Fall 2024

Mitigating Cyber Espionage: A Network Security Strategy Using Notifications

Claire Headland
cah212@uakron.edu

College of Engineering and Polymer Science

Department of Computer Science

The University of Akron

CIS Senior Cybersecurity Project's proposal to fulfill the requirement of the Computer Information Systems (CIS)- Cybersecurity Degree.

Mitigating Cyber Espionage: A Network Security Strategy Using Notifications

Claire A. Headland

Dr. Nadhem Ebrahim

CISS 491-801: CIS Senior Cybersecurity Project

ABSTRACT: Network security and its mitigation of cyber espionage is paramount to the confidentiality, integrity, and availability of data within the intelligence field. With the advancing efficacy of social engineering to execute cyber espionage attacks, further measures and fail-safe mechanisms have become necessary. If a malicious actor successfully penetrates the network, suspending confidential data transmissions over the compromised network becomes crucial. However, connected users need a platform to receive security notifications and, therefore, need to know that their continued network use compromises more data. This project eliminates this by achieving two primary objectives: designing a multi-layered, hardened, and segmented network environment and providing a fail-safe notification alert mechanism that informs all users that their network is compromised. This network security strategy designs and hardens a network and implements intrusion detection, prevention, and response through a security information and event management system (SIEM). This project integrates the SIEM alerts into Microsoft Teams to give centralized, accessible notifications to all network users. By executing various cyber attacks, this project finds the notification alert mechanism successful in providing network users with automatic, real-time notifications about the attack that instruct users to suspend network transmissions. Future research should explore how artificial intelligence and next-generation SIEMs can advance cyber espionage mitigation with user awareness.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I wish to express my genuine appreciation for the following individuals who dedicated their time and energy to help me navigate this journey. It is an honor to acknowledge the assistance of these individuals in the completion of this project. The continuing support of Professors Janet Kropff, Stanley Smith, and Richard Mehok enriched the ideation and development of my work. Additionally, the oversight, guidance, and comments given by Dr. Nadhem Ebrahim throughout the duration of this project was paramount to its completion. I thank them all for their indispensable contributions.

DEDICATION

I dedicate this work to my mom, dad, grandma, and grandpa, for their unwavering love, support, and open arms throughout my educational career. Thank you for believing in me and pushing me beyond the family farm.

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Cyberattacks are consistently advancing and evolving rapidly in complexity. This shift poses a significant concern for institutions and organizations managing confidential data, particularly in intelligence, where confidentiality directly impacts national security. Cyber espionage, also known as cyber spying, is a cyberattack in which an unauthorized actor attempts to gain sensitive data for various purposes [1]. This cyber attack specifically targets such confidential data. Furthermore, these attacks present a critical problem when an attacker has undetected long-term access to the data. Networks of confidential data are highly susceptible to cyber-spying attacks. As the Department of Defense defines, the "interdependent network of information technology infrastructures," such as network environments used by the intelligence field, is also commonly referred to as cyberspace [2]. Therefore, network security references the encompassing network devices, such as routers, switches, servers, and end devices. The mitigation of cyber espionage is paramount to combating the emerging threats exploiting even the most secure network of intelligence and confidential data.

1.1 — Overview of Cyber Espionage Attacks

Cyber espionage depicts a broad domain of several types of cyber attacks. The primary concerns involve attacks leading to data compromise, often conducted by exploiting human nature. One example includes advanced persistent threats (APTs), which describe attackers with long-term, undetected access to confidential data on the network [3]. Additionally, attackers may secretly conduct a man-in-the-middle (MITM) attack to intercept network transmissions [4]. Zero-day

attacks, however, pose one of the most severe threats to intelligence, where attackers unpreventably exploit an unknown vulnerability to access the environment [5]. These attack methodologies fall within the cyber espionage category and are increasingly concerning to the intelligence field and national security.

Successful cyber espionage attacks within the past decade have most commonly abused human nature to achieve network access, highlighting the need for security design specifically managing this abuse. Social engineering manipulates unaware users within the network into providing attackers with information or connectivity that grants them unauthorized network access [6]. Primarily, attackers conduct this through phishing emails containing links or attachments to hidden malicious programs and code [7]. Recent cyber espionage attacks have demonstrated their efficacy in using phishing and social engineering. In 2023, phishing attacks comprised approximately 33% of confidential data breaches [8].

Furthermore, the study estimates that 78% of cyber espionage attacks resulted from successful phishing attempts [8]. Cyber espionage attacks stemming from social engineering have not only raised concerns based on statistics; several current national security incidents have shown similar problems. In 2024, for instance, one group of hackers successfully penetrated a law firm connected to the Australian government, resulting in their access to 2.5 million government documents [9]. Additionally, in 2023, a different hacker organization planted malicious code onto government systems in the Philippines by sending phishing

emails, allowing the group to spy on the systems [9]. Another phishing attack occurred in 2023 when the Kuwait Ministry of Finance systems fell victim to phishing ransomware [9]. The drastic consequences of social engineering and its success in committing cyber espionage in the modern age depict the immediate need to secure networks based on these advances. The shift for intelligence analysts and managers of confidential data to the technology-forward age has encompassed a complete change in operations and procedures. Additionally, the policies, procedures, training, and user awareness have needed to evolve continuously since the shift to combat advancing social engineering techniques.

1.2 — General Security Principles

Proper network security necessitates securing the entire network environment, or cyberspace, which has become highly prevalent in mitigating cyber espionage. Within the past decade, the digital domain has witnessed a drastic increase in cyber attack efficacy, resulting in increased research and proposed security paradigms. Several models have proved successful and have become the standard for network environment design. However, with the more recent advancements in social engineering, a recent focus has been placed on principles addressing the abuse of human nature. Moreover, mindfulness of the unpatchable vulnerabilities it presents is quintessential to networks of intelligence and confidential data security.

The categorization of data security provides insight into how general security principles and concept deployment improve security. The CIA Triad defines these categories as confidentiality, integrity, and availability (CIA) [10]. The

model's primary objective is to ensure the fundamental security of the three components. In essence, security measures should safeguard data from unauthorized users and unauthorized modification but still remain available to only the necessary users and resources. Utilizing the CIA Triad in coalescence with the most recent security paradigms is critical for securing against the advancements of cyber espionage attacks.

1.2.1 — Security Lifecycle

The initial implementation of network environment security occurs during the environment's design and creation. The model published by the SANS Institute in [11] describes a security design approach with four steps: Identify, Assess, Protect, and Monitor. The model, as presented in [11, Fig. 1], depicts each step as a never-ending cycle, representing that security must be a continuous cycle.

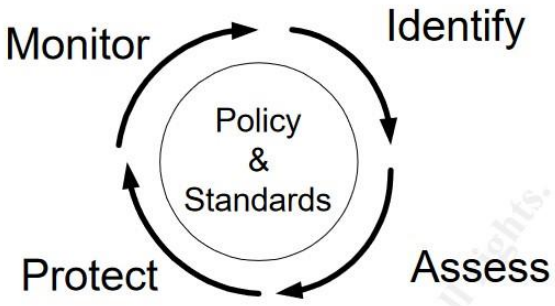


Fig. 1. Security Lifecycle Model [11].

Network engineers identify the assets to protect, analyze the identified targets to find the system vulnerabilities, create a protection plan to secure the current vulnerabilities, and continuously monitor the security implementation [11]. When applied to current security concerns, the primary vulnerabilities to safeguard

often involve human susceptibility to manipulation. Current security standards aid the creation of a protection plan in the Security Lifecycle model.

1.2.2 — Defense-in-Depth

The Defense-in-Depth (DiD) model is one of the primary strategies used to address the rise of cyber espionage through social engineering. The model recommends network environment security deployment through multiple comprehensive levels/layers [12]. Rather than implementing security as a single dimension, DiD introduces the framework of creating multiple levels that attackers must bypass to access the target of defense (ToD). This approach is beneficial for its mitigation of a singular point of failure. For instance, if an attack successfully intrudes the most external security layer, the attacker is met with other defense mechanisms that separate them from the core ToD. DiD recommends implementing some layers by securing multiple OSI model layers, such as the data link and network layers [13, 14]. However, the DiD model lacks security concepts that specifically address the vulnerabilities caused by human susceptibility to the cyber domain's advancing manipulation tactics. Therefore, network security designs require additional models addressing such concerns in addition to DiD.

1.2.3 — Zero Trust

One of the most prominent security paradigms the National Security Agency recommends is the Zero Trust (ZT) model. ZT and its corresponding ZT architecture (ZTA) are concerned with establishing security that prevents malicious actors from breaching the network through lateral movement [15]. The model seeks to minimize the impact of a successful network environment intrusion by

employing network micro-segmentation and isolating the ToD and users as much as possible while maintaining availability [16]. This model enhances DiD by creating security layers through the environment's design. Additionally, ZT inherently distrusts users on the network, accounting for the current abuses exploiting unaware network users. Moreover, if a cyber espionage attack succeeds, its damages are minimized to only the isolated, segmented area compromised. Thus, ZT principles have become highly essential in the battle against the growing cyber attack space.

1.2.4 — Secure by Design

The Secure by Design model, as documented by the Cybersecurity & Infrastructure Security Agency, recommends integrating security measures into the original design of a product [11]. However, this foundational concept applies to cybersecurity and network security, particularly with ZT model principles. Implementing multiple security layers through network environment design provides the network with enhanced, inherent security. The Kuwait government employed this approach after the cyber espionage attack against the Ministry of Finance, where they isolated confidential government data from unnecessary government agencies and branches [9]. The current standard security models ensure that, upon a successful malicious intrusion primarily caused by social engineering, other established security mechanisms can still safeguard the ToD.

1.3 — General Security Practices

Current security techniques that secure the network environment involve coalescing many previously independent mechanisms into one holistic approach.

The primary objective in such a combination is to centralize system management and ensure all the security mechanisms record potential malicious activity. The security techniques in the past involved network hardening and the configuration and deployment of an intrusion detection system (IDS), intrusion prevention system (ISP), and intrusion response system (IRS) [17]. However, the most recent mechanism to mitigate cyber espionage and other related attacks is implementing a Security Information and Event Management (SIEM) system.

1.3.1 — Network Hardening

User and end devices access networks via connections to switches and routers, making their security paramount. Network hardening encompasses the configurations of network devices that eliminate the devices' exploitable vulnerabilities. Furthermore, securing the devices at the data link and network layers ensures that both switches and routers have established security mechanisms [14]. To enhance data link layer security, port security, Dynamic Host Configuration Protocol (DHCP) snooping, and Dynamic Address Resolution Protocol (ARP) Inspection (DAI) are recommended [14]. Network layer security is also established through stateless firewall access control list (ACL) configurations [18]. These configurations do not pertain to cyber espionage specifically but prevent general, well-known vulnerabilities that could lead to such attacks.

1.3.2 — Intrusion Detection, Prevention, and Response Systems

There are three main intrusion mechanisms implemented to enhance network environment security: intrusion detection systems (IDSs), intrusion prevention systems (IPSs), and intrusion response systems (IRSs). The objective

of an IDS is to detect intrusions using anomaly or signature detection and generate alerts based on collected data [17]. An IPS serves the same purpose as an IDS but includes preventive action capabilities initiated by an alert [17]. The most encompassing utilization of these systems to mitigate cyber espionage attacks is using Host-based and Network-based systems alongside an IRS that additionally works to prevent detected attacks [17]. However, the enormous logs and data to analyze between all three leaves room for failure. Activity not correctly transmitted across all systems may result in undetected malicious activities. Therefore, with the most recent developments in cyber espionage, a centralized system is now pertinent to the security of network environments.

1.3.3 — Security Information and Event Management Systems

The evolving landscape of cyber-attacks has necessitated creating a comprehensive security approach. Hence, modern security standards emphasize implementing a security information and event management (SIEM) system. These systems offer a holistic solution by integrating hybrid intrusion detection (host-based and network-based), prevention, and response capabilities [20]. This approach has become indispensable to safeguarding network environments, emphasizing centralized analysis of logged data. Integrating an SIEM system into a network environment allows for a centralized platform for real-time intrusion detection, prevention, and response, often incorporating Extended Detection and Response (XDR) functionalities. Current SIEM options include Splunk and Microsoft Sentinel, both expensive investments. However, an increasingly popular solution is the open-source SIEM Wazuh, which provides SIEM and XDR

capabilities with many use cases, fortifying the network environment against numerous malicious activities, including cyber espionage [21]. Wazuh uses a manager to monitor and analyze data collected on end device agents, alerting them based on many configurations discussed in Section 3.8. Furthermore, Wazuh comes with pre-configured integrations and capabilities to create custom integrations. Implementing current security paradigms with an SIEM reduces the cascading effect of successful cyber espionage attacks.

Despite addressing the most recent successes of social engineering in executing cyber espionage attacks, these implementations do not inherently mitigate the vulnerability caused by user unawareness. For instance, if a malicious actor successfully breaches multiple levels of defense and gains access to intelligence or confidential data, the ZTA only minimizes the damages to all the data within that isolated area. However, users in that isolated area may unknowingly continue data transmissions to and from the network area, allowing the actor to access additional data. Therefore, creating a mechanism to minimize user unawareness proves vital to mitigating cyber espionage.

1.4 — Capstone Project Overview

This capstone project expounds upon the conceptual and skills-based objectives established through pursuing a Bachelor of Science in Computer Information Systems, Cybersecurity and a Bachelor of Arts in Political Science, Intelligence and National Security Studies. Moreover, the project explores the current network environment security standards and practices regarding confidential data and the intelligence field. As cyber espionage advancements

continue to escalate, this project calls for a security strategy to reduce the damaging effects of network users unintentionally compromising additional data after a successful malicious intrusion.

1.4.1 — Purpose and Motivations

This project aims to evaluate the contemporary network environment security landscape, focusing on addressing the persistent challenges of cyber espionage within the intelligence field. With the current efficacy of advanced social engineering tactics, intelligence analysts and managers of confidential data may further compromise data by continuing network transmissions and intelligence dissemination after an intrusion.

1.4.2 — Research Objectives

This project aims to review the advancements made in network security to minimize the advancing cyber espionage data compromises. Additionally, based on the current landscape, social engineering poses a significant threat to robust data confidentiality, integrity, and availability security. Therefore, this project develops a security strategy based on the following objectives:

- Design a secure, hardened intelligence field network based on DiD and ZT principles.
- Implement centralized analysis of several event sources to detect malicious activities and take preventive action.
- Contribute to current research dedicated to minimizing the effects of an attack by establishing fail-safe network security mechanisms.

These three primary objectives delineate this capstone project's essential goals.

1.4.3 — Project Contributions

The current cyberspace security landscape has highlighted the essential shift from previous practices to a Zero Trust model aimed at mitigating cyber espionage attacks and other related attacks and their effects. The holistic implementation of security paradigms, network hardening practices, the Wazuh SIEM, and Microsoft Teams alerts contributes to current countermeasures to cyber espionage through its holistic approach, efficiency, and the accessible awareness it brings. Thus, this project implements a security strategy to address user lack of awareness, enhancing the current strategies to minimize cyber attacks and potential damages.

CHAPTER 2: LITERATURE REVIEW

The complexity of evolving cyber threats poses an ongoing challenge to intelligence-related environments' network and cyber security. The necessity of fail-safe mechanisms, even within a ZTA deployment, cannot be over-emphasized when combating cyber espionage. The reliance on hardening techniques and intrusion detection systems alone is not enough to mitigate the omnipresent threat of social engineering that is leading to successful cyber spying attacks. Even with the implementation of an IDS, IPS, and IRS, if the detected activity generates an alert that goes unnoticed, it leaves the data on the network compromised for far too long. Therefore, securing data confidentiality, integrity, and availability through previous models and mechanisms is shifting to a much-needed, more efficient notification alerting process. Modern research now focuses on a non-traditional, heterogeneous combination of previous mechanisms from the traditional approaches [22]. This emphasis involves integrating fail-safe security processes that account for unpatchable vulnerabilities, including a lack of network or system compromise awareness.

2.1 — Intrusion Detection and Prevention System Alerts

Before the recent shift to SIEM systems, researchers worked towards advancing the capabilities of alerts generated by IDSs and IPSs. One devastating issue with alerts generated from an IDS/IPS is the drastically large number of inaccurate alerts the system may generate. Some studies seeking to mitigate that concern describe the alert generation process as a plethora of false positives, false negatives, inaccurate, duplicate, and/or unuseful event alerts [23, 24]. The

research documented in [23] mitigates the issues with alerts by developing CRIM, an IDS module that correlates and merges alerts generated by similar events. The study implements this module using the network-based IDSs (NIDSs), Snort, and e-Trust to merge alerts for the same event and reduce the influx of alerts and false positives/negatives, which proved successful [23]. The study in [22] expounds upon this research by implementing a mechanism to verify the accuracy of Snort alerts by integrating the scripts from the vulnerability scanner Nessus. Such research has paved the way for alerts to be accurate and a relied-upon tool for mitigating cyber attacks. Although recent approaches have diverged from independent deployment of IDSs and IPSs, these fundamental concepts apply to SIEM systems.

Other mechanisms relating to alerts have focused on advancing response systems that minimize cyber-attacks and damage based on IDS-generated alerts. For instance, research has sought to develop IRSs based on alerts by adapting dynamic capabilities and fail-safe mechanisms. The Adaptive Agent-based Intrusion Response System (AAIRS) dynamically adapts its response to cyber-attacks by measuring the success of its responses to previous cyber attacks [25]. As found by the study in [25] and affirmed in [26], IRS adaptation allows for a more efficient network intrusion response that can reduce the time data is compromised. The research in [26] added efficiency to IRSs by developing an automated IRS that immediately notifies system administrators of suspicious events and dynamically employs various responses determined by the type of attack detected. The dynamic and automated response mechanisms were successful in the two studies,

and approaches were developed that minimized data compromise caused by an attack using alerts for more efficient responses. As with the IDS and IPS developments, current security systems still use the basis of research within these studies.

By exploring additional capabilities to use notification alerts as a basis for response mechanisms, studies have also highlighted the necessity of accessible alerts in mitigating cyber attack damages. Research in [27] and [28] involved techniques to make malicious event alerts more accessible and faster. The system discussed in [27] enhances the Snort IDS alerts by creating a notification system that produces efficient warning messages with attack information to system administrators in real-time. The notification system tested in [28] created a system that also uses Snort alerts but diverges with its capability for system administrators to remotely receive alerts in real-time. The system administrator notification fail-safe mechanism allows administrators to patch detected security events and minimize how long the data is compromised more efficiently. However, some issues still exist for IPS, IDS, and notification alert responses. Therefore, a more centralized system has become the focal point of current research.

2.2 — Security Information and Event Management System Alerts

Current security emphasis highlights implementing an SIEM system over an IDS, IPS, or IRS, as the SIEM system integrates all its capabilities with additional security mechanisms. The development of a next-generation SIEM framework has recently introduced the integration of artificial intelligence (AI) with SIEMs to minimize the continuous problem of low-quality intrusion alerts [29].

Furthermore, AI-assisted notification systems may mitigate issues where the SIEM system detects too much data. While the framework is inaccessible for now, it underscores a genuine concern for alert fatigue. However, by implementing the Wazuh SIEM, alert fatigue is lessened through its easily configured alert and log configuration options. Therefore, many new studies have sought to customize Wazuh in ways that build upon the previous IDS, IPS, and IRS research.

2.2.1 — Wazuh Integrations to Produce Notification Alerts

Wazuh is capable of various pre-configured and customizable responses based on detected intrusions. However, focusing on cyber espionage mitigation techniques, Wazuh can also be integrated with third-party software to provide highly efficient and remote alerts about detected malicious activity. A brand-new study in [30] uses Wazuh's customizable integration capabilities to send detected event alerts through the messaging app Telegram. Outputting the alerts to Telegram allows the configured recipient(s) nearly immediate access to detected malicious activity with the corresponding data collected. However, this study does not discuss the applications of said integration to mitigate further data compromise caused by unaware users' continuing transmissions in a compromised network. Additional published research on Wazuh notification customization is currently minimal. However, two personal projects, [31] and [32], recently documented their integration of Wazuh with the messaging apps Discord and Microsoft Teams using Hypertext Transfer Protocol (HTTP) requests and webhooks, respectively. However, these projects only show a baseline webhook integration technique and do not expound upon custom integration's capabilities and use cases. A more

thorough analysis of the utilization of these functions, alongside more involved code, is necessary for network environment applications.

The advancements of IPSs, IDSs, IRSs, and the more current SIEMs have been substantial in cybersecurity and network security. However, cyber-attacks, such as cyber espionage, that threaten confidential data and intelligence have also advanced. Previously, studies have sought to enhance intrusion system alerts' quality, efficiency, and adaptability. However, with the most recent developments in the cybersecurity field, approaches have highlighted the need to implement and develop centralized SIEM systems for notification alerts. The holistic nature of the alerts generated from SIEM systems more accurately detects malicious activity within the entire network environment. The research on the Wazuh SIEM’s capabilities and notification customizations provides the advantages that previous research soughtgrouped alerts, accessibility, remote access, and high levels of efficiency. However, there is still a period between an initial intrusion and vulnerability patching. However, by enhancing the customizations to Wazuh’s alert integrations, centralized end-user awareness of malicious events can be accomplished.

CHAPTER 3: PROBLEM STATEMENT AND SOLUTION

Network environments serve as a ubiquitous cornerstone for intelligence operations today. Hence, network security ensures the confidentiality, integrity, and availability of confidential data and intelligence. Without it, these operations will fail, negatively affecting the intelligence field and national security. While traditional security measures enhance network resilience, they do not render networks impervious to attack. Even in a non-traditional, heterogeneous approach, which employs a mixture of hardening techniques and addresses unpatchable vulnerabilities, the risk of threat actors infiltrating the system persists. A robust network security framework has become even more imperative in recent years due to the growing complexity of advanced social engineering and phishing techniques successfully penetrating older environments. The holistic security of the network and its endpoints is vital to approach the modern digital domain. Furthermore, a recent shift to combat the efficacy of social engineering has emphasized inherent distrust of end devices and users and the implementation of an SIEM. By acting as the central nervous system to the network environment's security, these practices have allowed efficient notifications and responses to malicious events, minimizing the damage caused by an attack. However, even through this reduction, end users connected to segmented, compromised networks still have no channel to receive information about the security of their network.

3.1 — Problem Statement

Cyber espionage has become a primary concern within intelligence and other confidential data fields. With its advancing efficacy using social engineering

and phishing emails, there is still great concern regarding preventing such attacks and minimizing the damages of a successful attack. If a successful intrusion is detected, suspending confidential data and intelligence transmissions is paramount until system administrators patch the network. Recent shifts in security have emphasized fail-safe mechanisms and preparing the network environment for the event of an attack. However, many systems have yet to implement a centralized, accessible channel that educates end users on network compromises that occur. Previous research endeavors have focused on enhancing notification alert capabilities by developing adaptive, automatic, efficient, and fail-safe solutions [25, 26, 27]. Recent studies have underscored the significance of implementing holistic Security Information and Event Management (SIEM) solutions [29, 30]. Additionally, practices to segment end users and implement the Zero Trust model have risen, seeking to reduce the damages of successful social engineering. Nevertheless, the advancements in network security notification alerts and security practices do not inherently provide a mechanism that mitigates additional compromise caused by end users continuing transmissions across the network in the event of an attack.

3.1.1 — Problem Impact

Without applying mechanisms to minimize human error itself, even if users are segmented, and alerts are more efficient, current standards are not enough to reduce the loss of data confidentiality, integrity, and availability. There is still a delay between when the malicious actor penetrates the system and when the system administrators resolve the network intrusion. Within this time frame, segmented

users may still be unknowingly transmitting intelligence and confidential data across the insecure network segment, providing the malicious actor with even more data. Moreover, without informing end devices, the network is compromised, and any confidential data they communicate is at risk. While security practices have done well in advancing techniques and practices to safeguard against cyber espionage caused by social engineering, there are still risks caused by user unawareness.

3.2 — Proposed Solution

This project aims to mitigate cyber espionage by employing current security principles and techniques and alert end users in real-time of network intrusions. By implementing a centralized, accessible channel for network attack information, educated end users can suspend confidential data transmissions until administrators patch the network. The channel acts as a fail-safe mechanism for when unpatchable vulnerabilities, such as human error or unknown attack techniques, are used to execute successful cyber espionage attacks. The solution advocates for a holistic strategy that implements current DiD and ZT standards, such as a segmented design, device hardening, and the utilization of an IDS, IPS, and IDS via an SIEM with XDR capabilities, but also integrates the alerts generated into a channel that all end users can access. Therefore, cyber espionage is mitigated by isolating users and adding security layers, minimizing the network device exploitable vulnerabilities, and configuring an SIEM system with IDS, IPS, and IRS capabilities. Furthermore, the data compromises cyber espionage results, which are minimized by educating all users, not just system administrators, in real-

time. Alerting through a centralized, accessible platform thereby makes end users aware of the need to suspend confidential transmissions if their network segment has been compromised.

3.3 — Project Phases

In this project, the Security Lifecycle Model delineated in [14] works as the conceptual framework for composing the security strategy. This model depicts a structured outline integrated into the methodological approach described in Section 3.4. This project applies this model to a curated network environment overseeing intelligence and confidential data as part of its fundamental operations. Each Security Lifecycle phase contributes to the primary objective of fortifying the security of the network environment to mitigate cyber espionage. The following phases outline the concepts integrated with the project's methodology and approach:

i. *Identification Phase*: This initial phase identifies the intelligence and confidential data managed within the environment as the target of defense (ToD). These assets encompass any confidential, essential data, such as intelligence products composed by analysts, human intelligence (HUMINT), geospatial intelligence (GEOINT), signals intelligence (SIGINT), and more.
ii. *Assessment Phase*: After identification, the assessment phase involves analyzing the ToD to record all vulnerabilities. Minimal isolation, one-dimensional security, default network device configurations, unnecessary user access, human susceptibility to manipulation, and lack of awareness are the primary vulnerabilities exploited to execute cyber espionage. An

undetected cyber espionage attack exploiting these vulnerabilities poses a significant risk to data confidentiality, integrity, and availability.

iii. *Protection Phase*: Building upon insights from the assessment phase, this phase aims to implement security principles, practices, systems, and notification alerts to mitigate the identified vulnerabilities. The DiD and ZT paradigms guide the network design to implement micro-segmentation to limit access user access to only those necessary. Additionally, robust hardening techniques are applied to network devices, patching their vulnerabilities, and implementing stateless firewalls through access control lists (ACLs). Furthermore, deploying the Wazuh SIEM facilitates real-time monitoring and alerting of network and host activities. This project integrates a notification alert system with the SIEM to provide attack awareness to end users and prevent further utilization of a compromised network. Moreover, notifying all users of potential cyber espionage attacks prevents users from continuing ToD communications on the compromised network.

iv. *Monitoring Phase*: Subsequently, the effectiveness of the safeguards applied is monitored in this phase, verifying the successful operations of all security mechanisms deployed following the strategy. Monitoring the SIEM system and notification alerts sent in Microsoft Teams is the primary focus to ensure the detection, prevention, and responsiveness to security events.

3.4 — Project Approach and Methodology

The project executes the network security strategy utilizing two primary, essential software applications: Wazuh and Microsoft Teams. Wazuh, the open-

source SIEM with XDR capabilities, is particularly suited for the strategy because of its accessibility and customizability. The system, upon installation, has several use cases, monitors end devices with and without an agent service running, detects and alerts malicious activity in real-time, and can seamlessly integrate many attack and malware databases. Furthermore, Wazuh can be configured with customized integrations, such as one for Microsoft Teams. Teams is a communications platform utilized by over 1 million organizations today, according to statistics in [33], and can be configured with designated messaging channels within a ‘team,’ which can be configured to allow incoming webhooks, discussed further in Section 3.9.

Furthermore, Microsoft Teams is accessible through computers or mobile devices, allowing remote access to teams and channels. The strategy is cost-effective and integrates an already widely used platform with cybersecurity. Therefore, these two software applications are utilized to achieve the project’s main objectives. The network security strategy is conducted through five sequential phases, integrating the Security Lifecycle phases depicted in Section 3.3. The methodological phases organize the holistic security approach into each essential strategic component. The remainder of Chapter 3 illustrates each phase of execution and is structured as follows:

I. *Network Design*: This phase entails the conceptualization and design of the network topology. Layered security mechanisms and micro-segmentation are applied to integrate ZT and DiD principles.

II. *Program and Device Preparation*: The preparation phase encompasses configuring the software applications used to curate the project's virtualized network environment, which is comprised of virtual routers, switches, and end devices.

III. *Network Hardening*: During this phase, the SIEM system is deployed within the most internal network of focus, intended to monitor the environment, detect security incidents in real-time, respond to events, and generate alerts.

IV. *SIEM Implementation*: During this phase, the SIEM system is deployed within the most internal network of focus, intended to monitor the environment, detect security incidents in real-time, respond to events, and generate alerts.

V. *Microsoft Teams Alert Integration*: This phase configures the SIEM system to generate notification alerts. The centralized, accessible alert channel is implemented by using created scripts to integrate Wazuh's alerts into a dedicated Microsoft Teams.

VI. *Notification Testing*: The SIEM system's detection and response configurations are tested by conducting outlined attacks to trigger rules and generate alerts. Alert generation is verified by logs in the Wazuh Dashboard compared against messages sent in Teams. The timeframe between the detected attack and any notifications sent in Teams is also recorded.

This structured methodology and systematic approach depict the framework for creating and reproducing the proposed security strategy in the future.

3.5 — Network Design

A customized DiD model facilitates the layers the network security strategy implements. Fig. 2 illustrates the application of the DiD model within the project's environment. Configuring the monitoring and response features protects the ToD, with the Wazuh server cluster monitoring and recording the ToD agent data to generate alerts and check databases.

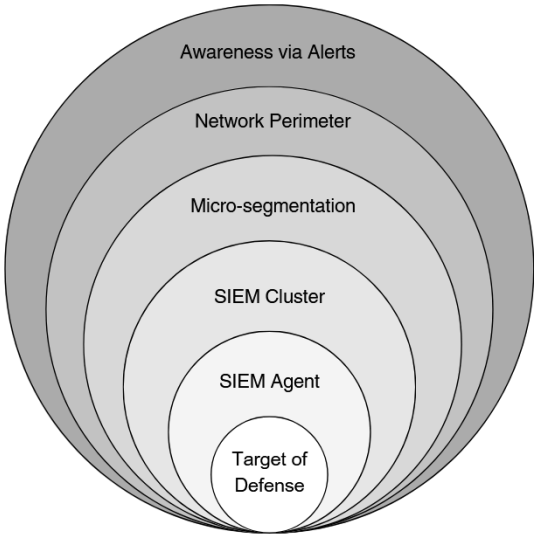


Fig. 2. Security strategy DiD model.

Micro-segmenting the environment into several subnetworks acts as an additional layer, with the hardened network perimeter following, and the last line of defense, or the fail-safe mechanism, minimizes cyber espionage comptonization by alerting all users. This initial model creates layers throughout the project execution, with the network perimeter and micro-segmentation being implemented within Sections 3.5.1 and 3.5.2, respectively.

3.5.1 —Network Perimeter Security

The strategy establishes the micro-segmentation and network perimeter layers utilizing the ZT model. The network splits into layers and micro-segments users into subnetworks containing only their most essential resources and devices. Fig. 3 illustrates the strategy's designed network topology.

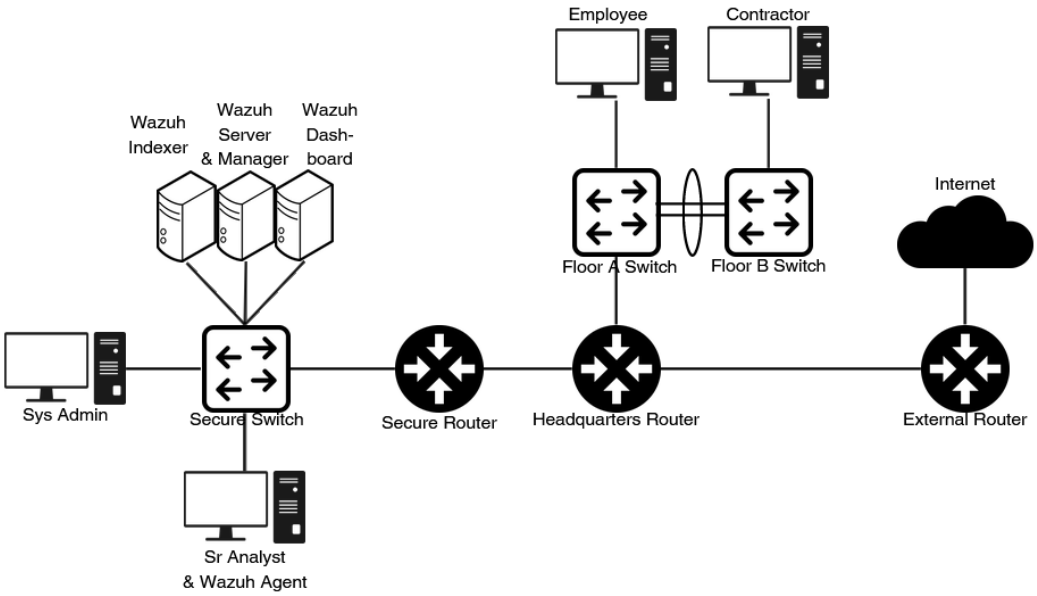


Fig. 3. Network topology design.

Three routers are deployed in the network environment to separate the subnetwork, containing the intelligence and confidential data, from the Internet as much as possible. This most internal local area network (LAN) is the focus of the stateless firewall, SIEM system deployment, and notification alerts, as it contains the ToD. The most external router has no connected end devices to enhance security layering. Between these routers, the middle router connects all other end devices within the organization that do not need direct access to intelligence and confidential data. The three router configurations are divided equally to add further

layers and mitigate a singular point of failure for three essential services. The External Router executes Network Address Translation (NAT) and Port Address Translation (PAT) to hide the internal addressing scheme from external networks. The Headquarters Router is a DHCP server that provides IP addresses for all internal devices. Then, the ACLs are implemented on the Secure Router to filter between the VLANs, as well as between the Secure Intelligence Branch and all other networks.

3.5.2 — Micro-Segmentation Security

To further network segmentation, the three switches within the topology are configured with various virtual LANs (VLANs), logically grouping end devices and acting as their own subnetwork with a differing Internet Protocol version 4 (IPv4, also referred to as just IP) pool. A Demilitarized Zone (DMZ) VLAN is configured within the Headquarters sector to represent an area external users must bypass before accessing any of the used subnetworks within the internal network. Implementing DiD and ZT creates many isolated network segments, serving as multiple layers that malicious actors must bypass before penetrating the subnetworks within the Secure Intelligence Branch, containing the Secure Router, Secure Switch, and connected end devices.

The environment is subneted using Variable Length Subnet Masking (VLSM) on IP number 10.0.0.0/8, resulting in the IP addressing scheme in Table 1. Appendix A contains the additional topology IP addressing information.

Table 1. Micro-segmented network IP addressing scheme.

VLAN Subnetworks						
Headquarters Subnetworks						
VLAN ID	VLAN Name	Network Address	CIDR	Subnet Mask	Usable Range	Wildcard Mask
112	Employees	10.0.0.0	/15	255.254.0.0	10.0.0.1-10.1.255.254	0.1.255.255
212	Contractors	10.2.0.0	/17	255.255.128.0	10.2.0.1-10.2.127.254	0.0.127.255
312	SrPersonnel	10.2.128.0	/18	255.255.192.0	10.2.128.1-10.2.191.254	0.0.63.255
412	TrustedClients	10.2.192.0	/19	255.255.224.0	10.2.192.1-10.2.223.254	0.0.31.255
512	DMZ	10.2.224.0	/22	255.255.252.0	10.2.224.1-10.2.227.254	0.0.3.255
612	Management	10.2.228.0	/23	255.255.254.0	10.2.228.1-10.2.229.254	0.0.1.255
Secure Intelligence Branch Subnetworks						
18	Analysts	10.2.230.0	/25	255.255.255.128	10.2.230.1-10.2.230.126	0.0.0.127
28	Administration	10.2.230.128	/26	255.255.255.192	10.2.230.129-10.2.230.190	0.0.0.63
38	Resources	10.2.230.192	/28	255.255.255.240	10.2.230.193-10.2.230.206	0.0.0.15
48	Management	10.2.230.208	/29	255.255.255.248	10.2.230.209-10.2.230.214	0.0.0.7
Wide Area Network (WAN) Subnetworks						
WAN Link		Network Address	CIDR	Subnet Mask	Usable Range	Wildcard Mask
Branch to Headquarters		10.2.230.216	/30	255.255.255.252	10.2.230.217-10.2.230.218	0.0.0.3
Headquarters to External		10.2.230.220	/30	255.255.255.252	10.2.230.221-10.2.230.222	0.0.0.3

With the network design established, the project executes the remaining methodological phases. The several IP pools ensure that devices cannot communicate directly without being routed through a router. Thus, if a cyber espionage attack were to succeed in one segment, the attack would likely need another attack to access other segments. However, if users in a compromised subnetwork continue their transmissions, it may allow attacks to collect data without additional attacks, necessitating the centralized user awareness channel.

3.6 — Program and Device Preparation

This project employs an entirely virtualized environment to implement the proposed security strategy. Specifically, VMware Workstation Pro 17.5.0 build-22583795 hosts the GNS3 version 2.2.44.1 server [33], which serves as the platform for running all network and end devices. The chosen security solution is Wazuh (version 4.7.2), the open-source SIEM and XDR software [21]. Within this virtualized environment, the following devices are utilized:

- (3) Cisco IOSv 15.9(3)M2 Virtual Routers
- (3) Cisco IOSv15.2 Virtual Switches
- (7) Ubtuntu 20.04 LTS Virtual Machines (VMs)

Additionally, Wazuh is integrated to send its alerts into the Microsoft Teams (version 24033.811.2738.2546) “Wazuh Alerts” channel within the “Secure Intelligence Branch” Team. It is essential to note that only the three routers, the Secure Switch, and the Secure Intelligence Branch VMs fall within the scope of the SIEM deployment to mitigate cyber espionage. For a detailed visual presentation of the GNS3 network topology, refer to Appendix B.

3.7 — Network Hardening

Network hardening measures are vital for mitigating the vulnerabilities of the default network device configurations. Without hardening the network perimeter, the layer is effectively useless. The security configurations recommended by [14] are applied to all network devices, enabling encrypted, remote access, and local database authentication. Additionally, connections to the

devices are restricted using limited authentication retries and session timers. Cisco Discovery Protocol (CDP) is disabled on all devices to mitigate reconnaissance attacks. The switch and router configurations differ when extending beyond basic configurations due to their differing functionalities. Therefore, the devices are hardened with different techniques, all of which eliminate as many vulnerabilities as possible that are present at default.

3.7.1 — Switch Hardening

The switch is highly susceptible to attack due to its essential functionalities used to forward frames received from the ingress port. Therefore, switch hardening is implemented by disabling default-enabled protocols and configurations, such as Dynamic Trunking Protocol (DTP) and VLAN 1. These, at default, are often exploited by attackers to launch a cyber spying attack. Additionally, all ports are configured so that they are highly restricted, only allowing intentional connections by enabling each end connection port, assigning all ports to used or unused VLANs, enabling PortFast, BPDUGuard, and configuring Port-Security. To minimize exploits used to conduct poisoning and spoofing attacks, which may lead to cyber spying, DHCP snooping and Dynamic Address Resolution Protocol (ARP) Inspection (DAI) are configured on all switches. The Secure Switch running configuration file contents are in Appendix C, with the Floor A and Floor B switch running configurations in Appendix D and E, respectively. By applying these configurations, default configurations are no longer exploitable, and access to the switch is heavily restricted, making the switch hardened. However, these do not affect the router’s security configurations.

3.7.2 — Router Hardening

Essential router hardening techniques mostly follow the basic hardening configurations described in Section 3.7, except for one key component: ACLs. The ACLs filter traffic being routed, acting as a stateless firewall. Three ACLs are configured on the Secure Router. The firewall configurations are contained within the Secure Routers running configuration file contents in Appendix F. The Headquarters and External Routers' running configurations are in Appendix G and H, respectively. The "Personnel-Firewall" and "Management-Firewall" are utilized on the corresponding inbound VLAN interfaces and the "External-Firewall," applied to the inbound WAN connection to the Headquarters Router, filters incoming from outside the Secure Intelligence Branch. The ACLs are configured only to permit essential services for the corresponding VLANs or external connections. The "Personnel-Firewall," filtering inbound traffic from the Analysts, Administration, and Resources VLANs, only permits necessary traffic, such as web traffic, DHCP services, mail services, and SIEM system services. The "Management-Firewall," filtering Management VLAN traffic (containing the Sys Admin, Wazuh server cluster, and Secure Switch), requires permittance of SecureShell (SSH) traffic. On the "External-Firewall," traffic incoming from outside the Secure Intelligence Branch is filtered similarly to the non-management VLANs, with further restriction of only permitting the Headquarters Router to send incoming DHCP traffic.

3.8 — SIEM Implementation

The project deploys the Wazuh SIEM system as an IDS, IPS, and IRS within the project environment. Its extensive capabilities facilitate a comprehensive

approach to safeguarding intelligence and sensitive data. Therefore, its deployment within the network is crucial for mitigating cyber espionage and serving as a basis for centralized notification alerts. The Wazuh SIEM operates through three central components within the Wazuh server cluster: the Wazuh Indexer, the Wazuh Server (housing the Wazuh manager), and the Wazuh Dashboard. The Wazuh agent, installed on the Sr Analyst host that manages intelligence, records event data on the host to send to the manager for analysis. The Wazuh Indexer keeps the data logs, and the Wazuh Dashboard is a platform used to access Wazuh's centralized, web-based user interface (web UI), which displays all network environment information.

3.8.1 — Wazuh SIEM Architecture

The project deploys Wazuh within the environment based on the primary objective of establishing a centralized, accessible notification platform that can aid in cyber espionage mitigation. The Sr Analyst host, running the Wazuh agent, follows the usual transmission processes to access external destinations, which routes its data through three routers before exiting the GNS3 environment. During this, however, the Wazuh agent sends data to the Wazuh manager to process in its analysis engine. Fig. 4. illustrates how data flows between the primary network environment components.

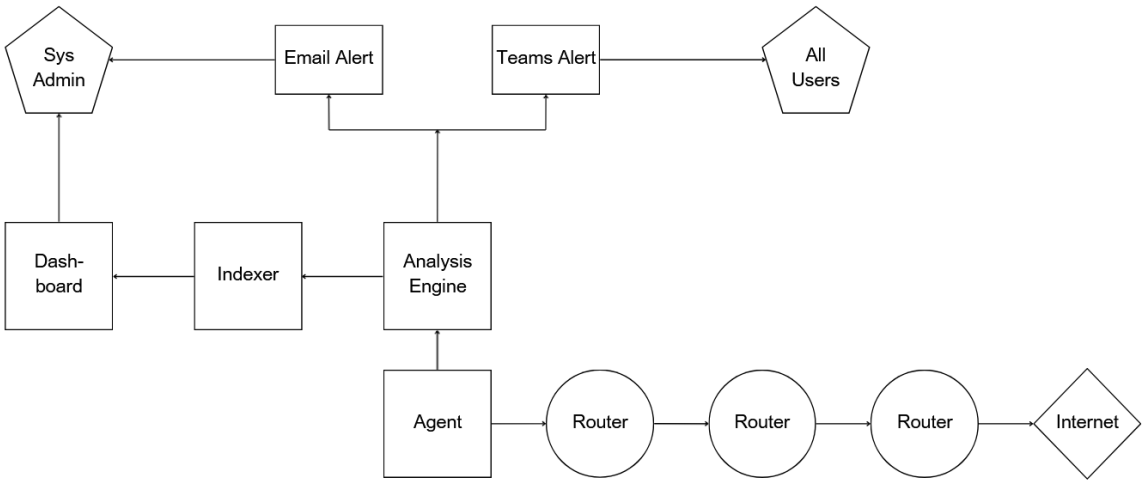


Fig. 4. Diagram of the data flow across the network environment.

The SIEM system engine either outputs a Teams alert to all users and/or an email alert to the Sys Admin's email address, `cyberalerts.sysad@gmail.com`. Additionally, the engine sends data to the Indexer, which provides the Dashboard with the necessary information to display on the web UI. From the Wazuh Dashboard, system administrators can see alerts and other generated data through the web UI. The framework additionally pertains to how the project tests the Teams message alerts. Comprehensive documentation in [35] guides installing, configuring, and implementing the system within this project. Additionally, Wazuh configurations in [36] that are not pre-configured are implemented to act as a basis for executing intrusions to test the Teams alert messages.

3.8.2 — Wazuh SIEM Configuration Overview

The primary configuration file for the Wazuh manager, the `/var/ossec/etc/ossec.conf` file on the Wazuh Server, holds all local server configurations in Extensible Markup Language (XML). The completed

configuration contents of the file are in Appendix I. The configuration file on the Sr Analyst for the Wazuh agent, under the same name and directory, serves a similar purpose and is in Appendix J. Both are customized to alert for the pre-determined tests outlined in Section 4.1.2. To emphasize cyber espionage mitigation, several of these customizations correspond to attacks leading to data confidentiality, integrity, and availability loss. The configurations added to the SIEM within the environment follow the “Proof of Concept” documentation in [36] and include the following:

i. Detecting and blocking a connection from a known attacker using the AlienVault IP Reputation Database [37].
ii. Detecting the execution of a malicious command.
iii. Detecting file integrity with Wazuh’s file integrity module (FIM) by monitoring file and directory modifications.
iv. Detecting and removing malware using the VirusTotal database [38].
v. Detecting unauthorized commands and/or processes.
vi. Detecting network intrusions by integrating the NIDS solution, Suricata.

Additional Wazuh capabilities, configured by default, aid in cyber espionage mitigation and are utilized for the testing processes documented in Section 4.1.2. These capabilities include the following:

i. Detecting rootkits and hidden processes running on the end device.
ii. Detecting web-based attacks, such as SQL injections and shellshocks.
iii. Detecting brute-force attacks to access the device via SSH.

Active responses are also added to the configuration file to correspond to some of the possible attacks, which are all to be executed during the testing phase in Section 4.2. The list of possible configurations for Wazuh is expansive, but the listed capabilities are necessary detection configurations that can detect and prevent techniques to execute cyber espionage.

Wazuh's detection capabilities provide robust security against several methods of cyber espionage, which are part of the pre-determined configurations outlined in [35] and [36]. The configurations are implemented into the environment through different tasks, such as adding custom rules to the `/var/ossec/etc/rules/local_rules.xml` Wazuh Server file (see Appendix K for the file contents), enabling active responses, integrating open-source databases and a NIDS, and adding files/directories to be monitored. For instance, its integration with the malware detection database VirusTotal is especially beneficial to cyber espionage mitigation, as over 70% of cyber espionage attacks are executed by successful phishing attacks [8]. Phishing attacks plant malicious, hidden code or malware onto the system, which this project's deployment is configured for its detection. For instance, commands and active responses can both be configured. The `<command>` configuration in the Wazuh Server's configuration file, depicted in Fig. 5., calls the installed script `remove-threat.sh`, documented in [36].

```
<command>
  <name>remove-threat</name>
  <executable>remove-threat.sh</executable>
  <timeout_allowed>no</timeout_allowed>
</command>
```

Fig. 5. Remove-threat command.

Then, the active response feature is implemented using the `<active-response>` tag. When added within the same file, the configuration enables Wazuh to run the `remove-threat` command, calling the `remover-threat.sh` script if rule identifier (ID) 87105 triggers. This rule relates to VirusTotal, which denotes that unusual file contents detected from a VirusTotal query will trigger the rule and generate an alert.

```
<active-response>
  <disabled>no</disabled>
  <command>remove-threat</command>
  <location>local</location>
  <rules_id>87105</rules_id>
</active-response>
```

Fig. 6. Configuration for the VirusTotal `<active-response>` action.

VirusTotal queries automatically execute when one of the custom rules in the `/var/ossec/etc/rules/local_rules.xml` file triggers are set to detect modifications to the `/root` directory. Therefore, if malicious files are unintentionally installed from cyber espionage-related phishing emails, not only are notification alerts expected to be output, but Wazuh will run the `remove-threat.sh` file to remove the file, preventing the attack. Another instance of Wazuh directly acting as a countermeasure against cyber espionage is its pre-defined configurations, such as rootkit, unauthorized process, and hidden process detection. Malicious

command detection is configured, additionally, by using the rules in [36] as a base to make rules that trigger when network sniffer tcpdump is run. All four attacks relate to common cyber espionage techniques, typically hiding and persisting on the system to gather data on it, often being initiated from some disguised, malicious attachments and/or links. With these configurations and more added to the machines, the alert integration process may begin.

3.9 — Customized Notification Alert Integration

The key feature of this project's strategy is that a fail-safe mechanism is configured to notify connected users of an attack. By integrating the alerts with a widely deployed messaging program, the notification component of the strategy can seamlessly integrate into many pre-exiting environments. The fundamental component of this custom integration is Microsoft Teams' compatibility with using incoming webhooks. Actions in one application, when using webhooks, can be sent into another via HTTP POST messages. Therefore, the integration of Teams and Wazuh can send generated security alerts into a centralized, accessible channel.

3.9.1 — Overview of Notification Alert Integration

Wazuh does not have integration scripts inherently designed for Microsoft Teams, but it does have a script that integrates Wazuh with Slack using webhooks. This pre-installed configuration is used as a foundation and guide for the custom Teams integration. Additionally, based on the research conducted to integrate Wazuh alerts into third-party software, customizable integrations such as these are certainly possible [30, 31, 32]. The primary objective is to send Microsoft Teams alerts utilizing a webhook to the target Teams channel, which receives its alert data

via HTTP Post requests. The `/var/ossec/integrations/slack` and `/var/ossec/integrations/slack.py` files on the Wazuh Server are used as a foundation to create the custom integration for Microsoft Teams. Modifying these files, as well as adding the integration into the Wazuh Server's configuration, allows alerts to be sent into Microsoft Teams, where all users on the network can be added.

3.9.2 — Microsoft Teams Webhook Configuration

The first phase of integration involves creating the Teams channel and the webhook to append to the Wazuh Server's configuration. In Microsoft Teams, a new team named "Secure Intelligence Branch" has been created, with a channel named "Wazuh Alerts" inside. In the channel's management settings, the "Incoming Webhook" feature, part of the "Connectors" category, is utilized to generate the webhook that allows Teams to receive Wazuh's alert data, where the unique webhook Uniform Resource Locator (URL) is recorded. Creating the webhook for the specific channel ensures that the alerts are only sent to the designated channel and will not clutter other essential Teams areas.

3.9.3 —Webhook-Based Integration Design

Once the webhook is created, the two base Slack files are customized to work with the Teams webhook, utilizing documentation in [31, 32, 35, 39] as a baseline reference. The Slack scripts created by Wazuh are copied into the files that are used for the Teams, `/var/ossec/integrations/custom-alerts` and `/var/ossec/integrations/custom-alerts.py`. The file `/var/ossec/integrations/custom-alerts` is only slightly adjusted by changing

the path stored in `WPYTHON_BIN` to `framework/python/bin/python3`. The `custom-alert` file's contents are in Appendix L. The remainder of the third-party integration customization is to the `/var/ossec/integrations/custom-alerts` file, with said adjustments contained in Appendix M.

Primarily, the code's payload function, which contains the Teams "Message Card" alert payload, requires the most customization. This function defines how the Microsoft Teams alert is sent within the channel. Microsoft Teams uses message cards as the base for incoming webhook message configurations. Hence, the documentation in [39] is referenced to understand the structure of message cards. Firstly, the level variable is defined in the function and retrieves its value from the alert data from Wazuh. Alert level ranges are adjusted to correlate to a different color, defined by a hex code for green, yellow, and red. The `msg = {} ` line defines the empty dictionary that is filled with the data to send into the Teams channel, and three empty lists, `facts`, `sections`, and `actions`, are also declared. The primary message card options, settings, type, theme color, and summary are added within the first section.

The message card is customized to display details about the detected event and alert. The Wazuh agent's name and agent ID of where the activity occurred are appended to the `facts` list. Then, `facts.extend` is used to add multiple values relating to the alert, such as the location of the alert, rule ID triggered, alert level, and event log contents, into the message card with corresponding labels. Therefore, the users obtain information regarding the context of the alert.

To add message card-specific contents, `sections.extend` is used to provide the message card with a title of “Wazuh Alert” and subtitle of “"Possible network environment security event. If alert level is 10 or higher, suspend all confidential communications until further notice.” Adding these details enhances user awareness even further, ensuring users know the network environment may be compromised. While some attacks may trigger alerts lower than ten, user discretion is necessary. Otherwise, operations may be too often halted. Regardless, the alert enables users to make such discretions. An attacker-related image is also incorporated into the message card for visual appeal.

A unique feature is integrated into the message card within the `actions` list. This section of code creates a functional button at the bottom of the message card. The following lines of code create this feature:

```
actions.append({
    '@type': "OpenUri",
    'name': "Learn More",
    'targets': [{
        'os': "default",
        'uri': "https://www.cio.gov/"
  }]
})
```

The `@type` declaration enables the button feature. The text on the button is set to “Learn More,” and upon clicking the button in Teams, the machine’s default web browser opens the specified website. For this integration, the `https://www.cio.gov/` URL serves as a placeholder for what could be an organization’s security policies and procedures. With all three lists being fully

adjusted, the function is compatible with Microsoft Teams' message cards. The debug functions follow the code in the original `slack.py' file, as well as the code in [32], which are not the focal areas that necessitate change. Furthermore, the additional configurations and customizations within the code are designed to reference the projects discussed in [31, 32, 35, 39].

3.9.4 — Wazuh Notification Alert Configuration

The final phase of integrating Microsoft Teams and Wazuh is adding the integration into the Wazuh Server's `/var/ossec/etc/ossec.conf` file. Within the file, under an `<integration>` tag, the name of the created files, `custom-alerts` is added, alongside the generated webhook, the alert level that should trigger the message to be sent in Teams, and the format. The alert level to trigger the integration is set to seven to prevent low-quality alerts from being sent to all users, considering the research concerns discussed in Chapter 2. Fig. 7 demonstrates how the custom integration appears in the Wazuh configuration file.

Fig. 7. Configuration of Microsoft Teams integration.

With the Wazuh Server configurations applied, upon restart of the Wazuh manager service, the custom integration is fully added to the Wazuh SIEM, and is expected to send notification alerts from Wazuh into the "Wazuh Alerts" channel.

CHAPTER 4: RESULTS AND ANALYSIS

After the project had completed a blended integration of an IPS, IDS, and IRS through the implementation of an SIEM, the results of the configurations were collected. All network hardening configurations operated as expected, however, the network security strategy's key focus was the implementation of a notification system for security administrators and connected end users. Hence, this project records multiple results from the strategy's deployment. The network security strategy utilizing notification alerts was operationally successful and successfully detected, reacted, and notified end users of attacks, aiding cyber espionage mitigation. When a malicious actor breaches all the network's outer security layers, the SIEM implementation and fail-safe notification alert mechanism still minimize cyber espionage. End users were notified in real-time of a detected attack and received instructions on how to react and able to access additional procedural resources. The notification alerts contained all the needed information for users to understand what security threat they were facing. The notification alert element of the strategy was successful due to the SIEM successfully monitoring, logging, detecting, reacting, and alerting malicious activities to the appropriate platforms in real time. These expected results intend to ensure that advancing cyber espionage attacks, exploiting humans and unpatchable vulnerabilities, are minimized.

4.1 — Notification Alert Testing Process

From the Wazuh SIEM, multiple configured and pre-defined capabilities successfully detected and mitigated, when applicable, cyber espionage attacks. The results were collected by defining variables that measured the operational

success and time efficiency of the notification alerts and the detection and response to a range of attacks that could lead to cyber espionage. The notification system's success was dependent on the SIEM's success; therefore, all alert recipients from Fig. 4 were tested. The variables and executed tests measured the strategy's cyber espionage mitigation when a malicious actor bypasses all other defense layers by examining the alert-success, time efficiencies, and the active response execution.

4.1.1 — Variable Selection

Several variables were selected to measure the strategy's ability to mitigate cyber espionage. For each attack the system successfully minimized, all the corresponding rule IDs were gathered. The notification system relied upon the success of Wazuh, and therefore, factors that exhibited the alerting success were measured:

i. *Alert via Dashboard (AvD)*: Recorded whether alerts levels two and higher were received via Dashboard as a Boolean expression.
ii. *Alert via Email (AvE)*: Recorded whether alerts levels three and higher were received via email as a Boolean expression.
iii. *Alert via Teams (AvT)*: Recorded whether alerts levels seven and higher were received via Teams as a Boolean expression.

Additionally, to measure time efficiency, the time between the attack launch and the time when a Microsoft Teams alert occurred was recorded in seconds as the Time to Alert (TTA). The TTA was gathered by utilizing a stopwatch to record the time between the attack's initiation and when Microsoft Teams first received an

alert for the attack. A formula was used to represent how the TTA was collected is outlined by the following:

Time Users are Unaware = (Time Attack Launches) - (Time Teams Alert Appears)

Successful attack mitigations using a configured response labeled Active Response Mitigation (ARM) were recorded as a Boolean expression. Attacks without one configured were recorded as “N/A” for “not applicable.”

4.1.2 — Test Procedures

Ten pre-selected attacks were executed to measure the selected variable documented in the previous section, with further procedures guiding them. The attacks were duplicated in a reloaded environment to confirm the recorded elements, denoted as Test I and Test II. Before the tests, slight modifications were made to the configurations and the network environment. To ensure the results were not affected by overlapping configurations, the stateless firewall ACLs were removed, also simulating the event of an attacker bypassing all other defense layers. Apache2 was configured on the Sr Analyst so that several of the ten attacks could be executed, as the addition of web-based attacks gave results of a broader capability range. Lastly, the email alert recipient was replaced with an alternative address in the configuration, as the previous one had too much previous data that would not provide a fresh environment for resulting alerts. The following descriptions list the ten executed attacks that were selected to showcase Wazuh’s range of capabilities, as described in [36]:

i. *Known Attacker*: The Wazuh Dashboard assumed the role of a known attacker. The IP address of the known attacker, IP 10.2.230.213, was added to AlienVault's database list and attempted to connect to the Sr Analyst using HTTP.
ii. *SQL Injection*: The Sys Admin assumed the role of an attacker and attempted to execute a well-known SQL injection pattern to the Sr Analyst.
iii. *Brute-Force*: The Sys Admin ran Hydra to attempt a brute-force attack to SSH to the Sr Analyst.
iv. *Shellshock*: The Sys Admin attempted a shellshock attack using a malicious HTTP request to the Sr Analyst.
v. *Malicious Command*: The network sniffer tcpdump was ran on the Sr Analyst.
vi. *File Integrity*: A file was modified in a monitored directory on the Sr Analyst.
vii. *Malware and Removal*: The EICAR test malware file was installed on the Sr Analyst.
viii. *Unauthorized Process*: The port scanner netcat was ran on the Sr Analyst.
ix. *Hidden Process*: The rootkit Diamorphine was installed and loaded on the Sr Analyst.
x. *Network Intrusion*: The Wazuh Server assumed the role of the attacker to make unauthorized pings to the Sr Analyst.

The attack execution and resulting findings showcased the success of Wazuh's several attack mitigation/minimization capabilities and the notification alert

mechanism's time efficiency. The findings from the tests are discussed and analyzed throughout the remainder of Chapter 4.

4.2 — Results and Discussion

Users managing intelligence and/or confidential data in environments beyond this project commonly have no platform to learn security risks, so they may be active components in mitigating cyber spying. Providing end users in such environments with security awareness enables them to suspend transmissions and prevent the malicious actor from obtaining additional data. Test I and II findings indicate that Wazuh successfully detected, alerted, and responded (when configured) to all the attacks. Furthermore, the results indicate that the Microsoft Teams alerts are received in real time, considering the project environment. The key findings from the execution of Test I and Test II are recorded in Section 4.2.1 and Section 4.2.2.

4.2.1 — Key Findings

From the conducted tests, several of the collected findings indicate the notification alert system was successful and time efficient, indicating that Wazuh's detection and alerting capabilities were also successful. Furthermore, Wazuh successfully executed its two configured responses, preventing cyber espionage-related attacks from occurring. The generated alerts all contained the necessary information to educate and instruct end users and the capability to mitigate cyber espionage through awareness. Test I and Test II's primary results, recorded in Table 2 and Table 3, are organized in a tabular format for readability.

Table 2. Notification alerts generated from Test I.

Notification Alerts Generated from Test I.						
Executed Attack	Rule ID(s)	Alert via Dashboard	Alert via Email	Alert via Teams	Time to Alert (TTA)	Active Response Mitigation (ARM)
Known Attacker	100100	True	True	True	14.04	True
SQL Injection	31103, 86601	True	True	True	10.25	N/A
Brute-Force	5551, 5763	True	True	True	11.78	N/A
Shellshock	31168, 86601	True	True	True	9.08	N/A
Malicious Command	80730	True	True	True	7.25	N/A
File Integrity	533	True	True	True	8.70	N/A
Malware and Removal	87105, 100092	True	True	True	8.07	True
Hidden Process	100210, 10051	True	True	True	10.68	N/A
Unauthorized Process	5132, 521	True	True	True	10.45	N/A
Network Intrusion	86601	True	True	True	7.54	N/A

Table 3. Notification alerts generated from Test II.

Notification Alerts Generated from Test II.						
Executed Attack	Rule ID(s)	Alert via Dashboard (AvD)	Alert via Email (AvE)	Alert via Teams (AvT)	Time to Alert (TTA)	Active Response Mitigation (ARM)
Known Attacker	100100	True	True	True	6.65	True
SQL Injection	31103, 86601	True	True	True	6.82	N/A
Brute-Force	5551, 5763	True	True	True	10.57	N/A
Shellshock	31168, 86601	True	True	True	5.15	N/A
Malicious Command	80730	True	True	True	4.98	N/A
File Integrity	533	True	True	True	3.70	N/A
Malware and Removal	87105, 100092	True	True	True	6.20	True
Hidden Process	100210, 10051	True	True	True	3.75	N/A
Unauthorized Process	5132, 521	True	True	True	5.63	N/A
Network Intrusion	86601	True	True	True	7.40	N/A

All twenty attacks generated at least one alert received on all three platforms, indicated by the “True” expressions for the AvD, AvE, and AvT. Each platform received alerts for the corresponding rule IDs that matched with the platform’s alert level configurations. Furthermore, Wazuh prevented the known attacker attack by successfully dropping the known malicious IP at the firewall for sixty seconds. Particularly relating to cyber espionage, Wazuh detected that malware had been installed and immediately removed the malware file.

In addition to these general key findings, the contents of the Teams alerts successfully educated users on all configured information fields regarding the attack and instructed users on how to proceed. Furthermore, the “Learn More” button successfully opened a web browser to the configured URL so that users could access additional information. Each of the alerts displayed the title, subtitle, image, button, and the corresponding attack data, one showcased in Fig. 8.

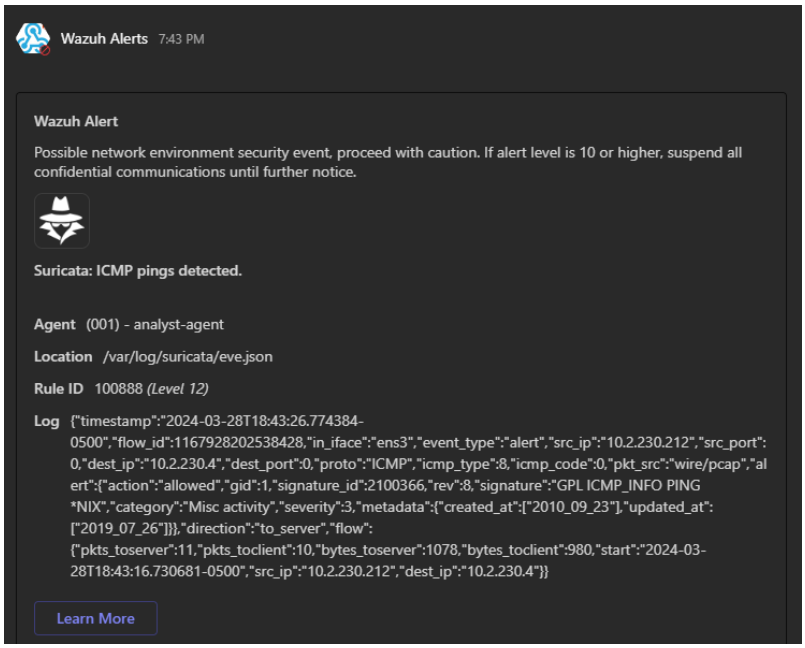


Fig. 8. Microsoft Teams Wazuh-generated alert.

The Wazuh email notifications displayed similar data contents, and the Wazuh Dashboard contained all possible details about each alert, per Wazuh's standard design. The Wazuh Dashboard also displayed statistical visualizations of the alerts generated, shown in Fig. 9, illustrating all alerts, including those from the twenty attack executions.

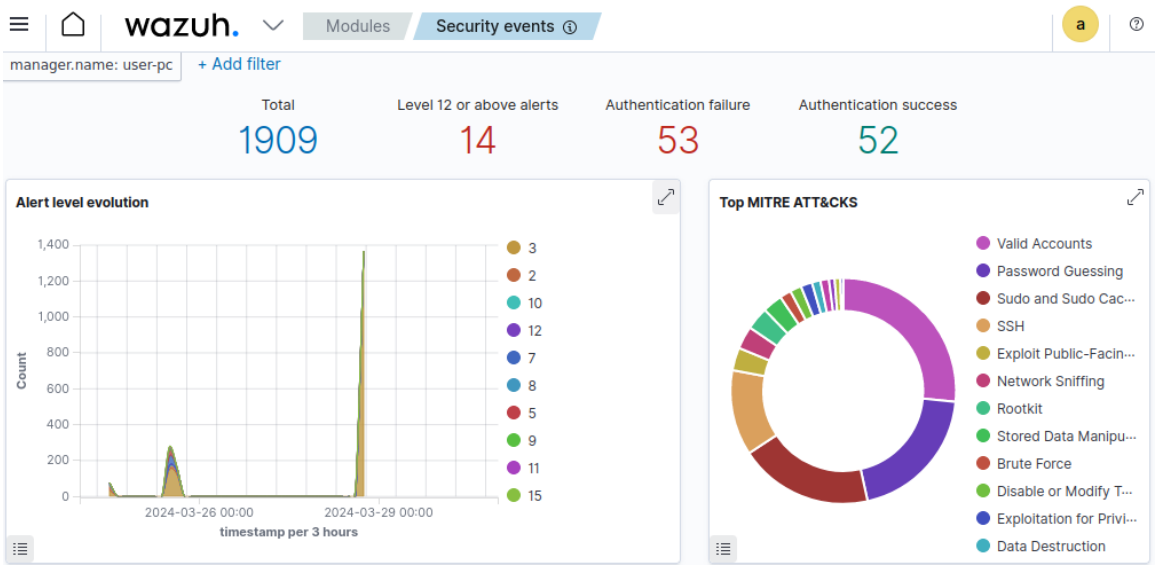


Fig. 9. Wazuh Dashboard alerts display.

From the Dashboard, attack attempts included password guessing, SSH access, network sniffing, rootkits, data manipulation, brute-force attacks, and data destruction. Thus, the findings demonstrate the success of Wazuh and the notification alert system in the event of a malicious actor executing attacks after bypassing all other defense layers.

The results indicated that the notification alerts in Microsoft Teams were successful and time-efficient for their environment. Referencing the TTA data collected, statistical measures were calculated to better depict the data set. Table 4 summarizes the statistical findings.

Table 4. Notification alert time range statistics.

Time to Alert (TTA) Statistics (in seconds)			
TTA Statistics	Test I (Ten TTAs)	Test II (Ten TTAs)	Test I and Test II (Twenty TTAs)
Average	9.78	6.09	7.93
Median	9.67	5.92	7.47
Range	6.79	6.87	10.34

The TTAs differed slightly between Test I and Test II. However, the primary purpose of executing two tests was to generally verify the results were not false representations. The slight variation between the two tests does not affect the overall findings that the notification alerts were time efficient. During each test, the time frames from when each attack launched, and an alert was sent in Teams only ranged approximately 7 seconds from one another. The attacks' TTAs are visualized in Fig. 10.

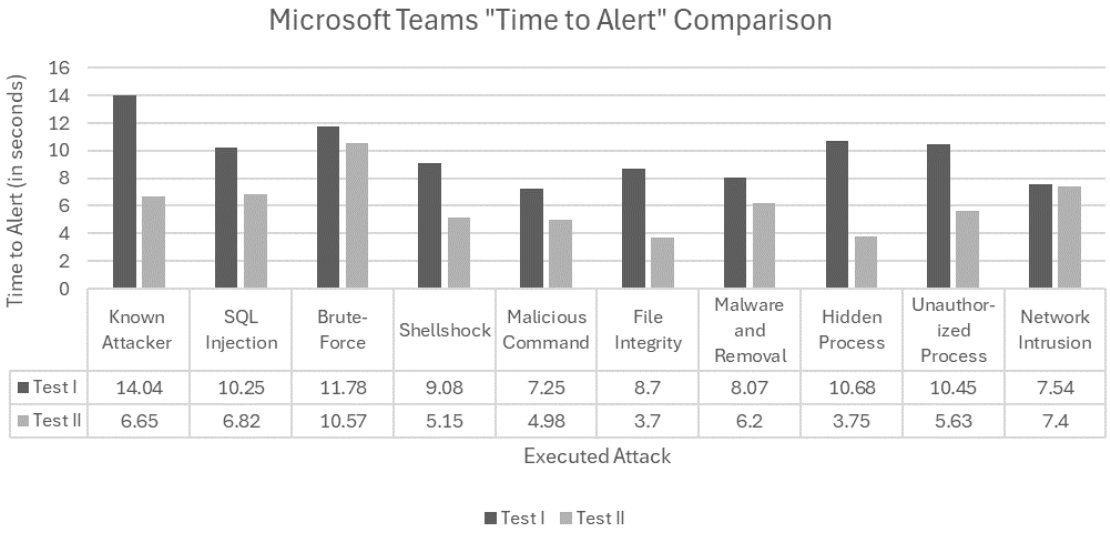


Fig. 10. Microsoft Teams notification alert efficiency comparison.

All Teams alerts were received within at least the first fifteen seconds of the attack. The TTA was shortened for all attacks during Test II, with the number of seconds

varying per attack. The longest TTA occurred during the known attacker attack, 14.04 seconds, while the shortest occurred during the file integrity attack, alerting within just 3.7 seconds. Slightly slower speeds are expected due to the project's environment, discussed further in Section 4.3. User notification across all twenty tested attacks occurred at an average of 7.93 seconds after the attack was launched. Therefore, the data set successfully documented the time efficiency of the network security strategy, indicating it minimized cyber espionage via real-time user awareness.

4.2.2 — Data Analysis

All the collected data sets and test results were expected and demonstrated that the SIEM successfully detected, alerted, and responded as configured. All three platforms were notified of the malicious activity, and the Microsoft Teams alerts contained information that educated users about the security risk, instructed users on how to respond, and provided a button to access additional resources. The alerts to the Dashboard and email included lower alert levels because they are expected to be seen by only system administrators managing the environment. Only high-level alerts (seven and higher) were sent to the Teams channel, minimizing the overflow of low-quality alerts. The network security strategy successfully mitigated cyber espionage because it notified end users about the attacks in real-time. It enabled users to immediately suspend transmissions, mitigating additional data loss until the network gets patched.

4.3 — Limitations

The project and network security strategy have several limitations. One of the primary limiting categories pertains to the strategy's implementation. The notification alert aspect of the strategy is limited to small-scale network segments, as alerting larger user pools may hinder unnecessary operations, pose a security concern, and are simply unfeasible. However, if applied to properly segmented subnetworks, a team can be made for each small subnetwork needing notifications, as all that would change in the strategy's configuration is additional integration sections in the Wazuh Server configuration file, listing an additional webhook to the extra team. The strategy is also limited by its reliance on users to take precautions upon security threat awareness. However, the strategy's value persists with proper user training about response procedures.

The additional limiting category encompasses factors of the project's environment that affected this strategy's capabilities and time efficiency. GNS3 utilized excessive Random Access Memory (RAM) on the physical computer, often causing it to crash. So, the VMs were configured with limited memory and space capacities that restricted Wazuh's functionality. Additionally, the environment limited the network devices to low bandwidths, lengthening the TTAs. However, the average TTA was still less than ten seconds, providing near-immediate awareness to users. These limitations do not discard the strategy's value to current practices, creating robust security designs to mitigate cyber spying. Instead, their acknowledgment further defines the strategy's overarching goals and the value it brings to them.

CHAPTER 5: CONCLUSIONS AND FUTURE WORK

Malicious events can be reported in real-time to security administrators and any connected users on the network. By doing so, users can make informed decisions as to whether their continued use of the network would compromise confidential data and intelligence. With the growing success of cyber espionage executed through social engineering and human manipulation, the push to mitigate such attacks and minimize their consequences has been profound. This project implements a coalescence of current security paradigms and practices but also introduces a fail-safe mechanism that mitigates cyber espionage by notifying all network users of an attack, instructing users to take preventive action, and suspending network usage.

The network security strategy employs the Zero Trust and Defense-in-Depth models, network hardening practices, and intrusion detection, prevention, and response through a security event and information management system (SIEM). Integrating the Wazuh SIEM system alerts into the messaging software Microsoft Teams, this project achieves centralized, accessible user notification that mitigates the success of cyber espionage. Furthermore, this project finds the following regarding the network security strategy:

i. *Current network security paradigms and practices are fundamental to cyber espionage mitigation. Eliminating exploitable network device vulnerabilities and micro-segmenting users are necessary strategy components, as they prevent the SIEM from being the singular line of defense.*

ii. *The deployment of an SIEM accounts for cyber espionage attacks exploiting unpatchable vulnerabilities and/or utilizing user manipulation by successfully detecting and preventing a broad range of attacks with its active response mechanisms.*

iii. *The integration of Wazuh and Microsoft Teams results in real-time, remotely accessible notifications to all end users. It also successfully instructs end users on how to prevent compromising intelligence and confidential data, thereby minimizing the success of the cyber espionage attack.*

Therefore, the network security strategy's integration of Microsoft Teams notification alerts furthers the current studies on cyber espionage mitigation by recommending a holistic security design that integrates user response into the overall mitigation measures.

5.1 — Project Implications

The project's network security strategy contributes to current paradigms and practices that mitigate cyber attacks and account for unpatchable vulnerabilities. These current standards do so by layering security and micro-segmenting users to contain successful attacks within a smaller network segment. However, this still leaves the segmented network at risk, as its users can still transmit confidential data, leaking it to the malicious actor. This project minimizes this occurrence by incorporating end users into the security response. Automatically notifying all users of the security risk the moment it occurs enables users to prevent compromising data by suspending their transmissions. This strategy applies a real-time

notification alert system to the current security paradigms and practices for holistic security.

5.2 — Future Work

The successful cohesion of integration projects and cyber espionage mitigation studies bring new research opportunities. Even through user awareness, susceptibility to social engineering persists. Additionally, users may not understand enough of the implications of what the security alerts indicate. As research attempts to integrate artificial intelligence (AI) with next-generation SIEM alerts, its application to end-user awareness and prevention should additionally be explored. AI-based SIEMs could provide end users with alert data that is intelligently re-formatted to provide personalized, easy-to-read instructions for each user. Additionally, new studies should research applying SIEM agents to more than end devices, but also a user's mail and cloud to prevent malicious programs from ever crossing onto the physical machine. As the implementation of next-generation SIEMs is explored, research can advance cyber espionage mitigation by furthering user-focused security mechanisms.

REFERENCES

[1] A. Civuli, S. Luma-Osmani, E. Rufati, and G. Arifi, “Cyber espionage consequences as growing threat,” Dept. Informatics Faculty of Natural Sciences and Mathematics, Univ. of Tetova, Tetovo, North Macedonia, 2022.

[2] C. A. Theohary, “Defense Primer: Cyberspace Operations,” Congressional Research Service, Washington, D.C., USA, 2023.

[3] “Advanced persistent threat.” National Institute of Science and Technology. https://csrc.nist.gov/glossary/term/advanced_persistent_threat (accessed Jan. 23, 2024)

[4] “Man in the middle”. National Institute of Science and Technology. https://csrc.nist.gov/glossary/term/mitm (accessed Jan. 23, 2024)

[5] “Zero day attack.” National Institute of Science and Technology. https://csrc.nist.gov/glossary/term/zero_day_attack (accessed Jan. 23, 2024)

[6] “Social Engineering.” National Institute of Science and Technology. https://csrc.nist.gov/glossary/term/social_engineering (accessed Mar. 21, 2024)

[7] “Phishing.” National Institute of Science and Technology. https://csrc.nist.gov/glossary/term/phishing (accessed Mar. 21, 2024)

[8] “2023 Data Breach Investigations Report.” Verizon. https://www.verizon.com/business/resources/reports/dbir/2023/summary-of-findings/ (accessed Mar. 17, 2024).

[9] “Significant Cyber Incidents.” Center for Strategic & International Studies. https://www.csis.org/programs/strategic-technologies-program/significant-cyber-incidents (accessed Mar. 17, 2024)

[10] “Election Security Spotlight – CIA Triad.” Center for Internet Security. https://www.cisecurity.org/insights/spotlight/ei-isac-cybersecurity-spotlight-cia-triad (accessed Mar. 4, 2024)

[11] R. Pfau, “The Security Lifecycle,” SANS Technology Institute, Rockville, MD, USA, 2003.

[12] K. Stouffer, T. Zimmerman, C. Tang, J. Lubell, J. Cichonski, J. McCarthy, “Cybersecurity Framework Manufacturing Profile,” *National Institute of Standards and Technology Internal Report 8183*, Sept. 2017, doi: doi.org/10.6028/NIST.IR.8183.

[13] K. Holl, “OSI Defense in Depth to Increase Application Security,” SANS Technology Institute, Rockville, MD, USA, 2003.

[14] Y. Bhaiji CCIE No. 9305, “Network Security Technologies and Solutions,” 1st ed., Indianapolis, IL, USA: Cisco Press, 2008

[15] “Advancing Zero Trust Maturity Throughout the Network and Environment Pillar,” National Security Agency, Fort Meade, MD, USA, 2024.

[16] S. Rose, O. Borchert, S. Mitchell, S. Connelly, “Zero Trust Architecture,” *NIST Special Publication 800-207*, Aug. 2020, doi: 10.6028/NIST.SP.800-207.

[17] T. Snoke, “Common Network Security Tools and Capabilities,” Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, 2019.

[18] “Hardening Network Devices,” National Security Agency. https://media.defense.gov/2020/Aug/18/2002479461/-1/-1/0/HARDENING_NETWORK_DEVICES.PDF (accessed Mar. 17, 2024).

[19] A.K. Pathan, “The State of the Art in Intrusion Prevention and Detection,” 1st ed., Boca Raton, FL, USA: CRC Press, 2014

[20] G.G. Granadillo, S. González-Zarzosa, R. Diaz, “Security Information and Event Management (SIEM): Analysis, Trends, and Usage in Critical Infrastructures,” *Sensors* 2021, vol. 21, no. 14:4759, doi: 10.3390/s21144759

[21] *Wazuh*, Wazuh, 2015.

[22] D. Borbor, L. Wang, S. Jajodia, and A. Singhal, “Surviving unpatchable vulnerabilities through heterogenous network hardening options,” *Journal of Computer Security*, vol. 26, no. 6, pp. 761-789, Oct. 2018, doi: 10.3233/JCS-171106.

[23] F. Cuppens and A. Miège, “Alert correlation in a cooperative intrusion detection framework,” in *Proceedings 2002 IEEE Symposium on Security and Privacy*, Berkeley, California, USA, May 2002, pp. 202-215, doi: 10.1109/SECPRI.2002.

[24] C. Kruegel and W. Robertson, “Alert verification determining the success of intrusion attempts,” in *Conference on Detection of Intrusions and Marlware & Vulnerability Assessment*, U. Flegel and M. Meier, Ed., Dortmund, Germany, July 6-7, 2004, pp. 23-38, doi: 10.17877/DE290R-2013.

[25] D.J. Ragsdale, C.A. Carver, J.W. Humphries, and U.W. Pooch, “Adaptation techniques for intrusion detection and intrusion response systems,” in *IEEE International Conference on Systems, Man and Cybernetics*, Nashville, TN, USA, Oct. 2000, doi: 10.1109/ICSMC.2000.884341.

[26] S. Anwar et al., “From intrusion detection to an intrusion response system: fundamentals, requirements, and future directions,” *Algorithms*, vol. 10, no. 2, Mar. 2017, doi: 10.3390/a10020039.

[27] M.A. Helmiawan, E. Julian, Y. Cahyan, and A. Saeppani, “Experimental evaluation of security mornitoring and notification on network intrustion detection system for server security,” in *9th International Conference on Cyber and IT Service Management*, Bengkulu, Indonedia, Sept. 2021, pp. 1-6, doi: 10.1109/CITSM52892.2021.9588988.

[28] R. Muwardi, H. Gao, H.U. Ghifarsyam, M. Yunita, A. Arriziki, and J. Andika, “Network security monitoring system via notification alert.” *Journal of Integrated and Advanced Engineering*, vol. 1, no. 2, pp. 113-122, Sept. 2021, doi: 10.51662/jiae.v1i2.22.

[29] T. Ban, T. Takahashi, S. Ndichu, and D. Inoue, “Breaking Alert Fatigue: AI-Assisted SIEM Framework for Effective Incident Response,” *Journal of Applied Sciences*, vol. 26, no. 6, pp. 761-789, Oct. 2018, doi: 10.3233/JCS-171106.

[30] F.I. Farrel, I. Mardianto, A.S. Quamar, “Implementation of Security Information & Event Management (SIEM) Wazuh with Active Response and Telegram Notification for Mitigating Brute Force Attacks on the GT-USAKTI Information System,” *Intelmatics*, vol. 13, no. 11, 6610, May 2023, doi: 10.3390/app13116610.

[31] E. Chaves, “Integrating Wazuh with Discord,” Github. https://eugenio-chaves.github.io/blog/2022/creating-a-custom-wazuh-integration/ (accessed Mar. 21, 2024).

[32] “Sending Alerts to Microsoft Teams from Wazuh,” Infopercept. https://www.infopercept.com/blogs/sending-alerts-to-microsoft-teams-from-wazuh/ (accessed Mar. 21, 2024).

[33] *GNS3.* SolarWinds Worldwide, 2024.

[34] “Microsoft Teams Statistics (2024) – Usage & Revenue,” Demandsage, https://www.demandsage.com/microsoft-teams-statistics/ (accessed Mar. 27, 2024)

[35] “Wazuh Documentation,” Wazuh. https://documentation.wazuh.com/current/index.html (accessed Mar. 1, 2024)

[36] “Wazuh Proof of Concept Guide,” Wazuh. https://documentation.wazuh.com/current/proof-of-concept-guide/index.html (accessed Mar. 1, 2024)

[37] *AlienVault*, AlienVault, 2007.

[38] *VirusTotal*, VirusTotal, 2004.

[39] “Create Incoming Webhooks,” Microsoft Teams Learn. https://learn.microsoft.com/en-us/microsoftteams/platform/webhooks-and-connectors/how-to/add-incoming-webhook?tabs=newteams%2Cdotnet (accessed Mar. 21, 2024)

APPENDICES

Appendix A. Network IP Addressing Scheme Table.

Table 5. Network IP addressing scheme.

HeadquartersRouter Interfaces

Description	Interface	IP Address	Subnet Mask	CIDR
Link to Headquarters LAN	Gi1	N/A	N/A	N/A
Link to VLAN 112	Gi1.112	10.0.0.1	255.254.0.0	/15
Link to VLAN 212	Gi1.212	10.2.0.1	255.255.128.0	/17
Link to VLAN 312	Gi1.312	10.2.128.1	255.255.192.0	/18
Link to VLAN 412	Gi1.412	10.2.192.1	255.255.224.0	/19
Link to VLAN 512	Gi1.512	10.2.224.1	255.255.252.0	/22
Link to VLAN 612	Gi1.612	10.2.228.1	255.255.254.0	/23
WAN Link to SecureRouter	Gi2	10.2.230.218	255.255.255.252	/30
WAN Link to ExternalRouter	Gi3	10.2.230.222	255.255.255.252	/30

Headquarters Device Interfaces

Device	Interface	IP Address	Subnet Mask	CIDR	Default Gateway
FloorASwitch	VLAN 612	10.2.228.2	255.255.254.0	/23	10.2.228.1
FloorBSwitch	VLAN 612	10.2.228.3	255.255.254.0	/23	10.2.228.1
Ubuntu-Employee	NIC1	DHCP	DHCP	DHCP	DHCP
Ubuntu-Contractor	eth0	DHCP	DHCP	DHCP	DHCP

Secure Router Interfaces

Description	Interface	IP Address	Subnet Mask	CIDR
Link to Secure Intel LAN	Gi1	N/A	N/A	N/A
Link to VLAN 18	Gi1.18	10.2.230.1	255.255.255.128	/25
Link to VLAN 28	Gi1.28	10.2.230.129	255.255.255.192	/26
Link to VLAN 38	Gi1.38	10.2.230.193	255.255.255.240	/28
Link to VLAN 48	Gi1.48	10.2.230.209	255.255.255.248	/29
WAN Link to HeadquartersRouter	Gi2	10.2.230.217	255.255.255.252	/30

Secure Intelligence Branch **Device Interfaces**

Device Name	Interface	IP Address	Subnet Mask	CIDR	Default Gateway
Secure Switch	VLAN 48	10.2.230.210	255.255.255.248	/29	10.2.230.209

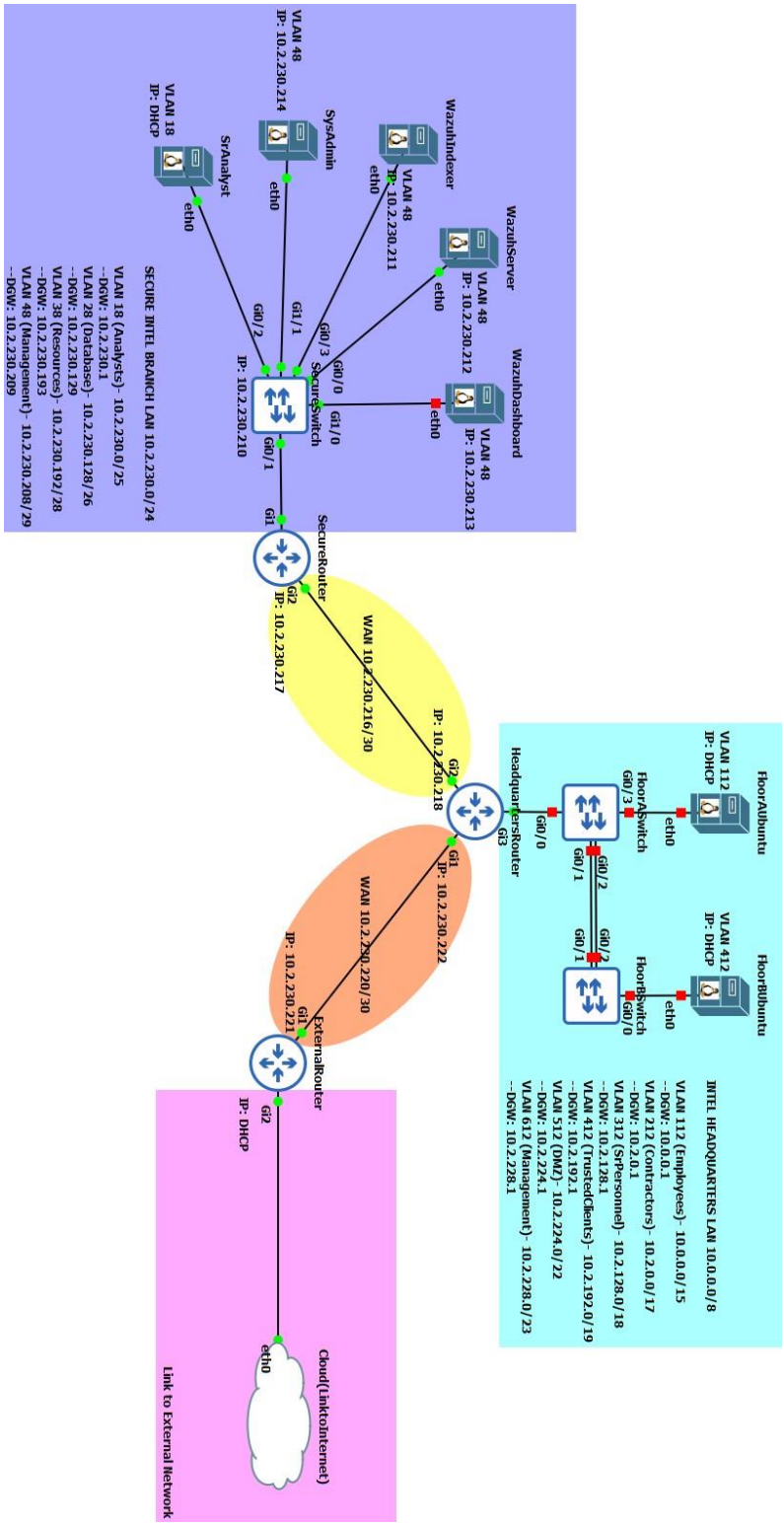
Table 5. *Cont*.

Indexer	ens3	10.2.230.211	255.255.255.248	/29	10.2.230.209
Server	ens3	10.2.230.212	255.255.255.248	/29	10.2.230.209
Dashboard	ens3	10.2.230.213	255.255.255.248	/29	10.2.230.209
Sys Admin	ens3	10.2.230.130	255.255.255.192	/26	10.2.230.129
Sr Analyst	ens3	DHCP	DHCP	DHCP	DHCP

External Router Interfaces

Description	Interface	IP Address	Subnet Mask	CIDR
Link to the Internet	Gi0/0	DHCP	DHCP	DHCP
WAN Link to HeadquartersRouter	Gi0/1	10.2.230.22	255.255.255.252	/30

Appendix B. GNS3 Network Topology.



Appendix C. Secure Switch Running Configuration.

```
SecureSwitch#sh run
Building configuration...

Current configuration : 6670 bytes
!
! Last configuration change at 21:02:15 UTC Mon Mar 25 2024
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service compress-config
!
hostname SecureSwitch
!
boot-start-marker
boot-end-marker
!
!
enable secret 5 $1$7JbD$6LLY/eTXmaD3yPl3No/JA.
!
username secureadminswitch password 7 073F331C440C1A1156
no aaa new-model
!
!
!
!
!
ip arp inspection vlan 18,28,38
ip arp inspection validate src-mac
!
!
```

```
!
ip dhcp snooping vlan 18,28,38
no ip dhcp snooping information option
ip dhcp snooping
ip domain-name cyberintelligence.com
ip cef
no ipv6 cef
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
no cdp run
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface GigabitEthernet0/0
 switchport access vlan 48
 switchport mode access
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0cc9.e42b.0000
```

```
 switchport port-security
 ip arp inspection limit rate 300
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 18,28,38,48
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 128
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 negotiation auto
 ip dhcp snooping trust
!
interface GigabitEthernet0/2
 switchport access vlan 18
 switchport mode access
 switchport port-security maximum 2
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0c36.9f35.0000
 switchport port-security mac-address sticky 0c94.83b9.0000
 switchport port-security
 ip arp inspection limit rate 300
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet0/3
 switchport access vlan 48
```

```
 switchport mode access
 switchport port-security maximum 2
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0c84.4a04.0000
 switchport port-security mac-address sticky 0cd9.5849.0000
 switchport port-security
 ip arp inspection limit rate 300
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet1/0
 switchport access vlan 48
 switchport mode access
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0cd6.f247.0000
 switchport port-security
 ip arp inspection limit rate 300
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet1/1
 switchport access vlan 48
 switchport mode access
 switchport port-security maximum 2
 switchport port-security mac-address sticky
 switchport port-security mac-address sticky 0c22.c61f.0000
 switchport port-security mac-address sticky 0cac.affa.0000
 switchport port-security
 ip arp inspection limit rate 300
```

```
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet1/2
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/3
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/0
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/1
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/2
 switchport access vlan 118
 switchport mode access
 shutdown
```

```
 negotiation auto
!
interface GigabitEthernet2/3
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/0
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/1
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/2
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/3
 switchport access vlan 118
 switchport mode access
 shutdown
 negotiation auto
!
interface Vlan1
```

```
 no ip address
 shutdown
!
interface Vlan48
 ip address 10.2.230.210 255.255.255.248
!
ip default-gateway 10.2.230.209
ip forward-protocol nd
!
ip http server
ip http secure-server
!
ip ssh time-out 60
ip ssh authentication-retries 2
ip ssh version 2
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
!
!
!
!
!
control-plane
!
banner exec ^C
**********************************************************************
***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
*
* education. IOSv is provided as-is and is not supported by Cisco's
*
* Technical Advisory Center. Any use or disclosure, in whole or in
part, *
* of the IOSv Software or Documentation to any third party for any
*
```

```
* purposes is expressly prohibited except as otherwise authorized by
*
* Cisco in writing.
*
**********************************************************************
***^C
banner incoming ^C
**********************************************************************
***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
*
* education. IOSv is provided as-is and is not supported by Cisco's
*
* Technical Advisory Center. Any use or disclosure, in whole or in
part, *
* of the IOSv Software or Documentation to any third party for any
*
* purposes is expressly prohibited except as otherwise authorized by
*
* Cisco in writing.
*
**********************************************************************
***^C
banner login ^C
**********************************************************************
***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
*
* education. IOSv is provided as-is and is not supported by Cisco's
*
* Technical Advisory Center. Any use or disclosure, in whole or in
part, *
* of the IOSv Software or Documentation to any third party for any
*
* purposes is expressly prohibited except as otherwise authorized by
*
* Cisco in writing.
*
**********************************************************************
***^C
banner motd ^CWarning: Authorized Access Only^C
```

```
!
line con 0
 exec-timeout 1 30
 password 7 107E1B490F12111F4D
 login
line aux 0
line vty 0 4
 password 7 073F331C440C1A1156
 login local
 transport input ssh
!
!
end

SecureSwitch#
```

Appendix D. Floor A Switch Running Configuration.

```
FloorASwitch#sh run
Building configuration...
Current configuration : 5944 bytes
!
! Last configuration change at 20:38:23 UTC Sun Mar 17 2024
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service compress-config
!
hostname FloorASwitch
!
boot-start-marker
boot-end-marker
!
!
enable secret 5 $1$7bEC$XWzYtys5JT8yulxRGruCr.
!
username flooraadmin password 7 08115E1E031C060353
no aaa new-model
!
!
!
!
!
ip arp inspection vlan 112,212,312,412,512
!
!
!
ip dhcp snooping vlan 112,212,312,412,512
```

```
ip dhcp snooping
ip domain-name intelligence-corp.com
ip cef
no ipv6 cef
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
no cdp run
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Port-channel1
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 ip dhcp snooping trust
```

```
!
interface GigabitEthernet0/0
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 negotiation auto
 ip dhcp snooping trust
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 negotiation auto
 channel-group 1 mode active
 ip dhcp snooping trust
!
interface GigabitEthernet0/2
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 negotiation auto
 channel-group 1 mode active
 ip dhcp snooping trust
!
```

```
interface GigabitEthernet0/3
 switchport access vlan 112
 switchport mode access
 switchport port-security maximum 3
 switchport port-security mac-address sticky
 switchport port-security
 ip arp inspection limit rate 300
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet1/0
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/1
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/2
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/3
 switchport access vlan 712
 switchport mode access
```

```
 shutdown
 negotiation auto
!
interface GigabitEthernet2/0
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/1
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/2
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/3
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/0
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
```

```
interface GigabitEthernet3/1
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/2
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/3
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan612
 ip address 10.2.228.2 255.255.254.0
!
ip default-gateway 10.2.228.1
ip forward-protocol nd
!
ip http server
ip http secure-server
!
ip ssh time-out 60
ip ssh authentication-retries 2
```

```
ip ssh version 2
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
!
!
!
!
!
control-plane
!
banner exec ^C
**********************************************************************
      ***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
      *
* education. IOSv is provided as-is and is not supported by Cisco's
      *
* Technical Advisory Center. Any use or disclosure, in whole or in
      part, *
* of the IOSv Software or Documentation to any third party for any
      *
* purposes is expressly prohibited except as otherwise authorized by
      *
* Cisco in writing.
      *
**********************************************************************
      ***^C
banner incoming ^C
**********************************************************************
      ***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
      *
* education. IOSv is provided as-is and is not supported by Cisco's
      *
* Technical Advisory Center. Any use or disclosure, in whole or in
      part, *
* of the IOSv Software or Documentation to any third party for any
      *
```

```
* purposes is expressly prohibited except as otherwise authorized by
      *
* Cisco in writing.
      *
**********************************************************************
      ***^C
banner login ^C
**********************************************************************
      ***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
      *
* education. IOSv is provided as-is and is not supported by Cisco's
      *
* Technical Advisory Center. Any use or disclosure, in whole or in
      part, *
* of the IOSv Software or Documentation to any third party for any
      *
* purposes is expressly prohibited except as otherwise authorized by
      *
* Cisco in writing.
      *
**********************************************************************
      ***^C
banner motd ^CWarning: Authroized Access Only^C
!
line con 0
 exec-timeout 1 30
 password 7 1522195C0E2F283069
 login
line aux 0
line vty 0 4
 password 7 046B195605244F5A48
 login local
 transport input ssh
!
end
```

Appendix E. Floor B Switch Running Configuration.

```
FloorBSwitch#sh run
Building configuration...

Current configuration : 5783 bytes
!
! Last configuration change at 20:38:18 UTC Sun Mar 17 2024
!
version 15.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service compress-config
!
hostname FloorBSwitch
!
boot-start-marker
boot-end-marker
!
!
enable secret 5 $1$tDf7$SOFNczwYDT5KG7szWJn3C.
!
username floorbadmin password 7 01231454510E051B60
no aaa new-model
!
!
!
!
!
ip arp inspection vlan 112,212,312,412,512
!
!
!
```

```
ip dhcp snooping vlan 112,212,312,412,512
ip dhcp snooping
ip domain-name intelligence-corp.com
ip cef
no ipv6 cef
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
no cdp run
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface Port-channel1
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
```

```
 ip dhcp snooping trust
!
interface GigabitEthernet0/0
 switchport access vlan 412
 switchport mode access
 switchport port-security maximum 3
 switchport port-security mac-address sticky
 switchport port-security
 ip arp inspection limit rate 300
 negotiation auto
 spanning-tree portfast edge
 spanning-tree bpduguard enable
 ip dhcp snooping limit rate 300
!
interface GigabitEthernet0/1
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 negotiation auto
 channel-group 1 mode active
 ip dhcp snooping trust
!
interface GigabitEthernet0/2
 switchport trunk allowed vlan 112,212,312,412,512,612
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 812
 switchport mode trunk
 switchport nonegotiate
 ip arp inspection trust
 negotiation auto
```

```
 channel-group 1 mode active
 ip dhcp snooping trust
!
interface GigabitEthernet0/3
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/0
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/1
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/2
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet1/3
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
```

```
interface GigabitEthernet2/0
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/1
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/2
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet2/3
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/0
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/1
 switchport access vlan 712
 switchport mode access
```

```
 shutdown
 negotiation auto
!
interface GigabitEthernet3/2
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface GigabitEthernet3/3
 switchport access vlan 712
 switchport mode access
 shutdown
 negotiation auto
!
interface Vlan1
 no ip address
 shutdown
!
interface Vlan612
 ip address 10.2.228.3 255.255.254.0
!
ip default-gateway 10.2.228.1
ip forward-protocol nd
!
ip http server
ip http secure-server
!
ip ssh time-out 60
ip ssh authentication-retries 2
ip ssh version 2
ip ssh server algorithm encryption aes128-ctr aes192-ctr aes256-ctr
ip ssh client algorithm encryption aes128-ctr aes192-ctr aes256-ctr
```

```
!
!
!
!
!
!
control-plane
!
banner exec ^C
**********************************************************************
      ***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
      *
* education. IOSv is provided as-is and is not supported by Cisco's
      *
* Technical Advisory Center. Any use or disclosure, in whole or in
      part, *
* of the IOSv Software or Documentation to any third party for any
      *
* purposes is expressly prohibited except as otherwise authorized by
      *
* Cisco in writing.
      *
**********************************************************************
      ***^C
banner incoming ^C
**********************************************************************
      ***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
      *
* education. IOSv is provided as-is and is not supported by Cisco's
      *
* Technical Advisory Center. Any use or disclosure, in whole or in
      part, *
* of the IOSv Software or Documentation to any third party for any
      *
* purposes is expressly prohibited except as otherwise authorized by
      *
* Cisco in writing.
      *
```

```
**************************************************************************
      ***^C
banner login ^C
**************************************************************************
      ***
* IOSv is strictly limited to use for evaluation, demonstration and IOS
      *
* education. IOSv is provided as-is and is not supported by Cisco's
      *
* Technical Advisory Center. Any use or disclosure, in whole or in
      part, *
* of the IOSv Software or Documentation to any third party for any
      *
* purposes is expressly prohibited except as otherwise authorized by
      *
* Cisco in writing.
      *
**************************************************************************
      ***^C
banner motd ^CWarning: Authroized Access Only^C
!
line con 0
 exec-timeout 1 30
 password 7 0334495B0C0A22580F
 login
line aux 0
line vty 0 4
 password 7 0334495B0C0A22580F
 login local
 transport input ssh
!
!
end
FloorBSwitch#
```

Appendix F. Secure Router Running Configuration.

```
SecureRouter#sh run
Building configuration...

Current configuration : 9406 bytes
!
! Last configuration change at 22:13:47 UTC Mon Mar 25 2024
!
version 17.3
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service call-home
platform qfp utilization monitor load 80
platform punt-keepalive disable-kernel-core
platform console serial
!
hostname SecureRouter
!
boot-start-marker
boot-end-marker
!
!
enable secret 9
      $9$WltUzvfmTbEhUE$HBFEza9IGN80XFT2mKCvkm0GjGHICpq63EZdXA6ss3I
!
no aaa new-model
!
!
!
!
!
!
```

```
!
ip name-server 192.168.26.1
ip domain name cyberintelligence.com
!
!
!
login on-success log
!
!
!
!
!
!
!
subscriber templating
!
!
!
!
!
!
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
crypto pki trustpoint TP-self-signed-2724869160
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-2724869160
 revocation-check none
 rsakeypair TP-self-signed-2724869160
!
crypto pki trustpoint SLA-TrustPoint
 enrollment pkcs12
 revocation-check crl
!
!
crypto pki certificate chain TP-self-signed-2724869160
 certificate self-signed 01
  30820330 30820218 A0030201 02020101 300D0609 2A864886 F70D0101
      05050030
  31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D
      43657274
  69666963 6174652D 32373234 38363931 3630301E 170D3234 30323230
      32333530
  32335A17 0D333430 32313932 33353032 335A3031 312F302D 06035504
      03132649
  4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32
      37323438
  36393136 30308201 22300D06 092A8648 86F70D01 01010500 0382010F
      00308201
  0A028201 01009BCE 90D8D700 C9C6DD66 E9F495EC 8F022EE2 F5D25BE8
      F0BCAB0E
  41C14835 DBEA0369 6B7639BC 8C48052B 49B8684E 86F39025 BDD648D8
      50812F69
  7A8483BC C73E5732 819CA48A 3217BB98 A7BD2250 5A270968 E3D445F9
      84A43172
  0FB72EB9 4C67D0ED 90C6DB64 A4BF1070 E81C612D 5A75A83F FB8A8EC6
      E749F31C
```

```
  BB4CF2B9 48A7052E ADD0557E B3C7E5B6 6C4732AA 3835CD6F 2774A895
      42D75C18

  CF19E53D C3281065 A00560B9 26383F63 3991B564 6A0E6239 9358DF05
      FFB188C0

  8F044875 9F12D0B4 8CA64B71 A172DB47 16668975 ED65FBBC 62260A85
      C2215722

  22AE0AAA 5807D320 844567FB EC8DDBC3 9CEEEECD 26BB97C7 9A49218C
      2FF37D18

  0782BA68 5DA50203 010001A3 53305130 0F060355 1D130101 FF040530
      030101FF

  301F0603 551D2304 18301680 14074C28 92C1EE2A 8F911C9E FE65863F
      1C64B78C

  AF301D06 03551D0E 04160414 074C2892 C1EE2A8F 911C9EFE 65863F1C
      64B78CAF

  300D0609 2A864886 F70D0101 05050003 82010100 9BC56383 B423F933
      B0BC0ED6

  FCDC12D1 5BCDCE81 FF41AE26 7C7593B0 4CEB0CF5 EAB6F2C6 47158D2F
      F151A1EF

  CE07D9FF 0BC41191 9DC63FC4 A8CC325D 2863D676 48E74B82 E0B77B6E
      B5F88539

  60B687CD 1AEC6EAB 1C9A5960 1DA74505 96DA2FB1 FD9F6253 84C0872F
      CF9CEAE3

  76DB1BDB F4081197 3CA72A01 978616E4 383FB169 6865A133 FF2087BE
      55D941BF

  DCDC53B0 5FE0C450 799C6EEF E4ADE653 019D4A96 0D7B09D8 CC09D5F2
      63A21030

  EDC1EF89 428301CD 46D4AC05 66E09EB5 A19F8532 B2D2D654 50AB0AEB
      7909CFEC

  2F459F3D 12D42062 8ADBBA5E 631A3649 580C1E49 1A9DBE9A ABA1604A
      4B0C4B5C

  E591FCCD F0553A0D 929BD1C7 A308410F 41BA03AE

        quit

crypto pki certificate chain SLA-TrustPoint

 certificate ca 01

  30820321 30820209 A0030201 02020101 300D0609 2A864886 F70D0101
      0B050030

  32310E30 0C060355 040A1305 43697363 6F312030 1E060355 04031317
      43697363

  6F204C69 63656E73 696E6720 526F6F74 20434130 1E170D31 33303533
      30313934
```

```
3834375A 170D3338 30353330 31393438 34375A30 32310E30 0C060355
    040A1305

43697363 6F312030 1E060355 04031317 43697363 6F204C69 63656E73
    696E6720

526F6F74 20434130 82012230 0D06092A 864886F7 0D010101 05000382
    010F0030

82010A02 82010100 A6BCBD96 131E05F7 145EA72C 2CD686E6 17222EA1
    F1EFF64D

CBB4C798 212AA147 C655D8D7 9471380D 8711441E 1AAF071A 9CAE6388
    8A38E520

1C394D78 462EF239 C659F715 B98C0A59 5BBB5CBD 0CFEBEA3 700A8BF7
    D8F256EE

4AA4E80D DB6FD1C9 60B1FD18 FFC69C96 6FA68957 A2617DE7 104FDC5F
    EA2956AC

7390A3EB 2B5436AD C847A2C5 DAB553EB 69A9A535 58E9F3E3 C0BD23CF
    58BD7188

68E69491 20F320E7 948E71D7 AE3BCC84 F10684C7 4BC8E00F 539BA42B
    42C68BB7

C7479096 B4CB2D62 EA2F505D C7B062A4 6811D95B E8250FC4 5D5D5FB8
    8F27D191

C55F0D76 61F9A4CD 3D992327 A8BB03BD 4E6D7069 7CBADF8B DF5F4368
    95135E44

DFC7C6CF 04DD7FD1 02030100 01A34230 40300E06 03551D0F 0101FF04
    04030201

06300F06 03551D13 0101FF04 05300301 01FF301D 0603551D 0E041604
    1449DC85

4B3D31E5 1B3E6A17 606AF333 3D3B4C73 E8300D06 092A8648 86F70D01
    010B0500

03820101 00507F24 D3932A66 86025D9F E838AE5C 6D4DF6B0 49631C78
    240DA905

604EDCDE FF4FED2B 77FC460E CD636FDB DD44681E 3A5673AB 9093D3B1
    6C9E3D8B

D98987BF E40CBD9E 1AECA0C2 2189BB5C 8FA85686 CD98B646 5575B146
    8DFC66A8

467A3DF4 4D565700 6ADF0F0D CF835015 3C04FF7C 21E878AC 11BA9CD2
    55A9232C

7CA7B7E6 C1AF74F6 152E99B7 B1FCF9BB E973DE7F 5BDDEB86 C71E3B49
    1765308B

5FB0DA06 B92AFE7F 494E8A9E 07B85737 F3A58BE1 1A48A229 C37C1E69
    39F08678

80DDCD16 D6BACECA EEBC7CF9 8428787B 35202CDC 60E4616A B623CDBD
    230E3AFB
```

```
  418616A9 4093E049 4D10AB75 27E86F73 932E35B5 8862FDAE 0275156F
      719BB2F0

  D697DF7F 28

        quit

!

license udi pid CSR1000V sn 9NJ0WY2R2J8

diagnostic bootup level minimal

memory free low-watermark processor 71489

!

!

spanning-tree extend system-id

!

username secureadminrouter password 7 08115E1E031C060353

!

redundancy

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!
```

```
!
!
!
!
!
!
interface GigabitEthernet1
 description Link to Secure Intel LAN
 no ip address
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet1.18
 description Link to VLAN 18
 encapsulation dot1Q 18
 ip address 10.2.230.1 255.255.255.128
 ip helper-address 10.2.230.218
 ip access-group Personnel-Firewall in
!
interface GigabitEthernet1.28
 description Link to VLAN 28
 encapsulation dot1Q 28
 ip address 10.2.230.129 255.255.255.192
 ip helper-address 10.2.230.218
 ip access-group Personnel-Firewall in
!
interface GigabitEthernet1.38
 description Link to VLAN 38
 encapsulation dot1Q 38
 ip address 10.2.230.193 255.255.255.240
 ip helper-address 10.2.230.218
 ip access-group Personnel-Firewall in
```

```
!
interface GigabitEthernet1.48
 description Link to VLAN 48
 encapsulation dot1Q 48
 ip address 10.2.230.209 255.255.255.248
 ip helper-address 10.2.230.218
 ip access-group Management-Firewall in
!
interface GigabitEthernet2
 description Link to HeadquartersRouter
 ip address 10.2.230.217 255.255.255.252
 ip access-group External-Firewall in
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
ip forward-protocol nd
ip http server
ip http authentication local
```

```
ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.2.230.218
ip route 10.0.0.0 255.254.0.0 10.2.230.218
ip route 10.2.0.0 255.255.128.0 10.2.230.218
ip route 10.2.128.0 255.255.192.0 10.2.230.218
ip route 10.2.192.0 255.255.224.0 10.2.230.218
ip route 10.2.224.0 255.255.252.0 10.2.230.218
ip route 10.2.228.0 255.255.254.0 10.2.230.218
ip route 10.2.230.220 255.255.255.252 10.2.230.218
ip route 10.2.230.221 255.255.255.255 10.2.230.218
!
ip access-list extended External-Firewall
 5 permit udp host 10.2.230.218 any eq bootps
 10 permit udp host 10.2.230.218 any eq bootpc
 15 permit tcp any any eq smtp established
 20 permit udp host 192.168.26.1 any
 25 permit tcp any any eq 443
 30 permit udp any any eq domain
 35 permit tcp any any eq www
 40 permit tcp any any eq 465
 45 permit tcp any any eq 587
 50 permit tcp any any eq domain
 55 permit tcp any any eq 993
 60 permit tcp any any eq 995
 100 deny   ip any any
ip access-list extended Management-Firewall
 5 permit tcp any any eq 443
 10 permit tcp any any eq domain
 15 permit udp any any eq domain
 20 permit tcp any any eq 465
 25 permit tcp any any eq 587
 30 permit udp any any eq bootps
```

```
 35 permit udp any any eq bootpc
 40 permit tcp any 10.2.230.208 0.0.0.7 eq 1515
 45 permit udp any host 192.168.26.1
 50 permit tcp any any eq www
 55 permit tcp any 10.2.230.208 0.0.0.7 eq 1514
 60 permit tcp host 10.2.230.214 10.2.230.208 0.0.0.7 eq 22
 65 permit tcp any any eq 8888
 70 permit tcp any any eq 993
 75 permit tcp any any eq 995
 80 permit tcp any any eq smtp established
 85 permit tcp any any eq 55000
 100 deny   ip any any
ip access-list extended Personnel-Firewall
 5 permit tcp any any eq 443
 10 permit tcp any any eq domain
 15 permit udp any any eq domain
 20 permit tcp any any eq 465
 25 permit tcp any any eq 587
 30 permit udp any any eq bootps
 35 permit udp any any eq bootpc
 40 permit tcp any 10.2.230.208 0.0.0.7 eq 1515
 45 permit udp any host 192.168.26.1
 50 permit tcp any any eq www
 55 permit tcp any 10.2.230.208 0.0.0.7 eq 1514
 60 permit tcp any any eq 993
 65 permit tcp any any eq 995
 70 permit tcp any any eq smtp established
 100 deny   ip any any
!
!
!
!
!
```

```
!
control-plane
!
!
!
!
!
banner motd ^CWarning: Authroized Access Only^C
!
line con 0
 password 7 12291747180E0F106B
 login
 stopbits 1
line vty 0 4
 password 7 12291747180E0F106B
 login local
 transport input ssh
!
call-home
 ! If contact email address in call-home is configured as sch-smart-
      licensing@cisco.com
 ! the email address configured in Cisco Smart License Portal will be
      used as contact email address to send SCH notifications.
 contact-email-addr sch-smart-licensing@cisco.com
 profile "CiscoTAC-1"
  active
  destination transport-method http
!
!
!
!
!
end
SecureRouter#
```

Appendix G. Headquarters Router Running Configuration.

```
HeadquartersRouter#sh run
Building configuration...

Current configuration : 9005 bytes
!
! Last configuration change at 21:07:57 UTC Mon Mar 25 2024
!
version 17.3
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service call-home
platform qfp utilization monitor load 80
platform punt-keepalive disable-kernel-core
platform console serial
!
hostname HeadquartersRouter
!
boot-start-marker
boot-end-marker
!
!
enable secret 9
      $9$sTSX/dzNko2nc.$P6h2D1JJAs6F9SxVtUWx6km8SIVc9dkBjxh.ie2pBYs
!
no aaa new-model
!
!
!
!
!
!
```

```
!
ip name-server 192.168.26.1
ip domain name intelligence-corp.com
ip dhcp excluded-address 10.0.0.1
ip dhcp excluded-address 10.2.0.1
ip dhcp excluded-address 10.2.128.1
ip dhcp excluded-address 10.2.192.1
ip dhcp excluded-address 10.2.224.1
ip dhcp excluded-address 10.2.230.1
ip dhcp excluded-address 10.2.230.129
ip dhcp excluded-address 10.2.230.193
ip dhcp excluded-address 10.2.230.209
ip dhcp excluded-address 10.2.230.210
!
ip dhcp pool VLAN112
 network 10.0.0.0 255.254.0.0
 default-router 10.0.0.1
 domain-name intelligence-corp.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN212
 network 10.2.0.0 255.255.128.0
 default-router 10.2.0.1
 domain-name intelligence-corp.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN312
 network 10.2.128.0 255.255.192.0
 default-router 10.2.128.1
 domain-name intelligence-corp.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN412
```

```
 network 10.2.192.0 255.255.224.0
 default-router 10.2.192.1
 domain-name intelligence-corp.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN512
 network 10.2.224.0 255.255.252.0
 default-router 10.2.224.0
 domain-name intelligence-corp.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN18
 network 10.2.230.0 255.255.255.128
 default-router 10.2.230.1
 domain-name cyberintelligence.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN28
 network 10.2.230.128 255.255.255.192
 default-router 10.2.230.129
 domain-name cyberintelligence.com
 dns-server 192.168.26.1
!
ip dhcp pool VLAN38
 network 10.2.230.192 255.255.255.240
 default-router 10.2.230.193
 domain-name cyberintelligence.com
 dns-server 192.168.26.1
!
!
!
login on-success log
!
```

```
!
!
!
!
!
!
subscriber templating
!
!
!
!
!
!
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
crypto pki trustpoint TP-self-signed-393707760
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-393707760
 revocation-check none
```

```
 rsakeypair TP-self-signed-393707760
!
crypto pki trustpoint SLA-TrustPoint
 enrollment pkcs12
 revocation-check crl
!
!
crypto pki certificate chain TP-self-signed-393707760
 certificate self-signed 01
  3082032E 30820216 A0030201 02020101 300D0609 2A864886 F70D0101
      05050030
  30312E30 2C060355 04031325 494F532D 53656C66 2D536967 6E65642D
      43657274
  69666963 6174652D 33393337 30373736 30301E17 0D323430 32323130
      30313235
  395A170D 33343032 32303030 31323539 5A303031 2E302C06 03550403
      1325494F
  532D5365 6C662D53 69676E65 642D4365 72746966 69636174 652D3339
      33373037
  37363030 82012230 0D06092A 864886F7 0D010101 05000382 010F0030
      82010A02
  82010100 B3A196B6 3CCD7BE4 940601D6 09735C4D A7526813 8D60600E
      07430C7F
  C59B2870 29647365 5C64451E E0DEBB10 8B44BF8B 42388BD9 4C2E1649
      FC7742E3
  9E0EE5CF AF66D804 FF8EF1C9 BDAA70FD 3861FB82 63C8D56C D7C9C837
      8E2174D9
  656F4E80 84A3857B 99F8AC82 7DA7859B 365E9607 54959560 BA9AFC9C
      CC6B5B61
  366188F7 E422B5CA BEDB481A 7A90DF64 84D9AB79 90912E33 68CCAE0B
      BBB108FD
  01ED871F 94814BA1 FDE09729 73D792B9 CB53703D 540D5EF1 7A2B48C3
      FA3E5C4E
  5133F78F CD44835D 56727A10 F3AFFCA7 3D65461B 571F37AC 364D0F41
      B0399ADC
  E50A6F49 DCB3874C 5206714C B83123CD 91667174 CEE39A68 006B6DA3
      2DF59C28
  D8E28DE7 02030100 01A35330 51300F06 03551D13 0101FF04 05300301
      01FF301F
```

```
  0603551D 23041830 1680142B 0AC572CF 8EC6C9A0 1A9E2EB0 C40EB00B
      9242F430

  1D060355 1D0E0416 04142B0A C572CF8E C6C9A01A 9E2EB0C4 0EB00B92
      42F4300D

  06092A86 4886F70D 01010505 00038201 01003E3D 9B88A3ED 53F05AC4
      A6CF6A59

  E46A697A B3432274 E74DB8A0 E25B0E58 6FBCB29F 5233578A A4D8BBC9
      7B2DE00B

  9348DDCF 25E058B3 F6A9D418 14074FA4 A5E84B24 81C67BD5 1E1AB492
      9AB7F4EF

  1FD525C4 A9257734 F149BBE7 24FBAB73 3448FFE3 C67021DB FF91F1E4
      514D680F

  B4923FB0 0E8DF6AB 076DA3F8 618A8621 4982AF86 7927BD27 E42F5E62
      3191A72D

  1D9E2593 C4282552 14E225D4 D1009462 26A512B2 30D6F602 3CF10877
      527D2E72

  EA4F0819 249FE30B 37D75D4B EC33BC01 2B8AC4D0 5A185886 42B524AC
      86EF6A20

  FAE2E19E 57F9DC91 7ACF890A 69A7C5E7 AC4FF90C 074331D0 DD1069C2
      545299C2

  C8A91E60 5189079F EC8A0662 40535B2D 2CC2

        quit

crypto pki certificate chain SLA-TrustPoint

 certificate ca 01

  30820321 30820209 A0030201 02020101 300D0609 2A864886 F70D0101
      0B050030

  32310E30 0C060355 040A1305 43697363 6F312030 1E060355 04031317
      43697363

  6F204C69 63656E73 696E6720 526F6F74 20434130 1E170D31 33303533
      30313934

  3834375A 170D3338 30353330 31393438 34375A30 32310E30 0C060355
      040A1305

  43697363 6F312030 1E060355 04031317 43697363 6F204C69 63656E73
      696E6720

  526F6F74 20434130 82012230 0D06092A 864886F7 0D010101 05000382
      010F0030

  82010A02 82010100 A6BCBD96 131E05F7 145EA72C 2CD686E6 17222EA1
      F1EFF64D

  CBB4C798 212AA147 C655D8D7 9471380D 8711441E 1AAF071A 9CAE6388
      8A38E520
```

```
  1C394D78 462EF239 C659F715 B98C0A59 5BBB5CBD 0CFEBEA3 700A8BF7
      D8F256EE

  4AA4E80D DB6FD1C9 60B1FD18 FFC69C96 6FA68957 A2617DE7 104FDC5F
      EA2956AC

  7390A3EB 2B5436AD C847A2C5 DAB553EB 69A9A535 58E9F3E3 C0BD23CF
      58BD7188

  68E69491 20F320E7 948E71D7 AE3BCC84 F10684C7 4BC8E00F 539BA42B
      42C68BB7

  C7479096 B4CB2D62 EA2F505D C7B062A4 6811D95B E8250FC4 5D5D5FB8
      8F27D191

  C55F0D76 61F9A4CD 3D992327 A8BB03BD 4E6D7069 7CBADF8B DF5F4368
      95135E44

  DFC7C6CF 04DD7FD1 02030100 01A34230 40300E06 03551D0F 0101FF04
      04030201

  06300F06 03551D13 0101FF04 05300301 01FF301D 0603551D 0E041604
      1449DC85

  4B3D31E5 1B3E6A17 606AF333 3D3B4C73 E8300D06 092A8648 86F70D01
      010B0500

  03820101 00507F24 D3932A66 86025D9F E838AE5C 6D4DF6B0 49631C78
      240DA905

  604EDCDE FF4FED2B 77FC460E CD636FDB DD44681E 3A5673AB 9093D3B1
      6C9E3D8B

  D98987BF E40CBD9E 1AECA0C2 2189BB5C 8FA85686 CD98B646 5575B146
      8DFC66A8

  467A3DF4 4D565700 6ADF0F0D CF835015 3C04FF7C 21E878AC 11BA9CD2
      55A9232C

  7CA7B7E6 C1AF74F6 152E99B7 B1FCF9BB E973DE7F 5BDDEB86 C71E3B49
      1765308B

  5FB0DA06 B92AFE7F 494E8A9E 07B85737 F3A58BE1 1A48A229 C37C1E69
      39F08678

  80DDCD16 D6BACECA EEBC7CF9 8428787B 35202CDC 60E4616A B623CDBD
      230E3AFB

  418616A9 4093E049 4D10AB75 27E86F73 932E35B5 8862FDAE 0275156F
      719BB2F0

  D697DF7F 28

        quit

!

license udi pid CSR1000V sn 98FPGDJCVNE

diagnostic bootup level minimal

memory free low-watermark processor 71489
```

```
!
!
spanning-tree extend system-id
!
username headquartersadmin password 7 1427005B0601293F65
!
redundancy
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
!
interface GigabitEthernet1
```

```
 description Link to ExternalRouter
 ip address 10.2.230.222 255.255.255.252
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 description Link to SecureRouter
 ip address 10.2.230.218 255.255.255.252
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 description Link to Headquarters LAN
 no ip address
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3.112
 description Link to VLAN 112
 encapsulation dot1Q 112
 ip address 10.0.0.1 255.254.0.0
!
interface GigabitEthernet3.212
 description Link to VLAN 212
 encapsulation dot1Q 212
 ip address 10.2.0.1 255.255.128.0
!
interface GigabitEthernet3.312
 description Link to VLAN 312
 encapsulation dot1Q 312
```

```
 ip address 10.2.128.1 255.255.192.0
!
interface GigabitEthernet3.412
 description Link to VLAN 412
 encapsulation dot1Q 412
 ip address 10.2.192.1 255.255.224.0
!
interface GigabitEthernet3.512
 description Link to Link to VLAN 512
 encapsulation dot1Q 512
 ip address 10.2.224.1 255.255.252.0
!
interface GigabitEthernet3.612
 description Link to VLAN 612
 encapsulation dot1Q 612
 ip address 10.2.228.1 255.255.254.0
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
ip forward-protocol nd
ip http server
ip http authentication local
ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.2.230.221
ip route 10.2.230.0 255.255.255.128 10.2.230.217
ip route 10.2.230.128 255.255.255.192 10.2.230.217
ip route 10.2.230.192 255.255.255.240 10.2.230.217
```

```
ip route 10.2.230.208 255.255.255.248 10.2.230.217
ip ssh time-out 60
ip ssh version 2
!
!
!
!
!
!
!
control-plane
!
!
!
!
!
banner motd ^CWarning: Authorized Access Only^C
!
line con 0
 exec-timeout 1 30
 password 7 046B195605244F5A48
 login
 stopbits 1
line vty 0 4
 password 7 133505420109073E6A
 login local
 transport input ssh
!
call-home
 ! If contact email address in call-home is configured as sch-smart-
      licensing@cisco.com
 ! the email address configured in Cisco Smart License Portal will be
      used as contact email address to send SCH notifications.
 contact-email-addr sch-smart-licensing@cisco.com
```

```
 profile "CiscoTAC-1"
  active
  destination transport-method http
!
!
!
!
!
end

HeadquartersRouter#
```

Appendix H. External Router Running Configuration.

```
ExternalRouter#sh run
Building configuration...

Current configuration : 7262 bytes
!
! Last configuration change at 21:08:34 UTC Mon Mar 25 2024
!
version 17.3
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
service call-home
platform qfp utilization monitor load 80
platform punt-keepalive disable-kernel-core
platform console serial
!
hostname ExternalRouter
!
boot-start-marker
boot-end-marker
!
!
enable secret 9
      $9$n/3JuzyDK8oYDk$X8v8Jb1YEw4sQBQHT2Pjvzy.pr4h/FKZTjZ7DUvXTP.
!
no aaa new-model
!
!
!
!
!
!
```

```
!
ip name-server 192.168.26.1
ip domain name external.com
!
!
!
login on-success log
!
!
!
!
!
!
!
subscriber templating
!
!
!
!
!
!
!
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
!
!
```

```
!
!
!
!
crypto pki trustpoint TP-self-signed-2029792552
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-2029792552
 revocation-check none
 rsakeypair TP-self-signed-2029792552
!
crypto pki trustpoint SLA-TrustPoint
 enrollment pkcs12
 revocation-check crl
!
!
crypto pki certificate chain TP-self-signed-2029792552
 certificate self-signed 01
  30820330 30820218 A0030201 02020101 300D0609 2A864886 F70D0101
      05050030
  31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D
      43657274
  69666963 6174652D 32303239 37393235 3532301E 170D3234 30323232
      31383337
  30315A17 0D333430 32323131 38333730 315A3031 312F302D 06035504
      03132649
  4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32
      30323937
  39323535 32308201 22300D06 092A8648 86F70D01 01010500 0382010F
      00308201
  0A028201 0100C852 BB134C71 594A8558 7A1434C7 D4F4EC83 92F7C376
      9D96DF04
  E142C673 F590CD24 9886A1B0 62771044 6AFEB6DB F1238931 3753D2C8
      2DB9D31D
  1E79544D 2549015E 99BD13D1 619CAB1B 1C2365D2 CD3995BA 61EB167C
      52713BF0
  66B9C99D CC30E0C6 60A9C8C8 0F1C8765 25AAD132 DA8CF366 E310FBE8
      75BAFF35
```

```
  8C9B64C4 0DBDA4C1 F6D25360 3BD9B6FE 582933CF A47C6AEB 9F96D9EE
      BB110A72

  57DDF25B 5D3910E8 6B09F874 55B014CD 89EC493E 487D3A60 1BA26442
      EF7DB77B

  3B4334E1 324219DB 4A3489F9 E7B1C9BE D25B86D1 B1147595 EE0D6011
      2AB2BD18

  497BF073 E49C53A3 F4A25E89 CD51992E 2DAF8C86 243FF8F3 97F57964
      2BEC6DDB

  3AF7AAEA F9F70203 010001A3 53305130 0F060355 1D130101 FF040530
      030101FF

  301F0603 551D2304 18301680 141903B0 4EDC5084 838E5A00 64018409
      ECB5DEFE

  56301D06 03551D0E 04160414 1903B04E DC508483 8E5A0064 018409EC
      B5DEFE56

  300D0609 2A864886 F70D0101 05050003 82010100 B1736FDA 5FA26A16
      031E8826

  BF05722F 36756E9A DB754F9A A84079F6 9741C4F3 7B28E5C8 4C10C93B
      05F69A8C

  45E5115A 80A0FB83 F9B2F4E4 B36A3AB9 D436891F C6193D95 757CB218
      7AD92548

  85142FF8 FD0549AB 1FA2BD9B 41A3D311 F4377F01 50587D60 8B93360B
      56429E6A

  090F0EF7 7ADF1B18 FCD16F89 BCF498C5 0A3A6FE4 F7E26D4B BFB883F1
      CF47E421

  600949B6 CE95031E DF1B49D3 B7C78CB5 B565917D 8C459B27 BC297B7C
      3C632397

  6D5149A0 0F8CD193 FDDE23C0 0AF5A3BE 8F04E90E 09A7A365 9942CBD6
      3C9675EC

  71A9A269 463A7FE3 3B8CA154 96832CC7 DAC07A94 004C803B 2EFA170E
      330FAA01

  DA60573C 377496D5 090B1AE0 29F69F69 C14005C6

        quit

crypto pki certificate chain SLA-TrustPoint

 certificate ca 01

  30820321 30820209 A0030201 02020101 300D0609 2A864886 F70D0101
      0B050030

  32310E30 0C060355 040A1305 43697363 6F312030 1E060355 04031317
      43697363

  6F204C69 63656E73 696E6720 526F6F74 20434130 1E170D31 33303533
      30313934
```

```
3834375A 170D3338 30353330 31393438 34375A30 32310E30 0C060355
    040A1305

43697363 6F312030 1E060355 04031317 43697363 6F204C69 63656E73
    696E6720

526F6F74 20434130 82012230 0D06092A 864886F7 0D010101 05000382
    010F0030

82010A02 82010100 A6BCBD96 131E05F7 145EA72C 2CD686E6 17222EA1
    F1EFF64D

CBB4C798 212AA147 C655D8D7 9471380D 8711441E 1AAF071A 9CAE6388
    8A38E520

1C394D78 462EF239 C659F715 B98C0A59 5BBB5CBD 0CFEBEA3 700A8BF7
    D8F256EE

4AA4E80D DB6FD1C9 60B1FD18 FFC69C96 6FA68957 A2617DE7 104FDC5F
    EA2956AC

7390A3EB 2B5436AD C847A2C5 DAB553EB 69A9A535 58E9F3E3 C0BD23CF
    58BD7188

68E69491 20F320E7 948E71D7 AE3BCC84 F10684C7 4BC8E00F 539BA42B
    42C68BB7

C7479096 B4CB2D62 EA2F505D C7B062A4 6811D95B E8250FC4 5D5D5FB8
    8F27D191

C55F0D76 61F9A4CD 3D992327 A8BB03BD 4E6D7069 7CBADF8B DF5F4368
    95135E44

DFC7C6CF 04DD7FD1 02030100 01A34230 40300E06 03551D0F 0101FF04
    04030201

06300F06 03551D13 0101FF04 05300301 01FF301D 0603551D 0E041604
    1449DC85

4B3D31E5 1B3E6A17 606AF333 3D3B4C73 E8300D06 092A8648 86F70D01
    010B0500

03820101 00507F24 D3932A66 86025D9F E838AE5C 6D4DF6B0 49631C78
    240DA905

604EDCDE FF4FED2B 77FC460E CD636FDB DD44681E 3A5673AB 9093D3B1
    6C9E3D8B

D98987BF E40CBD9E 1AECA0C2 2189BB5C 8FA85686 CD98B646 5575B146
    8DFC66A8

467A3DF4 4D565700 6ADF0F0D CF835015 3C04FF7C 21E878AC 11BA9CD2
    55A9232C

7CA7B7E6 C1AF74F6 152E99B7 B1FCF9BB E973DE7F 5BDDEB86 C71E3B49
    1765308B

5FB0DA06 B92AFE7F 494E8A9E 07B85737 F3A58BE1 1A48A229 C37C1E69
    39F08678

80DDCD16 D6BACECA EEBC7CF9 8428787B 35202CDC 60E4616A B623CDBD
    230E3AFB
```

```
  418616A9 4093E049 4D10AB75 27E86F73 932E35B5 8862FDAE 0275156F
      719BB2F0

  D697DF7F 28

        quit

!

license udi pid CSR1000V sn 9PCUYXDSZ3J

diagnostic bootup level minimal

memory free low-watermark processor 71489

!

!

spanning-tree extend system-id

!

username externaladmin password 7 06361D71464B0A0D44

!

redundancy

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!

!
```

```
!
!
!
!
!
!
interface GigabitEthernet1
 description Link to HeadquartersRouter
 ip address 10.2.230.221 255.255.255.252
 ip nat inside
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 description Link to Internet/External Network
 ip dhcp client client-id ascii 9PCUYXDSZ3J
 ip address dhcp
 ip nat outside
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 negotiation auto
 no mop enabled
```

```
 no mop sysid
!
ip forward-protocol nd
ip http server
ip http authentication local
ip http secure-server
ip http client source-interface GigabitEthernet2
!
ip nat inside source static 10.2.230.221 192.168.26.240
ip nat inside source list 8 interface GigabitEthernet2 overload
ip route 10.0.0.0 255.254.0.0 10.2.230.222
ip route 10.2.0.0 255.255.128.0 10.2.230.222
ip route 10.2.128.0 255.255.192.0 10.2.230.222
ip route 10.2.192.0 255.255.224.0 10.2.230.222
ip route 10.2.224.0 255.255.252.0 10.2.230.222
ip route 10.2.230.0 255.255.255.0 10.2.230.222
ip route 10.2.230.0 255.255.255.128 10.2.230.222
ip route 10.2.230.128 255.255.255.192 10.2.230.222
ip route 10.2.230.192 255.255.255.240 10.2.230.222
ip route 10.2.230.208 255.255.255.248 10.2.230.222
ip ssh time-out 60
ip ssh authentication-retries 2
ip ssh version 2
!
!
ip access-list standard 8
 10 permit 10.0.0.0 0.255.255.255
 20 permit 10.2.230.0 0.0.0.255
!
!
!
!
!
```

```
control-plane
!
!
!
!
banner motd ^CWarning: Authorized Access Only^C
!
line con 0
 exec-timeout 1 30
 password 7 003401560E5E08124E
 login
 stopbits 1
line vty 0 4
 password 7 003401560E5E08124E
 login local
 transport input ssh
!
call-home
 ! If contact email address in call-home is configured as sch-smart-
      licensing@cisco.com
 ! the email address configured in Cisco Smart License Portal will be
      used as contact email address to send SCH notifications.
 contact-email-addr sch-smart-licensing@cisco.com
 profile "CiscoTAC-1"
  active
  destination transport-method http
!
!
!
!
end
```

Appendix I. Wazuh Server Configuration File.

```
<!--
  Wazuh Server Configuration, Claire Headland
  Wazuh - Manager - Default configuration for ubuntu 20.04
  More info at: https://documentation.wazuh.com
  Mailing list: https://groups.google.com/forum/#!forum/wazuh
-->


<ossec_config>
  <global>
    <jsonout_output>yes</jsonout_output>
    <alerts_log>yes</alerts_log>
    <logall>yes</logall>
    <logall_json>no</logall_json>
    <email_notification>yes</email_notification>
    <smtp_server>10.2.230.212</smtp_server>
    <email_from>cyberalerts.wazuh@gmail.com</email_from>
    <email_to>cyberalerts.sysad@gmail.com</email_to>
    <email_maxperhour>50</email_maxperhour>
    <email_log_source>alerts.log</email_log_source>
    <agents_disconnection_time>10m</agents_disconnection_time>

<agents_disconnection_alert_time>0</agents_disconnection_alert_time>
  </global>


  <alerts>
    <log_alert_level>2</log_alert_level>
    <email_alert_level>3</email_alert_level>
  </alerts>


  <!-- Choose between "plain", "json", or "plain,json" for the format
of internal logs -->
  <logging>
    <log_format>plain,json</log_format>
```

```
  </logging>


  <remote>
    <connection>secure</connection>
    <port>1514</port>
    <protocol>tcp</protocol>
    <queue_size>150000</queue_size>
  </remote>


  <integration>
    <name>virustotal</name>

<api_key>dcafd3137e3b0991e2cbf765a0af902e09965c932349f43556b5ff0fedd02b
fb</api_key>
    <rule_id>100200,100201</rule_id>
    <group>syscheck</group>
    <alert_format>json</alert_format>
  </integration>


  <integration>
    <name>custom-alerts</name>
    <hook_url>https://uazips.webhook.office.com/webhookb2/43bbef63-
1a68-4591-aebb-a5a3716cec2f@e8575ded-d7f9-4ece-a4aa-
0b32991aeedd/IncomingWebhook/0acf2ac583d940ebacc880784d56ff6a/2e5f1abb-
9abf-40b5-a1df-8905e0226954/</hook_url>
    <level>7</level>
    <alert_format>json</alert_format>
  </integration>


  <!-- Policy monitoring -->
  <rootcheck>
    <disabled>no</disabled>
    <check_files>yes</check_files>
    <check_trojans>yes</check_trojans>
    <check_dev>yes</check_dev>
```

```
    <check_sys>yes</check_sys>
    <check_pids>yes</check_pids>
    <check_ports>yes</check_ports>
    <check_if>yes</check_if>

    <!-- Frequency that rootcheck is executed - every 12 hours -->
    <frequency>43200</frequency>

    <rootkit_files>etc/rootcheck/rootkit_files.txt</rootkit_files>

<rootkit_trojans>etc/rootcheck/rootkit_trojans.txt</rootkit_trojans>

    <skip_nfs>yes</skip_nfs>
  </rootcheck>

  <wodle name="cis-cat">
    <disabled>yes</disabled>
    <timeout>1800</timeout>
    <interval>1d</interval>
    <scan-on-start>yes</scan-on-start>

    <java_path>wodles/java</java_path>
    <ciscat_path>wodles/ciscat</ciscat_path>
  </wodle>

  <!-- Osquery integration -->
  <wodle name="osquery">
    <disabled>yes</disabled>
    <run_daemon>yes</run_daemon>
    <log_path>/var/log/osquery/osqueryd.results.log</log_path>
    <config_path>/etc/osquery/osquery.conf</config_path>
    <add_labels>yes</add_labels>
  </wodle>
```

```
<!-- System inventory -->
<wodle name="syscollector">
  <disabled>no</disabled>
  <interval>1h</interval>
  <scan_on_start>yes</scan_on_start>
  <hardware>yes</hardware>
  <os>yes</os>
  <network>yes</network>
  <packages>yes</packages>
  <ports all="no">yes</ports>
  <processes>yes</processes>

  <!-- Database synchronization settings -->
  <synchronization>
    <max_eps>10</max_eps>
  </synchronization>
</wodle>

<sca>
  <enabled>yes</enabled>
  <scan_on_start>yes</scan_on_start>
  <interval>12h</interval>
  <skip_nfs>yes</skip_nfs>
</sca>

<vulnerability-detector>
  <enabled>yes</enabled>
  <interval>5m</interval>
  <min_full_scan_interval>6h</min_full_scan_interval>
  <run_on_start>yes</run_on_start>

  <!-- Ubuntu OS vulnerabilities -->
  <provider name="canonical">
```

```
    <enabled>no</enabled>
    <os>trusty</os>
    <os>xenial</os>
    <os>bionic</os>
    <os>focal</os>
    <os>jammy</os>
    <update_interval>1h</update_interval>
  </provider>

  <!-- Debian OS vulnerabilities -->
  <provider name="debian">
    <enabled>no</enabled>
    <os>buster</os>
    <os>bullseye</os>
    <os>bookworm</os>
    <update_interval>1h</update_interval>
  </provider>

  <!-- RedHat OS vulnerabilities -->
  <provider name="redhat">
    <enabled>no</enabled>
    <os>5</os>
    <os>6</os>
    <os>7</os>
    <os>8</os>
    <os>9</os>
    <update_interval>1h</update_interval>
  </provider>

  <!-- Amazon Linux OS vulnerabilities -->
  <provider name="alas">
    <enabled>no</enabled>
    <os>amazon-linux</os>
```

```
  <os>amazon-linux-2</os>
  <os>amazon-linux-2022</os>
  <os>amazon-linux-2023</os>
  <update_interval>1h</update_interval>
</provider>

<!-- SUSE OS vulnerabilities -->
<provider name="suse">
  <enabled>no</enabled>
  <os>11-server</os>
  <os>11-desktop</os>
  <os>12-server</os>
  <os>12-desktop</os>
  <os>15-server</os>
  <os>15-desktop</os>
  <update_interval>1h</update_interval>
</provider>

<!-- Arch OS vulnerabilities -->
<provider name="arch">
  <enabled>no</enabled>
  <update_interval>1h</update_interval>
</provider>

<!-- Alma Linux OS vulnerabilities -->
<provider name="almalinux">
  <enabled>no</enabled>
  <os>8</os>
  <os>9</os>
  <update_interval>1h</update_interval>
</provider>

<!-- Windows OS vulnerabilities -->
```

```
    <provider name="msu">
      <enabled>yes</enabled>
      <update_interval>1h</update_interval>
    </provider>

    <!-- Aggregate vulnerabilities -->
    <provider name="nvd">
      <enabled>yes</enabled>
      <update_interval>1h</update_interval>
    </provider>

  </vulnerability-detector>

  <!-- File integrity monitoring -->
  <syscheck>

    <disabled>no</disabled>

     <directories check_all="yes" realtime="yes">/root</directories>

     <directories check_all="yes"
realtime="yes">/var/ossec/logs/alerts/alerts.log</directories>

    <!-- Frequency that syscheck is executed default every 12 hours -->
    <frequency>43200</frequency>

    <scan_on_start>yes</scan_on_start>

    <!-- Generate alert when new file detected -->
    <alert_new_files>yes</alert_new_files>

    <!-- Don't ignore files that change more than 'frequency' times -->
    <auto_ignore frequency="10" timeframe="3600">no</auto_ignore>
```

```
<!-- Directories to check  (perform all possible verifications) -->
<directories>/etc,/usr/bin,/usr/sbin</directories>
<directories>/bin,/sbin,/boot</directories>

<!-- Files/directories to ignore -->
<ignore>/etc/mtab</ignore>
<ignore>/etc/hosts.deny</ignore>
<ignore>/etc/mail/statistics</ignore>
<ignore>/etc/random-seed</ignore>
<ignore>/etc/random.seed</ignore>
<ignore>/etc/adjtime</ignore>
<ignore>/etc/httpd/logs</ignore>
<ignore>/etc/utmpx</ignore>
<ignore>/etc/wtmpx</ignore>
<ignore>/etc/cups/certs</ignore>
<ignore>/etc/dumpdates</ignore>
<ignore>/etc/svc/volatile</ignore>

<!-- File types to ignore -->
<ignore type="sregex">.log$|.swp$</ignore>

<!-- Check the file, but never compute the diff -->
<nodiff>/etc/ssl/private.key</nodiff>

<skip_nfs>yes</skip_nfs>
<skip_dev>yes</skip_dev>
<skip_proc>yes</skip_proc>
<skip_sys>yes</skip_sys>

<!-- Nice value for Syscheck process -->
<process_priority>10</process_priority>

<!-- Maximum output throughput -->
```

```
    <max_eps>50</max_eps>

    <!-- Database synchronization settings -->
    <synchronization>
      <enabled>yes</enabled>
      <interval>5m</interval>
      <max_eps>10</max_eps>
    </synchronization>
  </syscheck>

  <!-- Active response -->
  <global>
    <white_list>127.0.0.1</white_list>
    <white_list>^localhost.localdomain$</white_list>
    <white_list>127.0.0.53</white_list>
  </global>

  <command>
    <name>disable-account</name>
    <executable>disable-account</executable>
    <timeout_allowed>yes</timeout_allowed>
  </command>

  <command>
    <name>remove-threat</name>
    <executable>remove-threat.sh</executable>
    <timeout_allowed>no</timeout_allowed>
  </command>

  <command>
    <name>restart-wazuh</name>
    <executable>restart-wazuh</executable>
  </command>
```

```
<command>
  <name>firewall-drop</name>
  <executable>firewall-drop</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>

<command>
  <name>host-deny</name>
  <executable>host-deny</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>

<command>
  <name>route-null</name>
  <executable>route-null</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>

<command>
  <name>win_route-null</name>
  <executable>route-null.exe</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>

<command>
  <name>netsh</name>
  <executable>netsh.exe</executable>
  <timeout_allowed>yes</timeout_allowed>
</command>

<active-response>
  <disabled>no</disabled>
```

```
  <command>remove-threat</command>
  <location>local</location>
  <rules_id>87105</rules_id>
</active-response>

<active-response>
  <disabled>no</disabled>
  <command>firewall-drop</command>
  <location>local</location>
  <rules_id>100100</rules_id>
  <timeout>60</timeout>
</active-response>

<active-response>
  <disabled>no</disabled>
  <command>host-deny</command>
  <location>local</location>
  <rules_id>100001</rules_id>
  <timeout>60</timeout>
</active-response>

<active-response>
  <disabled>no</disabled>
  <command>disable-account</command>
  <location>local</location>
  <rules_id>100200,100201,80792</rules_id>
  <timeout>60</timeout>
</active-response>

<!-- Log analysis -->
<localfile>
  <log_format>command</log_format>
  <command>df -P</command>
```

```
    <frequency>360</frequency>
  </localfile>


  <localfile>
    <log_format>full_command</log_format>
    <command>netstat -tulpn | sed 's/\([[:alnum:]]\+\)\
\+[[:digit:]]\+\ \+[[:digit:]]\+\ \+\(.*\):\([[:digit:]]*\)\ \+\([0-
9\.\:\*]\+\).\+\ \([[:digit:]]*\/[[:alnum:]\-]*\).*/\1 \2 == \3 == \4
\5/' | sort -k 4 -g | sed 's/ == \(.*\) ==/:\1/' | sed 1,2d</command>
    <alias>netstat listening ports</alias>
    <frequency>360</frequency>
  </localfile>


  <localfile>
    <log_format>full_command</log_format>
    <command>last -n 20</command>
    <frequency>360</frequency>
  </localfile>


  <ruleset>
    <!-- Default ruleset -->
    <decoder_dir>ruleset/decoders</decoder_dir>
    <rule_dir>ruleset/rules</rule_dir>
    <rule_exclude>0215-policy_rules.xml</rule_exclude>
    <list>etc/lists/audit-keys</list>
    <list>etc/lists/amazon/aws-eventnames</list>
    <list>etc/lists/suspicious-programs</list>
    <list>etc/lists/security-eventchannel</list>


    <!-- User-defined ruleset -->
    <decoder_dir>etc/decoders</decoder_dir>
    <rule_dir>etc/rules</rule_dir>


    <decoder_dir>ruleset/decoders</decoder_dir>
    <rule_dir>ruleset/rules</rule_dir>
```

```
    <rule_exclude>0215-policy_rules.xml</rule_exclude>
    <list>etc/lists/audit-keys</list>
    <list>etc/lists/amazon/aws-eventnames</list>
    <list>etc/lists/security-eventchannel</list>
    <list>etc/lists/blacklist-alienvault</list>

    <!-- User-defined ruleset -->
    <decoder_dir>etc/decoders</decoder_dir>
    <rule_dir>etc/rules</rule_dir>
  </ruleset>

  <rule_test>
    <enabled>yes</enabled>
    <threads>1</threads>
    <max_sessions>64</max_sessions>
    <session_timeout>15m</session_timeout>
  </rule_test>

  <!-- Configuration for wazuh-authd -->
  <auth>
    <disabled>no</disabled>
    <port>1515</port>
    <use_source_ip>no</use_source_ip>
    <purge>yes</purge>
    <use_password>no</use_password>

<ciphers>HIGH:!ADH:!EXP:!MD5:!RC4:!3DES:!CAMELLIA:@STRENGTH</ciphers>
    <!-- <ssl_agent_ca></ssl_agent_ca> -->
    <ssl_verify_host>no</ssl_verify_host>
    <ssl_manager_cert>etc/sslmanager.cert</ssl_manager_cert>
    <ssl_manager_key>etc/sslmanager.key</ssl_manager_key>
    <ssl_auto_negotiate>no</ssl_auto_negotiate>
  </auth>
```

```
<cluster>
  <name>wazuh</name>
  <node_name>node01</node_name>
  <node_type>master</node_type>
  <key></key>
  <port>1516</port>
  <bind_addr>0.0.0.0</bind_addr>
  <nodes>
      <node>NODE_IP</node>
  </nodes>
  <hidden>no</hidden>
  <disabled>yes</disabled>
</cluster>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/ossec/logs/active-responses.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/auth.log</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/syslog</location>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/dpkg.log</location>
</localfile>
```

```
  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/kern.log</location>
  </localfile>

</ossec_config>
```

Appendix J. Wazuh Agent Configuration File.

```
<!--
  Sr Analyst Configuration, Claire Headland
  Wazuh - Agent - Default configuration for ubuntu 20.04
  More info at: https://documentation.wazuh.com
  Mailing list: https://groups.google.com/forum/#!forum/wazuh
-->

<ossec_config>
  <client>
    <server>
      <address>10.2.230.212</address>
      <port>1514</port>
      <protocol>tcp</protocol>
      <max_retries>5</max_retries>
      <retry_interval>5</retry_interval>
    </server>
    <config-profile>ubuntu, ubuntu20, ubuntu20.04</config-profile>
    <notify_time>10</notify_time>
    <time-reconnect>60</time-reconnect>
    <auto_restart>yes</auto_restart>
    <crypto_method>aes</crypto_method>
  </client>

  <client_buffer>
    <!-- Agent buffer options -->
    <disabled>no</disabled>
    <queue_size>5000</queue_size>
    <events_per_second>500</events_per_second>
  </client_buffer>

  <!-- Policy monitoring -->
  <rootcheck>
```

```
    <disabled>no</disabled>
    <check_files>yes</check_files>
    <check_trojans>yes</check_trojans>
    <check_dev>yes</check_dev>
    <check_sys>yes</check_sys>
    <check_pids>yes</check_pids>
    <check_ports>yes</check_ports>
    <check_if>yes</check_if>

    <!-- Frequency that rootcheck is executed - every 12 hours -->
    <frequency>120</frequency>

    <rootkit_files>etc/shared/rootkit_files.txt</rootkit_files>
    <rootkit_trojans>etc/shared/rootkit_trojans.txt</rootkit_trojans>

    <skip_nfs>yes</skip_nfs>
  </rootcheck>

  <wodle name="cis-cat">
    <disabled>yes</disabled>
    <timeout>1800</timeout>
    <interval>1d</interval>
    <scan-on-start>yes</scan-on-start>

    <java_path>wodles/java</java_path>
    <ciscat_path>wodles/ciscat</ciscat_path>
  </wodle>

  <!-- Osquery integration -->
  <wodle name="osquery">
    <disabled>yes</disabled>
    <run_daemon>yes</run_daemon>
    <log_path>/var/log/osquery/osqueryd.results.log</log_path>
```

```
    <config_path>/etc/osquery/osquery.conf</config_path>
    <add_labels>yes</add_labels>
  </wodle>

  <!-- System inventory -->
  <wodle name="syscollector">
    <disabled>no</disabled>
    <interval>1h</interval>
    <scan_on_start>yes</scan_on_start>
    <hardware>yes</hardware>
    <os>yes</os>
    <network>yes</network>
    <packages>yes</packages>
    <ports all="no">yes</ports>
    <processes>yes</processes>

    <!-- Database synchronization settings -->
    <synchronization>
      <max_eps>10</max_eps>
    </synchronization>
  </wodle>

  <sca>
    <enabled>yes</enabled>
    <scan_on_start>yes</scan_on_start>
    <interval>12h</interval>
    <skip_nfs>yes</skip_nfs>
  </sca>

  <!-- File integrity monitoring -->
  <syscheck>
    <disabled>no</disabled>
```

```
    <directories realtime="yes"
whodata="yes">/specialdir3</directories>


    <directories  whodata="yes" report_changes="yes"
realtime="yes">~/Downloads</directories>


    <directories check_all="yes" realtime="yes" report_changes="yes"
whodata="yes">/root</directories>


    <directories check_all="yes" whodata="yes"
report_changes="yes">/etc/passwd</directories>


    <!-- Frequency that syscheck is executed default every 12 hours -->

    <frequency>43200</frequency>


    <scan_on_start>yes</scan_on_start>


    <!-- Directories to check  (perform all possible verifications) -->

    <directories>/etc,/usr/bin,/usr/sbin</directories>

    <directories>/bin,/sbin,/boot</directories>


    <!-- Files/directories to ignore -->

    <ignore>/etc/mtab</ignore>

    <ignore>/etc/hosts.deny</ignore>

    <ignore>/etc/mail/statistics</ignore>

    <ignore>/etc/random-seed</ignore>

    <ignore>/etc/random.seed</ignore>

    <ignore>/etc/adjtime</ignore>

    <ignore>/etc/httpd/logs</ignore>

    <ignore>/etc/utmpx</ignore>

    <ignore>/etc/wtmpx</ignore>

    <ignore>/etc/cups/certs</ignore>

    <ignore>/etc/dumpdates</ignore>

    <ignore>/etc/svc/volatile</ignore>
```

```
    <!-- File types to ignore -->
    <ignore type="sregex">.log$|.swp$</ignore>

    <!-- Check the file, but never compute the diff -->
    <nodiff>/etc/ssl/private.key</nodiff>

    <skip_nfs>yes</skip_nfs>
    <skip_dev>yes</skip_dev>
    <skip_proc>yes</skip_proc>
    <skip_sys>yes</skip_sys>

    <!-- Nice value for Syscheck process -->
    <process_priority>10</process_priority>

    <!-- Maximum output throughput -->
    <max_eps>50</max_eps>

    <!-- Database synchronization settings -->
    <synchronization>
      <enabled>yes</enabled>
      <interval>5m</interval>
      <max_eps>10</max_eps>
    </synchronization>
  </syscheck>

  <!-- Log analysis -->
  <localfile>
    <log_format>command</log_format>
    <command>df -P</command>
    <frequency>360</frequency>
  </localfile>

  <localfile>
```

```
    <location>/var/log/test.log</location>
    <log_format>syslog</log_format>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/apache2/access.log</location>
  </localfile>

  <localfile>
    <log_format>full_command</log_format>
    <command>netstat -tulpn | sed 's/\([[:alnum:]]\+\)\
\+[[:digit:]]\+\ \+[[:digit:]]\+\ \+\(.*\):\([[:digit:]]*\)\ \+\([0-
9\.\:\*]\+\).\+\ \([[:digit:]]*\/[[:alnum:]\-]*\).*/\1 \2 == \3 == \4
\5/' | sort -k 4 -g | sed 's/ == \(.*\) ==/:\1/' | sed 1,2d</command>
    <alias>netstat listening ports</alias>
    <frequency>360</frequency>
  </localfile>

  <localfile>
    <log_format>full_command</log_format>
    <command>last -n 20</command>
    <frequency>360</frequency>
  </localfile>

  <localfile>
    <log_format>full_command</log_format>
    <alias>process list</alias>
    <command>ps -e -o pid,uname,command</command>
    <frequency>30</frequency>
  </localfile>

  <localfile>
    <log_format>audit</log_format>
    <location>/var/log/audit/audit.log</location>
```

```
  </localfile>


  <!-- Active response -->
  <active-response>
    <disabled>no</disabled>
    <ca_store>etc/wpk_root.pem</ca_store>
    <ca_verification>yes</ca_verification>
    <command>firewall-drop</command>
    <location>local</location>
    <rules_id>100100</rules_id>
    <timeout>60</timeout>
  </active-response>


  <!-- Choose between "plain", "json", or "plain,json" for the format
of internal logs -->
  <logging>
    <log_format>plain</log_format>
  </logging>


  <localfile>
    <log_format>json</log_format>
    <location>/var/log/suricata/eve.json</location>
  </localfile>


  <localfile>
    <log_format>syslog</log_format>
    <location>/var/ossec/logs/active-responses.log</location>
  </localfile>


  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/auth.log</location>
  </localfile>
```

```
  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/syslog</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/dpkg.log</location>
  </localfile>

  <localfile>
    <log_format>syslog</log_format>
    <location>/var/log/kern.log</location>
  </localfile>

</ossec_config>
```

Appendix K. Wazuh `local_rules.xml` Local Rules File.

```
<!-- Wazuh Server Local Rules, Claire Headland -->


<!-- Modify it at your will. -->
<!-- Copyright (C) 2015, Wazuh Inc. -->


  <rule id="100001" level="5">
    <if_sid>5716</if_sid>
    <srcip>10.2.230.214</srcip>
    <description>sshd: authentication failed from IP
10.2.230.214.</description>
    <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</group>
  </rule>


</group>


<group name="attack,">
  <rule id="100100" level="10">
    <if_group>web|attack|attacks</if_group>
    <list field="srcip" lookup="address_match_key">etc/lists/blacklist-
alienvault</list>
    <description>IP address found in AlienVault reputation
database.</description>
    <options>alert_by_email</options>
  </rule>
</group>


<group name="ossec,">
  <rule id="100050" level="0">
    <if_sid>530</if_sid>
    <match>^ossec: output: 'process list'</match>
    <description>List of running processes.</description>
    <group>process_monitor,</group>
  </rule>
```

```
  <rule id="100051" level="7" ignore="900">
    <if_sid>100050</if_sid>
    <match>nc -l</match>
    <description>netcat listening for incoming
connections.</description>
    <group>process_monitor,</group>
  </rule>
</group>

<group name="syscheck,pci_dss_11.5,nist_800_53_SI.7,">
    <!-- Rules for Linux systems -->
    <rule id="100200" level="7">
        <if_sid>550</if_sid>
        <field name="file">/root</field>
        <description>File modified in /root directory.</description>
    </rule>
    <rule id="100201" level="7">
        <if_sid>554</if_sid>
        <field name="file">/root</field>
        <description>File added to /root directory.</description>
    </rule>
</group>

<group name="virustotal,">
  <rule id="100092" level="12">
    <if_sid>657</if_sid>
    <match>Successfully removed threat</match>
    <description>$(parameters.program) removed threat located at
$(parameters.alert.data.virustotal.source.file)</description>
  </rule>

  <rule id="100093" level="12">
    <if_sid>657</if_sid>
```

```
    <match>Error removing threat</match>

    <description>Error removing threat located at
$(parameters.alert.data.virustotal.source.file)</description>

  </rule>

</group>


<group name="audit">

  <rule id="100210" level="12">

      <if_sid>80792</if_sid>

  <list field="audit.command" lookup="match_key_value"
check_value="orange">etc/lists/suspicious-programs</list>

    <description>Audit: Highly Suspicious Command executed:
$(audit.exe)</description>

      <group>audit_command,</group>

  </rule>

</group>


<group name="suricata">

  <rule id="100888" level="12">

      <if_sid>86601</if_sid>

      <match>PING</match>

    <description>Suricata: ICMP pings detected.</description>

  </rule>

</group>
```

Appendix L. Custom Integration `custom-alerts` Script File.

```
#!/bin/sh

# Created by Claire Headland

# Based on Wazuh's Slack integration using webhooks

# Mitigating Cyber Espionage, April 1, 2024

#

# Copyright (C) 2015, Wazuh Inc.

# Created by Wazuh, Inc. <info@wazuh.com>.

# This program is free software; you can redistribute it and/or modify
it under the terms of GPLv2


WPYTHON_BIN="framework/python/bin/python3"


SCRIPT_PATH_NAME="$0"


DIR_NAME="$(cd $(dirname ${SCRIPT_PATH_NAME}); pwd -P)"

SCRIPT_NAME="$(basename ${SCRIPT_PATH_NAME})"


case ${DIR_NAME} in

    */active-response/bin | */wodles*)

        if [ -z "${WAZUH_PATH}" ]; then

            WAZUH_PATH="$(cd ${DIR_NAME}/../..; pwd)"

        fi


        PYTHON_SCRIPT="${DIR_NAME}/${SCRIPT_NAME}.py"

    ;;

    */bin)

        if [ -z "${WAZUH_PATH}" ]; then

            WAZUH_PATH="$(cd ${DIR_NAME}/..; pwd)"

        fi


        PYTHON_SCRIPT="${WAZUH_PATH}/framework/scripts/$(echo
${SCRIPT_NAME} | sed 's/\-/_/g').py"

    ;;
```

```
     */integrations)
        if [ -z "${WAZUH_PATH}" ]; then
            WAZUH_PATH="$(cd ${DIR_NAME}/..; pwd)"
        fi

        PYTHON_SCRIPT="${DIR_NAME}/${SCRIPT_NAME}.py"
    ;;
esac
${WAZUH_PATH}/${WPYTHON_BIN} ${PYTHON_SCRIPT} "$@"
```

Appendix M. Custom Integration `custom-alerts.py` Python File.

```
#!/usr/bin/env js
# Created by Claire Headland
# Based on Wazuh's Slack integration using webhooks
# Mitigating Cyber Espionage, April 1, 2024
#
# Copyright (C) 2015, Wazuh Inc.
#
# This program is free software; you can redistribute it
# and/or modify it under the terms of the GNU General Public
# License (version 2) as published by the FSF - Free Software
# Foundation.

import json
import sys
import time
import os

try:
    import requests
    from requests.auth import HTTPBasicAuth

except Exception as e:
    print("No module 'requests' found. Install: pip install requests")
    sys.exit(1)

# Global variables
debug_enabled   = False
pwd                                                                    =
os.path.dirname(os.path.dirname(os.path.realpath(__file__)))
json_alert      = {}
now = time.strftime("%a %b %d %H:%M:%S %Z %Y")
```

```
log_file = '{0}/logs/integrations.log'.format(pwd)


def main(args):
    debug("[Teams-Alerts] Starting")
    alert_file_location = args[1]
    webhook = args[3]
    debug("[Teams-Alerts] Webhook")
    debug(webhook)
    debug("[Teams-Alerts] File location")
    debug(alert_file_location)

    with open(alert_file_location) as alert_file:
        json_alert = json.load(alert_file)
    debug("[Teams-Alerts] Processing alert")
    debug(json_alert)
    debug("[Teams-Alerts] Generating message")
    msg = generate_msg(json_alert)
    if isinstance(msg, str):
        if len(msg) == 0:
            return
        debug(msg)
    debug("[Teams-Alerts] Sending message")
    send_msg(msg, webhook)


def debug(msg):
    if debug_enabled:
      msg = '{0}: {1}\n'.format(now, msg)
      print(msg)
      f = open(log_file, 'a')
      f.write(msg)
      f.close()


def generate_msg(alert):
```

```
    level = alert['rule']['level']
    if (level <= 6):
            color = "00ff00"
    elif (level >= 7 and level <= 9):
            color = "ffff00"
    else:
            color = "ff0000"
    msg = {}
    sections = []
    msg['@type'] = "MessageCard"
    msg['themeColor'] = color
    msg['summary'] = "Wazuh Alert:" + \
       alert['rule']['description'] if 'description' in alert['rule']
else "N/A"
    actions = []
    facts = []

    if 'agent' in alert:
      facts.append({
       'name': 'Agent',
           'value': "({0}) - {1}".format(
            alert['agent']['id'],
            alert['agent']['name']
          )})

    if 'agentless' in alert:
        facts.append({
            'name': 'Agentless host',
            'value': alert['agentless']['host']
        })

    facts.extend([
      {
        'name': 'Location',
```

```
        'value': alert['location']},
      {
        'name': 'Rule ID',
        'value':    "{0}    _(Level    {1})_".format(alert['rule']['id'],
level)},
      {
        'name': 'Log',
        'value': alert.get('full_log')}
    ])
    sections.extend([
      {
        'activityTitle': "Wazuh Alert"},
      {
        'activitySubtitle': "Possible network environment security event,
proceed with caution. If alert level is 10 or higher, suspend all
confidential communications until further notice."},
      {
        'activityImage':
"https://upload.wikimedia.org/wikipedia/commons/3/34/Icon_hacker.png"}
    ])
    if 'description' in alert['rule']:
         sections.append({
            'title': alert['rule']['description'],
        })
    sections.append({
        'facts': facts,
        'markdown': 'true'
    })
    actions.append({
        '@type': "OpenUri",
        'name': "Learn More",
        'targets': [{
            'os': "default",
            'uri': "https://www.cio.gov/"
      }]
```

```
    })
    msg['sections'] = sections
    msg['potentialAction'] = actions
    return json.dumps(msg)


def send_msg(msg, url):
    headers = {'content-type': 'application/json', 'Accept-Charset':
'UTF-8'}
    res = requests.post(url, data=msg, headers=headers)
    debug(res)


if __name__ == "__main__":
    try:
        # Read arguments
        bad_arguments = False
        if len(sys.argv) >= 4:
            msg = '{0} {1} {2} {3} {4}'.format(
                now,
                sys.argv[1],
                sys.argv[2],
                sys.argv[3],
                sys.argv[4] if len(sys.argv) > 4 else '',
            )

            debug_enabled = (len(sys.argv) > 4 and sys.argv[4] ==
'debug')
        else:
            msg = '{0} Wrong arguments'.format(now)
            bad_arguments = True

        # Logging the call
        f = open(log_file, 'a')
        f.write(msg + '\n')
        f.close()
```

```
        if bad_arguments:
            debug("[Teams-Alerts] Exiting: Bad arguments.")
            sys.exit(1)

        # Main function
        main(sys.argv)
    except Exception as e:
        debug(str(e))
        raise
```