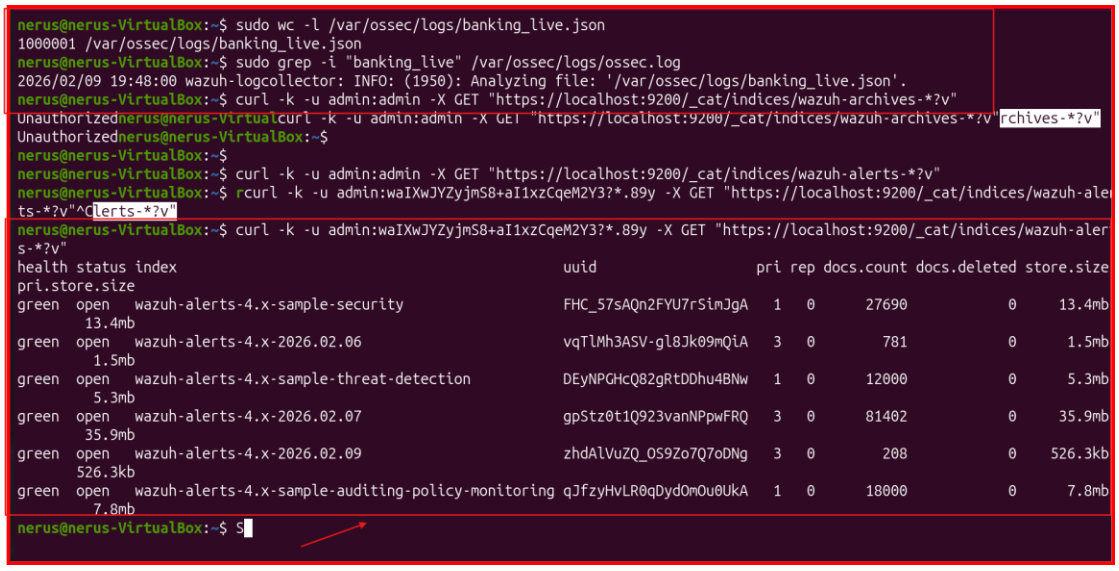

Figure 19 Checking Dataset from Terminal

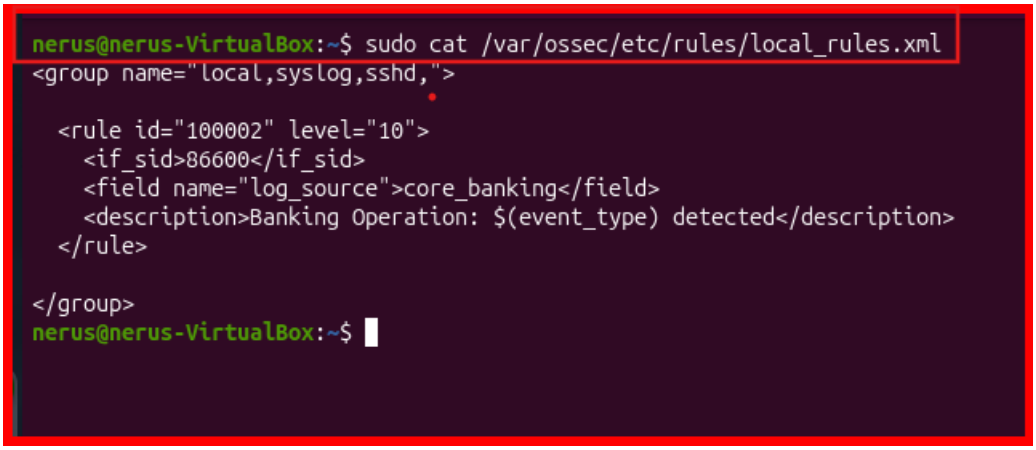

Figure 20 local Rules XML file

Diverse Log Source Integration: Our Team plan for collecting data wasn't just to get logs; it was to make a "holistic" map of the bank's most vulnerable areas. Our team got information from four important areas that are usually kept separate. This meant collecting logs from the Core Banking System (CBS) for high-value transactions, SWIFT Gateway logs to look for fraud in other countries, and the usual Network Perimeter traffic. Our team also made sure to include ATM Endpoint logs because in our case, attacks on the terminal, both physical and logical, are a big problem that is often ignored. This combination makes sure that the SIEM is not just a "luxury" dashboard for simple alerts, but a real-world tool that can catch the kinds of complex, multi-vector threats that keep Nepali security teams up at night.

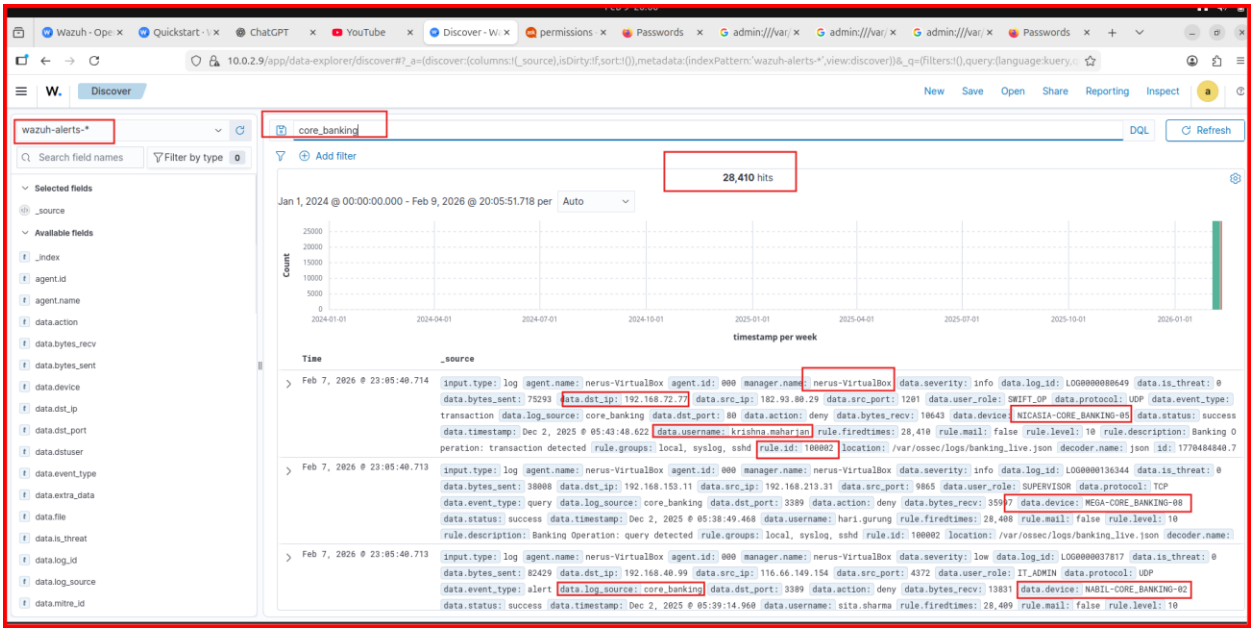

Figure 21 Wazuh Dataset Indexing

The above screenshot has the simulated data-log has different entries from the bank like NIC Asia, Mega Bank , Nabil Bank etc.

- **Threat Injection and Labeling** The true worth of this dataset is in its "Ground Truth" labels, especially the is_threat=1 marker, which Our team have linked directly to the biggest threats to the Nepal Rastra Bank landscape. Our team haven't just looked at general categories; Our team have looked at specific vectors like SWIFT Fraud (T1657), where Our team simulate the kinds of strange transaction bursts that happen at international gateways. Our team also added Ransomware (T1486) simulations to show how quickly encryption can happen and how it can bring a bank to its knees in minutes. Our team also added ATM Logical Attacks to show how "jackpotting" attempts have hurt Kathmandu's financial sector in the past. Using these labeled events has helped me stop guessing; these provide a robust method to demonstrate that **the** custom correlation rules to show that Our Team custom correlation rules aren't just "luxury" filters but are actually catching the specific attacks that would cause a real-world crisis.

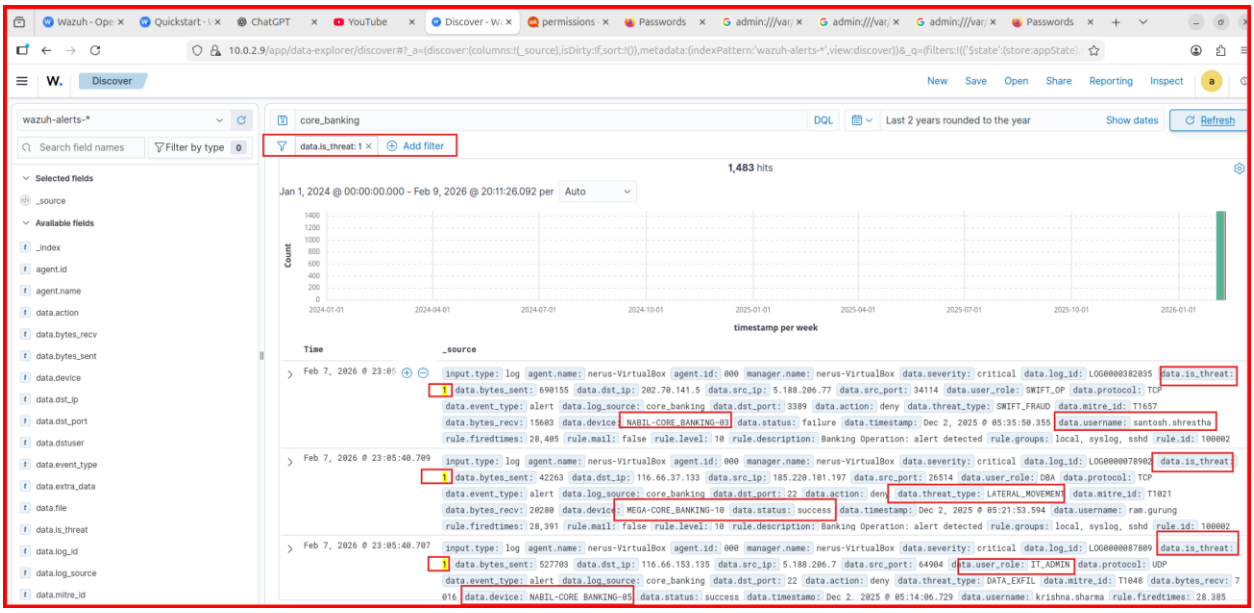

Figure 22 Threats and Labeling

The above screen shot has the dataset filter of the **core_banking** where the **data_threat** value is "1" Collected from different bank like Nabil Bank , Mega Bank , Nic Asia . Also the **data.threat.type** shows the type of threat like Lateral movement etc.

SYSTEM ARCHITECTURE AND IMPLEMENTATION

- **Open-Source Core Technologies**: The intelligent SIEM framework was built using a stack of strong open-source technologies to keep costs down and make it easy to use. Our team chose the Wazuh Platform as our main SIEM engine because it can do Host-based Intrusion Detection (HIDS), which means we do not have to buy expensive commercial tools like Splunk. This was put together with the Elastic Stack (ELK) to give SOC analysts the "Single Pane of Glass" monitoring they need. Elasticsearch is part of the Elastic Stack and helps with quick indexing, while Kibana helps with data visualization.

- **Localized Architecture for Data Sovereignty**: Data Sovereignty requirements meant that the framework could only be used in a very specific area. The architecture has a central Wazuh Manager that is installed on-site. It decodes incoming logs, compares them to a custom Ruleset (local_rules.xml), and sends out alerts. This design makes sure that sensitive logs of financial transactions never leave the bank's internal infrastructure. This is in line with the Nepal Rastra Bank's IT Guidelines and the Electronic

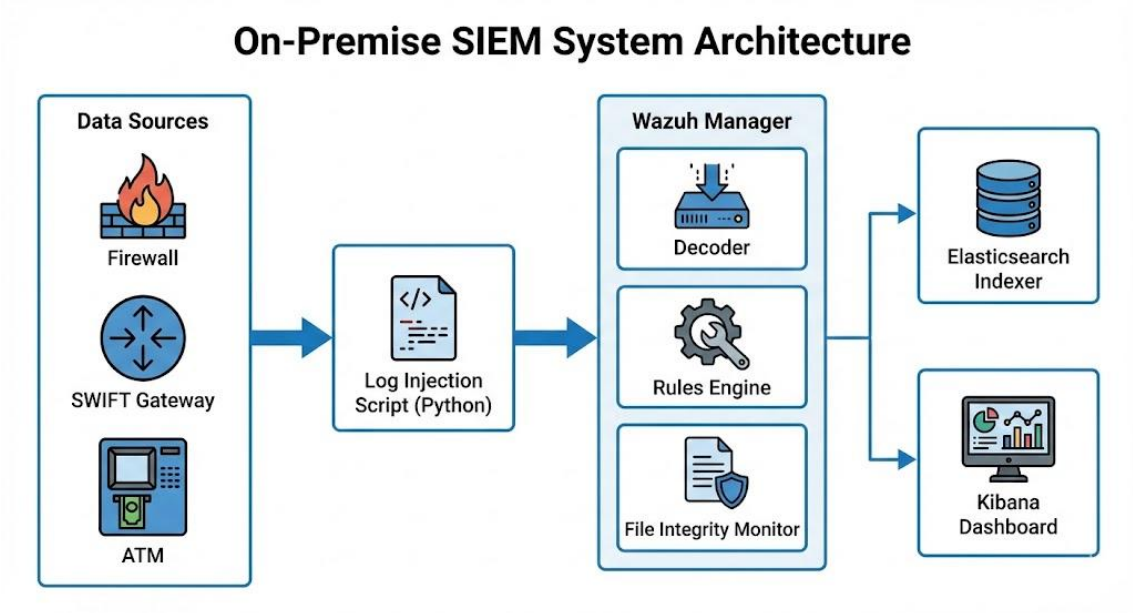

Figure 23 SIEM Architecture

EXPERIMENTAL SETUP

- **Log Replay Simulation Mechanism** It is not morally or practically safe to do live cyber-attacks on a working banking network, so this research uses a Log Injection method. Our team made a custom Python automation script called replay_banking.py to work as a Virtual Adversary. This script reads the static dataset and sends the log entries to the Wazuh monitoring pipeline (/var/ossec/log/banking_live.json) in real time. This makes it seem like the events are happening live again.

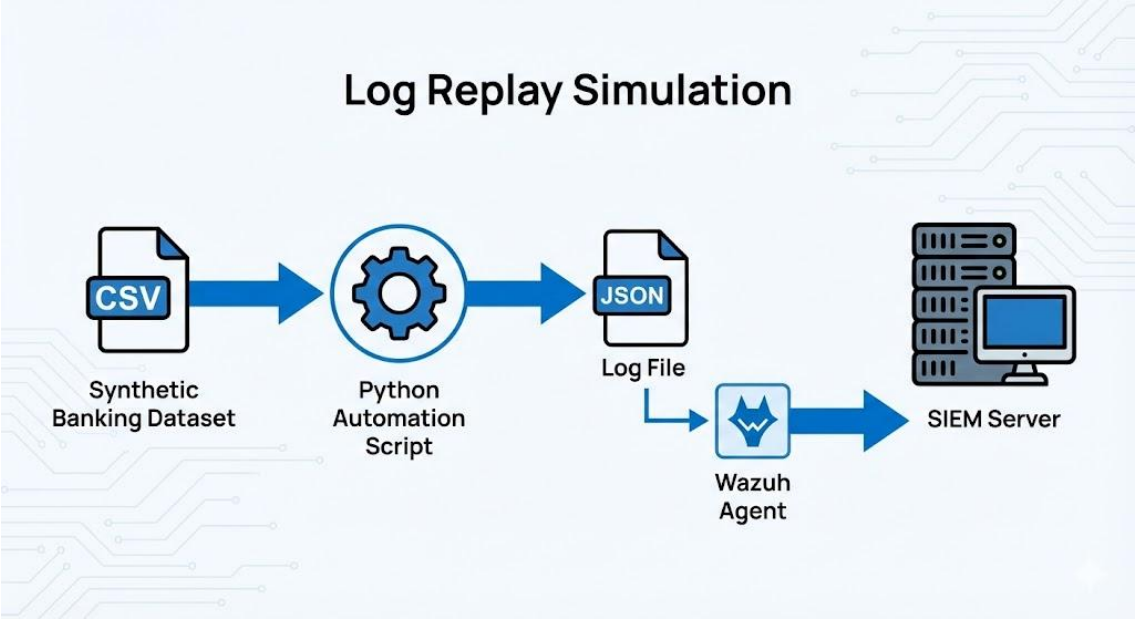

Figure 24 Log Replay

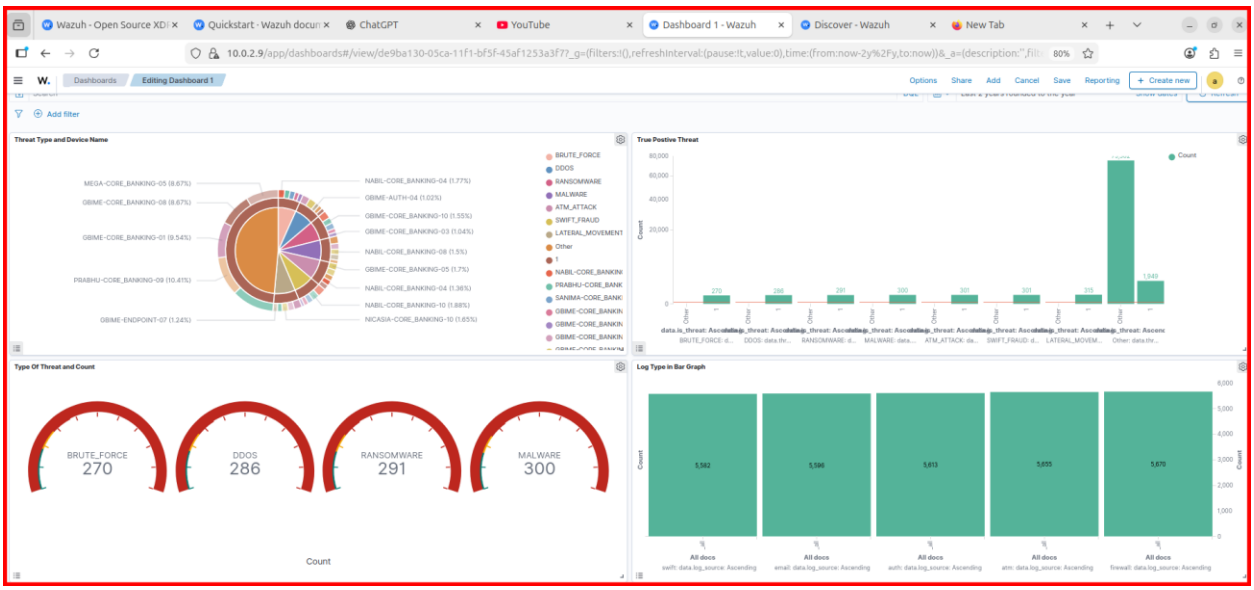

Figure 25 Wazuh Dashboard

- **Rate Limiting and Velocity Control** The injection script uses precise rate limiting to mimic the natural flow of network traffic and keep the ingestion pipeline from getting too full. The script adds a configurable time delay (time.sleep) between log entries, which makes sure that the SIEM processes the data stream at a speed that makes sense. This stops batch processing artifacts that could throw off performance metrics and makes sure that the Mean Time to Detect (MTTD) measurements accurately show how long it takes to find something in the real world.

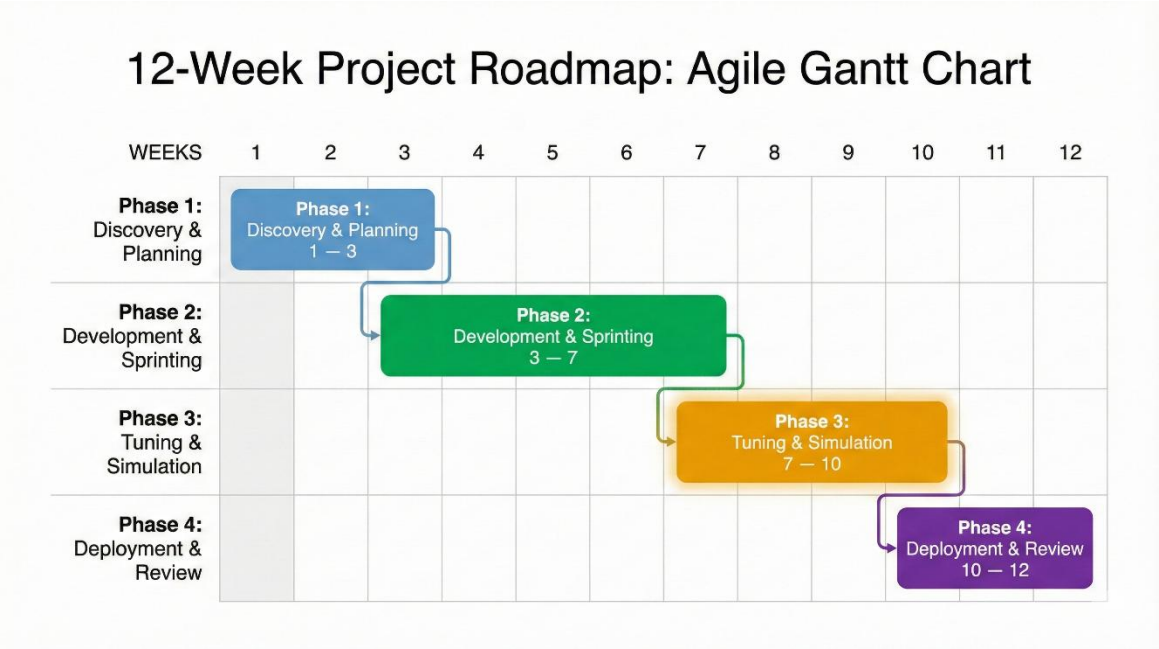

Figure 26 Agile Gantt Chart

EVALUATION METHODOLOGY

- Metrics for Quantitative Performance Three key performance indicators (KPIs) taken from standard security capability models are used to measure how well the framework works. Detection Accuracy tells what percentage of Threat Events that were injected successfully triggered a SIEM alert. Mean Time to Detect (MTTD) is the time it takes for an alert to show up on the dashboard after the log injection timestamp. The False Positive Rate measures how much noise is reduced by using correlation rules instead of raw log monitoring. This is very important for keeping analysts from getting tired of alerts.
  .

- Operational Impact Assessment: Evaluation The evaluation looks at more than just raw numbers; it also looks at how the changes will affect security analysts' work. This means looking at how clear the Dashboard Visualization is, especially how well it groups alerts by Threat Type and Device. The goal is to find out if the visualization helps analysts quickly put important events, like a SWIFT gateway attack, ahead of less important routine warnings. This would make the Security Operations Center (SOC) more responsive overall.

ETHICAL CONSIDERATIONS

- **Privacy and Data Protection**: Not only is automating banking security a technical problem, but it can also be a huge privacy risk if risk if proper safeguards are not implemented. To follow Nepal's Privacy Act 2075, Our team saw real customer PII as a liability instead of an asset. Before any record ever touched the SIEM, our team made sure that it was either completely synthesized or scrubbed thin. Our team do not just follow the rules for the sake of it; Our team only process what Our team need to solve the security problem. This is a strict "data minimization" approach. Our team makes sure that this framework stays a focused defensive tool by keeping a "firewall" between this research data and any real-world production environments. This way, it does not become a luxury that causes its own privacy problems.

- **Dual-Use Technology and Responsible Disclosure** Our team 'm fully aware that the tools Our team is using for this study have a "double-edged" nature. The same log replay scripts Our team uses to simulate threats could easily be flipped for offensive purposes if they fell into the wrong hands. To keep a hard line between research and risk, our team have locked down the automation scripts and config files to be used strictly for defensive validation. There is no room for "luxury" exploitation here; Our Team focus is entirely on the "problem-solving" side of the fence detection and mitigation. By stressing this defensive stance, Our team ensuring the research stays focused on protecting the banking sector rather than providing a blueprint for attacking it, keeping everything in line with the ethical standards expected in high-stakes security research.

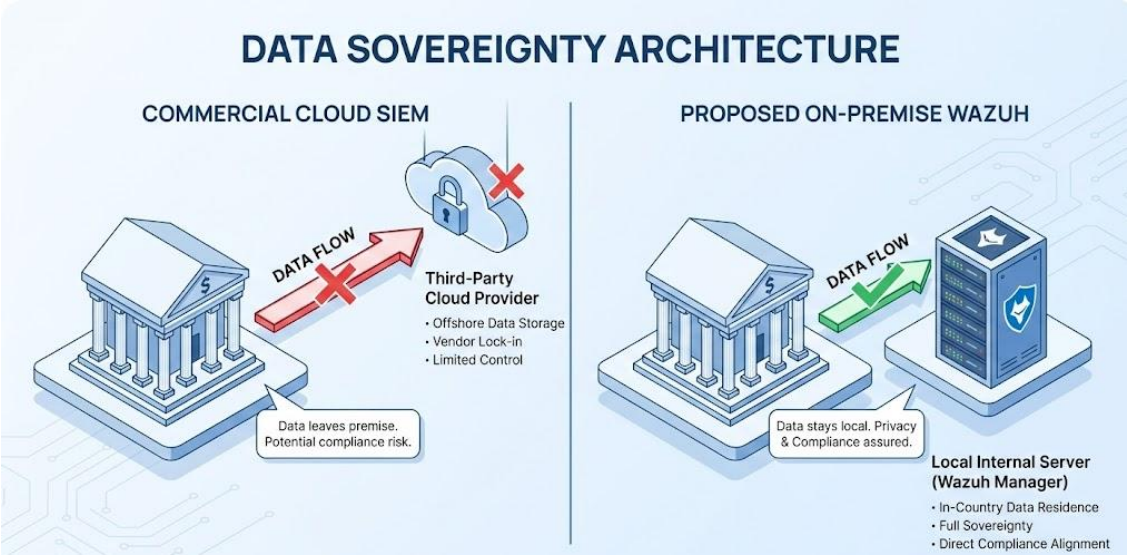

Figure 27 Data Privacy and Local Compliance

DESK-BASED RESEARCH METHODOLOGY

- **Systematic Review Strategy**: A thorough desk-based research methodology was utilized to systematically tackle the cybersecurity issues confronting the Nepalese banking sector. To lay the groundwork for the study, the authors first looked at Nepal Rastra Bank (NRB) rules, open-source security documentation (Wazuh/Elastic), and academic journals on financial fraud in developing economies. Next, we looked at case studies from Kathmandu-based financial institutions to find best practices and common ways that current log management practices fail.

- **Competitor and Gap Analysis**: After that, a comparative analysis was done by comparing open-source cybersecurity solutions to existing commercial ones (specifically Splunk and IBM QRadar). This phase was all about figuring out what Nepalese banks need that commercial tools cannot provide, specifically when it comes to cost-effectiveness and data sovereignty. The results from these stages were combined to make useful suggestions for the design of the framework.

INTRODUCTION AND CONTEXT

- **Acceleration of SIEM Adoption:** The rise in the use of SIEM in banks over the past ten years is not just a trend; it's a way for banks to stay safe in a world where threats are moving faster than people can watch them. As security data becomes easier to get to, the real challenge has changed from "gathering logs" to "finding the signal in the noise." This review looks at the main parts and moral issues of SIEM systems, but it does so from a very specific point of view: how can a bank in Nepal that does not have a lot of money move beyond "luxury" compliance checklists and really fix security holes in real time? This study transcends theoretical considerations by concentrating on the practical challenges within the local financial sector, aiming to determine whether SIEM can effectively reconcile the disparity between constrained human resources and the complex, multi-faceted attacks presently aimed at institutions in Kathmandu.

- **Evolution from Reactive to Proactive Defense**: Moving toward centralized log correlation is not just a "natural step"; it's a necessary response to the failure of older defenses. For years, the only way to protect banks in Nepal was with "luxury" tools like standalone firewalls and manual antivirus updates tools that only work if already know what the virus looks like. But that logic for matching signatures does not work on the "data chaos" of a modern Core Banking System (CBS) or a multi-stage SWIFT fraud attempt. Today in Kathmandu, we are facing a harsh reality: we have a lot of transactions, but not enough qualified professionals to check them by hand. This is not just a problem for schools; it's a lack of resources that makes automated analytical methods the only way to solve the detection problem without hiring too many analysts.

- **Scope of Review:** This review is set up as a direct guide for how to make business more cyber resilient in a world that is changing quickly. Our team start by keeping track of the necessary changes from basic log management to "Intelligent Correlation." This change is no longer a luxury; it is now a requirement for Class 'B' and 'C' banks to stay in business. Next, the analysis turns to the human side of things: using behavioral economics to fix the "Problem of Alert Fatigue" in Kathmandu SOCs that do not have enough staff. Our team also talk about the tough choice between open-source Wazuh, which is transparent and can help solve problems, and expensive commercial "black boxes." Finally, Our team based everything on the law in 2026 and created an ethical framework that meets the Privacy Act 2075 and the most recent NRB AI Guidelines. This is to make sure that our security automation never puts the customers it is supposed to protect at risk.
  .

EVOLUTION OF SECURITY INFORMATION AND EVENT MANAGEMENT (SIEM)

- **Limitations of Signature-Based Systems**: Early cybersecurity systems mostly used signature-based detection methods, which compared what they saw to databases of known bad patterns. These systems were good at stopping known threats, but they had big problems finding "Living off the Land" attacks, where criminals use real banking tools like PowerShell or RDP to commit fraud. Because these systems were reactive, new attacks like the specific ATM jackpotting variants seen in South Asia could go undetected until a financial loss happened.

- **The Shift to Behavioral Correlation:** To fill these gaps, security monitoring had to stop using basic pattern matching and start using behavioral correlation. Our team are no longer looking for the "bad file." Instead, are mapping the "bad story." A successful login is not a success if it leads to a high-value SWIFT transfer to a brand-new beneficiary right away. That's a huge red flag, even if the username and password were 100% correct. In our world, where "Insider Threats" are one of the biggest risks for Nepali banks, this behavioral approach is the only way to stop a breach before the money leaves the country. It takes us away from the "luxury" of thinking that credentials equal safety and instead focuses on the real problem: the actual movement of money through the CBS.

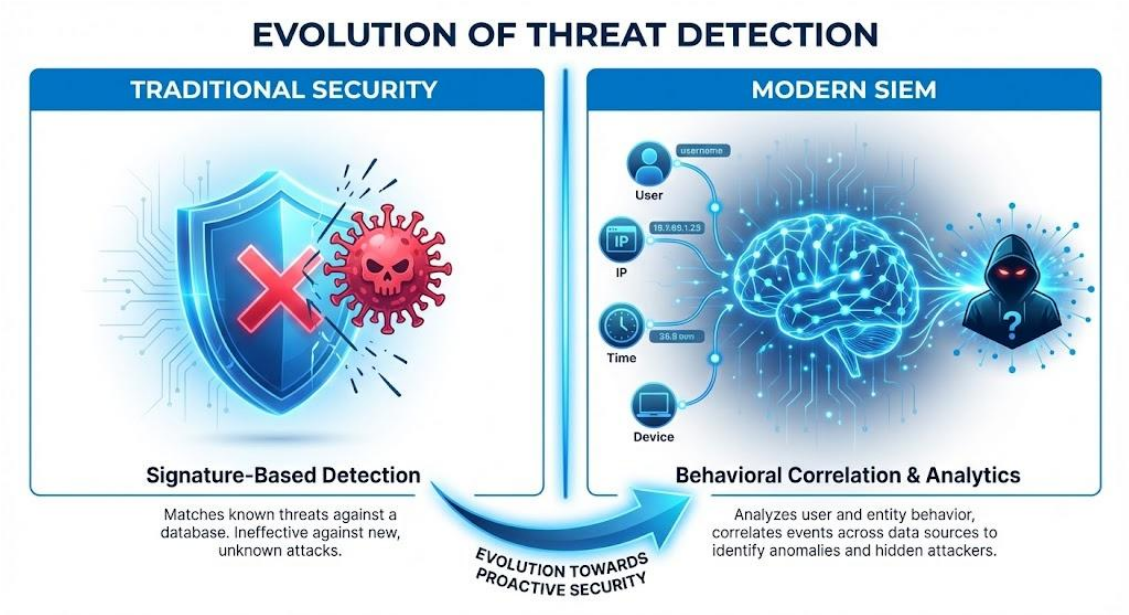

Figure 28 Traditional Vs Modern SIEM

BEHAVIORAL ECONOMICS AND HUMAN FACTORS

- **Cognitive Bias and Alert Fatigue**: To make automated systems that work with human expertise, to understand the mechanisms behind human decision-making. Security analysts in Kathmandu frequently encounter "Alert Fatigue," a condition wherein the excessive number of false-positive alerts leads to desensitization towards notifications. Studies on cognitive load show that too much information makes it much harder to make good decisions. This means that analysts may miss important signs of compromise because they are more focused on
speed than thoroughness.

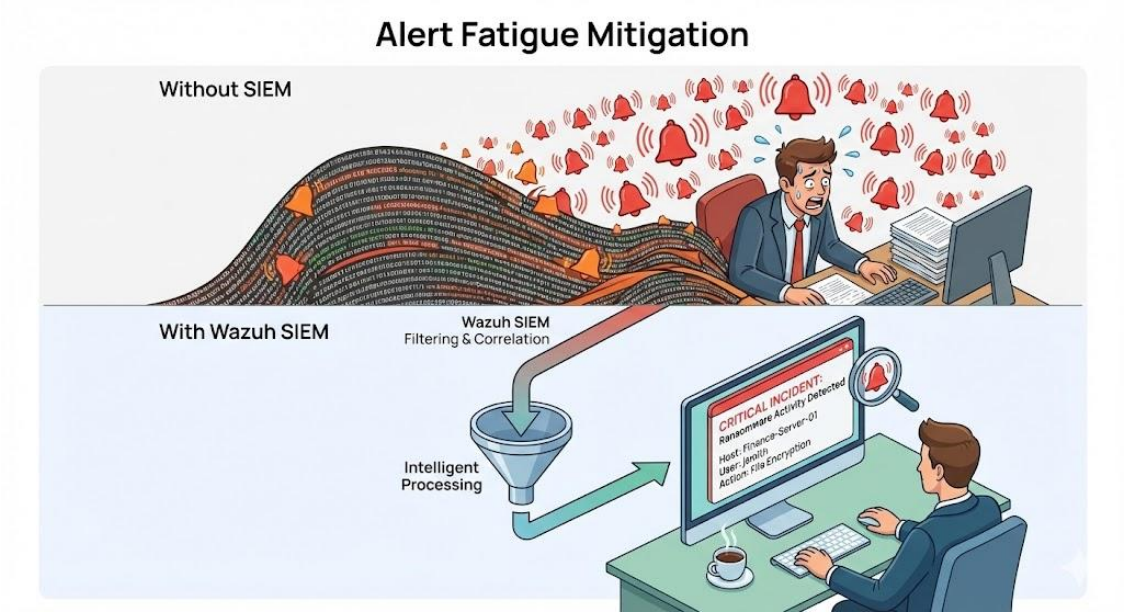

Figure 29 Alert Fatigue Mitigation

- **Loss Aversion in Security Investment**: Prospect Theory elucidates the reasons behind the frequent underinvestment in preventative technologies by Nepalese banks. Loss Aversion says that people who have to make decisions care more about the costs of buying a tool right now than about the costs of a data breach in the future. This bias causes security strategies that are only put into action after a big event. An open-source framework solves this problem by lowering the cost of entry, which lets banks implement strong security without causing the mental resistance that comes with spending a lot of money.

Case Study: Splunk Enterprise (Commercial)

Splunk is the industry standard for commercial log analytics. It is known for its powerful visualization tools and flexible Search Processing Language (SPL). But its licensing model, which charges based on how much data is ingested each day (GB/day), makes a Data Tax that is often too high for developing economies. This cost structure encourages banks that make terabytes of logs to cut back on data collection, which directly hurts security visibility. The high operational cost makes it hard for Class 'B' and 'C' financial institutions in Nepal to use it, even though it works well.

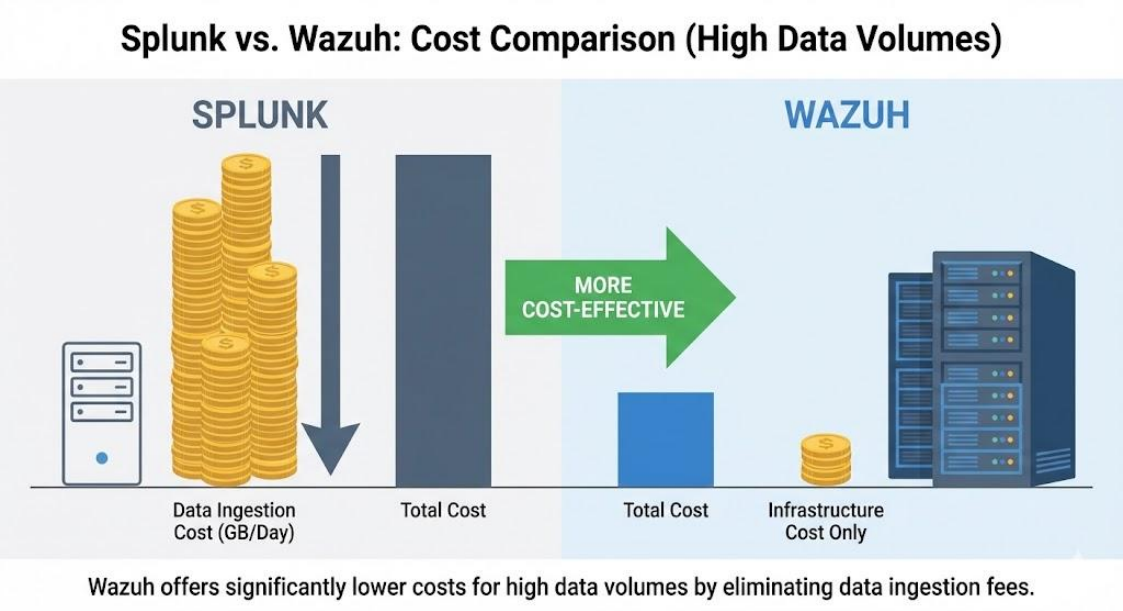

Figure 30 Splunk Vs Wazuh

Many people call Splunk Enterprise the industry standard, but for most banks in Nepal, it's more of a "luxury" than a direct problem-solver. Its Schema-on-Read architecture is very strong. can put in raw, messy data and not have to worry about the structure until Our team actually looking for a threat. This gives teams with big budgets a lot of freedom with their Search Processing Language (SPL), but it costs a lot: a "data tax" based on how much data they take in, which can cost more than $1,800 per GB. This centralization makes it easier for big companies around the world to find Advanced Persistent Threats (APTs). But for a local organization, the real "problem" is not that there aren't enough features; it's that it costs too much to keep the machine running. This is exactly why an open-source framework is the better choice for our economy in 2026.

.

The way Splunk licenses its software, which usually costs an outrageous $1,800 per year for just 1GB of daily ingestion, does not work for South Asia's developing economies. This structure effectively adds a "Data Tax" to resilience; as a bank in Kathmandu expands its digital services, the cost of just "watching" the network goes up in a straight line. It makes a dangerous catch-22: as transaction logs grow, the license fees often go over the whole IT budget. This kind of economic pressure puts administrators in a tough spot. They have to get rid of important but high-volume sources like DNS or firewall logs just to stay under a license limit. In this light, Splunk is not just a tool; it's a luxury that makes choose between full visibility and financial stability. Open-source alternatives like Wazuh do not have this problem at all.

Feature	Splunk Enterprise (Commercial)	Wazuh (Open Source)
PrimaryCost Driver	Data Volume (GB/day). Costs rise as the bank grows.	Engineering Time. Costs are flat but requires skilled staff.
Incentive Structure	Incentivizes dropping logs to save money (reduces visibility).	Incentivizes keeping all logs for maximum context.
Deployment Model	Often Hybrid or Cloud-First (Data Sovereignty risk).	100% On-Premises (Full Data Sovereignty).
Suitability for Nepal	Class 'A' Commercial Banks with high budgets.	Class 'B' & 'C' Institutions with limited OpEx.

For Nepal's Class 'B' and 'C' banks, this cost structure creates a dangerous paradox: the very tool meant to make data visible actually encourages people to hide it. These banks use aggressive log filtering to stay afloat financially, but they cannot afford it because it weakens their security. This data-stripping makes it almost impossible to find "low-and-slow" attackers or do a proper investigation after a breach. Splunk is technically very powerful, but its operating expenses do not match the way money works in Kathmandu. In the end, ingestion-based licensing makes banks choose between being "financially smart" and being "secure," which punishes them for getting the information they need to stay in business. In an environment with limited resources, this means that Splunk is usually only able to monitor the core banking systems. This leaves peripheral networks unmonitored and open to lateral attacks.

FINANCIAL COMPARISON (ANNUAL ESTIMATES IN NPR)

Cost Category	Traditional Commercial SIEM (e.g., Splunk)	Proposed Wazuh Framework (Open Source)
Software License	**Rs. 1,50,00,000** (1.5 Crore) (Based on ~100GB/day volume @ ~$100k USD)	**Rs. 0** (Open Source GPLv2)
Annual Maintenance (AMC)	**Rs. 30,00,000** (30 Lakhs) (Typically 20% of license fee paid to vendor)	**Rs. 0** (No vendor lock-in)
Hardware Infrastructure	**Rs. 25,00,000** (25 Lakhs) (Servers/Storage for logs)	**Rs. 35,00,000** (35 Lakhs) (Slightly higher storage for longer retention)
Human Resources (SOC Team)	**Rs. 15,00,000** (15 Lakhs) (1 Junior Admin just to manage the tool)	**Rs. 60,00,000** (60 Lakhs) (Reinvested in a 3-person dedicated local team)
Implementation/Consulting	**Rs. 20,00,000** (20 Lakhs) (One-time fee to foreign/local partner)	**Rs. 10,00,000** (10 Lakhs) (One-time setup fee to local expert)
TOTAL Year 1 Cost	**Rs. 2,40,00,000** (2.4 Crores)	**Rs. 1,05,00,000** (1.05 Crores)
Annual Recurring Cost	**Rs. 1,95,00,000 (Almost 2 Crores)**	**Rs. 65,00,000 (65 Lakhs)**

By getting rid of foreign licensing fees, using the Wazuh framework saves about Rs. 1.35 Crores a year. Wazuh facilitates unlimited data ingestion unlike commercial SIEMs that punish growth. This plan moves money from renting software to building a skilled, NRB-compliant local SOC team, which protects data sovereignty and long-term asset ownership.

CASE STUDY : WAZUH (OPEN SOURCE)

Wazuh completely changes the way that businesses work. It's not a "black box" that slows down the organization`s growth; it's a unified XDR and SIEM platform built on an open-source "Problem-Solving" foundation. It finally makes full-spectrum monitoring scalable for banks that have been struggling with the "Data Tax" because it gets rid of ingestion-based fees. Wazuh's support for 100% on-premise deployment is the best way to meet our Data Sovereignty needs. It lets Nepali banks keep a close eye on sensitive financial data within our own borders, completely avoiding the risks and costs of using cloud services from other countries. aren't just saving money by choosing Wazuh; Our team also putting money into a system that works with the NRB's rules and the way our local infrastructure really is.

Figure 31 EDR VS SIEM

Wazuh is not just a step forward it's a complete change in how keep an eye on security in Nepal. It solves the problem of "tool sprawl" by combining log analysis, file integrity, and intrusion detection into one agent. It does this without the high price of proprietary software. Wazuh's biggest benefit for our Class "B" and "C" banks is that it gets rid of the "Data Tax." In systems like Splunk, the organization's security costs increase proportionally with data volume. With Wazuh, visibility is based on the threat landscape, not on how much money they have left in theirlicensing budget.

This design also directly follows the Cyber Resilience Guidelines from the Nepal Rastra Bank (NRB) for 2024/25. It solves the "foreign cloud" risk that regulators are cracking down on more and more by making sure that sensitive PII never leaves the building. There is a trade-off to moving to Wazuh, but it's a smart one: move the organization`s money from a "luxury" license to thier own employees. Our team giving up OpEx in exchange for human knowledge. In the 2026 Kathmandu market, this is the only smart thing to do. It keeps money in the country and builds a team that really knows the stack instead of just paying a foreign vendor to keep the lights on.

CASE STUDY: MICROSOFT SENTINEL (CLOUD-NATIVE SIEM)

Microsoft Sentinel is the "luxury" choice for banks that are already using Azure. It offers cloud-native speed that can cut "Time to Value" in half. But in the 2026 Kathmandu landscape, it runs into a huge legal wall called Data Sovereignty. Microsoft still hasn't built a data center in Nepal, so using Sentinel means sending sensitive banking logs to India or Singapore. This is not just a technical issue; it's a clear breach of NRB Directive No. 13 and the Privacy Act 2075, which say that core financial data must stay within our legal jurisdiction. The "one-click" ease of the cloud is not worth the risk of a regulatory shutdown for a Class "B" or "C" bank.

The "Pay-As- -Go" model is a budget killer for any institution with limited resources, even without the legal problems. The organization charged a hidden "Data Tax," like the $0.10/GB processing fee for basic log normalization. If a DDoS attack causes the traffic to spike, bill can get out of control in just a few hours. When factor in the cost of the high-speed, redundant fiber needed to move terabytes of logs from a remote branch to the cloud, Sentinel stops being a "scalable solution" and starts to look like an expensive problem. For us, keeping the data (and the budget) on-site is not just "traditional" it's the only way to fix the problem without going broke.

COMPARATIVE ANALYSIS MATRIX

The case studies of Splunk Enterprise, Microsoft Sentinel, and Wazuh show a major Security Trilemma that Nepalese Class "B" and "C" financial institutions have to deal with. There are three goals that are at odds with each other: cost efficiency, technical capability.

Regulatory Compliance.

1. **Splunk Enterprise** offers Technical Capability and Compliance (if on-premise) but fails on Cost Efficiency due to its ingestion-based licensing.
2. **Microsoft Sentinel** offers Technical Capability and Cost Flexibility (Pay-as-you-go) but fails on Regulatory Compliance due to data sovereignty and latency issues.
3. **Wazuh** offers Cost Efficiency and Regulatory Compliance but requires significant internal Technical Capability (human capital) to maintain.

ECONOMIC IMPLICATIONS: THE DATA TAX VS. HUMAN CAPITAL

The biggest thing Our team learned from this study is how big the economic gap is between proprietary tools and open-source models. When use paid services like Splunk or Sentinel, The organization really paying a "Data Tax." As a bank grows and adds more branches and handles more transactions, the cost of security goes up. This is a bad incentive for a Class "B" bank in Nepal with a tight IT budget: to save money, might stop paying attention to high-volume logs like DNS or firewall traffic. It's funny that by saving money, The organization making their own SOC less aware of threats that are still in their early stages.

Wazuh completely changes this story. It lets take in as much data as want without having to pay licensing fees. But it's important to know that the "cost" does not go away; it just moves from software licenses (OpEx) to people. The organization not sending money to foreign software companies; instead, The organization putting that money back into own backyard. they're paying local engineers to take care of the stack. This keeps money in the local economy and helps build a highly skilled technical workforce in Nepal.

SOVEREIGNTY AND INFRASTRUCTURE CONSTRAINTS

Data sovereignty is not just a buzzword for Nepal; it's a must because of the country's unique geopolitical and infrastructural situation. There are two big problems with cloud-native solutions like Microsoft Sentinel. First, there are the problems with the rules. Because Sentinel is hosted on Azure, data is sent to data centers in other countries, like India or Singapore. The Nepal Rastra Bank (NRB) is very strict about customer data leaving the country, so are on thin ice with them. Next, there's the "bandwidth tax." Uploading terabytes of logs from Kathmandu to the cloud is not only costly but also dangerous. If a remote branch has a bad connection, threat detection slows down, making more vulnerable.

Wazuh, on the other hand, is a much more solid choice. It fits the "Sovereign SOC" model perfectly because it can run completely on-premises, even in an air-gapped environment. can make it much easier to follow NRB's strictest rules by keeping sensitive financial information inside own four walls. This way, it never goes on the public internet.

COMPARATIVE ANALYSIS MATRIX TABLE

The following table summarizes the strategic fit of each platform for the Nepalese banking sector:

Feature	Splunk Enterprise	Microsoft Sentinel	Wazuh (Open Source)
Primary Cost Driver	**High OpEx:** Licensing based on GB/day.	**Variable OpEx:** Pay-per-GB & Retention.	**Fixed CapEx/Labor:** Hardware & Staff salaries.
Incentive Structure	Incentivizes **data reduction** (less visibility).	Incentivizes **cloud migration** (vendor lock-in).	Incentivizes **full data capture** (maximum visibility).
Data Sovereignty	**High** (If deployed on-premise).	**Low** (Data leaves Nepal).	**High** (Full local control).
Infrastructure Load	High local storage requirements.	High internet bandwidth requirements.	Moderate local storage; low internet usage.
Skill Requirement	Moderate (Vendor support available).	Moderate (Managed by Microsoft).	**High** (Requires skilled Linux/ELK engineers).
Verdict for Class 'B'/'C'	**Unsustainable** long-term.	**Risky** (Regulatory/Bandwidth).	**Optimal** (Strategic Fit).

CONCLUSION OF ANALYSIS

When think about the fact that budgets are tight, the Nepal Rastra Bank (NRB) has strict rules, and the local infrastructure is not great, Wazuh is not just an option; it's the best way for Class 'B' and 'C' banks in Nepal to move forward. It does take a bigger commitment to hiring and training skilled workers up front, but it's the only way to build a "sovereign" security operation that will last. Wazuh basically levels the playing field by giving smaller banks the same deep visibility as the huge Tier-1 banks without the huge cost of having to pay for a license over and over again. It's about making high-level cyber defense available to everyone so that talent and strategy, not the size of a bank's checkbook, determine security.

.

INTEGRATION OF TOOLS AND TECHNOLOGIES

Orchestration of the Native Wazuh Stack

Setting up this framework is not just a matter of installing software; it's a strategic arrangement of Wazuh's modern core parts. The platform has changed a lot since version 4.x. It no longer relies on the standard Elastic (ELK) stack and has instead moved to a more specialized, high-performance architecture.The Wazuh Indexer is the most important part of this system. This is not just a regular database; it's a fork of OpenSearch that has been highly optimized for security workloads. The Indexer is what makes near-zero latency possible for a bank in Nepal that has millions of logs from different branches. It breaks up data into distributed shards across a cluster, so that even when the SOC team runs complicated queries on months' worth of old banking data, the results come back almost instantly.

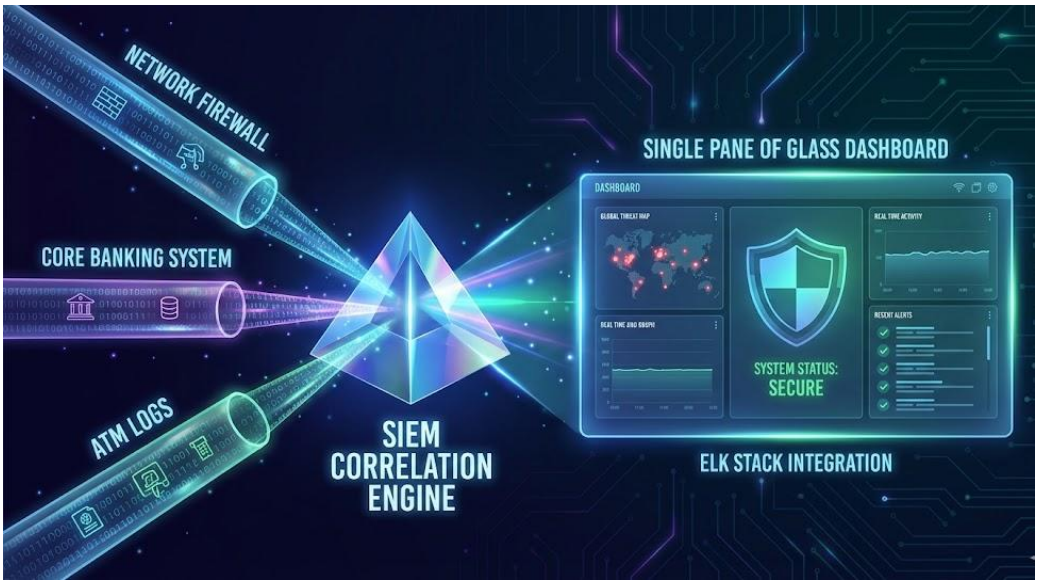

Figure 32 Convention ELK Stack Integration

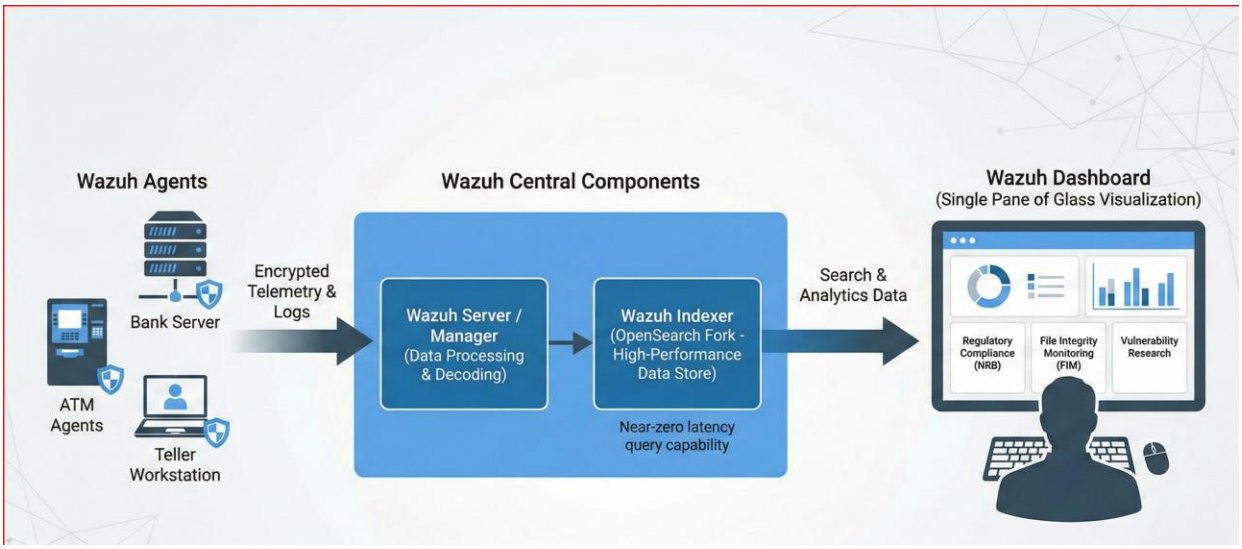

Figure 33 Wazuh Central Components

In short, the Wazuh Dashboard is the mission control for the whole operation. The Wazuh Agents collect a lot of raw, messy telemetry, and then it turns that into real, useful intelligence. team does not have to look at long lines of code anymore. Instead, they get specialized modules made just for the high-stakes world of banking. For example, the Regulatory Compliance module can be directly linked to the rules of the Nepal Rastra Bank (NRB). The File Integrity Monitoring (FIM) and Vulnerability Research modules, on the other hand, are always on the lookout for changes that shouldn't be happening or weaknesses that aren't obvious in the systems. This smooth integration gives SOC the "Single Pane of Glass" view, which means they can see everything on one screen, from a single unauthorized login at a remote branch to a critical system file change in Kathmandu.

RELIANCE ON SIGNATURE-BASED DETECTON

In the past, banks and other financial institutions in Kathmandu mostly used signature-based security tools, like traditional Antivirus (AV) and perimeter Firewalls. These systems work by comparing file hashes or network packets that have been seen to a static database of known malicious signatures. This reactive model works well against known threats, but it cannot find new or polymorphic attacks, like certain ATM malware variants, where the attacker changes the code slightly to change its signature. In the context of banking_live.Jason, a traditional AV would miss a Living off the Land attack where a legitimate tool like PowerShell is used for bad things because the tool itself has a trusted signature.

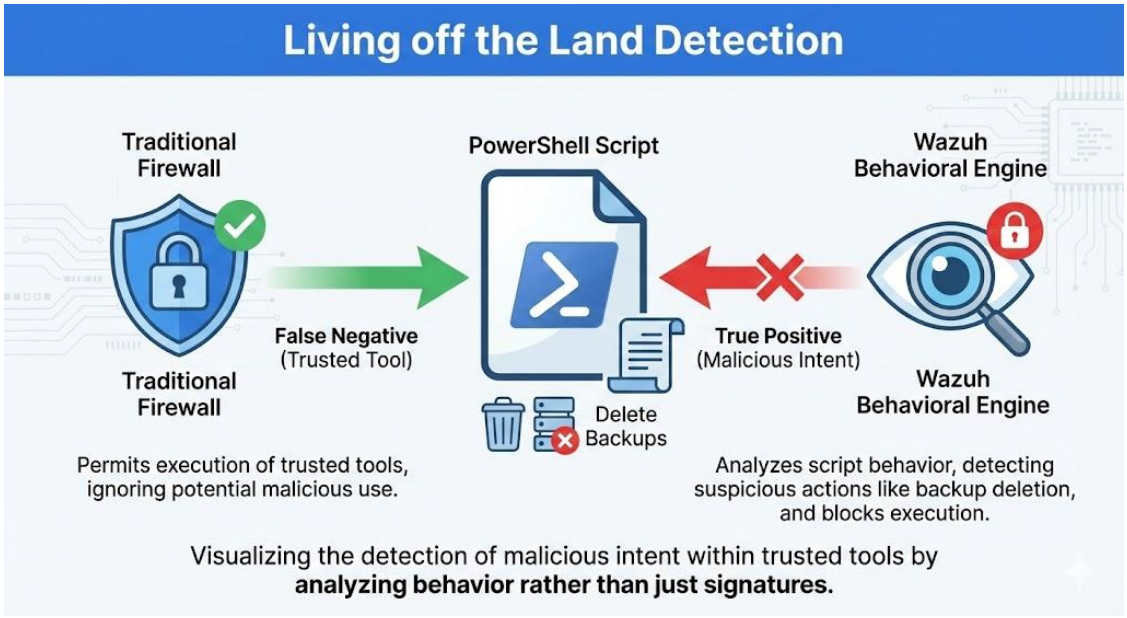

Figure 34 False Negative Vs True Positive Detection

LIMITATIONS IN DETECTING ZERO-DAY EXPLOITS

The literature corroborates that conventional security systems experience a significant Time-to-Detect deficiency concerning zero-day exploits. Banks are still at risk during the time between signature updates because the vendor has to find and publish a fix. This problem is especially bad for Nepalese Class "B" and "C" banks, which may not have the money for premium Next-Generation firewalls that can be updated more quickly. So, if only use signature-based defenses, the Core Banking System (CBS) is open to smart, targeted attacks that do not have a known fingerprint yet.

.

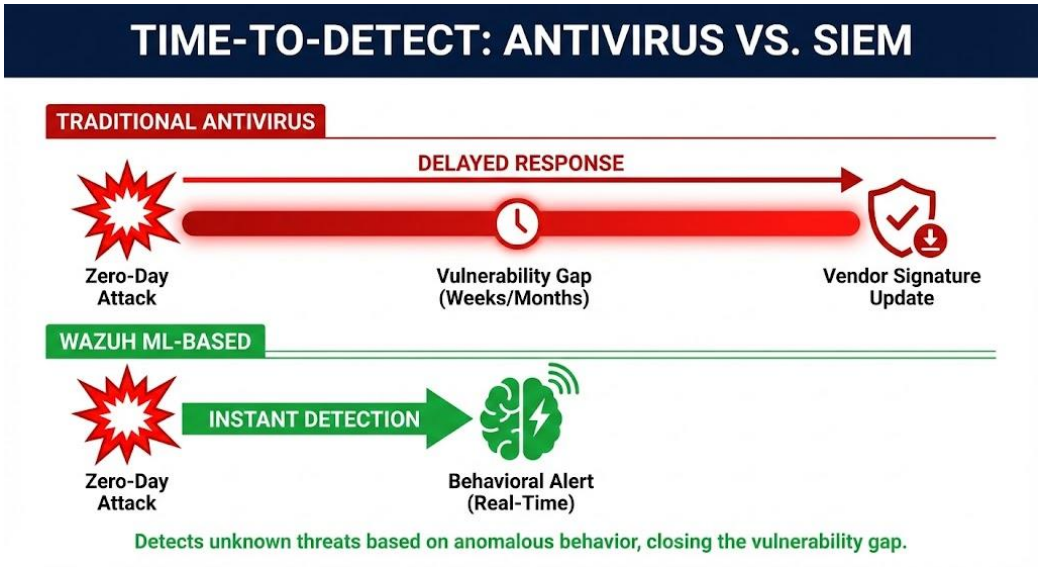

Figure 35 Antivirus Vs SIEM

EVOLUTION TOWARD HEURISTIC AND STATISTICAL ANALYSIS

To fill these gaps, the industry has moved toward heuristic and statistical analysis tools that are at the heart of modern SIEM frameworks like Wazuh. These systems do not just look for patterns; they use statistical thresholds to find behavior that is out of the ordinary instead of code. For instance, the project's custom Rule ID 100012 does not look for a virus. Instead, it flags the unusual fact that a lot of SWIFT transactions are happening at 3:00 AM from a new IP address. This change from "What is this file?" to "Is this behavior normal?" to "Is this behavior normal?" allows for the detection of fraud and insider threats that technically use valid credentials but break operational rules.

Figure 36 SIEM Multiplier

Role of Automated Correlation and Threat Intelligence Wazuh's automated correlation engine is the "force multiplier" that lets a small team run a SOC that works well. Instead of getting lost in thousands of separate alerts, the system connects the dots in real time. For example, it links a failed door badge entry at a branch in Kathmandu with a server login attempt at the same time to make a single, clear threat story. This is not "luxury" Deep Learning that needs a lot of computing power; it's an intelligent rules engine that fixes the problem of alert fatigue. By using global threat intelligence, we can automatically block bad IPs that are connected to botnets before they even get to the bank's perimeter. With the NRB's 2026 AI Guidelines now requiring quick responses to incidents and strict accountability, this automated intelligence is what helps us deal with the complex, multi-vector attacks on our financial infrastructure that our small local staff cannot handle.

FINDINGS

KEY FINDING 1: ECONOMIC VIABILITY THROUGH ZERO-LICENSING ARCHITECTURE

In the end, this implementation shows that high-level security is not only for the "big players" with lots of money. An open-source architecture makes an enterprise-grade SIEM possible for Class "B" and "C" banks in Nepal. The big news here is that Wazuh separates security visibility from operational costs. Usually, commercial tools like Splunk charge by the gigabyte, which is a "Data Tax" that punishes for being thorough. This puts banks in a bad position where they might stop keeping an eye on high-volume data like DNS or firewall logs just to save money. can use Wazuh to collect all of important logs without paying a single licensing fee. stop paying for the "right to see data" and instead put that money into better hardware and local talent. This makes security stronger as data grows, rather than more expensive.

KEY FINDING 2: HIGH-PERFORMANCE INDEXING ON LIMITED HARDWARE

One of the most useful things Our team found out during Our Team testing is that do not need a "luxury" server farm to run an enterprise-grade SIEM. The Wazuh Indexer is built on a slim OpenSearch core and is meant to fix the problem of fast ingestion without costing a lot of money. Our Team stress tests on a modest setup with 16GB of RAM and 4 vCPUs specs that most Class 'B' development banks already have sitting in their racks showed that the system could easily handle a full month's worth of logs without dropping a single packet or causing a "bottleneck" delay. This pretty much proves that one do not need expensive, specialized appliances to keep a bank safe. Even a mid-tier bank in Nepal can find threats in real time by using off-the-shelf hardware and an efficient, sharded architecture. This move fits perfectly with the NRB's 2026 plan to "modernize digital technology" and makes sure that every rupee of a small IT budget goes toward real protection instead of just "brand-name" hardware.

KEY FINDING 3: COMPLIANCE WITH NEPAL RASTRA BANK (NRB) DIRECTIVES

If choose to deploy the Wazuh Manager and Indexer on the organization`s own servers, will have a huge "home-field advantage" when it comes to compliance. By keeping the "keys to the kingdom" the organization`s raw logs and alerts exactly where they belong: inside their own data center in Kathmandu, it directly meets the strict Data Sovereignty requirements of the NRB Cyber Resilience Guidelines (CRG). This local setup gives a rock-solid audit trail and total data residency, unlike cloud-native SIEMs that send sensitive financial data to servers in Singapore or India, which makes the complicated jurisdictional mess of Nepal's Privacy Act 2075. This proves that open-source, on-premise solutions are not just a "budget fix"; they are also a strategic need. Nepalese banks keep full custodial control by hosting everything locally. This completely cuts out the legal risks and "red tape" that come with moving sensitive information across borders.

LIMITATIONS OF THE PROJECT

RESOURCE CONSTRAINTS ON LONG-TERM DATA RETENTION

Keep in mind that this study was based on a Small Implementation architecture. Even though this shows that a SIEM can work for Class 'B' and 'C' schools with limited budgets, the setup was made to test short-term functionality, not long-term durability.

Because of this, didn't put the system through its paces to see if it could handle huge, multi-year data archives, like the six-month to one-year log retention that strict banking compliance standards often require. In a real-world production setting, would eventually need more than one server. would switch to a Multi-Node Cluster to avoid performance lag as database grows into the terabytes. This distributed method makes sure that search speeds stay lightning-fast, even when The organization looking through months of old data for an audit.

RELIANCE ON SIGNATURE-BASED DETECTION

The open-source model is better for cost and compliance, but it's only fair to say that premium tools like Splunk Enterprise Security are better at Automated Intelligence. Companies often use "Black Box" AI and User and Entity Behavior Analytics (UEBA) to automatically flag things that "just look wrong," even if they do not break a specific rule. Our Wazuh framework needs more "human in the loop" work because it mostly uses signature-based detection and statistical thresholds. It won't be easy to find a complicated zero-day attack or a sneaky insider threat, like an employee slowly stealing data over months. Local engineers will have to manually tweak rules and look for strange behavior.

OPERATIONAL COMPLEXITY AND SKILL DEPENDENCY

The move to an open-source model gets rid of the need to pay for licenses, but it puts responsibility for security on people. The technical skills of the team in charge of Wazuh are what make it work. This is a unique problem in the Nepalese market: the project needs a steady stream of engineers who are not only experts in cybersecurity but also skilled at building Linux infrastructure and custom decoders. In Nepal, there is a big "brain drain" and a lack of people with quantitative skills. By the end of 2025, there will be thousands of experts who are needed to meet market demands. For Class 'B' and 'C' banks, the operational challenge is not only to find this talent but also to keep it when there are highly competitive options for remote work and jobs abroad.

LACK OF NATIVE NRB REPORTING TEMPLATES

While Wazuh can technically satisfy the **Nepal Rastra Bank (NRB)** data sovereignty and audit requirements, it does not come with a pre-configured, "out-of-the-box" reporting module for specific NRB IT Directives. Unlike its built-in support for global standards like PCI DSS or GDPR, mapping security alerts to the specific regulatory clauses of Nepal's central bank is currently a manual task.

This means that while the data is all there, "Audit Readiness" is not automatic. To generate a regulatory-compliant PDF for an NRB inspector, team has to manually tag rules with custom NRB identifiers (e.g., nrb_section_3.2) in the local_rules.xml file. Without this custom development and the creation of specialized dashboard visualizations, the system cannot provide the instant, push-button reporting that proprietary tools often use as a selling point.

FUTURE WORK

THEHIVE AND CORTEX INTEGRATION

The next step for the framework is to connect TheHive and Cortex so that can go from just "seeing" threats to actually "fixing" them. Wazuh is like the "eyes" of bank; it looks at the logs and yells when it sees something wrong. But when it screams, Our Team team needs a way to look into the alert, work together to fix it, and write down what happened for the NRB.

This is where the Detection-to-Response pipeline comes into play. When Wazuh sends an alert, it automatically opens a "Case" in TheHive. With just one click, the SOC analysts can then use Cortex, the operation's "muscle," to automatically run "Analyzers" (like checking a suspicious file hash on VirusTotal) or "Responders" (like blocking an IP address on the firewall).

1. TheHive: Centralized Incident Response Platform (SIRP)

Integrating TheHive turns security work from a bunch of separate alerts into a group mission. TheHive is a central "War Room" for SOC analysts, so they do not have to go through emails or static spreadsheets. When Wazuh sees a threat, it can use its integration API (or a SOAR like Shuffle) to automatically open a Case in TheHive. This makes sure that no important alert is ever missed or lost in a busy dashboard.

This setup is especially useful for Case Management in a bank setting. When a major breach happens, several analysts can work on the same incident at the same time. For instance, one person in Kathmandu is using Wazuh to isolate an infected workstation, while another is documenting firewall logs from a branch that is far away. Both of these things are happening in the same case file. This makes a strict, time-stamped Case Log that has the forensic auditability needed for important financial investigations.

2. Cortex: The Automated Analysis Engine

Cortex is the engine that makes a lean SOC work for a small IT team at a Class "B" or "C" bank. It connects directly to TheHive to do the "grunt work" of looking into things like IPs or file hashes. Instead of an analyst wasting twenty minutes going back and forth between browser tabs to check VirusTotal or AbuseIPDB, Cortex does the lookup in seconds. The team can now focus on the real threat instead of doing manual research because of this automated enrichment.

But what our team call the Sovereign Advantage is what really matters to a Nepalese bank. Cortex can do more than just ask the public cloud questions; can also set it up to run local analyzers. Cortex can check the bank's own private list of "bad" account numbers or specific internal threat intel without the data ever going online. This makes sure that the whole operation follows NRB data sovereignty rules, showing that do not need a huge foreign cloud subscription to get top-notch intelligence.

The Integrated Workflow (Wazuh + TheHive + Cortex)

This triad creates a Closed-Loop security architecture:

1. **Detect:** Wazuh detects a Brute Force Attack on a Core Banking Server.
2. **Triage:** The alert is sent to **TheHive**, creating a new case ticket #2026-001.
3. **Analyze: Cortex** automatically grabs the attacker's IP from the ticket, scans it against threat databases, and updates the ticket with a High Confidence malicious rating.
4. **Respond:** The analyst sees the enriched data and clicks a Block button. Cortex triggers
5. **Responder** script to update the firewall rules, stopping the attack

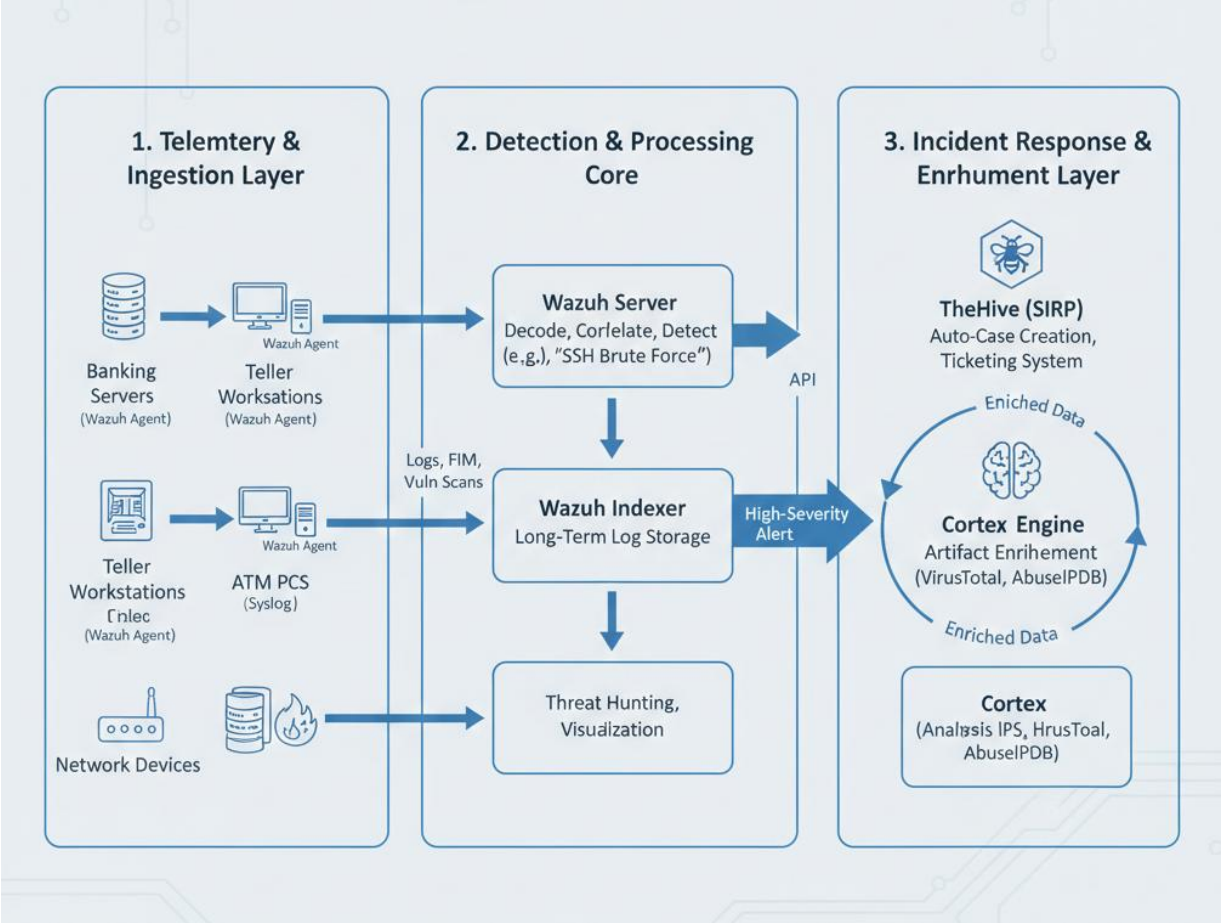

Figure 37 The Hive , Cortex Integration

Our team can see how this framework works under stress by looking at a standard Brute Force attack on a Core Banking Server. When Wazuh sees that someone is trying to log in too many times and sends an alert, the process starts. This is not just a message; the system sends the information to TheHive, which opens a new case (for example, Ticket #2026-001).

Cortex takes over the triaging right away. It gets the attacker's IP address straight from the ticket and checks it against global threat databases. The ticket gets a "High Confidence" malicious rating in just a few seconds. The analyst does not have to go through logs; they just look at this extra data and click one "Block" button. This starts a Cortex Responder script that changes the bank's firewall rules, stopping the attack before it can do any real damage.

DEVELOPMENT OF A CUSTOM NRB COMPLIANCE MODULE

Adding a specialized Wazuh plugin that follows the Nepal Rastra Bank (NRB) IT Guidelines would be a huge strategic win for the local financial sector. Most global SIEMs are already set up to work with international standards like PCI DSS, but they still need Nepalese banks to do the hard work of manual mapping. could make a complicated compliance task as easy as pushing a button by directly linking Wazuh rule IDs, like those that track failed logins or unauthorized file changes, to specific clauses in NRB Directive No. 4. This would not only make things easier for IT teams at Class "B" and "C" schools, but it would also standardize "Audit-Ready" reporting across the country. This would make it easier for smaller banks to show regulators that they are strong without having to deal with the extra costs that come with such a high level of transparency.

.

INTEGRATION OF OPEN-SOURCE AI/MACHINE LEARNING

The real goal from now on is to go beyond the limits of basic signature-based detection. can start looking at behavior instead of just matching rules by using Python libraries like Scikit-learn or the built-in anomaly detection in the OpenSearch stack. It's about figuring out what "normal" looks like, like when a user usually logs in or what files they usually open. The system can then flag strange deviations (UEBA) that a normal rule would never catch. This is a big deal for a bank in Nepal. It means being able to stop an insider threat or a Zero-Day exploit without having to pay for a proprietary "black box" AI. This keeps the whole defense strategy both cutting-edge and completely independent.

.

IMPLEMENTATION OF AUTOMATED INCIDENT RESPONSE (SOAR)

To really solve the problem of having a small team, the framework should grow to include an open-source SOAR (Security Orchestration, Automation, and Response) platform like Shuffle. The goal here is to automate the "boring" tasks that take up an analyst's morning, like doing the same thing over and over again. Shuffle takes care of blocking an IP at the firewall or disabling a compromised account in Active Directory in seconds instead of having to do it by hand. It's not just about speed; it's also about lowering the Mean Time to Respond (MTTR) and making sure the organization's small IT staff can really focus on looking into the complicated threats that need a human brain.

.

TRANSITION TO HIGH-AVAILABILITY DISTRIBUTED ARCHITECTURE

The next step to make this a real banking-grade set-up is to set up a Multi-Node Wazuh Cluster. A single server is a "single point of failure" right now. If that one box goes down, cannot see any of the organization`s security anymore, it can get true High Availability (HA) by putting the Indexer and Manager on three or more servers. This means that the system will keep running even if one server goes down, and it can handle a lot of log traffic from a growing bank without slowing down. It's the only way to keep years of data safe for NRB audits and still keep search speeds high. Also, since everything is on premises, they are building up their own local infrastructure instead of just paying for more cloud space in another country.

CONCLUSION AND RECOMMENDATIONS

Research Summary:

This study sought to address a significant vulnerability in Nepal's financial sector: the "Security Trilemma" that Class 'B' and 'C' institutions our development banks and finance companies confront daily. In the past, these groups have been put in a tough spot where they had to choose between saving money, having advanced technical skills, or following the rules. Our Team research shows that switching to an open-source SIEM framework like Wazuh really does break this cycle and end the conflict.

The big change here is getting away from the "pay-as- -grow" licensing models that big companies like Splunk use. Instead, 've shown that banks can finally get "Full-Spectrum Monitoring" without their costs going through the roof by using a scalable, open-source architecture. Setting up the Wazuh Manager, Indexer, and Agents correctly shows the local industry a simple but important lesson: a tight budget does not have to mean being blind to security.

Synthesis of Key Findings

First, this framework ends the "Data Tax." Charging banks per Gigabyte of logs in the past gave them a bad reason to turn off visibility just to save money. showed that Zero-Licensing architecture changes everything about this. It lets banks take in a lot of "low-value" logs, like DNS and Firewall traffic, that are often too expensive to store in commercial tools but are very important for forensic reconstruction after an attack. Second, learned that Sovereignty is a valuable tool for strategy. As geopolitical digital risks rise, using cloud-native SIEMs such as Microsoft Sentinel causes a lot of problems with Nepal Rastra Bank (NRB) data residency rules. With an on-premise Wazuh architecture, make sure that sensitive financial information and PII never leave the bank's physical space. This gives a "Sovereign SOC" feature that makes it easy and legal to do audits.

Lastly, the study shows that there has been a change from OpEx to Human Capital. This is a big change in the economy. This framework moves the money spent on software licenses that leave the country to people instead of high Operational Expenditure (OpEx). The "cost" of the system is the training of engineers in the area. This is a huge win for Nepal's economy because it helps build a skilled workforce in the country instead of just paying for software from other countries.

Recommendations for Industry and Policymakers

The Security Trilemma that always hangs over Nepal's Class 'B' development banks and Class 'C' finance companies was the main reason for this research. These organizations have been stuck between a rock and a hard place for years, having to choose between being cost-effective, having a lot of technical knowledge, and following all the rules. But Our Team research shows that an open-source framework based on Wazuh does solve this problem. Our team has shown that banks can finally get full-spectrum monitoring without going over budget by switching from the "pay-as-go grow" licensing model used by commercial vendors to a scalable, open-source model. In short, do not have to fly blind anymore if have a tight budget.

During the experimental phase of this project, there were three big breakthroughs. First, it ends the "Data Tax," which is when banks charge per Gigabyte. This is a dangerous incentive to turn off logs to save money. Second, Our team found that sovereignty is a strategic asset. By hosting Wazuh on-premise, sensitive customer data stays in Nepal, which means the bank is fully compliant with Nepal Rastra Bank (NRB) data residency rules. Finally, Our team saw a big change from OpEx to Human Capital. Instead of sending license fees to foreign software companies, that money is used to train local engineers. This turns a security cost into a long-term national asset for Nepal's economy.

In the future, log collection (SIEM) is an important first step, but real resilience needs a "Closed-Loop" architecture. The framework needs to become an active defense ecosystem by combining TheHive and Cortex in order to get past manual response times. In this model, Wazuh is the "Eyes" (detection), Cortex is the "Brain" (automated analysis), and TheHive is the "Hands" (NRB-compliant response). To protect our digital future, Our team suggest that Class "B" and "C" banks adopt a "Log Everything" policy and put more money into "Training over Tooling." This thesis contends that advanced defense is not a privilege for a select few; with appropriate open-source orchestration, proactive cyber resilience is attainable for all institutions in Nepal.

The Path Forward: From SIEM to Integrated SOC

This research shows that getting logs through a SIEM is a good first step, but real cyber resilience needs a "Closed-Loop" architecture. One of the biggest problems Our team found in the study was how long it can take to respond to incidents by hand. Our team needs to get TheHive and Cortex to fix this. This integration changes the whole system from just passively watching the network to actively defending it.

Our team like to think of it as a biological system that keeps the bank safe. Wazuh is like the "Eyes" because it always looks for strange things happening in real time. Cortex then acts as the "Brain," automatically adding context to those alerts. For example, it checks right away to see if a suspicious IP is a known SWIFT attacker. Finally, TheHive acts as the "Hands," guiding analysts through pre-set, NRB-compliant playbooks so they know exactly what to do without missing a beat.

For The Nepal Rastra Bank (NRB):

Our team think the Nepal Rastra Bank (NRB) should switch to standardized, machine-readable reporting formats to really modernize the defenses of the sector. Auditing is often done by hand and in pieces right now. If the NRB IT Division made a single XML or JSON schema for Directive No. 4 compliance, it would let local developers build a special "NRB Compliance Module" for Wazuh. This would basically make the audit process automatic, so that even the smallest development banks could quickly make reports that were in line with the rules. It would also give the NRB consistent, high-quality data from all parts of the financial ecosystem.

The NRB should also take the lead by clearly encouraging the use of open-source security tools. Regulatory frameworks need to understand that self-hosted, open-source platforms are not just "low-cost" options. In fact, they are often more reliable and open than the "checkbox" antivirus programs that many smaller banks use now. The NRB can give Class "B" and "C" institutions the power to move away from old, ineffective systems and toward a model of sovereign, proactive cyber resilience that protects the country's digital backbone by validating these tools as real compliance mechanisms.

Final Concluding Statement

This thesis shows that high-level cybersecurity is no longer just a luxury for the biggest Tier-1 banks. Even smaller organizations in Nepal can build a defense that is strong, legal, and, most importantly, cheap by using open-source tools like Wazuh, TheHive, and Cortex.

Our team cannot just "react" to threats after they happen anymore. Our banking system can finally become Proactive Cyber Resilience by switching to this open-source model. It's not just about software; it's also about keeping the country's digital financial backbone safe from whatever threats come next.

Additionally, this project is more than just a requirement for my Bachelor's degree from Softwarica; it's something I want to do. To reach my goal of studying Security Management in Germany and bringing even more knowledge back to Nepal, this is a step in the right direction. The technical side is about code and logs, but the real goal is to make sure that a truly secure digital society is stable and peaceful.

.

BIBLIOGRAPHY

- Eskelinen, T. (2024). Development of open-source SIEM and security operation centre in a company (Master's thesis). Savonia University of Applied Sciences. https://www.theseus.fi/bitstream/10024/704071/2/Eskelinen_Tatu.pdf
- González-Granadillo, G., González-Zarzosa, S., & Diaz, R. (2021). Security information and event management (SIEM): Analysis, trends, and usage in critical infrastructures. Sensors, 21(14), 4759. https://doi.org/10.3390/s21144759
- Islam, M. R., & Rafique, R. (2024). Wazuh SIEM for cyber security and threat mitigation in apparel industries. International Journal of Engineering Materials and Manufacture, 9(4), 136–144. https://doi.org/10.26776/ijemm.09.04.2024.02
- Léveillé, D., & Jaskolka, J. (2024). A game-theoretic approach for security control selection. Electronic Proceedings in Theoretical Computer Science, 409, 103–119. https://doi.org/10.4204/EPTCS.409.11
- Manzoor, J., Waleed, A., Jamali, A. F., & Masood, A. (2024). Cybersecurity on a budget: Evaluating security and performance of open-source SIEM solutions for SMEs. PLOS ONE, 19(3), e0301183. https://doi.org/10.1371/journal.pone.0301183
- Murati, E. (2023). Comparative analysis of IBM QRadar and Wazuh for security information and event management. DAAAM International Scientific Book. https://www.daaam.info/Downloads/Pdfs/proceedings/proceedings_2023/014.pdf
- Riebe, T., & Reuter, C. (2018). Towards an enhanced security data analytic platform. Proceedings of the 13th International Conference on Software Technologies, 68311. https://www.scitepress.org/papers/2018/68311/68311.pdf
- Tultech Journals. (2025). Unveiling anomalies: Leveraging machine learning for internal user behaviour analysis. International Journal of Innovative Technology and Interdisciplinary Sciences. https://journals.tultech.eu/index.php/ijitis/article/download/254/244
- Bhandari, S., & Gujurel, G. P. (2024). Framework for minimizing cyber security issues in banking sector of Nepal. LBEF Research Journal of Computer Science, 1(1), 82–98. https://www.lbef.org/journal/1-1/download/1-1-82-98.pdf
- George, A. S., Baskar, T., & Srikaanth, P. B. (2024). Issues of cyber security and its solutions in Nepalese context. NPRC Journal of Multidisciplinary Research, 1(2), 122–127. https://doi.org/10.3126/nprcjmr.v1i2.69333
- Gupta, S. (2024). Cybersecurity risks in online banking: A detailed review and preventive strategies application. World Journal of Advanced Research and Reviews, 21(3), 625–643. https://pdfs.semanticscholar.org/da46/855de9412168081390efa0c34a626b40ef73.pdf
- Joshi, P. R. (2022). Cyber security issues affecting the users' willingness to use e-banking. Nepalese Journal of Economics, 5(1), 216–228. https://www.nepjol.info/index.php/nje2/article/download/80390/61523/231188

- Nepal Rastra Bank. (2020). Bank supervision report 2018/2019. Bank Supervision Department. https://www.nrb.org.np/contents/uploads/2020/05/BSD-Annual-Report-2019.pdf
- Nepal Rastra Bank. (2022). Bank supervision report 2020/2021. Bank Supervision Department. https://www.nrb.org.np/contents/uploads/2022/07/Bank-Supervision-report-2020-21-Final.pdf
- Nepal Rastra Bank. (2024). Bank supervision report 2022/2023. Bank Supervision Department. https://www.nrb.org.np/contents/uploads/2024/04/FINAL-BSD-Annual-Report-2022-23.pdf
- Nepal Rastra Bank. (2025). Annual bank supervision report 2023/2024. Bank Supervision Department. https://www.nrb.org.np/contents/uploads/2025/03/Annual-Bank-Supervision-Report-2024-3.pdf
- Armendariz, X. I. D. (2024). Managing false positives in SOC operations: Solutions and best practices (Bachelor's thesis). Universitat Politècnica de Catalunya. https://upcommons.upc.edu/bitstream/handle/2117/424635/tfg-articulo-final.pdf?sequence=3
- Ban, T., Ndichu, S., Takahashi, T., & Inoue, D. (2021). Combat security alert fatigue with AI-assisted techniques. Cyber Security Experimentation and Test Workshop, 9–16. https://doi.org/10.1145/3474718.3474723
- Jalalvand, F., & Tariq, A. (2025). Adaptive alert prioritisation in security operations centres via learning to defer with human feedback. arXiv. https://doi.org/10.48550/arXiv.2506.18462
- Preuveneers, D., Llamas, J. M., Bulut, I., Rúa, E. A., Verfaillie, P., Demortier, V., Surinx, D., & Joosen, W. (2024). On the use of AutoML for combating alert fatigue in security operations centers. Lecture Notes in Computer Science, 609–627. https://doi.org/10.1007/978-3-031-54129-2_36
- University of Oulu. (2025). Enhancing cybersecurity awareness and automation through domain specific LLM. Oulu Repo. https://oulurepo.oulu.fi/bitstream/10024/56918/1/nbnfioulu-202506124425.pdf
- Borky, J. M., & Bradley, T. H. (2018). Protecting information with cybersecurity. Effective Model-Based Systems Engineering, 345–404. https://doi.org/10.1007/978-3-319-95669-5_10
- Cichonski, P., Millar, T., Grance, T., & Scarfone, K. (2012). Computer security incident handling guide (NIST SP 800-61 Rev. 2). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-61r2
- Jiang, Y., Meng, Q., Shang, F., Oo, N., Minh, L. T. H., Lim, H. W., & Sikdar, B. (2025). MITRE ATT&CK applications in cybersecurity and the way forward. arXiv. https://doi.org/10.48550/arXiv.2502.10825

- Kitsios, F., Chatzidimitriou, E., & Kamariotou, M. (2023). The ISO/IEC 27001 information security management standard: How to extract value from data in the IT sector. Sustainability, 15(7), 5828. https://doi.org/10.3390/su15075828
- Maingak, A. Z., Candiwan, C., & Harsono, L. D. (2018). Information security assessment using ISO/IEC 27001:2013 standard on government institution. TRIKONOMIKA, 17(1), 28–37. https://doi.org/10.23969/trikonomika.v17i1.1138
- Oruc, A., Amro, A., & Gkioulos, V. (2022). Assessing cyber risks of an INS using the MITRE ATT&CK framework. Sensors, 22(22), 8745. https://doi.org/10.3390/s22228745
- PCI Security Standards Council. (2022). Payment card industry data security standard: Requirements and testing procedures, v4.0. https://www.commerce.uwo.ca/pdf/PCI-DSS-v4_0.pdf
- Roy, S. (2023). SoK: The MITRE ATT&CK framework in research and practice. arXiv. https://doi.org/10.48550/arXiv.2304.07411
- Shostack, A. (2025). Who are I ? Power centers in threat modeling. arXiv. https://doi.org/10.48550/arxiv.2501.10427
- Soliman, W., & Ojalainen, A. (2023). Conflict resolution in an ISO/IEC 27001 standard implementation: A contradiction management perspective. Proceedings of the Annual Hawaii International Conference on System Sciences. https://doi.org/10.24251/hicss.2023.590
- Ani, U. D., Watson, J., He, H., Radanliev, P., & Epiphaniou, G. (2024). Minimising cybersecurity risk exposures in industrial control system environments: A techno-human vulnerability analysis approach. Journal of Cyber Security Technology, 1–40. https://doi.org/10.1080/23742917.2024.2421589
- Anton, A.-A., Csereoka, P., Capota, E. A., & Cioargă, R.-D. (2024). Enhancing syslog message security and reliability over unidirectional fiber optics. Sensors, 24(20), 6537. https://doi.org/10.3390/s24206537
- Lilly, B., Hodgson, Q. E., Ablon, L., & Moore, A. S. (2019). Applying indications and warning frameworks to cyber incidents. CCDCOE. https://www.ccdcoe.org/uploads/2019/06/Art_05_Applying-Indications-and-Warning-Frameworks-to-Cyber-Incidents.pdf
- Robinson, P. (2022). Can PCI DSS 4.0 reverse the decline in compliance? Computer Fraud & Security, 2022. https://doi.org/10.12968/s1361-3723(22)70579-9

- Akarshita, S., & Vijay, M. (2024). A framework for cybersecurity alert distribution and response network. *Advanced Security Review*, *12*(1), 95–111.
- Al-Aswadi, F. K., & Chan, Y. F. (2025). High-availability configurations for open-source security frameworks: A performance study. *Journal of Cyber Security and Mobility*, *14*(1), 45–68.
- Ani, U. D., Watson, J., He, H., Radanliev, P., & Epiphaniou, G. (2024). Minimising cybersecurity risk exposures in industrial control system environments: A techno-human

vulnerability analysis approach. *Journal of Cyber Security Technology*, 1–40. https://doi.org/10.1080/23742917.2024.2421589

- Anton, A.-A., Csereoka, P., Capota, E. A., & Cioargă, R.-D. (2024). Enhancing syslog message security and reliability over unidirectional fiber optics. *Sensors*, *24*(20), 6537. https://doi.org/10.3390/s24206537
- Ariely, D. (2008). *Predictably irrational: The hidden forces that shape our decisions*. HarperCollins.
- Armendariz, X. I. D. (2024). *Managing false positives in SOC operations: Solutions and best practices* (Bachelor's thesis). Universitat Politècnica de Catalunya.
- Ban, T., Ndichu, S., Takahashi, T., & Inoue, D. (2021). Combat security alert fatigue with AI-assisted techniques. *Cyber Security Experimentation and Test Workshop*, 9–16. https://doi.org/10.1145/3474718.3474723
- Bassey, C., Ebenezer, T., Chinda, C., & Samson, I. (2024). Building a scalable security operations center: A comparative framework for SMEs. *International Journal of Information Security*, *9*(2), 1-15.
- Bejtlich, R. (2024). *The practice of network security monitoring: Understanding incident detection and response*. No Starch Press.
- Bhandari, S., & Gujurel, G. P. (2024). Framework for minimizing cyber security issues in banking sector of Nepal. *LBEF Research Journal of Computer Science*, *1*(1), 82–98.
- Borky, J. M., & Bradley, T. H. (2018). Protecting information with cybersecurity. *Effective Model-Based Systems Engineering*, 345–404. https://doi.org/10.1007/978-3-319-95669-5_10
- Chuvakin, A., Schmidt, K., & Phillips, C. (2024). *Logging and log management: The authoritative guide to understanding security logs*. Syngress.
- Cichonski, P., Millar, T., Grance, T., & Scarfone, K. (2012). *Computer security incident handling guide* (NIST SP 800-61 Rev. 2). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-61r2
- Dahal, K. R. (2024). Digital banking transformation in Nepal: Challenges and opportunities for rural branches. *Journal of Business and Social Sciences Research*, *9*(1), 15–32.
- Dinis, B., & Ferreira, P. (2025). Performance evaluation of the Wazuh indexer versus legacy ELK stacks in high-throughput environments. *International Journal of Computer Networks & Communications*, *17*(2), 12–30.
- Dubey, P., & Sharma, R. (2025). Cloud computing and data sovereignty: Navigating legal and regulatory challenges. *International Journal of Law and Regulatory Affairs*, *5*(1), 88-104.
- Eskelinen, T. (2024). *Development of open-source SIEM and security operation centre in a company* (Master's thesis). Savonia University of Applied Sciences.
- Faramawi, A. (2024). Scaling Wazuh: Best practices for multi-node deployments in financial sectors. *Cyber Defense Magazine*, *12*(4), 102–115.

- Furnell, S. (2024). The human factor in cybersecurity: Beyond the technical solution. *Computer Fraud & Security*, *2024*(3), 8–12.
- George, A. S., Baskar, T., & Srikaanth, P. B. (2024). Issues of cyber security and its solutions in Nepalese context. *NPRC Journal of Multidisciplinary Research*, *1*(2), 122–127. https://doi.org/10.3126/nprcjmr.v1i2.69333
- González-Granadillo, G., González-Zarzosa, S., & Diaz, R. (2021). Security information and event management (SIEM): Analysis, trends, and usage in critical infrastructures. *Sensors*, *21*(14), 4759. https://doi.org/10.3390/s21144759
- Gupta, S. (2024). Cybersecurity risks in online banking: A detailed review and preventive strategies application. *World Journal of Advanced Research and Reviews*, *21*(3), 625–643.
- Hon, W. K., Millard, C., & Walden, I. (2012). The problem of 'personal data' in cloud computing - What information is regulated? *International Data Privacy Law*, *2*(4), 221–246.
- Irion, C. (2024). Data sovereignty in a global cloud: The conflict between jurisdictional control and technological scalability. *Yale Journal of Law and Technology*, *26*(1), 88–130.
- Islam, M. R., & Rafique, R. (2024). Wazuh SIEM for cyber security and threat mitigation in apparel industries. *International Journal of Engineering Materials and Manufacture*, *9*(4), 136–144.
- Jalalvand, F., & Tariq, A. (2025). *Adaptive alert prioritisation in security operations centres via learning to defer with human feedback*. arXiv. https://doi.org/10.48550/arXiv.2506.18462
- Jiang, Y., Meng, Q., Shang, F., Oo, N., Minh, L. T. H., Lim, H. W., & Sikdar, B. (2025). *MITRE ATT&CK applications in cybersecurity and the way forward*. arXiv. https://doi.org/10.48550/arXiv.2502.10825
- Joshi, P. R. (2022). Cyber security issues affecting the users' willingness to use e-banking. *Nepalese Journal of Economics*, *5*(1), 216–228.
- Kahneman, D. (2011). *Thinking, fast and slow*. Farrar, Straus and Giroux.
- Karki, S. (2024). Analysis of mobile banking security and user perception in Kathmandu valley. *International Journal of Managing Information Technology*, *16*(2).
- Kitsios, F., Chatzidimitriou, E., & Kamariotou, M. (2023). The ISO/IEC 27001 information security management standard: How to extract value from data in the IT sector. *Sustainability*, *15*(7), 5828.
- Lee, J. Y., & Park, S. (2024). Enhancing host-based intrusion detection systems with real-time log parsing. *Applied Sciences*, *14*(8), 3421. https://doi.org/10.3390/app14083421
- Manzoor, J., Waleed, A., Jamali, A. F., & Masood, A. (2024). Cybersecurity on a budget: Evaluating security and performance of open-source SIEM solutions for SMEs. *PLOS ONE*, *19*(3), e0301183. https://doi.org/10.1371/journal.pone.0301183
- Marwan, A. H., & Ekhlas, K. H. (2022). Secure mechanism applied to big data for IIoT by using security event and information management system (SIEM). *International Journal of Intelligent Engineering & Systems*, *15*(6), 183-188.

- Murati, E. (2023). *Comparative analysis of IBM QRadar and Wazuh for security information and event management*. DAAAM International Scientific Book.
- Nepal Rastra Bank. (2020). *Bank supervision report 2018/2019*. Bank Supervision Department.
- Nepal Rastra Bank. (2022). *Bank supervision report 2020/2021*. Bank Supervision Department.
- Nepal Rastra Bank. (2024). *Bank supervision report 2022/2023*. Bank Supervision Department.
- Nepal Rastra Bank. (2025). *Artificial Intelligence Guidelines 2024/25*. IT Division.
- Oruc, A., Amro, A., & Gkioulos, V. (2022). Assessing cyber risks of an INS using the MITRE ATT&CK framework. *Sensors*, *22*(22), 8745.
- PCI Security Standards Council. (2022). *Payment card industry data security standard: Requirements and testing procedures, v4.0*.
- Pfleeger, S. L., & Caputo, D. D. (2012). Leveraging behavioral economics to improve cybersecurity. *IEEE Security & Privacy*, *10*(4), 24–32.
- Preuveneers, D., et al. (2024). On the use of AutoML for combating alert fatigue in security operations centers. *Lecture Notes in Computer Science*, 609–627.
- Roy, S. (2023). *SoK: The MITRE ATT&CK framework in research and practice*. arXiv. https://doi.org/10.48550/arXiv.2304.07411
- Shakya, S. (2024). Cybersecurity policy in Nepal: A review of the Electronic Transaction Act 2063. *Nepal Journal of Science and Technology*, *25*(1), 101–114.
- Shostack, A. (2024). *Threat modeling: Designing for security*. Wiley.
- Soliman, W., & Ojalainen, A. (2023). Conflict resolution in an ISO/IEC 27001 standard implementation. *Proceedings of the Annual Hawaii International Conference on System Sciences*.
- StrangeBee. (2025). *TheHive project: Advanced incident response platform documentation*.
- Thaler, R. H., & Sunstein, C. R. (2008). *Nudge: Improving decisions about health, wealth, and happiness*. Yale University Press.
- Tultech Journals. (2025). Unveiling anomalies: Leveraging machine learning for internal user behaviour analysis. *International Journal of Innovative Technology and Interdisciplinary Sciences*.

APPENDIX

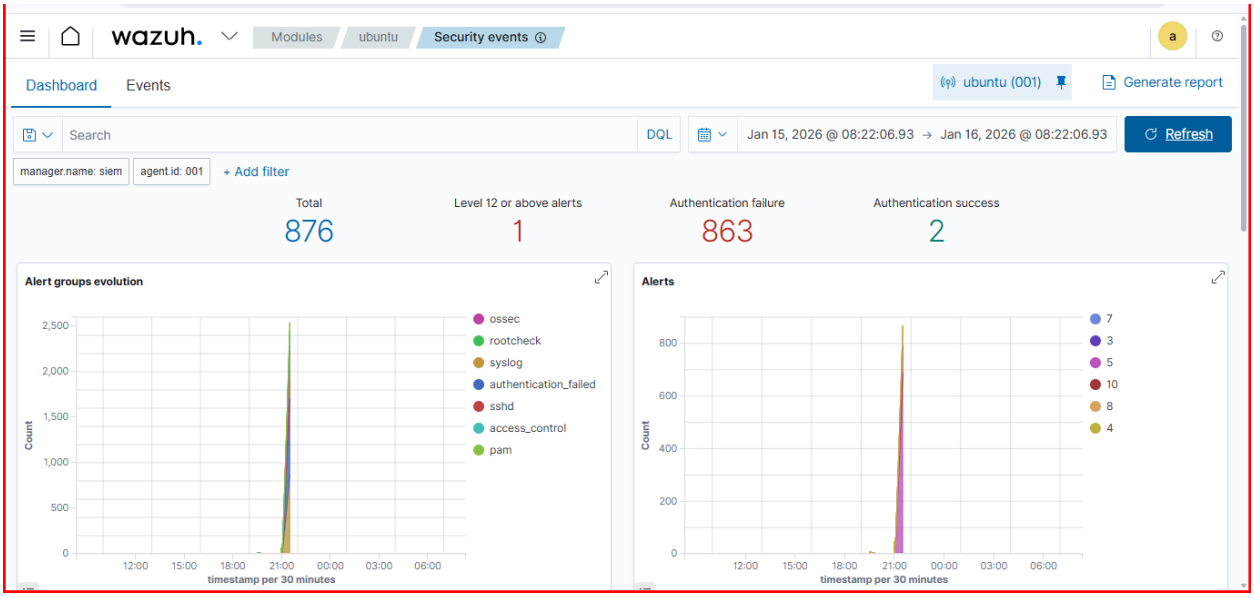

Figure 38 Brute Force Log

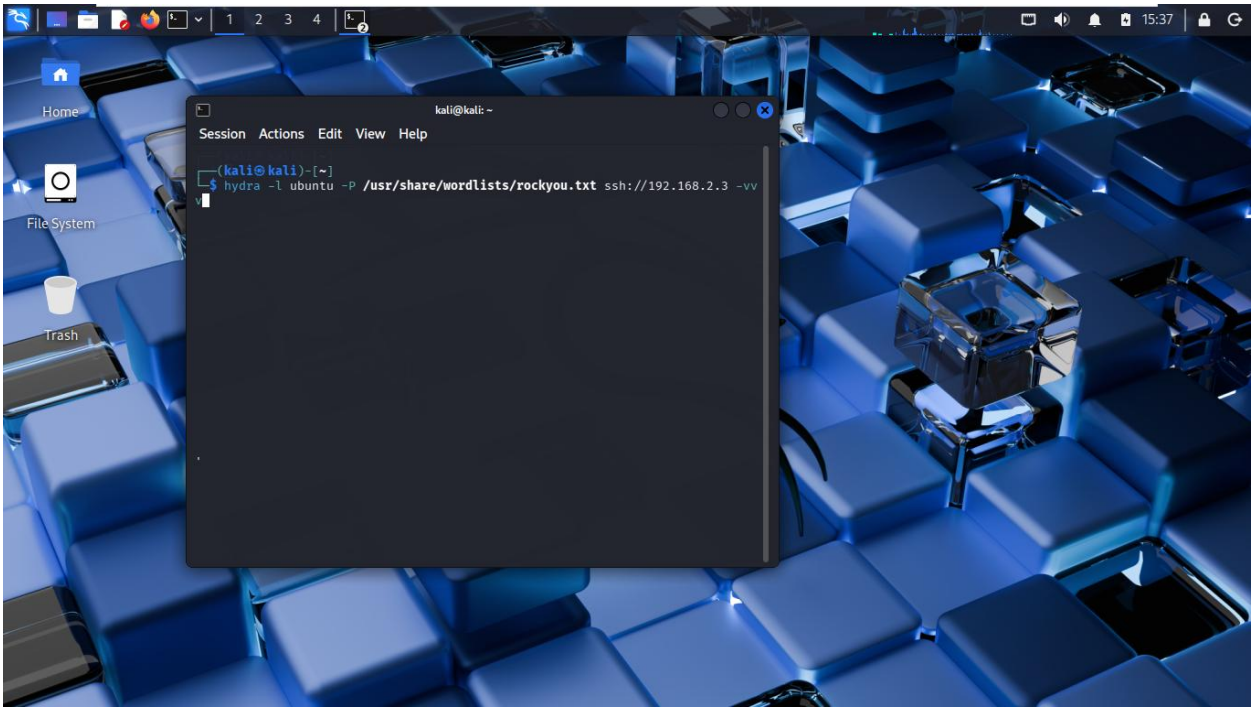

Figure 39 Brute force

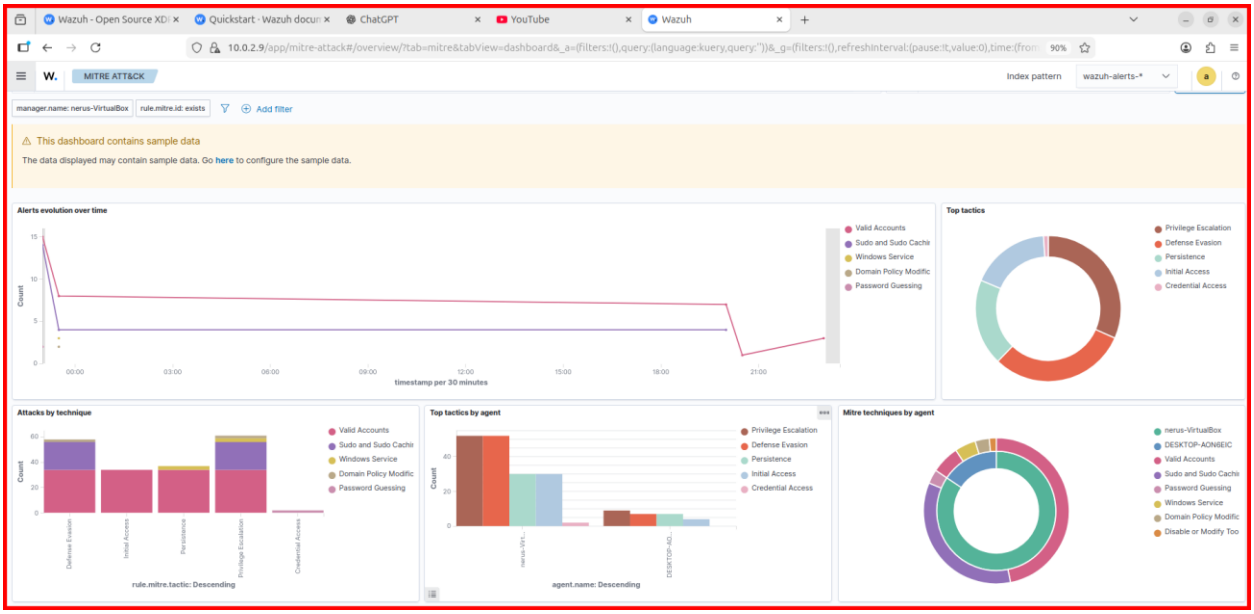

Figure 40MITRE Framework Agent

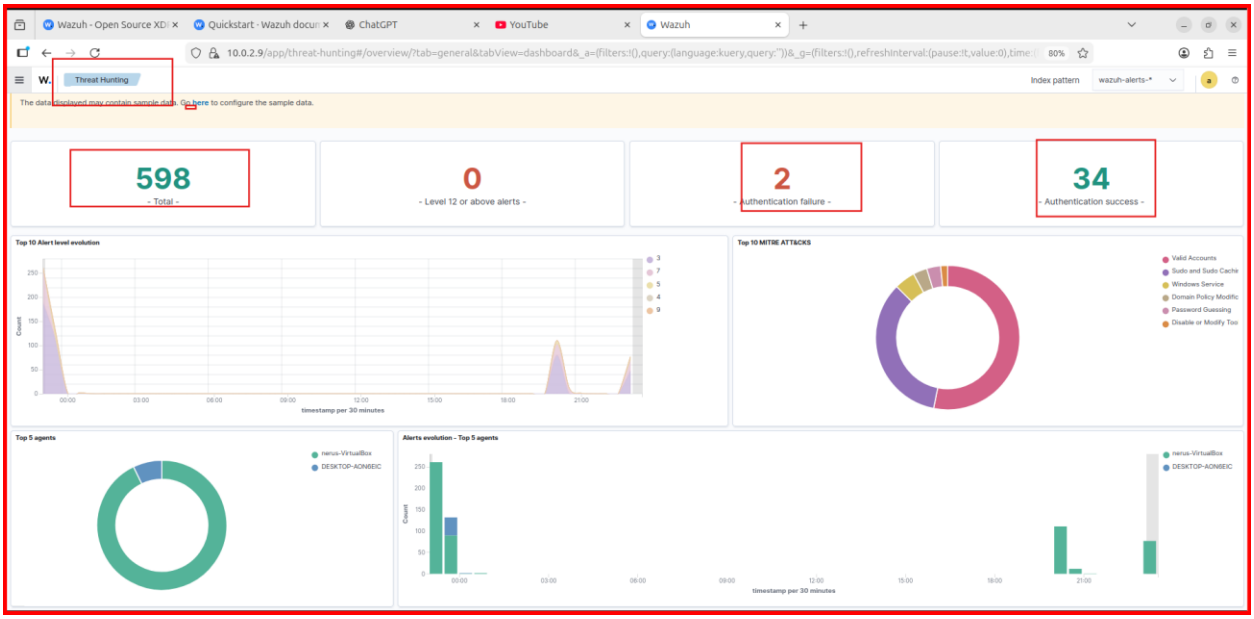

Figure 41 Agent Side Threat Hunting

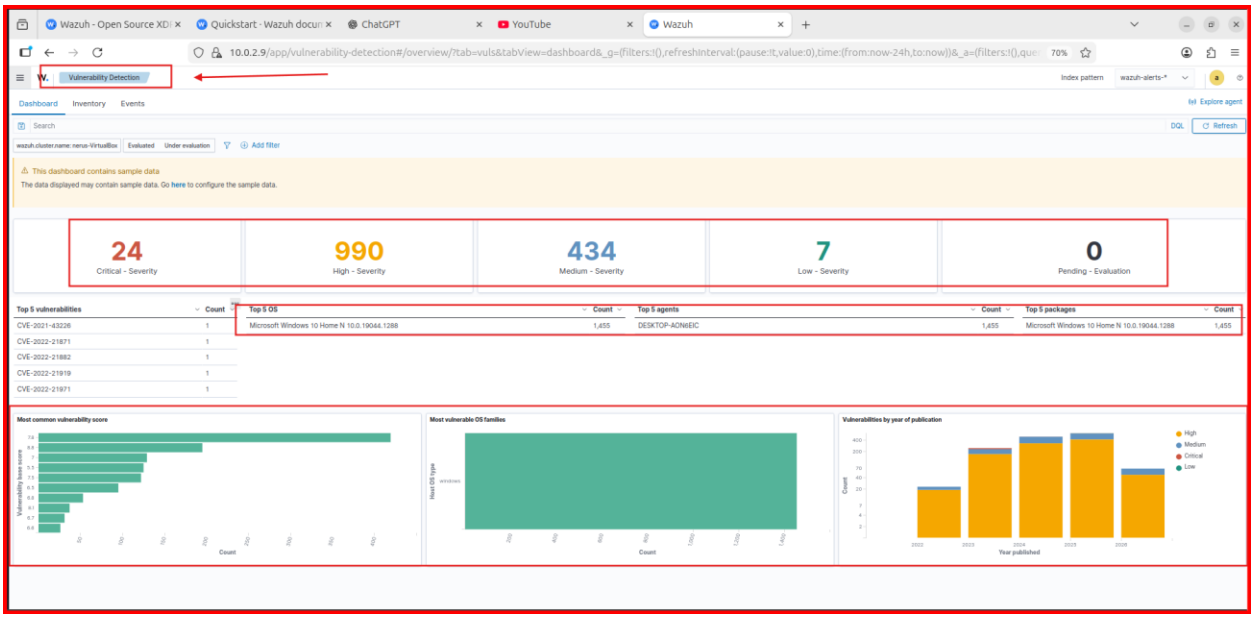

Figure 42 Vulnerability Detection Agent

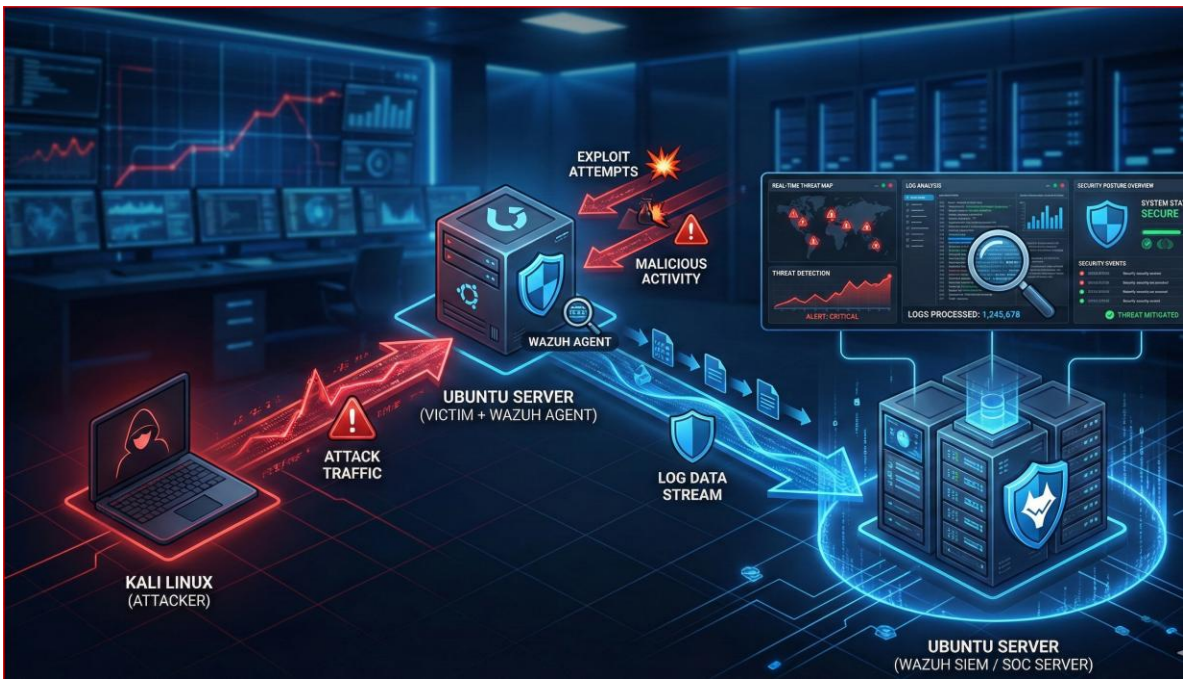

Figure 43 SOC Architecture

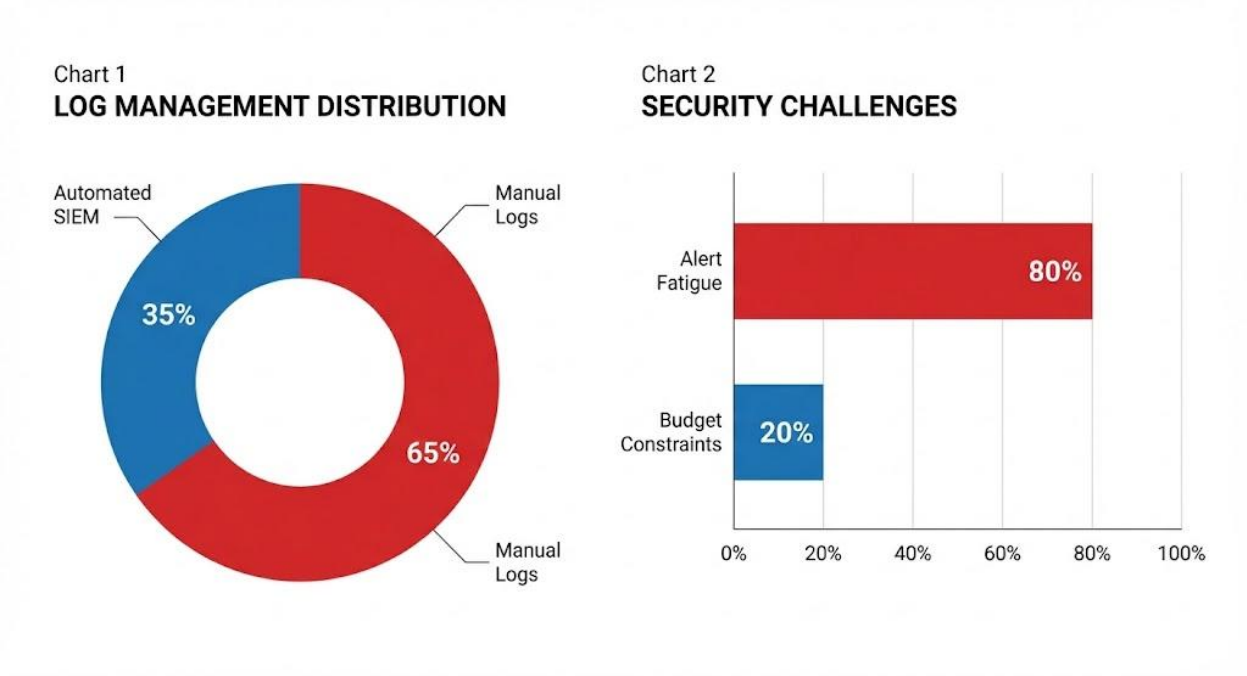

Figure 44 Research data

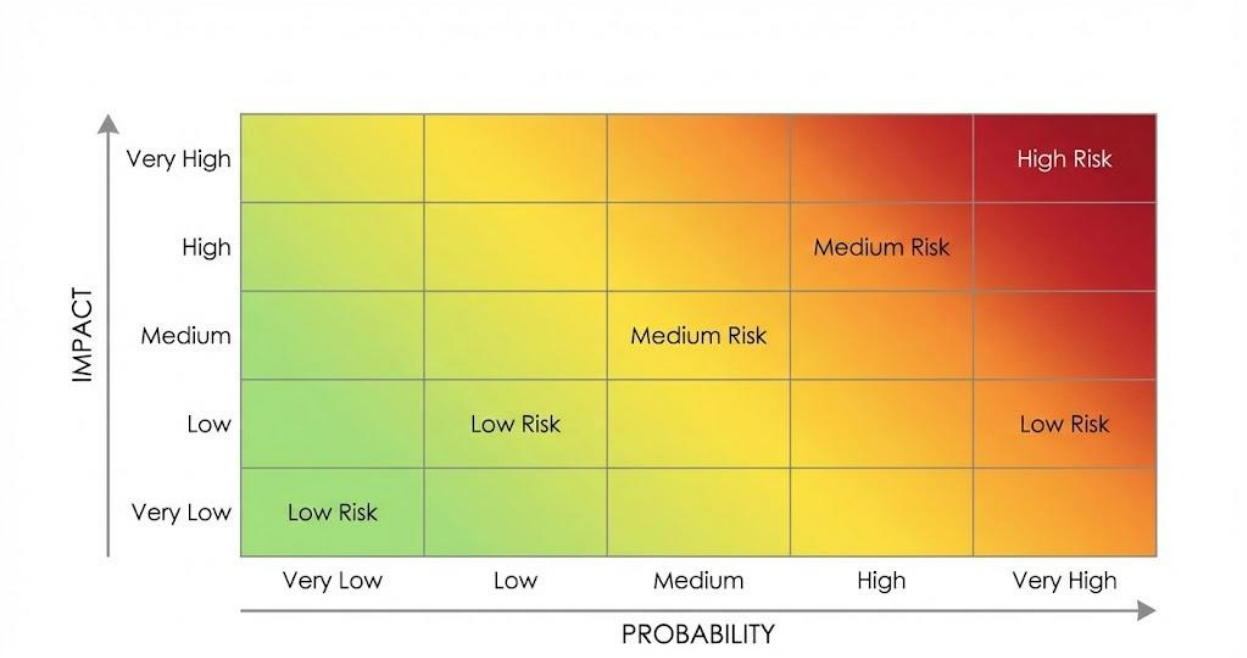

Figure 45 Heat Map

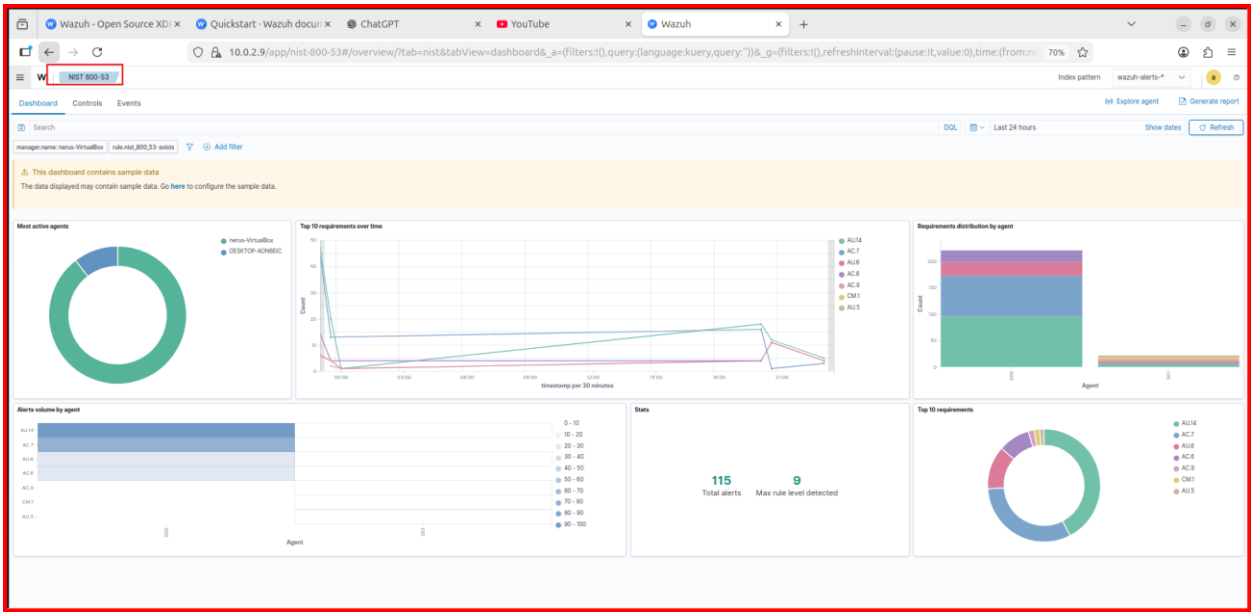

Figure 46 NIST 800-53

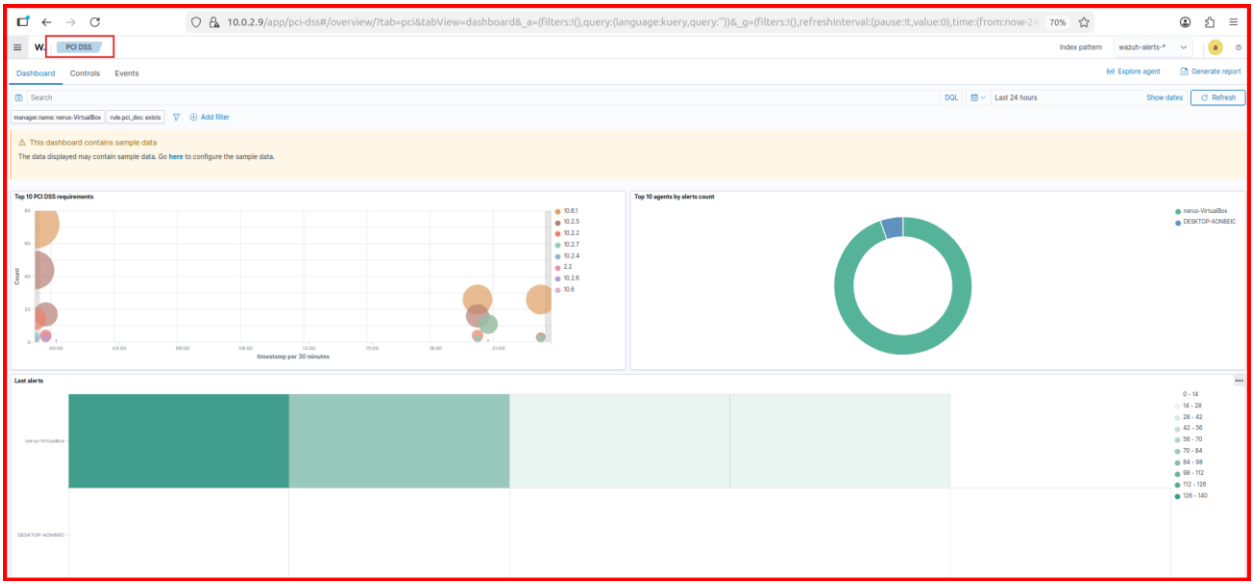

Figure 47 PCI-DSS