Network Segmentation and Enhancing Security in Hybrid On-Premises and Cloud Infrastructure

University of Essex Online

Abstract.

This project investigates security enhancements through network segmentation in hybrid cloud and on-premises environments. It examines Identity-Based Network Segmentation, and Micro-Segmentation approaches to tackle security vulnerabilities in hybrid environments. The project highlights the benefits of network segmentation in enhancing security, isolating critical assets, and ensuring regulatory compliance. The central project question focuses on the effectiveness of micro-segmentation compared to identity-based segmentation in augmenting security within hybrid infrastructures. The project aims to shed light on their relative effectiveness in mitigating risks and protecting critical assets against unauthorized access and lateral movement. The findings will provide valuable insights for organizations to make informed decisions about implementing network segmentation in their hybrid infrastructures to enhance security.

Table of Contents

ABSTRACT. **2**

1. INTRODUCTION. **6**

2. LITERATURE REVIEW. **8**

3. METHODOLOGY. **13**

4. NETWORK SEGMENTATION TECHNIQUES. **15**

4.1 About network Segmentation **15**

4.2 Identity-Based Segmentation **16**

4.3 Micro-Segmentation **17**

5. IMPLEMENTATION. **18**

5.1 Cloud setup. **18**

5.2 ESXi on premises setup. **18**

5.3 Interconnectivity. **19**

5.4 Host deployment. **20**

5.5 Vulnerable testing hosts in AWS. **20**

5.6 On premises scanners and information gathering. **22**

5.7 Segmentation Implementation – Identity Based. **22**

5.8 Segmentation Implementation – Micro-Segmentation. **23**

6. TESTING SCENARIOS. 25

6.1.1 Identity-Based Segmentation 25

6.1.2 Micro-Segmentation 26

7. RESULTS INTERPRETATION AND EVALUATION. 27

7.1 Evaluation Criteria 27

7.1.1 Identity-Based Segmentation 27

7.1.2 Micro-Segmentation 28

7.2 Security Effectiveness and Risk Mitigation: 30

7.2.1 Identity Based Segmentation 31

7.2.2 Micro-Segmentation 34

7.3 Validation and Verification 36

7.4 Discussion and Evaluation 37

7.5 Lessons Learned 37

7.6 Ethics 38

7.7 Future work and next steps 39

8. LIMITATIONS: 40

9. CONCLUSION. 42

10. REFERENCES: 44

11. ANNEX 1- TECHNICAL DETAILS 51

1. Introduction.

This project aims to investigate the security enhancements that network segmentation can bring to hybrid infrastructures both in the cloud and on-premises and how different approaches to segmentation can enhance security and reduce attack vectors in such environments.

This work's central project question or hypothesis investigates the efficacy of micro-segmentation in augmenting security within hybrid infrastructures, specifically in comparison to identity-based segmentation deployed in similar environments. The project question guiding this investigation is: "Does the implementation of micro-segmentation will result in the effectiveness of attack surface reduction, mitigation of the impact of breaches, host vulnerabilities reduction, as well as lateral movement in hybrid infrastructures, and how does it compare to identity-based segmentation within the same type of environment?"

This project is because there are multiple ways to implement segmentation on a given network, while a more complex one in cases such as a hybrid one can have its challenges. The best ways to implement such security methods and leverage the data outcome from this project are so that a guideline can be created on how different implementation methods can be advantageous compared to each other.

The focus of this project is to look at the most efficient ways of performing segmentation in hybrid environments where hosts can be compromised from a security standpoint and measure which way of implementing segmentation will provide better results with a focus on a hybrid cloud environment targeting cloud instances. The project focuses primarily on two selected ways of performing segmentation: micro-segmentation and identity-based segmentation, which are applied to the same hybrid environment when measuring the results.

The tests performed for both scenarios are going to be done by performing vulnerability scanning using an easily accessible tool called Nmap as well as a more in-depth type of scanner called Legion and Metasploit. The purpose of the tests is to capture the data needed for demonstrating the reduction of attack vectors and reducing vulnerability by minimizing access to vulnerable hosts or vulnerable protocol's running on vulnerable hosts.

This project aims to provide insights into the best practices and challenges associated with network segmentation, which can be a guideline for corporations with such environments. The project will investigate critical factors driving the adoption of network segmentation, the methodologies employed in its implementation, and the tangible benefits realized in terms of improved security posture, reduced attack surface, and enhanced incident response capabilities. Additionally, the project will examine network segmentation's potential drawbacks or limitations and identify strategies to address them effectively.

To accomplish these objectives, a mixed-methods approach will be employed, combining both qualitative and quantitative research methods and performing experimental research on the effectiveness of each type of network segmentation. A testing environment will be leveraged by connecting the AWS landing zone with an on-premises data center hosting a virtualized server interconnected with a redundant VPN tunnel.

The findings of this project will contribute to the existing body of knowledge on network segmentation and its role in enhancing cybersecurity. By examining a real-world case study, this project aims to provide practical insights and recommendations that organizations across different sectors can apply to strengthen their cybersecurity defenses.

Overall, the project presented in this paper underscores the importance of network segmentation as a vital component of a comprehensive cybersecurity strategy.

2. Literature review.

The safety of confidential information and vital resources is a primary responsibility for organizations deploying hybrid networks in the frantic digital environment of the modern connected world. (PricewaterhouseCoopers, 2019) (IBM Security. 2020). Comprehensive safety protocols must be implemented to protect against complex cyber-attacks and guarantee company continuity. (National Institute of Standards and Technology (NIST), 2020). Micro-segmentation and identity-based segmentation have both gained popularity as viable solutions. A comprehensive analysis of the body of research highlights that micro-segmentation offers significant benefits over identity-based segmentation in improving the effectiveness of security and operations inside hybrid infrastructures.

Mhaskar et al. (2021) defined network segmentation as a security measure that enforces the segregation of activities by grouping resources with comparable security rules under a typical firewall. Utilizing Product Family Algebra (PFA) and Guarded Commands, they offered a formal method of network segmentation, creating a robust, reliable network with improved access control (Mhaskar et al., 2021).

Network-based cybersecurity threats have recently increased in volume and intensity (Symantec, 2022), outpacing traditional defense strategies. These crimes result from several malevolent actors, from lone hackers to well-organized cyber-gangs with various goals such as financial gain, disruption of services, espionage, propaganda or influence operations, ransom attacks, and cyber warfare. (EUROPOL, 2023).

Rose et al. (2020) investigated the Zero Trust Architecture (ZTA), a new transformational security approach that puts individuals, resources, and assets ahead of network-based edges. It executes under the tenet that no implicit assurance should be given based on the possession of assets or placement within a network. They talked about Micro-

Segmentation and Enhanced Identity Governance. Although Enhanced Identity Governance provides first networking access, continuous surveillance is necessary to reduce risks since nefarious actors may still try to conduct network reconnaissance. In contrast, micro-Segmentation proactively safeguards resources from unauthorized access and excels at delivering strong security and flexibility (Rose et al., 2020).

Defenders must be alert and adaptable due to the constantly shifting terrain of attackers and networks. Threat hunting takes an aggressive approach instead of SOC analysts' reactive strategies. To define threat hunting, highlight its importance, and provide improvement measures, the SANS 2019 Threat Hunting Survey gathered information from 575 respondents. Results reveal that 35% handle SOC alerts, 56% utilize threat intelligence, and 34% use hypotheses. 71% of people say that technology is their primary concern, and as a result, 61% say that their security posture has improved by at least 11%. The research identifies misunderstandings around threat hunting and provides helpful advice. Assessing the consequences of threat hunting is difficult for organizations. The key findings support specialized methodology, competent staff, and hypothesis-driven hunts to improve security (Fuchs & Lemon, 2019).

Network defense uses both software and hardware solutions to resist harmful operations. Businesses have historically depended on a "fortress" concept, where a distinct line delineates the trusted interior from the distrustful exterior. On top of this idea, network segmentation creates a tiered fortress framework that has smaller forts inside. Segmentation has drawbacks despite improving defense levels and reducing threat maneuverability. The need to link policies with resource restrictions complicates deployment.

Furthermore, sticking to the fortress strategy has flaws. In their investigation, Simpson and Foltz attempted to integrate segmentation with Zero Trust Architecture (ZTA). The border

security elements of segmentation collide with a complete ZTA implementation. It is conceivable to use a hybrid strategy, with ZTA enhancing security inside segments. ZTA is the goal of a hybrid system for security, with segmentation delivering non-security advantages, including performance, reduced broadcast traffic, cost savings, and other efficiency (Simpson & Foltz, 2021).

In their research, Azodolmolky et al. examined IaaS architecture and investigated the substantial difficulties associated with network virtualization and cloud federation. IaaS provides economical options based on computational consumption, storage, and runtime; nevertheless, for maximum performance, it is essential to solve challenges, including application speed, device installation, enforcement of policies, and multi-layer network difficulties. By separating the data transmission and control planes and providing virtual networking solutions like VLAN and Nicira NVP, SDN positions itself as a possible option. However, additional research is needed to address adaptability, collaboration, and efficiency (Azodolmolky et al., 2013).

A flexible, adaptable, and dynamic manner of resource delivery to systems and their users is provided by cloud computing. In their study, Jeuk et al. pointed out the drawbacks of currently used segmentations, including VLAN, VxLAN, and GRE. The authors created a brand-new IPv6-only architecture that uses IPv6's Universal Cloud Classification (UCC) and Interface Identifiers (IIDs) for segmentation. The proposed structure makes possible precise endpoint recognition and cloud-specific traffic isolation, as shown using OpenStack. It enables network equipment to divide traffic into groups according to endpoint affiliation, tenants, and cloud service. The suggested method can be used outside OpenStack in different cloud frameworks to deliver effective and trustworthy results (Jeuk et al., 2015).

Utilizing hardware and software-based computing assets offered as a service across a network is called "cloud computing." As possible risks to cloud computing, security breaches, lost data, account theft, insecure APIs, denial of service (DOS) attacks, malevolent employees, misuse of cloud services, inadequate due diligence, and problems with shared technology were all covered by Prakash and Surajit. These dangers highlight the requirement for strong security measures to safeguard priceless data and preserve service integrity. In multi-tenant cloud settings, safety and separation are improved via a framework built on the Xen platform that uses page coloring-based cache partitioning and strategies including introspection, automation, and segmentation. In addition to being used as deception traps to entice and find potential attackers, honeypot devices are also employed to protect technological infrastructure within organizations (Prakash & Dasgupta, 2016).

In a similar vein, Mushtaq et al. investigated the complex framework of cloud computing, discovered security threats and holes in data transfer to the cloud, and suggested a workable solution using Single-Sign-On (SSO), Trusted Third Party (TTP) with Public Key Infrastructure (PKI), and Lightweight Directory Access Protocol (LDAP). Cloud computing can address difficulties with data and communication genuineness, accessibility, reliability, and secrecy by integrating SSO, LDAP, and PKI. A thorough safety and confidentiality trust assessment control mechanism for cloud services could be developed due to further investigations of this study, thereby enhancing overall service quality and consumer confidence in using clouds (Mushtaq et al., 2017).

Data center architecture has been the subject of significant research as cloud-based applications have expanded quickly. Systems with exceptional efficiency and minimal downtime are required for data centers with many servers. Mujib et al. believed network architecture is crucial for maintaining stability and integrity to satisfy changing data center demands. The authors used micro-segmentation to construct the zero-trust security paradigm

because they thought it was a helpful strategy. The study assessed the efficacy of the network in the data center utilizing software-defined networking and a zero-trust security approach. The findings showed that micro-segmentation improves protection without materially reducing network speed, making it a vital data center solution (Mujib & Sari, 2020).

NFV, SDN, and SDP are all used in micro-segmentation in the Internet of Things (IoT) to safeguard resources. This method has drawbacks, including a single breakdown point, and needs substantial network upgrades. According to Syed et al.'s research, a durable and distributed segmentation and SDP approach is required (Syed et al., 2022). For different interaction levels, a hybridized micro-segmentation strategy is recommended. Via SDN-based micro-segmentation, which uses independent device proxies to regulate access privileges for other users in IoT devices, customized access management is made possible. By properly routing user requests to the correct device proxies, the SDN-based network infrastructure improves safety and accessibility control.

To prevent data breaches and ransomware attacks, the Zero Trust architecture (ZTA) continually authenticates users' access to their devices and apps. Farook et al. claim that Zero Trust technologies assist IT managers lower vulnerabilities by enhancing access control, implementing dynamic rules, and securing endpoints. ColorTokens promotes the use of Zero Trust and provides security tools for networks. Cooperation within the organization, discussing the ZTA plan with all parties involved, and implementing password-less identification is necessary for successful deployment. The authors' main recommendations for implementing the zero-trust protection strategy embrace coaching, security evaluation, flow tracking, choosing appropriate methodologies, and continual monitoring for improved security examination (Farook et al., 2022).

3. Methodology.

This project examines the security enhancements associated with network segmentation in hybrid cloud and on-premises environments. It explores various approaches to network segmentation, including Identity-Based Network Segmentation and Micro-Segmentation. The project focuses on capturing efficient mechanisms against security vulnerabilities within the scope of network segmentation in hybrid environments.

The following project will involve a mixture of qualitative and quantitative methodologies, which will be applied to the data collected on the artifact created for the testing of each method of segmentation. The testing environment was created using Amazon Web Services (AWS) cloud for the cloud instances and an ESXi server instance storing all the on-premises virtual machines connected using a redundant VPN tunnel between the data center and the cloud provider.

The investigation design employed in the following project is based on an experimental approach, targeting qualitative and quantitative measurements. The reason is that the test results in the lab environment target both reduction of the attack vector and the level of efficiency that needs to be evaluated.

The project's objective is to measure the effectiveness of the measurements tested in the network topology created and measure the results against baseline testing and each other evaluating efficiency and quality.

The data collection process will primarily be from the testing done on the hybrid environment created and, in the process, identifying best practices. The project will investigate the traffic logs generated by the tests performed on the environment and the actual output of the testing done.

The variables and measurements for this project are predominantly focused on security applied to the selected scenarios: identity-based segmentation and micro-segmentation. Network segmentation's benefits in improving security, isolating critical assets, and enhancing regulatory compliance are discussed.

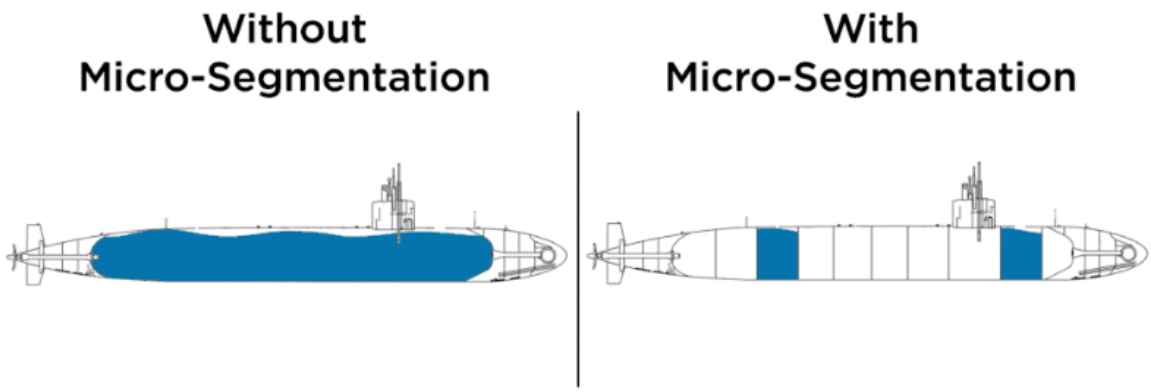
The experimental configuration comprises a cloud environment, specifically Amazon Web Services, and an ESXi server located in a data center. Two redundant site-to-site Virtual Private Network (VPN) tunnels connect these components. The cloud provider manages the VPN connections on the cloud side. In contrast, the termination of the VPN tunnels on the data center side is facilitated by a Palo Alto virtual machine hosted on the server itself. Customized virtual machines were created using a base Ubuntu operating system to execute the experimental setup. These virtual machines were equipped with specific secure and unsecure applications by industry standards. The scanning activities were conducted using the Kali Linux distribution and built-in tools like Metasploit, Legion, and Nmap.

Overall, this project will contribute to understanding network segmentation in hybrid environments and reference organizations in different sectors implementing network segmentation in their respective domains.

4. Network Segmentation Techniques.

4.1 About network Segmentation

Fortytwo provides a simplified analogy for network segmentation. (n.d.) that divides a submarine into segments which, if any of it is compromised, does not affect the entire warship. The concept of network segmentation is similar as it is known as dividing networks into multiple segments to reduce lateral movement or limiting the overall areas that might be affected by a malicious actor.



Fortytwo. (n.d.)

According to Simpson, W.R. and Foltz, K.E., (2021), "is a term for dividing a network into multiple subnetworks, or segments, and managing access to these segments. Typically, it involves segregating traffic between the network segments and enforcing segment policies with firewalls or other security appliances".

There are multiple ways to achieve segmentation in an organization. According to Simpson, W.R. and Foltz, K.E., (2021), it can be achieved by either looking at the low-level instances connected to the network, which is called micro-segmentation also having a higher-level approach at the other end of the spectrum macro-segmentation where we look at an entire network rather than individual hosts.

Zero trust architectures are a strategic approach to cybersecurity where networks in organizations are defining so, simply put, nothing is trusted by default, at the core of which lies different ways of performing network segmentation. That is achieved by breaking up the security perimeter into smaller zones by performing traditional VLAN (Virtual Local Area Network) segmentation to maintain access to separate network parts. (Cloudflare, n.d.).

The National Institute of Standards and Technology (2020) suggests that to implement zero-trust architectures effectively, we must enable and deploy micro-segmentation. Organizations can adopt the deployment of a host-based software agent or firewall on the endpoint device. This approach involves utilizing specialized software or firewall solutions installed directly on individual devices, such as workstations or servers, to enforce strict access controls and continuously verify trustworthiness. By leveraging these endpoint-level security measures, organizations can create a granular security framework where each device is treated as untrusted by default, regardless of location or network connectivity. This strategy enables organizations to achieve enhanced protection against potential threats, minimize lateral movement within the network, and ensure access privileges are granted on a need-to-know basis.

4.2 Identity-Based Segmentation

According to CrowdStrike. (n.d.) identity segmentation is a crucial pillar of the Zero Trust Security framework and is a method of restricting access to applications or resources by leveraging identities. The technique goes beyond traditional ways of looking at networks as it applies security policies not bound to IP addresses or specific access lists. It improves security by preventing authorized access to sensitive resources that can be spoofed by getting the correct IP address. It also reduces the attack surface (Cytracom, n.d.) by limiting the scope of the network while simultaneously limiting lateral movement. Elisity. (n.d.).

According to Gartner. (2021) there are three identity segmentation styles: agent-based, hypervisor-based, and network-based. In this project, we will be looking at network-based identity segmentation by leveraging the capabilities of the Palo Alto Next Gen Firewall, which is application aware but also capable of allocating identities to specific hosts through authentication.

4.3 Micro-Segmentation

Micro-segmentation is a security technique that allows organizations to divide their network into workload-level segments, providing the possibility to create security controls for each unique segment (VMware. n.d.). The significant difference compared to other segmentation techniques is that it allows the creation of specific security policies for the application-level workflow and creates flexibility when enforcing them. It limits the attack surface by restricting the ability to spoof specific attack network ports using different applications (Palo Alto Networks. n.d.)

There are several ways of implementing micro-segmentation by either leveraging dedicated tools such as Illumio that will allow users to whitelist traffic between specific hosts (Sheikh et al., 2021) as well as creating reliable firewall policies at the host level, which restricts traffic based on need, the later having scalability limitation as in environments where hosts grow exponentially it will be challenging maintaining such firewall rulesets. This research project will investigate applying micro-segmentation in the cloud to multiple hosts grouped based on the purpose of the host by employing security groups that can be used for numerous hosts grouped for specific workloads.

5. Implementation.

5.1 Cloud setup.

AWS (Amazon Web Services) was chosen as the cloud provider to facilitate the deployment and examination of proposed scenarios. The experimental environment encompasses a VPC (Virtual Private Cloud) partitioned into three subnets. One subnet permits public access to the internet. In contrast, the remaining two subnets are exclusively accessible within the confines of the VPC or via the VPN (Virtual Private Network) connection established from the data center.

5.2 ESXi on premises setup.

The on-premises infrastructure configuration entails an ESXi server housed within a data center equipped with an internet uplink. Additionally, a Palo Alto Next-Generation Firewall VM (Virtual Machine) has been deployed to fulfill the role of an endpoint VPN (Virtual Private Network) to establish connectivity with the cloud provider. At the ESXi level, the network configuration is structured into four distinct VLANs (Virtual Local Area Networks) or port groups, each serving specific purposes as outlined below:

Outside: This network segment is directed towards the internet and serves as the gateway to the cloud provider. It facilitates communication and connectivity between the on-premises setup and the external network.

DMZ: This network segment is designated for hosting virtual machines intended for testing lateral movement. It provides a segregated environment to analyze and assess potential security risks associated with lateral network traversal.

Inside Zone 1: This Zone is allocated for hosting virtual machines that serve specific purposes within the infrastructure. It provides a controlled environment for deploying and managing virtual machines for various functionalities.

Inside Zone 2: Like Inside Zone 1, this Zone is reserved for hosting virtual machines within the on-premises infrastructure. It enables the segregation and management of virtual machines, ensuring distinct functionalities and security boundaries.

In the Palo Alto Next-Generation Firewall, the zoning configuration follows a comparable approach, with the exception that the inside zones are consolidated into a single entity. Specifically, the rulesets governing the desired traffic, facilitating communication between the on-premises and cloud environments, are established based on the specific testing scenario.

Multiple virtual machines are hosted within the data center, including a jump server running Windows Server 2022. The jump server is a centralized access point to reach other virtual machines within the infrastructure. Additionally, virtual machines dedicated to information gathering and testing, such as the Kali Linux instance, are deployed to facilitate security assessments and exploration.

5.3 Interconnectivity.

The Connection between the cloud provider and the on-premises data center is established through a site-to-site VPN. This VPN connection is configured to terminate on the Palo Alto virtual machine, serving as the endpoint within the on-premises environment. The site-to-site VPN ensures secure and encrypted communication between the cloud provider and the data center, facilitating seamless connectivity and data exchange between the two domains.

5.4 Host deployment.

The deployment of hosts on the cloud provider is accomplished by utilizing the EC2 service, ensuring that they are appropriately positioned within the corresponding Virtual Private Cloud (VPC). On the other hand, in the ESXi environment, host deployment is executed using specific images tailored to the particular testing requirements dictated by the selected scenario.

5.5 Vulnerable testing hosts in AWS.

To ensure the instances on the AWS side meet the testing objectives, a foundational Ubuntu image has been chosen as the base. Subsequently, specific services are installed on these instances to intentionally introduce vulnerabilities, enabling the measurement of each segmentation's effectiveness. A suite of applications, including HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), TFTP (Trivial File Transfer Protocol), and Telnet, along with secure protocols like SSH (Secure Shell) and HTTPS (Hypertext Transfer Protocol Secure), has been selected for conducting the designated tests on these hosts.

The instances used for vulnerability scanning are deployed using the steps provided in Annex 1, section 1-f.

After deploying the instances, additional configurations are required to prepare them for testing. This entails installing various applications and protocols, both open-source and insecure, as well as secure applications like HTTPS. The following steps were undertaken to accomplish this setup:

- Installation of insecure applications and protocols.
- Implementation of secure applications, including HTTPS.

The instances were adequately configured by following these steps to facilitate the intended testing procedures.

This is achieved by running the following commands:

```
sudo apt update

sudo apt install -y net-tools apache2 vsftpd tftpd-hpa telnetd

sudo a2enmod ssl

sudo systemctl restart apache2
```

As a result, our hosts will be configured to listen on specific ports for incoming connections, encompassing both secure and insecure protocols. The list of ports includes:

Secure Ports:

HTTPS (Port 443)

Secure Shell (SSH) (Port 22)

Unsecure Ports:

Hypertext Transfer Protocol (HTTP) (Port 80)

File Transfer Protocol (FTP) (Port 21)

Telnet (Port 23)

These ports have been designated to facilitate communication and testing within the experimental setup.

5.6 On premises scanners and information gathering.

The initial phase of the project involves conducting information gathering through the use of two distinct scanners. The scanners used as part of the information gathering against the hosts are carried out using a scanner integrated into the Kali Linux distribution known as Legion. This scanner utilizes Nmap as its underlying port scanning tool. It offers insights into Common Vulnerabilities and Exposures (CVEs) along with capturing screenshots of the web pages running on the scanned machines. These scanners are employed to perform a comprehensive scan across the entire subnet hosted within the cloud provider's CIDR range, explicitly targeting the 11.0.0.0/24, 11.0.1.0/24, and 11.0.2.0/24 subnet.

The purpose of conducting the initial scan with both tools is to establish a foundational assessment of the existing hosts, serving as a benchmark for subsequent measurements. These measurements will be compared against the results obtained from different segmentation techniques explored in this project.

5.7 Segmentation Implementation – Identity Based.

The implementation of identity-based segmentation entails a step-by-step process that ensures effective control and management of network traffic. The first crucial step involves defining the identity of the application within the ruleset of the on-premises firewall. This identity serves as a unique identifier for the host and forms the basis for applying specific policies governing network communication.

Once the identity is established, tailored policies have been crafted and enforced to regulate the traffic associated with this identity. These policies were designed to align with the network environment's desired security requirements and operational needs. They determine whether incoming or outgoing traffic from the identified host should be allowed or denied based on factors such as source, destination, protocol, and port.

To evaluate the effectiveness of segmentation, a comparative analysis was conducted. This analysis involves implementing different restriction policies for the same identity, explicitly targeting the virtual machines hosted on the cloud provider side. By applying these distinct policies, the network environment undergoes a controlled transformation, enabling a before-and-after measurement of the segmentation's impact.

5.8 Segmentation Implementation – Micro-Segmentation.

In implementing micro-segmentation, the project approach involves leveraging the security group functionality offered by the cloud provider. This feature is a vital firewall ruleset applied to each host within the Virtual Private Cloud network infrastructure. By utilizing security groups, the aim is to establish granular control over the network traffic flowing to and from these hosts.

A single security group has been allocated to each host during the initial deployment phase to ensure a streamlined implementation process. This security group acts as a foundational configuration that can be subsequently modified to accommodate the specific requirements of the before and after testing stages. The assigned security group currently permits unrestricted communication between the on-premises data center and the cloud instances. This permissive policy allows for comprehensive visibility and analysis of all the vulnerable hosts during the initial scanning phase, which also serves as a baseline result for evaluating the segmentations implemented.

By strategically manipulating the security group rules and policies, we can effectively enforce segmentation and restrict network traffic flow between host entities stored in the Virtual Private Cloud and on-premises data center hosts, including scanners. This controlled approach enables assessment of the impact of micro-segmentation by comparing the outcomes before and after implementing more stringent security measures. Through this

iterative process, we can evaluate the efficacy of the applied security policies, identify any vulnerabilities or gaps, and refine our approach to enhance network security and protect against potential threats.

6. Testing scenarios.

The testing scenarios have been selected (ESRB, 2022) to highlight the security mitigations and risk mitigation for each type of segmentation evaluated. Each method was evaluated against the other as well as against a baseline evaluation done without any segmentation at all. The results gathered are stored in Annex 1, section 2a, and serve as a reference point for further assessment if needed. The selected testing will be performed using integrated scanners as part of the Kali Linux distribution as one of the industry's go-to tools for information gathering. (Allen, et al., 2014)

6.1.1 Identity-Based Segmentation

The first testing scenario focuses on identity-based segmentation, which restricts access to resources based on application identities rather than IP addresses or specific access lists. The goal is to evaluate the effectiveness of this segmentation method in preventing unauthorized access and reducing the attack surface. The following steps have been taken:

Identity Establishment: Unique identities has been defined for each application within the ruleset of the on-premises firewall. These identities will serve as the basis for applying specific network communication policies. The application selected for this test is the secure ones meaning SSH running over TCP port 22 and HTTPS running over port TCP 443.

Policy Enforcement: Tailored policies have been crafted and enforced to regulate traffic associated with each identity. These policies determine whether incoming or outgoing traffic should be allowed or denied based on the application identity selected.

Comparative Analysis: The restriction policies will be implemented for the same identity, targeting the cloud instances. The impact of identity-based segmentation will be evaluated by comparing the outcomes before and after the implementation of segmentation measures against the baseline evaluation.

Assessment of Security: The effectiveness of identity-based segmentation will be assessed based on its ability to prevent authorized access, reduce the attack surface, and limit lateral movement.

6.1.2 Micro-Segmentation

The second testing scenario focuses on micro-segmentation, which allows us to divide their network into segments at the instance level and create specific security controls for each segment. The objective is to evaluate the effectiveness of micro-segmentation in securing application-level workflows and reducing the attack surface. The following steps have been followed:

Security Group Allocation: Each host within the network infrastructure has assigned a security group, which acts as a firewall rule applied to that host. The security group controls network traffic flow to and from the host.

Manipulation of Security Group Rules: The security group rules and policies have been strategically modified to enforce segmentation and restrict network traffic flow between different host entities. In the scenario selected, only traffic for HTTPS running over TCP port 443 and SSH running over TCP port 22 has been chosen to replicate the scenario for identity-based segmentation.

Impact Assessment: The impact of micro-segmentation has been assessed by comparing the outcomes before and after the implementation of stricter security measures done against the baseline testing. The reduction of attack vectors, efficiency, and quality has been evaluated.

7. Results interpretation and evaluation.

7.1 Evaluation Criteria

The evaluation criteria used to assess the results will prioritize security effectiveness and risk mitigation. The following criteria will be considered for each testing scenario:

Security Effectiveness: The extent to which the segmentation technique effectively reduces attack vectors, mitigates breaches, and enhances overall network security.

Risk Mitigation: The ability of the segmentation method to limit lateral movement, prevent unauthorized access, protect critical network assets, and minimize potential vulnerabilities.

7.1.1 Identity-Based Segmentation

The results obtained from the identity-based segmentation testing scenario have been interpreted based on the evaluation criteria. The interpretation focuses on the security effectiveness and risk mitigation capabilities of identity-based segmentation, considering the reduction of attack vectors, efficiency, quality, and alignment with the CIA Triad (Confidentiality, Integrity, and Availability) principles.

In the assessment conducted to examine identity-based segmentation, the data obtained was acquired through a conventional methodology after the implementation of a ruleset on the Palo Alto Next-Generation Firewall. The implemented rules were specifically designed to authorize only HTTPS and SSH as identity applications originating from the data center towards the AWS cloud environment instances, which initially allowed unrestricted traffic for these hosts. The scanning procedure involved the utilization of both Legion and Nmap tools to assess TCP and UDP ports. These tools were leveraged by using Kali Linux,

and a comprehensive outcome of this process can be found in the designated section of Annex 1 section 2-b.

The scanning activity encompassed all subnets associated with the Virtual Private Cloud (VPC), including Public (11.0.2.0/24), Private 1(11.0.0.0/24), and Private 2 (11.0.1.0/24) subnets. The collected results were consolidated simultaneously for all instances. Initial observations indicated that the stringent ruleset effectively filtered out traffic directed towards ports other than SSH and HTTPS, as these were the only permitted types of traffic based on the application identity established at the data center perimeter firewall. The entirety of the remaining traffic, along with the vulnerabilities inherently associated with it, was effectively eliminated. Consequently, the corresponding attack surface associated with these removed elements was successfully reduced compared to the baseline configuration.

7.1.2 Micro-Segmentation

The results obtained from the micro-segmentation testing scenario were interpreted based on the evaluation criteria. The interpretation focused on the security effectiveness and risk mitigation capabilities of micro-segmentation, considering the reduction of attack vectors, efficiency, quality, and alignment with the CIA Triad (Confidentiality, Integrity, and Availability) principles.

The interpretation of results involved analyzing the effectiveness of micro-segmentation in providing granular security controls at the workload level, reducing the attack surface, and preventing lateral movement within the network. The evaluation considers the ability of micro-segmentation to protect critical network assets, minimize potential vulnerabilities, and uphold the confidentiality, integrity, and availability of network resources.

The outcomes related to micro-segmentation were obtained using a similar methodology, with the addition of an extra scanning technique and specifically by utilizing

Metasploit as a scanner. This inclusion was made to leverage the superior security offered by this implementation. Notably, the host-level filtering of ICMP (Internet Control Message Protocol) rendered Legion, which relies on Nmap for host discovery, incapable of identifying the hosts operating within the cloud environment, irrespective of the ports in use.

To address this limitation, an auxiliary scanner provided by Metasploit was employed for an additional scan. This scan targeted the subnets within the AWS cloud provider, focusing on the HTTPS and SSH ports, as was reflected in the security group rules update. The obtained results were duly recorded and can be examined in the specified section of Annex 1, section 2-c.

Micro-segmentation emerged as the preferred approach due to its inherent flexibility, offering the ability to fine-tune security measures for each individual host through the utilization of dedicated security groups. In this project, a single security group was employed as a practical means of testing scalability and feasibility. However, it is essential to recognize that the true potential of micro-segmentation lies in its capability to allocate additional security groups tailored to specific host roles within the infrastructure. (Klein, 2019)

For instance, different types of host instances, such as database instances or frontend instances, may necessitate distinct security configurations to address their unique requirements. By assigning separate security groups to these hosts, it becomes possible to adjust the security group rules based on the specific needs and communication matrix mandated by the associated application. This fine-grained control enables organizations to implement access permissions that precisely align with the operational demands and security considerations of each individual host. (Sarkar et. al. 2022)

Moreover, the flexibility of this approach in micro-segmentation facilitates the dynamic allocation of security groups as the infrastructure scales. As new hosts are added to the environment, security groups can be provisioned and tailored to suit their intended purpose and role. This ensures that each host receives the appropriate level of access control and safeguards, thereby bolstering the overall security posture of the micro-segmented infrastructure.

In summary, micro-segmentation's advantage lies in its ability to offer granular control over security measures through the utilization of dedicated security groups. By allocating specific security groups to hosts based on their roles and adjusting the associated rules, organizations can tailor access permissions to meet the unique needs of each host and effectively manage the communication requirements of the applications they support. This flexibility not only enhances security but also enables seamless scalability and adaptability in dynamic hybrid environments.

7.2 Security Effectiveness and Risk Mitigation:

Through the interpretation of results, the aim is to gain insights into the effectiveness of each segmentation method in reducing attack vectors, preventing unauthorized access, and protecting critical network assets. Furthermore, evaluating their capabilities in mitigating breaches, limiting lateral movement, and minimizing potential vulnerabilities.

By analyzing the security effectiveness and risk mitigation of these segmentation techniques, the aim is to provide a comprehensive understanding of the strengths and weaknesses of such segmentation. This evaluation will facilitate informed decision-making for organizations seeking to bolster their network security in hybrid infrastructures, enabling them to select the most suitable approach based on the specific security requirements and risk tolerance of their environments.

The evaluation criteria prioritize the extent to which each technique reduces risks and mitigates potential threats within hybrid infrastructures while considering the alignment with the CIA Triad principles of confidentiality, integrity, and availability.

7.2.1 Identity Based Segmentation

Within the scope of identity-based segmentation, a crucial aspect of this project involved leveraging the advanced capabilities of the Palo Alto Next-Generation Firewall. This firewall solution incorporates features specifically designed to identify and control applications using a variety of techniques. By implementing this functionality, the project aimed to establish granular control over network traffic based on the identity of the applications themselves. (Francis, 2018)

In this scenario, the concept of identity revolves around the applications running within the network infrastructure from on-premises to the cloud environment. Rather than focusing solely on IP addresses, the project prioritized the identification and management of applications as the basis for granting or denying traffic permissions. By associating unique identities with each application, the firewall could discern and differentiate between various applications, enabling precise control over their network communication. This can be expanded to encompass applications running on nonstandard protocols and port numbers. In such scenarios, the firewall will still filter the traffic based on the application's characteristics rather than the port itself.

Throughout the testing phase, the project employed strict policies to regulate the permitted protocols within the network environment. Specifically, only secure protocols such as SSL (TCP 443) and SSH (TCP 22) were authorized to traverse the network. This approach aimed to enhance the security posture by prioritizing secure communication channels and reducing the potential exposure to vulnerabilities associated with insecure protocols.

By adopting this identity-based segmentation approach, the project targeted to evaluate its effectiveness in enforcing strict access controls and mitigating the risk of unauthorized access to critical resources, which in our case are the cloud instances. The focus on secure protocols not only ensured the confidentiality and integrity of data transmission but also aligned with industry best practices for secure network communication. (McKay and Cooper, 2019)

The observations made during the project revealed a notable reduction in the accessible ports as perceived by the scanner. Specifically, the firewall implemented in the identity-based segmentation framework restricted access to only two ports based on the identity of the selected application. In the conducted testing, the permitted ports were limited to SSL (TCP 443) and SSH (TCP 22) as per the attached screenshot of the rule base on the Palo Alto Next-Generation Firewall in Annex 1 Section 2-b. This stands in contrast to the original reference scan, which allowed access to a total of seven ports: SSL (TCP 443), SSH (TCP 22), FTP (TCP 21), Telnet (TCP 23), HTTP (TCP 80), DHCP (UDP 68), and TFTP (UDP 69).

	Baseline Scan	**Identify based scan**	**Delta ports mitigated/filtered**
Secure ports	HTTPS (TCP 443), SSH (TCP 22)	HTTPS (TCP 443), SSH (TCP 22)	
Unsecure ports	FTP (TCP 21), Telnet (TCP 23), HTTP (TCP 80), DHCP (UDP 68), TFTP (UDP 69). ICMP		FTP (TCP 21), Telnet (TCP 23), HTTP (TCP 80), DHCP (UDP 68), TFTP (UDP 69).

The outcome of this implementation resulted in a significant reduction in the attack surface by approximately 71.4%. By limiting access to a smaller number of ports, specifically those associated with secure protocols, the network environment became more resilient against potential threats. This reduction in attack surface denotes a substantial enhancement in security, as fewer entry points are available for malicious actors to exploit. A further detail of the decline of ports accessible can be found in Annex 1 Section 2-b.

The findings indicate the effectiveness of the implemented identity-based segmentation approach in mitigating risks and fortifying the network's security posture. By allowing access exclusively to SSL and SSH ports, which are widely recognized for their encryption capabilities and secure communication standards, this implementation of identity-based segmentation successfully decreased the potential attack surface and minimized the exposure to vulnerabilities associated with other open ports.

It should be noted that this particular implementation did not decrease the vulnerabilities linked to the protocol on which the application operates on the scanned hosts. The limitation of this assessment was solely centered on reducing the attack opportunities and restricting unauthorized access to the application. However, further research is necessary to address the evaluation of application versions separately.

These outcomes highlight the significance of identity-based segmentation as a powerful security mechanism, enabling organizations to tailor their access controls based on application identities. The observed reduction in the attack surface compared to the base assessment reaffirms the efficacy of this approach in enhancing network security and aligning with the principles of risk mitigation and safeguarding critical assets.

7.2.2 Micro-Segmentation

The implementation of micro-segmentation demonstrates notable security effectiveness in this scenario. By restricting management access over SSH exclusively to the data center, the network environment adopts a zero-trust approach, ensuring that potential exploits cannot infiltrate the organization from any zone, including the management one (Rose et al. 2020). This restrictive policy significantly enhances the security posture by reducing the attack surface and mitigating the risk of unauthorized access from both internal and external sources as per the zero-trust architecture approach. (Rose et al., 2020)

Furthermore, by adjusting the rules in the security group to remove All Permitted Traffic and other ports, leaving only SSH and HTTPS towards the cloud instances, the micro-segmentation implementation effectively isolates the cloud-hosted instances that can serve as website hoisting serving cloud clients from unauthorized scanning attempts. The removal of All traffic permitted from the security group access lists prevents the information-gathering scanner from discovering the end hosts stored in the cloud, enhancing security by limiting the visibility of potential targets. To validate the open ports on the instance level, additional scanners had to be employed to perform the information gathering and analysis of the implementation. Metasploit was used to scan the instances for the open ports, which revealed that SSH was accessible from the on-premises network and HTTPS was only accessible from within the cloud environment as per the policy defined in the security group. More technical details about the outcome of the micro-segmentation results can be found in Annex 1 Section 2-c.

The micro-segmentation approach enables us to apply traffic filtering at the individual instance level rather than at the network perimeter through the firewall in the data center. This grants the flexibility to define more stringent rules within the same subnet or environment and

limit traffic at the instance level, which also ensures that lateral movement is significantly reduced as a compromised host is unable to communicate with others, except over HTTPS, as per the defined rules. However, implementing this scenario in identity-based segmentation is not feasible due to the flow of traffic between the environments.

Micro-segmentation serves as a robust risk mitigation strategy in this scenario. By compartmentalizing the network and enforcing strict access controls, the risk of lateral movement within the network environment is significantly reduced due to the limitation of traffic. The isolation of the cloud-hosted websites and instances from the data center through the micro-segmentation implementation ensures that any potential security breaches or compromises in one segment do not propagate to other segments.

Moreover, the removal of unnecessary ports from the security group rules mitigates the risk of unauthorized access attempts and potential exploits. By allowing only SSH traffic from the data center for management purposes and HTTPS traffic only from cloud instances, the cloud network environment minimizes the exposure to vulnerabilities associated with other open ports. This risk mitigation strategy limits the potential attack vectors and fortifies the security posture of the micro-segmented infrastructure.

	Baseline Scan	**Micro Segmentation scan**	**Delta ports mitigated/filtered**
Secure ports	HTTPS (TCP 443), SSH (TCP 22)	SSH (TCP 22)	HTTPS (TCP 443),
Unsecure ports	FTP (TCP 21), Telnet (TCP 23), HTTP (TCP 80), DHCP (UDP 68), TFTP (UDP 69). ICMP		FTP (TCP 21), Telnet (TCP 23), HTTP (TCP 80), DHCP (UDP 68), TFTP (UDP 69).

The combination of security effectiveness and risk mitigation offered by micro-segmentation in this scenario ensures a robust defense against potential threats and unauthorized access attempts. By isolating critical resources and enforcing strict access controls, the network environment attains a higher level of security, reducing the likelihood of security breaches and minimizing the potential impact of such incidents. (Borky and Bradley, 2018)

7.3 Validation and Verification

Indeed, the project question has been effectively addressed, and the analysis of the gathered data indicates that the micro-segmentation technique surpasses identity-based segmentation in terms of its efficacy in reducing the attack surface, mitigating the impact of breaches, reducing host vulnerabilities, and limiting lateral movement within the network. The findings reveal that micro-segmentation offers superior security outcomes compared to identity-based segmentation in the context of hybrid infrastructures.

The artifact development process was conducted with unwavering integrity, ensuring that no deliberate manipulation of hosts took place to influence the project outcomes. Throughout the experimentation, the same set of hosts with identical vulnerabilities was consistently employed for testing both segmentation techniques. This approach was diligently adopted to maintain a high level of consistency and eliminate any potential bias that might arise from using different hosts for each method.

Moreover, meticulous documentation of the firewall rules governing the communication between the data center and the AWS environment was diligently recorded in Annex 1. This comprehensive documentation serves as tangible evidence, attesting to the accurate representation of the network environment and the measures taken to ensure the validity and reliability of the project results. By employing the same hosts and capturing the

firewall rules, the project aimed to establish a fair and robust comparison between the two segmentation techniques, further reinforcing the credibility and rigor of the findings.

7.4 Discussion and Evaluation

While this project prioritized cost-minimization in the artifact for the implementation and evaluation of segmentations, a more comprehensive and detailed project approach could have been pursued if budget constraints were not a factor. In such a scenario, leveraging custom paid tools developed by the industry could have led to more accurate and precise results. Additionally, the testing environment artifact was tailored specifically for this project's requirements, but using a replicated production environment for testing these segmentations could have provided valuable insights into implementation limitations and potential availability risks associated with each approach. The use of a replicated production environment would have allowed for a more realistic evaluation of the segmentations in real-world conditions and scenarios.

7.5 Lessons Learned

Upon completing the project implementation, valuable insights have been gained, prompting consideration of certain improvements for future research endeavors. To enhance cost-effectiveness, utilizing simulation techniques instead of real-live implementations with AWS and a datacenter setup could prove advantageous. Simulation methodologies offer comparable outcomes at significantly lower costs.

For future host selections, expanding the range of applications tested and exploring various protocol stacks would contribute to a more comprehensive evaluation of segmentation techniques. Additionally, incorporating additional information gathering tools and conducting detailed analyses of lateral movement reduction would augment the depth of the research.

Effective time management and a well-structured project breakdown are essential for successful research execution. By allocating ample time to each phase and adhering to a systematic project plan, the project process can be streamlined and optimized.

Recognizing that security is an ongoing process, the research underscores the importance of continuous improvement and adaptation of strategies to address emerging threats effectively. Implementing robust security measures requires constant vigilance and the flexibility to adapt to evolving threat landscapes.

In future projects, a greater emphasis on evaluating user experience, as well as assessing the complexity and difficulty of implementation, would provide valuable insights into the practicality and user-friendliness of the segmentation techniques deployed. Understanding the end-users' perspectives and experiences can inform further refinements to optimize the efficacy of implemented strategies.

In conclusion, this project has provided valuable lessons, encouraging a focus on cost-effectiveness through simulation techniques, comprehensive host selections, efficient time management, continuous security improvement, and an enhanced emphasis on user experience and implementation complexities. By integrating these lessons learned, future research endeavors can advance the understanding and implementation of network segmentation techniques in hybrid infrastructures.

7.6 Ethics

The main objective of this project is to improve network security through the implementation of micro-segmentation and identity-based segmentation techniques. To ensure the integrity of the evaluation, these segmentation approaches were tested in a carefully controlled environment expressly set up for this project. However, it is essential to

acknowledge that when deploying these techniques in a production environment, the potential risk of unintended service disruptions should be carefully evaluated and addressed.

Throughout the project, a commitment to full disclosure and transparency was maintained. This approach ensured that all project findings, methodologies, and results were fully documented, making it possible for others to trace and replicate the experiments with confidence.

7.7 Future work and next steps

Although this project's primary focus was on implementing micro-segmentation and identity-based segmentation in hybrid cloud environments, there are several potential areas for future work and next steps to consider:

Multi-Cloud Environment: Future research could explore the application of similar segmentation techniques in multi-cloud environments, where different cloud providers are interconnected and provide a testing sandbox for parallel implementation. This would provide valuable insights into how segmentation can be effectively deployed in complex, distributed cloud architectures.

Dynamic Segmentation: While the segmentations employed in this project utilized static mapping of specific policies to enhance security, an alternative approach worth exploring is dynamic segmentation. Further research and analysis could be conducted to investigate the benefits and challenges of dynamically adjusting segmentation rules based on real-time risk assessments and network conditions in similar environments.

8. Limitations:

Limited Scope: This project focuses on comparing the effectiveness of micro-segmentation and identity-based segmentation in a specific hybrid infrastructure setup. The findings may not be directly applicable to other network environments or different cloud providers, or different environments other than the one used in the project. Further projects are needed to explore the generalizability of these findings across a broader range of scenarios.

Resource Constraints: The testing scenarios in this project were conducted with a limited number of hosts and applications. The effectiveness of segmentation techniques may vary when applied to larger-scale environments with numerous interconnected systems as well as different applications or vulnerabilities. The resource limitations in the testing setup may have influenced the outcomes and should be taken into consideration when interpreting the results.

Simplified Threat Model: The project focused on evaluating the reduction of attack vectors and risk mitigation capabilities of segmentation techniques. However, the analysis did not consider the full range of potential threats and attack vectors that organizations may face in real-world scenarios. Future projects could explore the impact of segmentation techniques on specific types of attacks and vulnerabilities, as this project looked at a generic approach to vulnerabilities.

Single Testing Environment: The evaluation and comparison of segmentation techniques were conducted in a specific testing environment with predefined network configurations. The results may not fully capture the diverse range of network setups and designs found in various organizations. Future Projects could include multiple testing

environments to provide a more comprehensive analysis of the performance and effectiveness of segmentation techniques across different hybrid scenarios.

Time Constraint: The research project was conducted within a limited timeframe, which may have restricted the depth and breadth of the experiments and analysis. Given more time, additional scenarios and variations could be explored to provide a more comprehensive understanding of the strengths and limitations of micro-segmentation and identity-based segmentation in hybrid infrastructures.

Assumptions and Simplifications: The research project made certain assumptions and simplifications to facilitate the experimentation and analysis process. These assumptions, such as the selection of specific protocols and applications for testing, may not fully capture the complexity and diversity of real-world network environments. Future research could consider a broader range of protocols, applications, and network configurations to obtain more comprehensive insights.

9. Conclusion.

Based on the results gathered and evaluated in this project, it can be concluded that micro-segmentation emerges as a superior candidate compared to identity-based segmentation in hybrid infrastructures.

Micro-segmentation, as demonstrated in the testing scenario, showcased its effectiveness in reducing attack vectors, enhancing security controls, and limiting lateral movement within the network. By implementing granular security policies at the instance level and leveraging cloud provider security groups, this implementation of micro-segmentation provided a robust defense mechanism that significantly mitigated risks and protected critical network assets for the cloud environment instances.

The findings clearly highlight the advantages of micro-segmentation over identity-based segmentation in terms of security effectiveness, risk mitigation, and operational feasibility. Micro-segmentation allows for tailored security policies at the instance level, ensuring that each unique segment is protected with specific controls based on its requirements regardless of the traffic permitted on the network perimeter. This approach minimizes the attack surface, prevents unauthorized access, and enables organizations to respond effectively to potential security breaches.

While identity-based segmentation has its merits, particularly in environments where the number of hosts and applications is limited, it may not provide the same level of flexibility as micro-segmentation. The ability of micro-segmentation to address the specific security needs of each instance, targeting its roles, and its adaptability to changing network environments make it a more suitable choice for organizations seeking comprehensive network security in hybrid infrastructures.

In conclusion, based on the results gathered and evaluated in this project, micro-segmentation emerges as the preferred choice over identity-based segmentation in hybrid on-premises and cloud infrastructures. Its effectiveness in reducing attack vectors, enhancing security controls, and accommodating scalability and flexibility requirements positions it as a superior candidate for organizations aiming to bolster their network security defenses. By adopting micro-segmentation as a critical strategy, organizations can establish a robust security framework that aligns with the evolving threat landscape and protects essential assets of hybrid infrastructures.

10. References:

Advanced Computer Science and Applications, 8(10). https://doi.org/10.14569/IJACSA.2017.081025

Allen, L., Heriyanto, T. and Ali, S., 2014. Kali Linux–Assuring security by penetration testing. Packt Publishing Ltd. Available from: https://books.google.com/books?hl=en&lr=&id=QcBGAwAAQBAJ&oi=fnd&pg=PT2&dq=Kali+Linux+distribution+as+one+of+the+industries+go+to+tool+for+information+gathering.&ots=s78RVeUi49&sig=Cj2WdnnQBDJdt74aOdPik6RJjI8

AWS Whitepapers - Network segmentation and hardening (no date) Amazon. Available at: https://docs.aws.amazon.com/whitepapers/latest/architecting-hipaa-security-and-compliance-on-amazon-eks/network-segmentation-and-hardening.html

Azodolmolky, S., Wieder, P. and Yahyapour, R., 2013. Cloud computing networking: Challenges and opportunities for innovations. IEEE Communications Magazine, 51(7), pp.54-62. Available from: https://ieeexplore.ieee.org/abstract/document/6553678/

Azodolmolky, S., Wieder, P., & Yahyapour, R. (2013). Cloud computing networking: Challenges and opportunities for innovations. IEEE Communications Magazine, 51(7), 54–62. https://doi.org/10.1109/MCOM.2013.6553678

Borky, J.M. and Bradley, T.H. (2018) 'Protecting information with cybersecurity', Effective Model-Based Systems Engineering, pp. 345–404. doi:10.1007/978-3-319-95669-5_10. Availalbe from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7122347/

Cloudflare. (n.d.). What is Zero Trust? Available from: https://www.cloudflare.com/learning/security/glossary/what-is-zero-trust/

CrowdStrike. (n.d.). Identity Segmentation. Retrieved from https://www.crowdstrike.com/cybersecurity-101/identity-segmentation/#:~:text=Gartner%27s%20identity%2Dbased%20segmentation%2C%20on,to%20do%20with%20workforce%20identities.

Cytracom. (n.d.). Micro-Segmentation and Identity-Based Networking. Available from: https://www.cytracom.com/post/micro-segmentation-and-identity-based-networking

Elisity. (n.d.). The Benefits of Identity-Based Microsegmentation for Network Security. Retrieved from https://blog.elisity.com/the-benefits-of-identity-based-microsegmentation-for-network-security

EUROPOL (2023) Internet Organised Crime Assessment (IOCTA) 2023, europol.europa.eu. Available at: https://www.europol.europa.eu/publication-events/main-reports/internet-organised-crime-assessment-iocta-2023#downloads (Accessed: 31 July 2023).

ESRB (2022) Mitigating systemic cyber risk. Available at: https://www.esrb.europa.eu/pub/pdf/reports/esrb.SystemiCyberRisk.220127~b6655fa027.en.pdf (Accessed: 01 August 2023).

Farook, M., Macklin, T., Ahmadinia, A., & Tyagi, S. (2022). Zero Trust Evolution & Transforming Enterprise Security. https://scholarworks.calstate.edu/concern/theses/41687p91q

Fortytwo. (n.d.). What is a PCI Segmentation Test? Available from: https://fortytwo.nl/what-is-a-pci-segmentation-test/](https://fortytwo.nl/what-is-a-pci-segmentation-test/

Francis, J.A.S.J., 2018. Let's Learn Palo Alto NGFW: A Case Study of Checkpoint, Juniper, Cisco, Hacking and Knowing Thyself. Available from https://dl.acm.org/doi/abs/10.5555/3294176

Fuchs, M. and Lemon, J., 2019. Sans 2019 threat hunting survey: The differing needs of new and experienced hunters. SANS Institute Information Reading Room. Available from https://images.g2crowd.com/uploads/attachment/file/123389/SANSreport-A8-20191029.pdf

Gartner. (2021). Three Styles of Identity-Based Segmentation. Available from: https://static1.squarespace.com/static/5eab3148a60be427b8a4a09c/t/613d1fd2bf41e601dbe80c56/1631395795113/2021+Q1+Gartner+Report+Three+Styles+of+Identity+Based+Segementation.pdf

Gilman, E. & Barth, D. (2017) Zero trust networks : building secure systems in untrusted networks. 1st edition. Beijing, [China: O'Reilly. Accessed from https://essex.primo.exlibrisgroup.com/permalink/44UOES_INST/1vvl5tg/alma991008470583707346

IBM Security. (2020). IBM Security: Cost of a Data Breach Report 2020. Retrieved from https://www.ibm.com/security/digital-assets/cost-data-breach-report/\#/

Jeuk, S., Salgueiro, G., Baker, F. and Zhou, S., 2015, October. Network segmentation in the cloud a novel architecture based on UCC and IID. In 2015 IEEE 4th International Conference on Cloud Networking (CloudNet) (pp. 58-63). IEEE. Available from: https://ieeexplore.ieee.org/abstract/document/7335280/

Klein, D., 2019. Micro-segmentation: securing complex cloud environments. Network Security, 2019(3), pp.6-10. Available from https://www.magonlinelibrary.com/doi/abs/10.1016/S1353-4858%2819%2930034-0

McKay, K.A. and Cooper, D.A. (2019) 'Guidelines for the selection, configuration, and use of Transport Layer Security (TLS) implementations', Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations [Preprint]. doi:10.6028/nist.sp.800-52r2. Available from: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r2.pdf

MedCity News. (2022, June). How Identity Segmentation Can Reduce the Attack Surface for Healthcare Organizations. Available from: https://medcitynews.com/2022/06/how-identity-segmentation-can-reduce-the-attack-surface-for-healthcare-organizations/

Mehdizadeha, A., Suinggia, K., Mohammadpoorb, M. and Haruna, H., 2017, December. Virtual Local Area Network (VLAN): Segmentation and Security. In The Third International Conference on Computing Technology and Information Management (ICCTIM2017) (Vol. 78, p. 89). Available at: https://www.researchgate.net/profile/Natalie-Walker-15/publication/322077322_Proceedings_of_the_Third_International_Conference_on_Computing_Technology_and_Information_Management_ICCTIM2017_Thessaloniki_Greece_2017/links/5a4369dda6fdcce19716a967/Proceedings-of-the-Third-International-Conference-on-Computing-Technology-and-Information-Management-ICCTIM2017-Thessaloniki-Greece-2017.pdf#page=80

Mhaskar, N., Alabbad, M., & Khedri, R. (2021). A Formal Approach to Network Segmentation. Computers & Security, 103, 102162. https://doi.org/10.1016/J.COSE.2020.102162

Mujib, M., & Sari, R. F. (2020). Performance Evaluation of Data Center Network with Network Micro-segmentation. ICITEE 2020 - Proceedings of the 12th International

Conference on Information Technology and Electrical Engineering, 27–32. https://doi.org/10.1109/ICITEE49829.2020.9271749

Mushtaq, M. F., Akram, U., Khan, I., Khan, S. N., Shahzad, A., & Ullah, A. (2017). Cloud Computing Environment and Security Challenges: A Review. International Journal of Advanced Computer Science and Applications, 8(10). Available from: https://www.researchgate.net/profile/Muhammad-Mushtaq-20/publication/320802850_Cloud_Computing_Environment_and_Security_Challenges_A_Review/links/59fc20da458515d07062864c/Cloud-Computing-Environment-and-Security-Challenges-A-Review.pdf

Mushtaq, M. F., Akram, U., Khan, I., Khan, S. N., Shahzad, A., & Ullah, A. (2017). Cloud Computing Environment and Security Challenges: A Review. International Journal of

National Institute of Standards and Technology (NIST). (2020). NIST Special Publication 800-53: Security and Privacy Controls for Information Systems and Organizations. Retrieved from: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf

National Institute of Standards and Technology. (2020). Zero Trust Architecture. Available from: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf

Palo Alto Networks. (n.d.). What is a Zero Trust Architecture? Available from: https://www.paloaltonetworks.com/cyberpedia/what-is-a-zero-trust-architecture

Palo Alto Networks. (n.d.). What is Microsegmentation? Available from: https://www.paloaltonetworks.com/cyberpedia/what-is-microsegmentation

PricewaterhouseCoopers. (2019). Safeguarding businesses in a digital world. Retrieved from https://www.pwc.com/gx/en/issues/cyber-security/information-security-survey.html

Prakash, C., & Dasgupta, S. (2016). Cloud computing security analysis: Challenges and possible solutions. International Conference on Electrical, Electronics, and Optimization Techniques, ICEEOT 2016, 54–57. https://doi.org/10.1109/ICEEOT.2016.7755626

Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero Trust Architecture. NIST Special Publication 800-207. https://doi.org/10.6028/NIST.SP.800-207

Roth, J.D., Lutton, C.E. and Michael, J.B., 2019. Security Through Simplicity: A Case Study in Logical Segmentation Inference. Computer, 52(7), pp.76-79. Available from: https://ieeexplore.ieee.org/abstract/document/8747213/

Sarkar, S., Choudhary, G., Shandilya, S.K., Hussain, A. and Kim, H., 2022. Security of zero trust networks in cloud computing: A comparative review. Sustainability, 14(18), p.11213. Available from: https://www.mdpi.com/2071-1050/14/18/11213

Secci, S. and Murugesan, S., 2014. Cloud networks: Enhancing performance and resiliency. Computer, 47(10), pp.82-85. Available from https://www.computer.org/csdl/magazine/co/2014/10/mco2014100082/13rRUwcS1y7

Sheikh, N., Pawar, M. and Lawrence, V., 2021, May. Zero trust using network micro segmentation. In IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) (pp. 1-6). IEEE. Available from: https://ieeexplore.ieee.org/abstract/document/9484645

Syed, N. F., Shah, S. W., Shaghaghi, A., Anwar, A., Baig, Z., & Doss, R. (2022). Zero Trust Architecture (ZTA): A Comprehensive Survey. IEEE Access, 10, 57143–57179. https://doi.org/10.1109/ACCESS.2022.3174679

Simpson, W.R. and Foltz, K.E., 2021. Network Segmentation and Zero Trust Architectures. In Lecture Notes in Engineering and Computer Science, Proceedings of the World Congress on Engineering (WCE) (pp. 201-206). Available from http://www.iaeng.org/publication/WCE2021/WCE2021_pp201-206.pdf

Symantec (2022) The threat landscape in 2021, Symantec Enterprise Blogs. Available at: https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/threat-landscape-2021 (Accessed: 29 July 2023).

VMware. (n.d.). Micro-segmentation. Available from https://www.vmware.com/topics/glossary/content/micro-segmentation.html#:~:text=Micro%2Dsegmentation%20is%20a%20network,services%20for%20each%20unique%20segment

11. Annex 1- Technical Details

1. Environment Configuration and Implementation:

    a. Definition of the VPC and routing table as per the bellow screenshot:

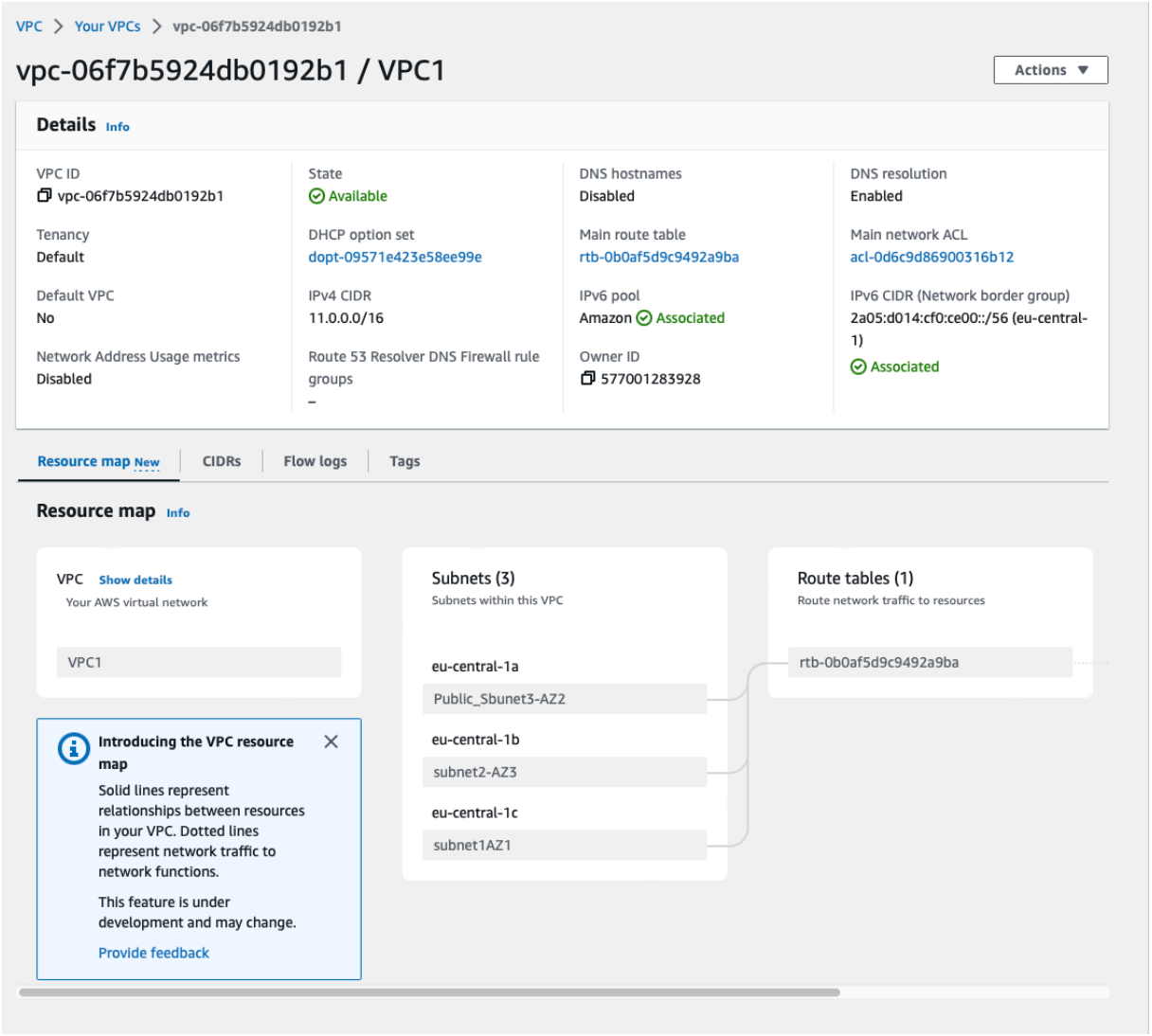
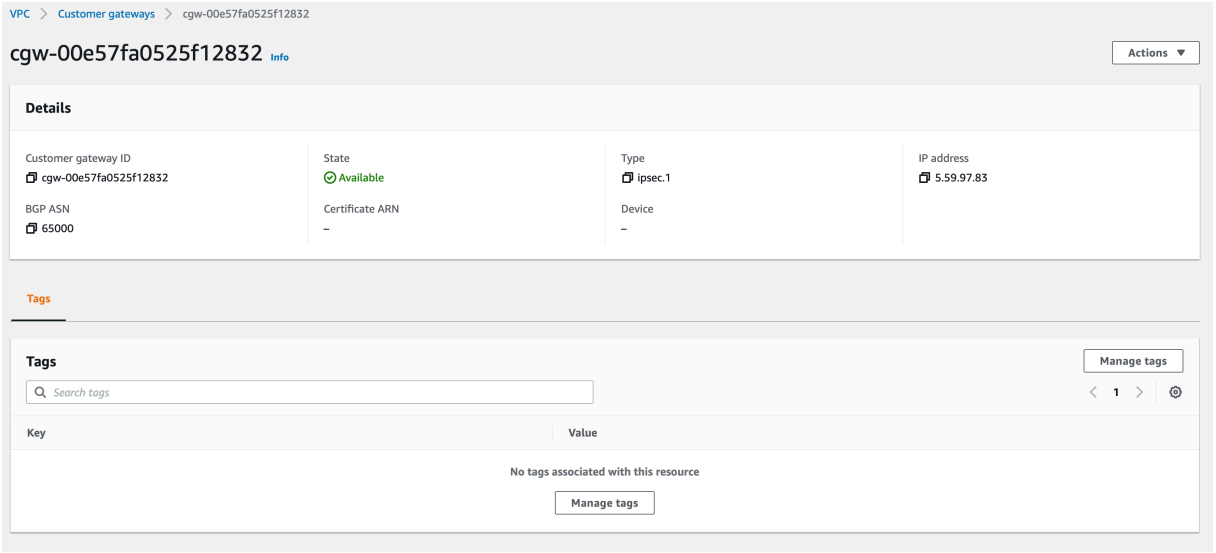


Figure 1. (VPC Configuration and routing table)

Defining the Virtual Private Network (VPN) configuration from the cloud provider environment towards the on-premises data center.

b. Creation of the endpoint VPN on the cloud environment:



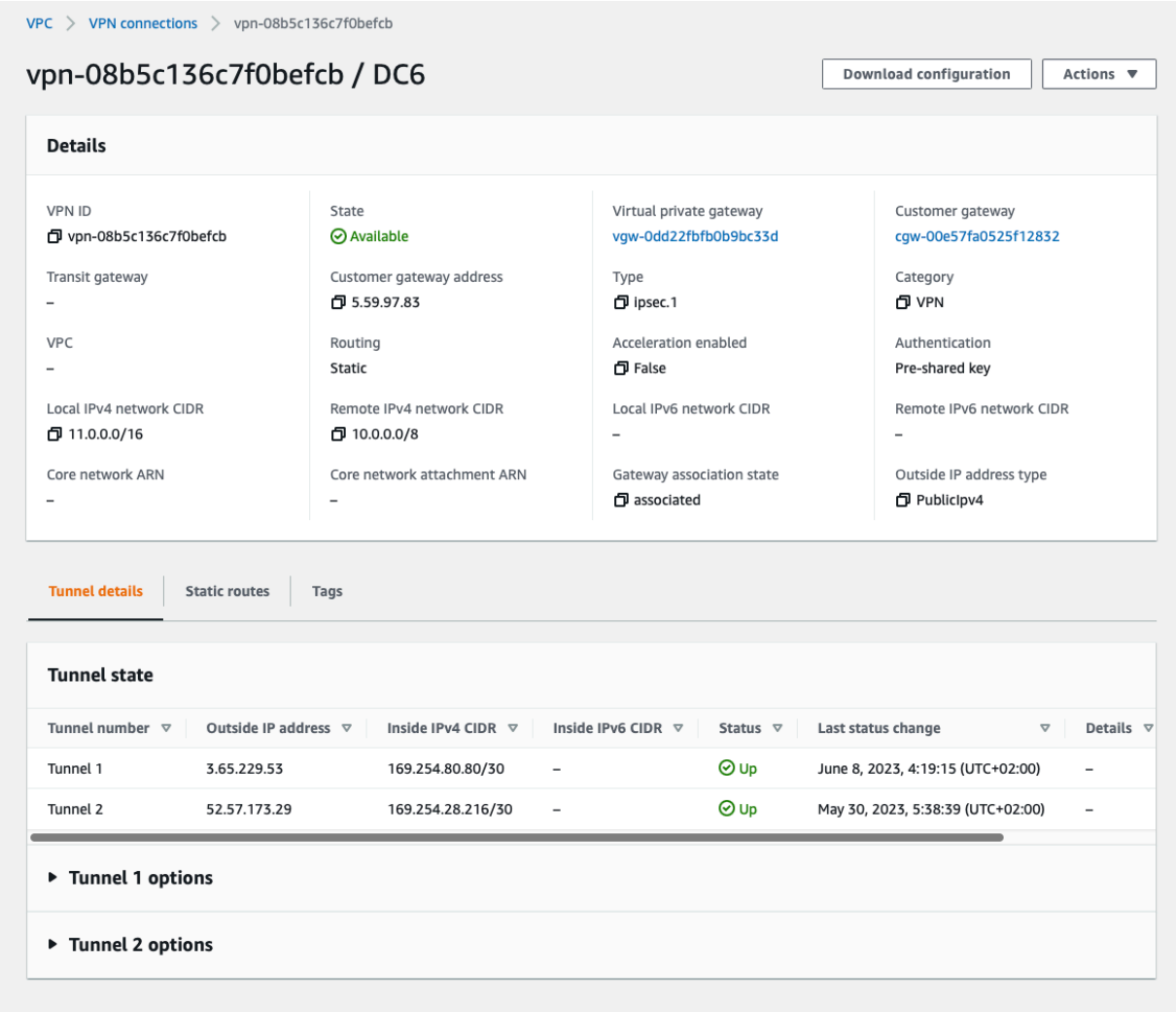
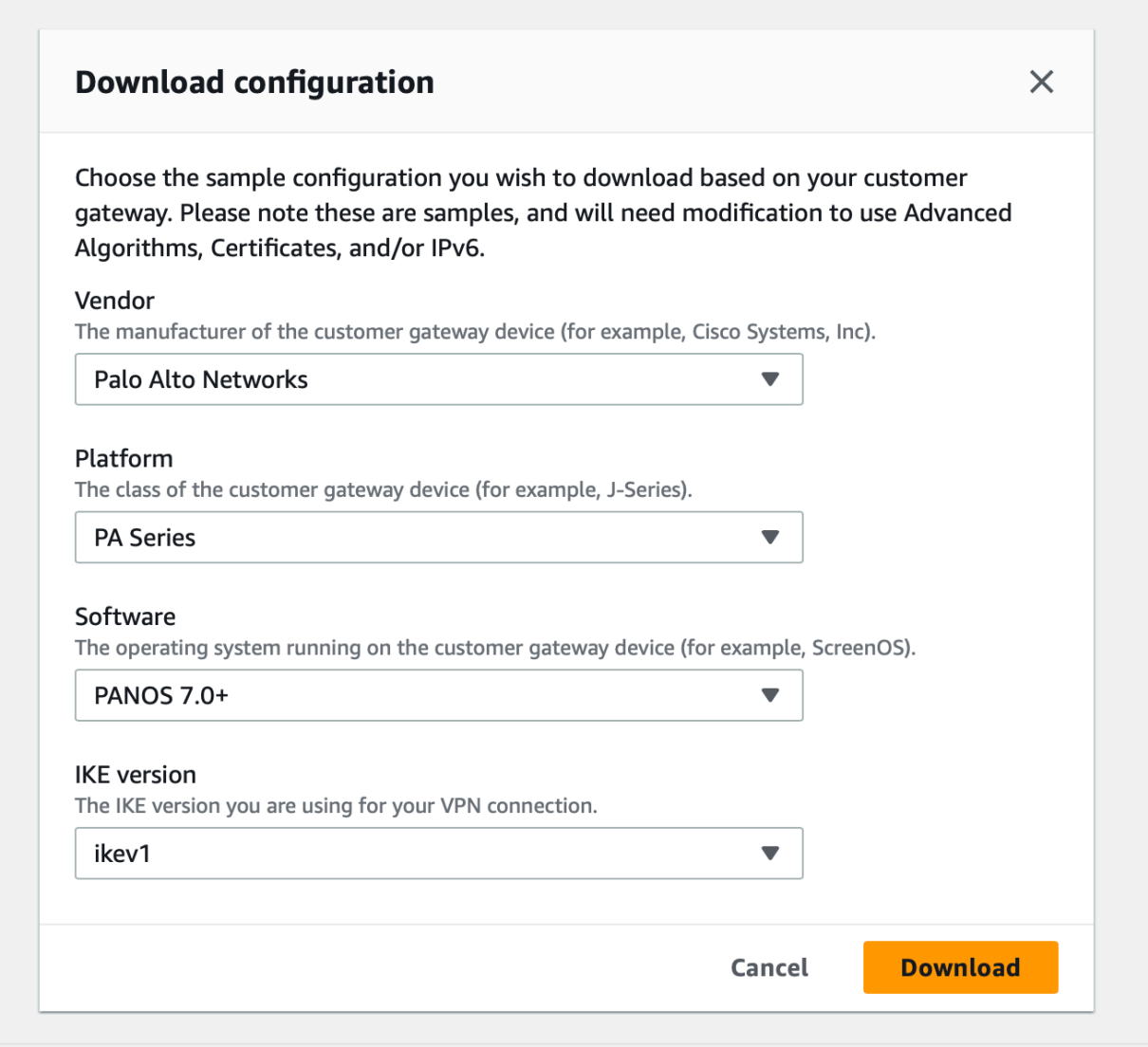


Figure 2. (vpn configuration on the cloud provider)

c. Downloading the configuration for the data center firewall configuration from the cloud environment, which is generated automatically by the cloud provider.



d. Configuration to run on the Palo Alto Virtual Machine hosted in the on-premises data center:

---------- BEGINNING OF CONFIGURATION ----------

! Amazon Web Services

! Virtual Private Cloud

```
! AWS utilizes unique identifiers to manipulate the configuration of
! a VPN Connection. Each VPN Connection is assigned an identifier and is
! associated with two other identifiers, namely the
! Customer Gateway Identifier and Virtual Private Gateway Identifier.
!
! Your VPN Connection ID                  : vpn-08b5c136c7f0befcb
! Your Virtual Private Gateway ID         : vgw-0dd22fbfb0b9bc33d
! Your Customer Gateway ID                : cgw-00e57fa0525f12832
!
!
! This configuration consists of two tunnels. Both tunnels must be
! configured on your Customer Gateway.
!
! ------------------------------------------------------------------------
! IPSec Tunnel #1
! ------------------------------------------------------------------------
! #1: Internet Key Exchange (IKE) Configuration
!
! A policy is established for the supported ISAKMP encryption,
! authentication, Diffie-Hellman, lifetime, and key parameters.
! Please note, these sample configurations are for the minimum requirement of AES128, SHA1, and DH Group 2.
! Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.
! You will need to modify these sample configuration files to take advantage of AES256, SHA256, or other DH groups like
2, 14-18, 22, 23, and 24.
```

```
! NOTE: If you customized tunnel options when creating or modifying your VPN connection, you may need to modify these sample configurations to match the custom settings for your tunnels.

!

! Higher parameters are only available for VPNs of category "VPN," and not for "VPN-Classic".

! The address of the external interface for your customer gateway must be a static address.

! Your customer gateway may reside behind a device performing network address translation (NAT).

! To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall !rules to unblock UDP port 4500.

! If not behind NAT, and you are not using an Accelerated VPN, we recommend disabling NAT-T. If you are using an Accelerated VPN, make sure that NAT-T is enabled.

!

 configure

 edit network ike crypto-profiles ike-crypto-profiles vpn-08b5c136c7f0befcb-0

  set dh-group group2

  set hash sha1

  set lifetime seconds  28800

  set encryption aes-128-cbc

  top


! With local-address IP please append the configured subnet mask (i.e, /30) on the VPN initiating interface (i.e., ethernet 1/1)

! For example if you have /30 as subnet mask the local-address ip should be 5.59.97.83/30


edit network ike gateway ike-vpn-08b5c136c7f0befcb-0

set protocol version ikev1

set protocol ikev1 ike-crypto-profile vpn-08b5c136c7f0befcb-0 exchange-mode main

set protocol ikev1 dpd enable yes interval 10 retry 3
```

```
set authentication pre-shared-key key 6c2EALT2fWmcOf4n_dp2Dy7fxl99o7yu

set protocol-common nat-traversal enable yes/no

set protocol-common fragmentation enable yes/no

set local-address ip 5.59.97.83

 set local-address interface ethernet1/1

 set peer-address ip 3.65.229.53

 top

! #2: IPSec Configuration

!

! The IPSec transform set defines the encryption, authentication, and IPSec

! mode parameters.

!

! Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.

! Please note, you may use these additionally supported IPSec parameters for encryption like AES256 and other DH groups

like 2, 5, 14-18, 22, 23, and 24.

! Higher parameters are only available for VPNs of category "VPN," and not for "VPN-Classic".

!

 edit network ike crypto-profiles ipsec-crypto-profiles ipsec-vpn-08b5c136c7f0befcb-0

  set esp authentication sha1

  set esp encryption aes-128-cbc

  set dh-group group2

  set lifetime seconds 3600

 top
```

```
! ------------------------------------------------------------------------
! #3: Tunnel Interface Configuration
!
! A tunnel interface is configured to be the logical interface associated
! with the tunnel. All traffic routed to the tunnel interface will be
! encrypted and transmitted to the VPC. Similarly, traffic from the VPC
! will be logically received on this interface.
!
! Association with the IPSec security association is done through the
! "tunnel protection" command.
!
! The address of the interface is configured with the setup for your
! Customer Gateway. If the address changes, the Customer Gateway and VPN
! Connection must be recreated with Amazon VPC.
!



 edit network interface tunnel units tunnel.1
  set ip 169.254.80.82/30
  set mtu 1427
 top


!
! Tunnel interface needs to be associated to Zone, we are using untrust zone as an example, please adjust according
```

```
!


set zone untrust network layer3 tunnel.1


!
! Tunnel interface needs to be associated to a virtual router, we are using default as an example, please adjust accordingly
!


set network virtual-router default interface tunnel.1


 edit network tunnel ipsec ipsec-tunnel-1
  set auto-key ipsec-crypto-profile ipsec-vpn-08b5c136c7f0befcb-0
  set auto-key ike-gateway ike-vpn-08b5c136c7f0befcb-0
  set tunnel-interface tunnel.1
  set anti-replay yes
 top


! ------------------------------------------------------------------------------
! #4 Static Route Configuration
!
! Your Customer Gateway needs to set a static route for the prefix corresponding to your
! VPC to send traffic over the tunnel interface.
!
! Static routing does not allow for failover of traffic between tunnels. If there is a problem with one of the
! tunnels, we would want to failover the traffic to the second tunnel. This is done by creating a tunnel monitor
```

```
! profile in Palo Alto networks device. This profile pings the other end of the tunnel, and check if the tunnel is up.
! If ping fails, it will remove the policy-based static route from the routing table, and the second route in the table will
! become active.

! You need to set the interval and Threshold as a part of the profile. Interval is number of seconds
! between pings. Threshold is the number of lost consecutive pings. Using the respective values of 2 and 5, your tunnel
! will failover in 10 seconds.
! The following command shows how to set up a profile named 'tunnelmonitor'.
edit network profiles monitor-profile tunnelmonitor
  set interval 2 threshold 5 action fail-over
 top

! LAN-CIDR is an object which contains your Local LAN IP addresses.
! VPC-CIDR is an object which contains your VPC CIDR addresses.
! If your VPC-CIDR is 10.0.0.0/16, you can configure an object using the following:
!
! set address VPC-CIDR ip-netmask 10.0.0.0/16
! set address LAN-CIDR ip-netmask 192.168.0.0/16
!

! To allow for failover between tunnels, we use policy based routing. We bind the tunnelmonitor profile
! to this policy. When the tunnelmonitor reaches its threshold, the policy is removed , and the backup
! policy becomes active, please adjust from zone/interface accordingly.

 edit rulebase pbf rules pbf-vpn-vpn-08b5c136c7f0befcb-0
```

```
 set action forward nexthop ip-address 169.254.80.81
 set action forward egress-interface tunnel.1
 set action forward monitor profile tunnelmonitor disable-if-unreachable yes ip-address 169.254.80.81
 set source LAN-CIDR source-user any destination VPC-CIDR application any service any
 set from zone trust
 set disabled no
top

! Please note that using above PBF based static route configuration, you can't ping
! (by specifying source) from the CGW via VPN tunnel and you would need a LAN side
! resource to test VPN connectivity.

! ----------------------------------------------------------------------------
! IPSec Tunnel #2
! ----------------------------------------------------------------------------
! #1: Internet Key Exchange (IKE) Configuration
!
! A policy is established for the supported ISAKMP encryption,
! authentication, Diffie-Hellman, lifetime, and key parameters.
! Please note, these sample configurations are for the minimum requirement of AES128, SHA1, and DH Group 2.
! Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.
! You will need to modify these sample configuration files to take advantage of AES256, SHA256, or other DH groups like
2, 14-18, 22, 23, and 24.
! NOTE: If you customized tunnel options when creating or modifying your VPN connection, you may need to modify these
sample configurations to match the custom settings for your tunnels.
```

!

! Higher parameters are only available for VPNs of category "VPN," and not for "VPN-Classic".

! The address of the external interface for your customer gateway must be a static address.

! Your customer gateway may reside behind a device performing network address translation (NAT).

! To ensure that NAT traversal (NAT-T) can function, you must adjust your firewall !rules to unblock UDP port 4500.

! If not behind NAT, and you are not using an Accelerated VPN, we recommend disabling NAT-T. If you are using an Accelerated VPN, make sure that NAT-T is enabled.

!

configure

edit network ike crypto-profiles ike-crypto-profiles vpn-08b5c136c7f0befcb-1

set dh-group group2

set hash sha1

set lifetime seconds 28800

set encryption aes-128-cbc

top

! With local-address IP please append the configured subnet mask (i.e, /30) on the VPN initiating interface (i.e., ethernet 1/1)

! For example if you have /30 as subnet mask the local-address ip should be 5.59.97.83/30

edit network ike gateway ike-vpn-08b5c136c7f0befcb-1

set protocol version ikev1

set protocol ikev1 ike-crypto-profile vpn-08b5c136c7f0befcb-1 exchange-mode main

set protocol ikev1 dpd enable yes interval 10 retry 3

set authentication pre-shared-key key HxSoVp3pYd0wnmqfS4_oS9Zkn0HZeW3k

set protocol-common nat-traversal enable yes/no

```
set protocol-common fragmentation enable yes/no

set local-address ip 5.59.97.83

 set local-address interface ethernet1/1

 set peer-address ip 52.57.173.29

 top



! #2: IPSec Configuration

!

! The IPSec transform set defines the encryption, authentication, and IPSec

! mode parameters.

!

! Category "VPN" connections in the GovCloud region have a minimum requirement of AES128, SHA2, and DH Group 14.

! Please note, you may use these additionally supported IPSec parameters for encryption like AES256 and other DH groups

like 2, 5, 14-18, 22, 23, and 24.

! Higher parameters are only available for VPNs of category "VPN," and not for "VPN-Classic".

!

 edit network ike crypto-profiles ipsec-crypto-profiles ipsec-vpn-08b5c136c7f0befcb-1

  set esp authentication sha1

  set esp encryption aes-128-cbc

  set dh-group group2

  set lifetime seconds 3600

 top

! ------------------------------------------------------------------------------

! #3: Tunnel Interface Configuration
```

```
!
! A tunnel interface is configured to be the logical interface associated
! with the tunnel. All traffic routed to the tunnel interface will be
! encrypted and transmitted to the VPC. Similarly, traffic from the VPC
! will be logically received on this interface.
!
! Association with the IPSec security association is done through the
! "tunnel protection" command.
!
! The address of the interface is configured with the setup for your
! Customer Gateway. If the address changes, the Customer Gateway and VPN
! Connection must be recreated with Amazon VPC.
!
 edit network interface tunnel units tunnel.2
  set ip 169.254.28.218/30
  set mtu 1427
 top
!
! Tunnel interface needs to be associated to Zone, we are using untrust zone as an example, please adjust according
!
set zone untrust network layer3 tunnel.2
!
! Tunnel interface needs to be associated to a virtual router, we are using default as an example, please adjust accordingly
!
```

```
set network virtual-router default interface tunnel.2


 edit network tunnel ipsec ipsec-tunnel-2
  set auto-key ipsec-crypto-profile ipsec-vpn-08b5c136c7f0befcb-1
  set auto-key ike-gateway ike-vpn-08b5c136c7f0befcb-1
  set tunnel-interface tunnel.2
  set anti-replay yes
 top


! ------------------------------------------------------------------------------
! #4 Static Route Configuration
!
! Your Customer Gateway needs to set a static route for the prefix corresponding to your
! VPC to send traffic over the tunnel interface.
!
! Static routing does not allow for failover of traffic between tunnels. If there is a problem with one of the
! tunnels, we would want to failover the traffic to the second tunnel. This is done by creating a tunnel monitor
! profile in Palo Alto networks device. This profile pings the other end of the tunnel, and check if the tunnel is up.
! If ping fails, it will remove the policy-based static route from the routing table, and the second route in the table will
! become active.


! You need to set the interval and Threshold as a part of the profile. Interval is number of seconds
! between pings. Threshold is the number of lost consecutive pings. Using the respective values of 2 and 5, your tunnel
! will failover in 10 seconds.
! The following command shows how to set up a profile named 'tunnelmonitor'.
```

```
edit network profiles monitor-profile tunnelmonitor
  set interval 2 threshold 5 action fail-over
 top

! LAN-CIDR is an object which contains your Local LAN IP addresses.
! VPC-CIDR is an object which contains your VPC CIDR addresses.
! If your VPC-CIDR is 10.0.0.0/16, you can configure an object using the following:
!
! set address VPC-CIDR ip-netmask 10.0.0.0/16
! set address LAN-CIDR ip-netmask 192.168.0.0/16
!

! To allow for failover between tunnels, we use policy based routing. We bind the tunnelmonitor profile
! to this policy. When the tunnelmonitor reaches its threshold, the policy is removed , and the backup
! policy becomes active, please adjust from zone/interface accordingly.

 edit rulebase pbf rules pbf-vpn-vpn-08b5c136c7f0befcb-1
  set action forward nexthop ip-address 169.254.28.217
  set action forward egress-interface tunnel.2
  set action forward monitor profile tunnelmonitor disable-if-unreachable yes ip-address 169.254.28.217
  set source LAN-CIDR source-user any destination VPC-CIDR application any service any
  set from zone trust
  set disabled no
 top
```

```
! Please note that using above PBF based static route configuration, you can't ping
! (by specifying source) from the CGW via VPN tunnel and you would need a LAN side
! resource to test VPN connectivity.

! If tunnel and LAN side network interfaces are in different security zones,
! we need to configure NAT examption and put at the top, so that actual IP sources
! show up on the VPC side for proper route back via tunnel as follows when
! LAN side zone is considered as "trust" and tunnel interface being part of "untrust" zone,
! please change accordingly:
!

edit rulebase nat
  set rules No_NAT_LAN_VPC to untrust
  set rules No_NAT_LAN_VPC from trust
  set rules No_NAT_LAN_VPC source LAN-CIDR
  set rules No_NAT_LAN_VPC destination VPC-CIDR
  set rules No_NAT_LAN_VPC service any
  set rules No_NAT_LAN_VPC disabled no
 top
 move rulebase nat rules No_NAT_LAN_VPC top

! *** NOTE *** :
! If tunnel and LAN side network interfaces are in the different security zones,
! we need to configure a firewall policy to allow inter-zone communication as well.
```

! You can use VPC-CIDR/LAN-CIDR object groups to create firewall policy as well.

!

! Please also note that you will need to commit the configuration using "commit" command.

! Additional Notes and Questions

! - Amazon Virtual Private Cloud Getting Started Guide:

! http://docs.amazonwebservices.com/AmazonVPC/latest/GettingStartedGuide

! - Amazon Virtual Private Cloud Network Administrator Guide:

! http://docs.amazonwebservices.com/AmazonVPC/latest/NetworkAdminGuide

---------- END OF CONFIGURATION ----------

e. Defining security groups to be used for the cloud instances, deploying VPN configuration in the data center Palo Alto Firewall as well as defining the based network access control list needed for basic functionality and scanning.

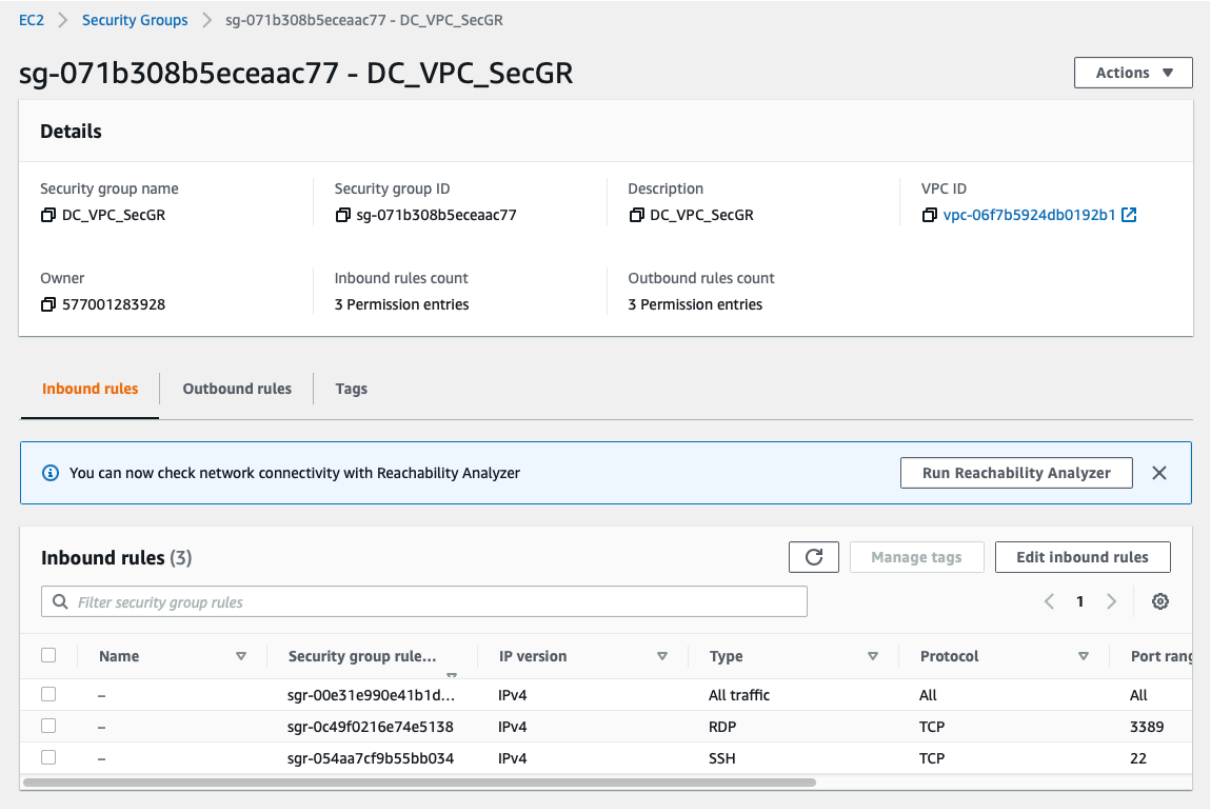


Figure 3. (Security group used on the instances in the cloud)

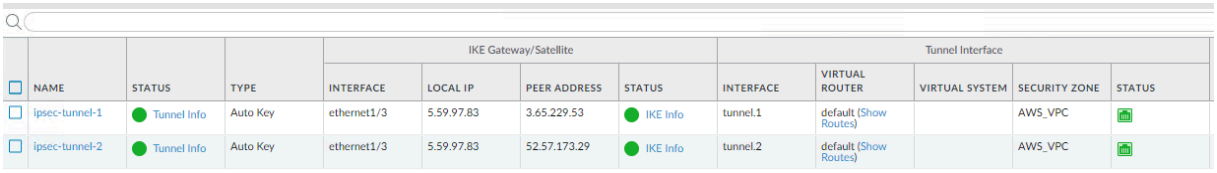


Figure 4. (VPN tunnel configuration and status from data center)

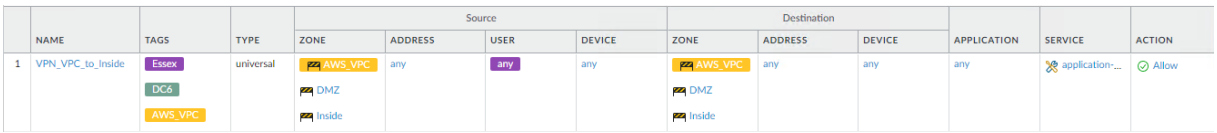


Figure 5. (Firewall rule policies in the data center)

f. In order to deploy the cloud instances, we navigate on the cloud environment and select the EC2 service select Launch instance as bellow:

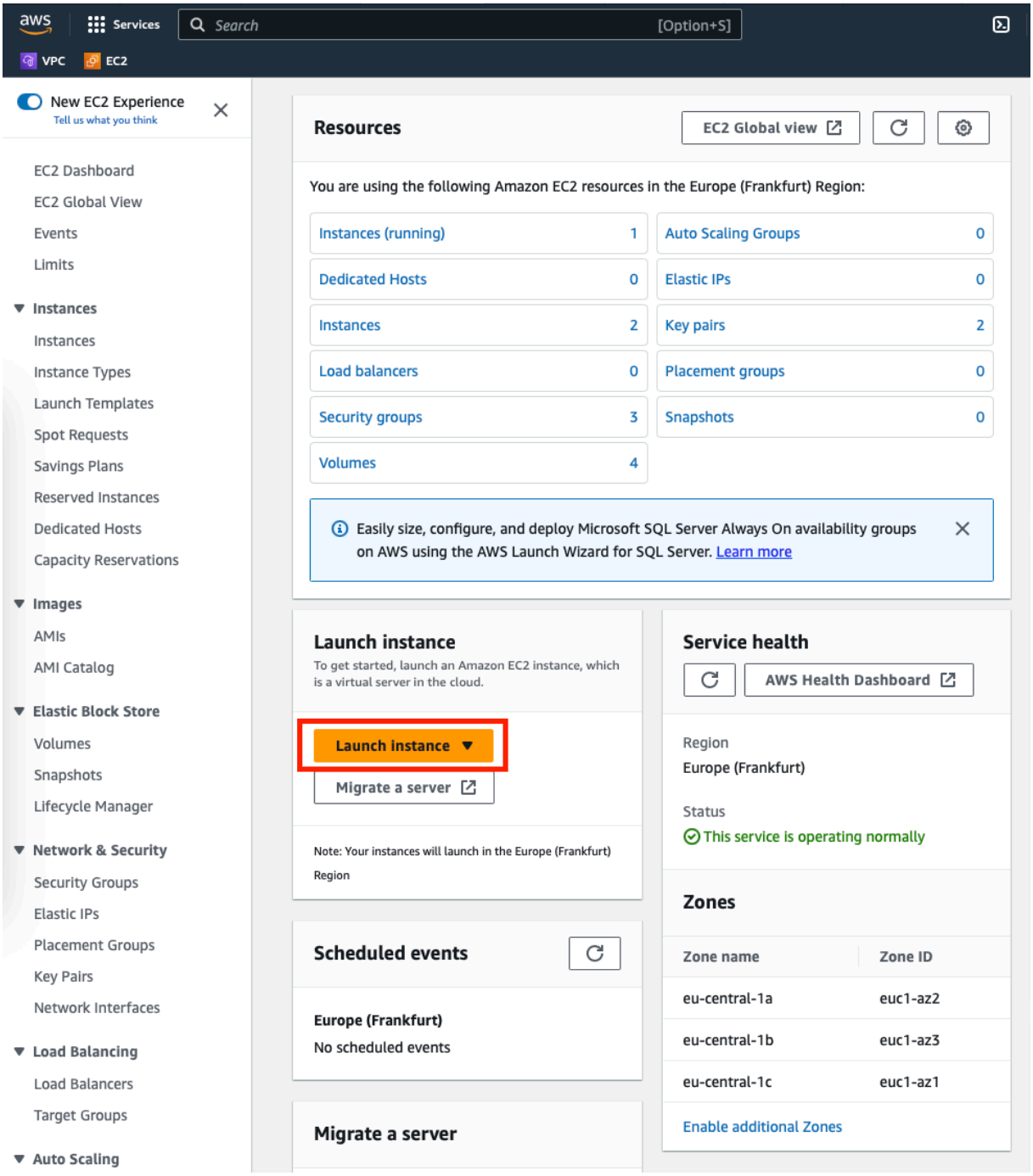


Figure 6. (launching a new instance)

Name your instances and select the number of hosts you want to deploy and at the same time select the operating system for the virtual machine:

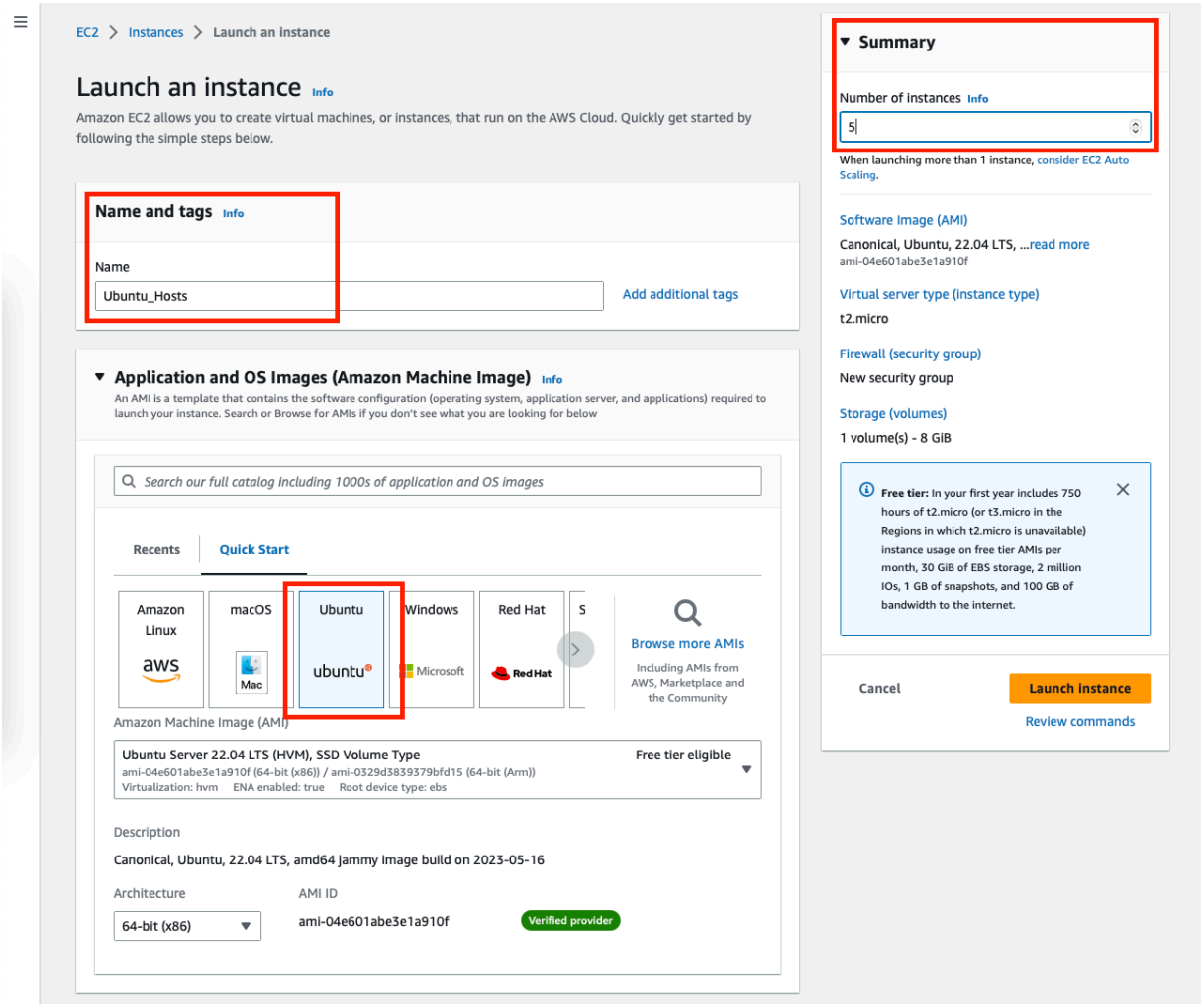


Figure 7. (selecting the number of instances, naming them and selecting os)

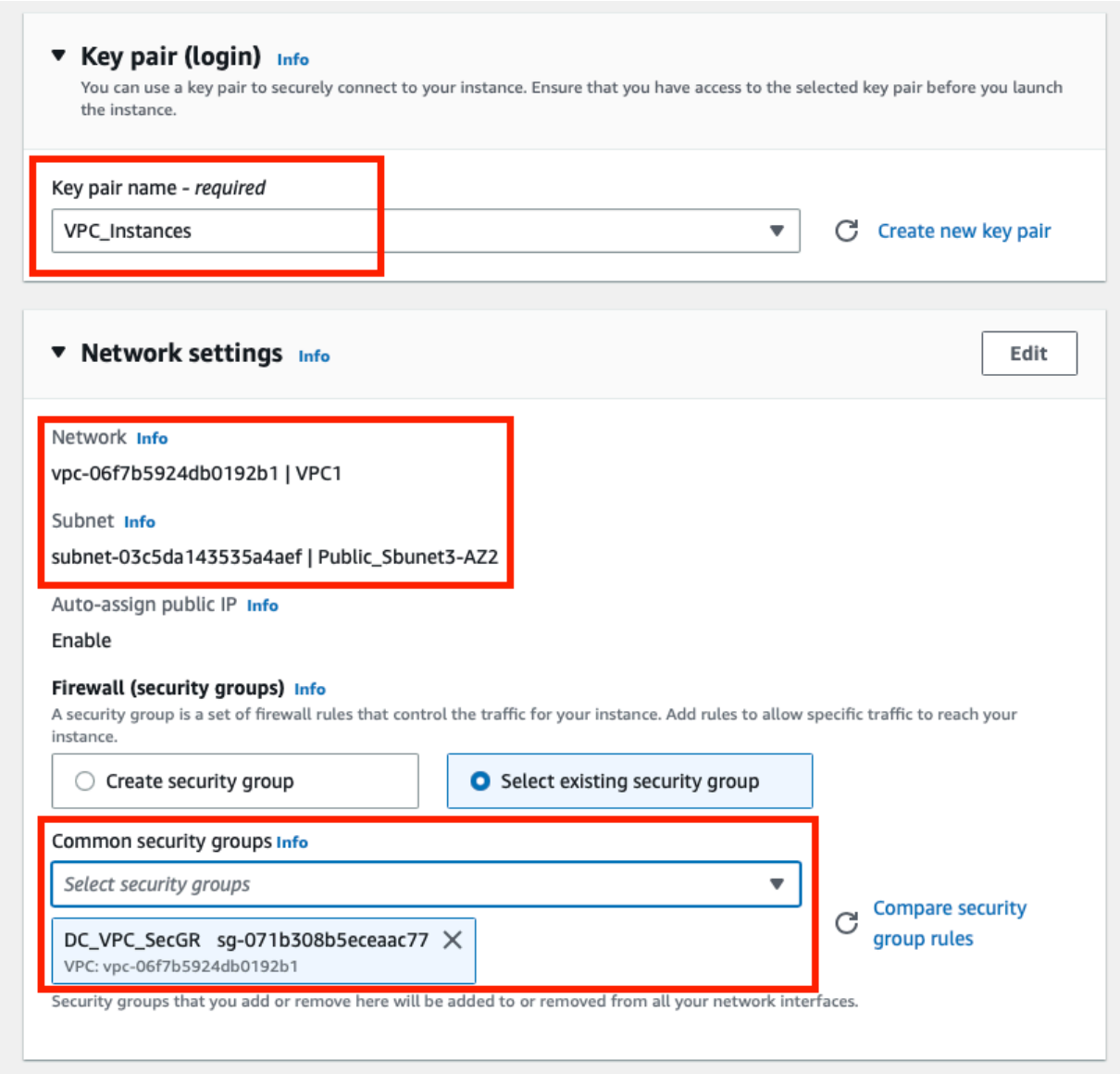


Figure 8. (select key pairs for facilitating Connection to the hosts, select your appropriate VPC and subnet, as well as proper security group)

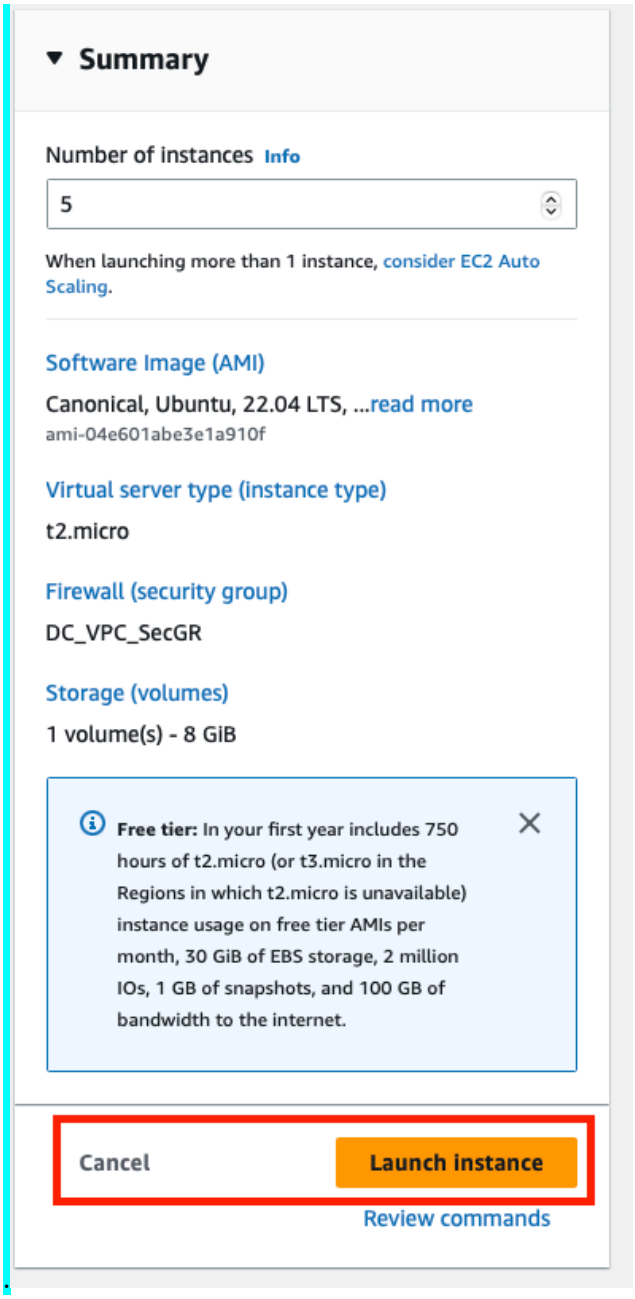


Figure 9. (Finalizing the deployments)

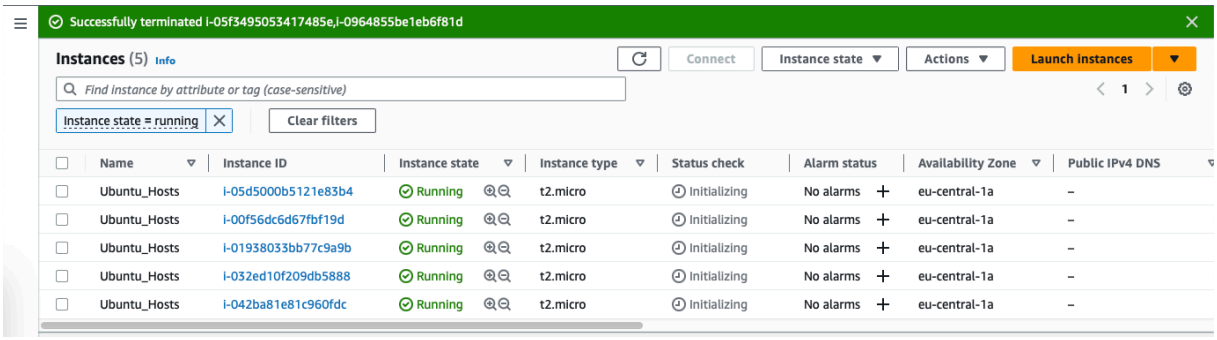


Figure 10. (instances are deployed)

g. Verifying that the host is listening on the incoming connections for the defined ports.

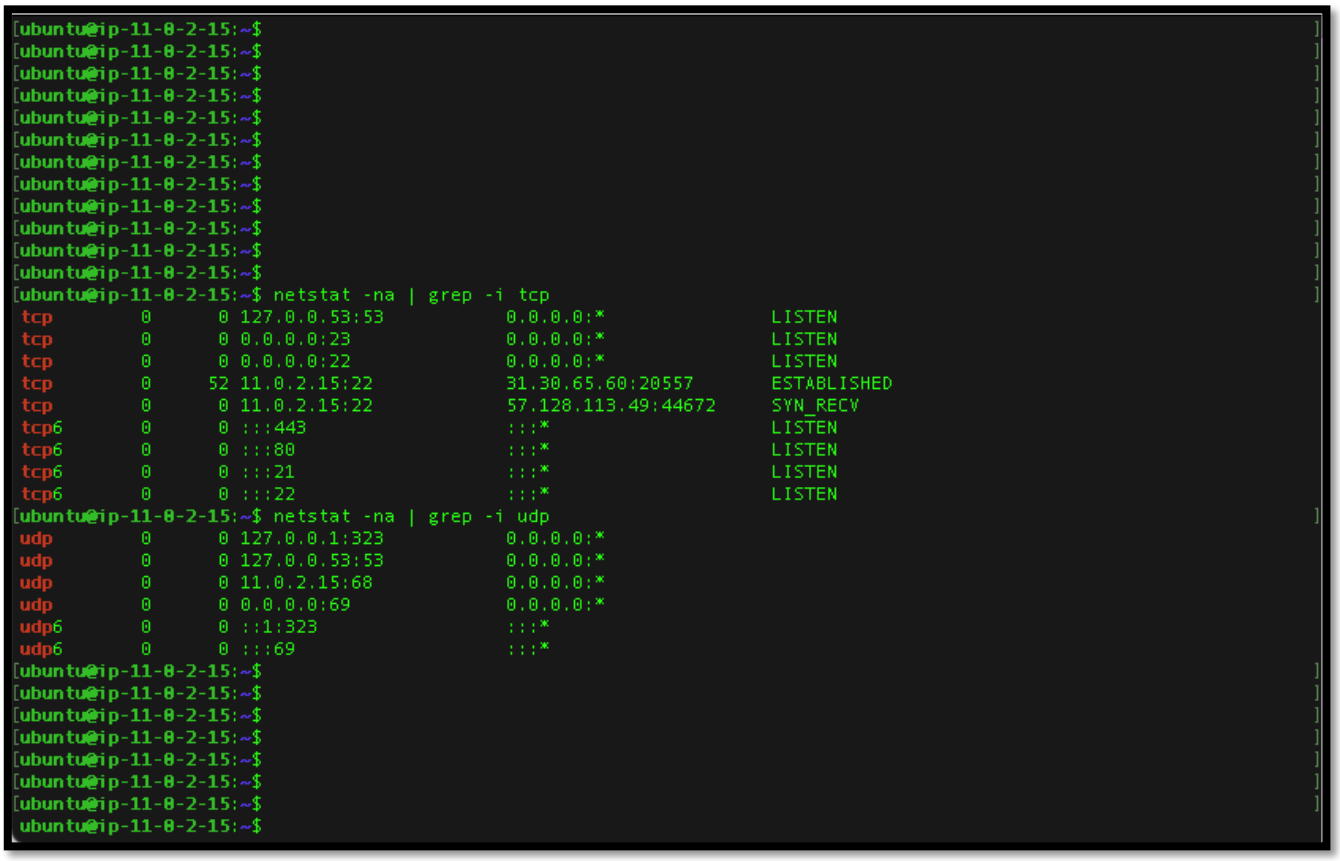
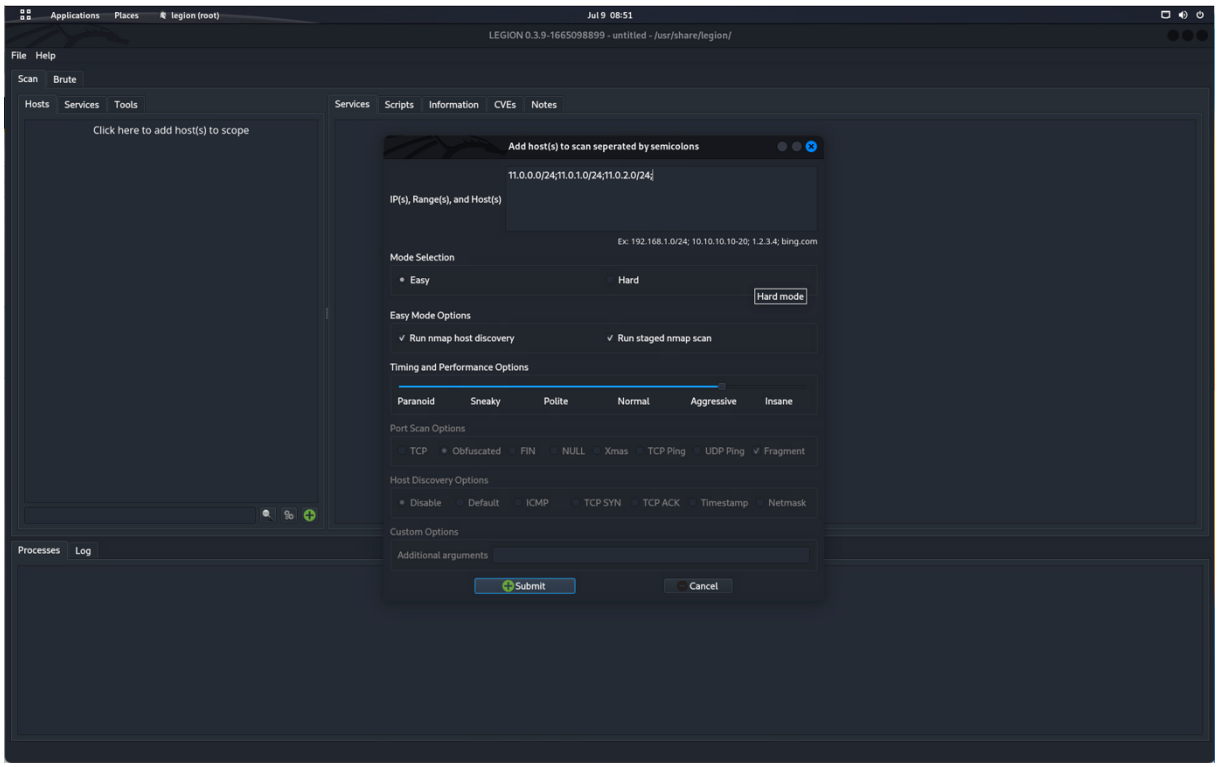


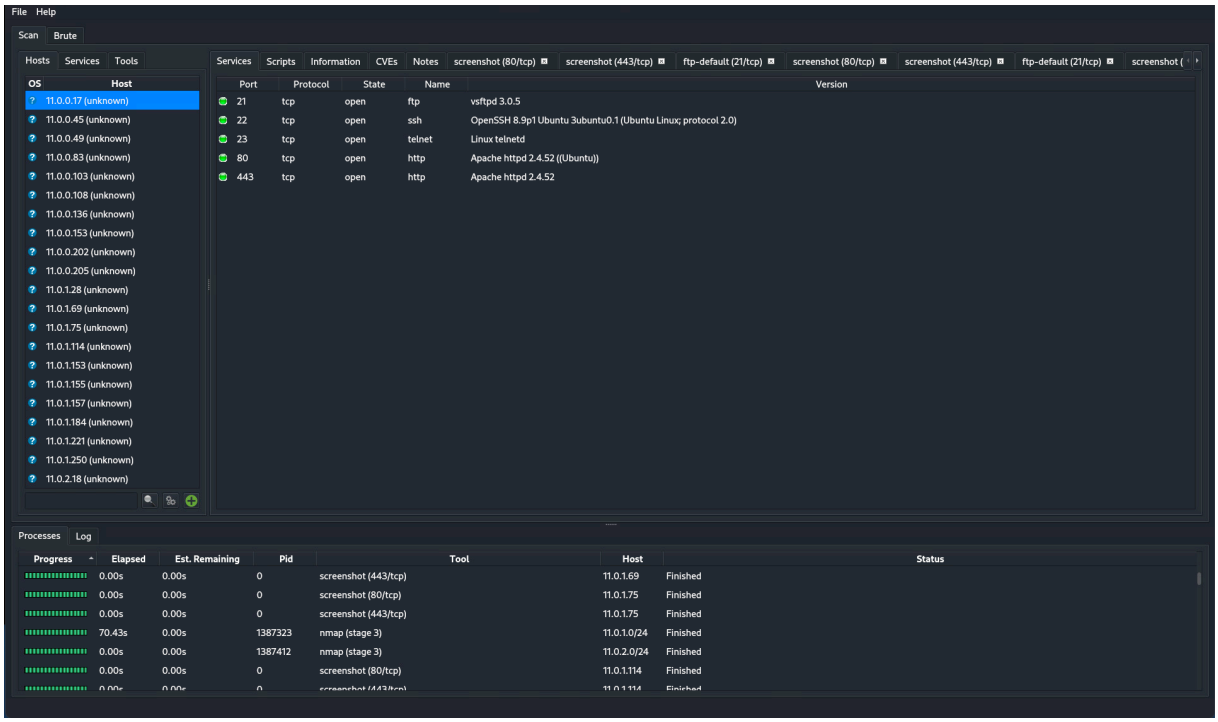
Figure 11 (hosts configure for incoming connections)

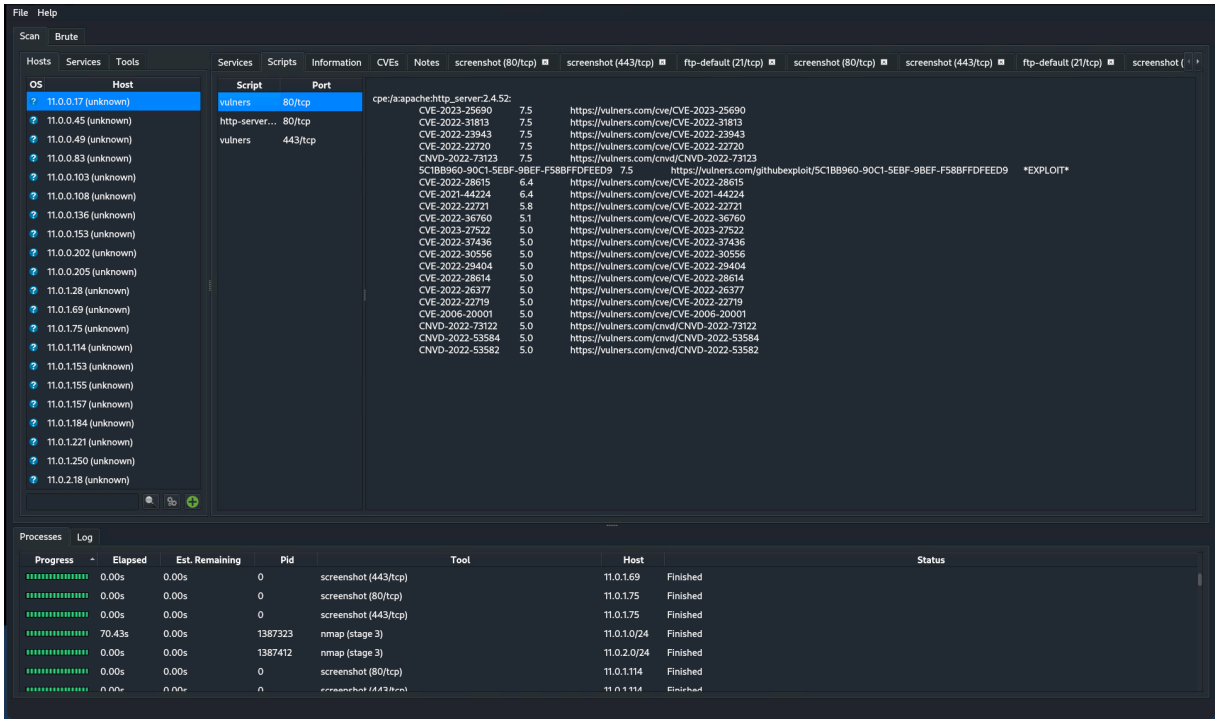
2. Scanning results outcome:

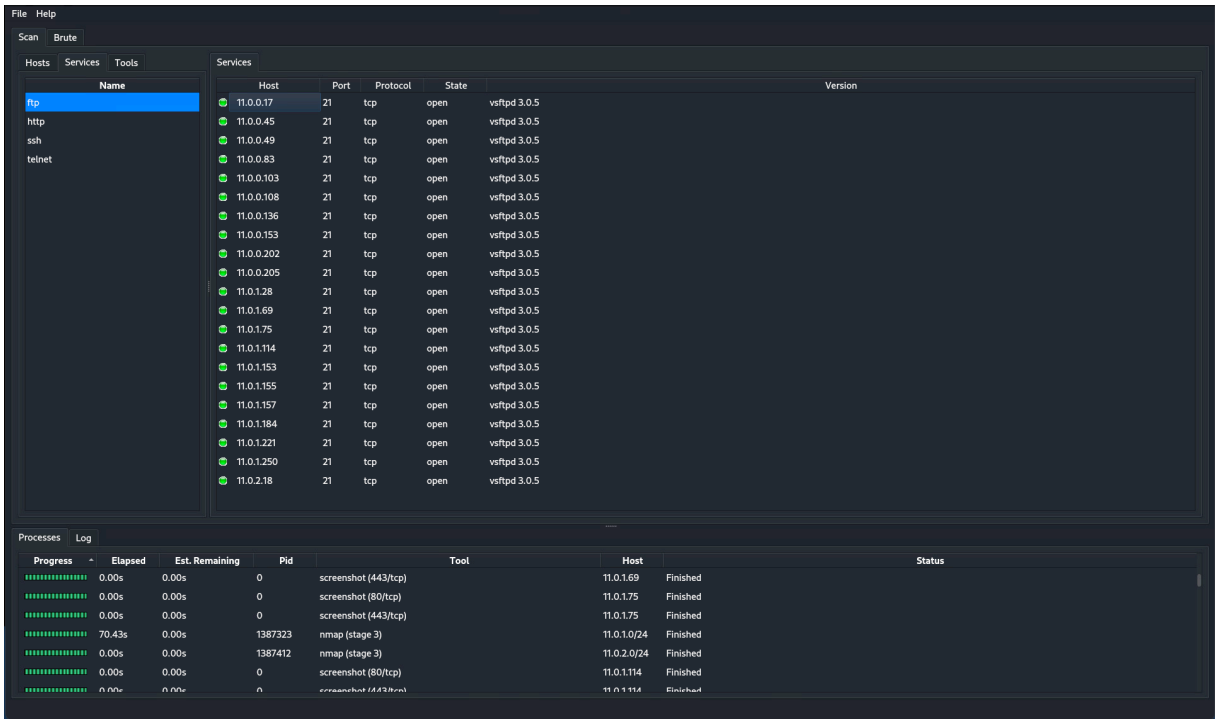
    a. Base Scanning:

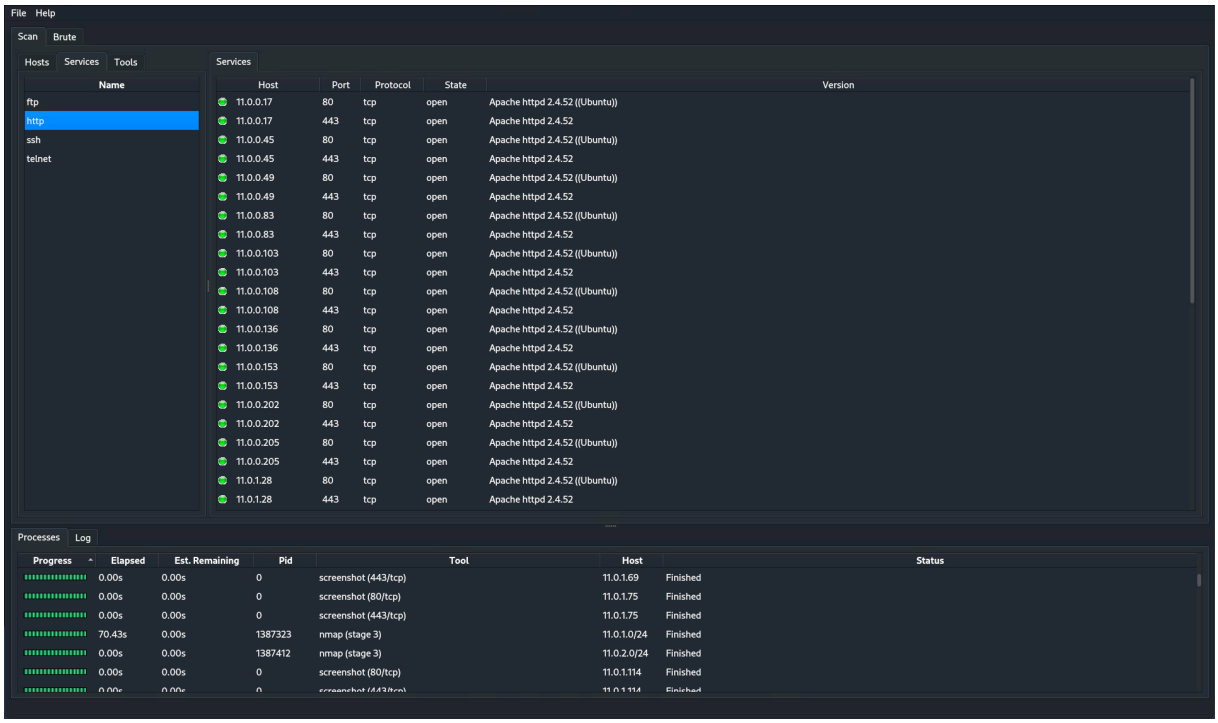
Legion scan for TCP ports of the hosts in the cloud from the on-premises without any segmentation or restrictions:

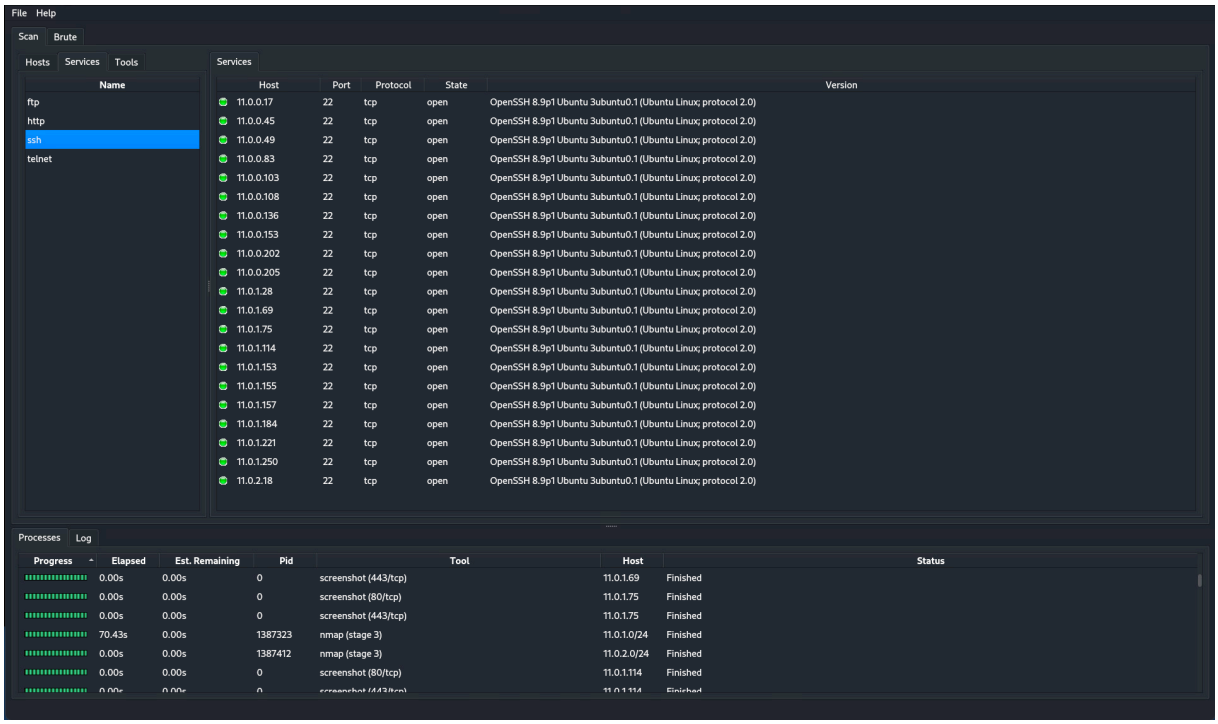


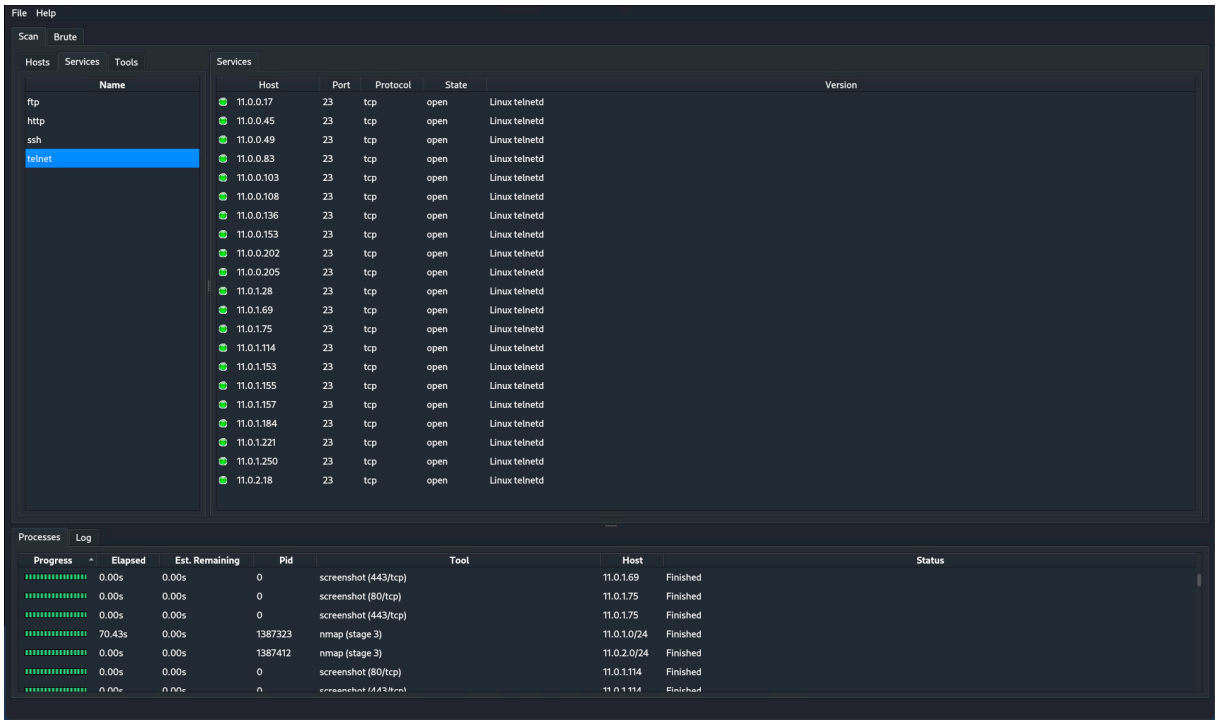

File Help
Scan Brute
Hosts Services Tools
OS Host
11.0.0.17 (unknown)
11.0.0.45 (unknown)
11.0.0.49 (unknown)
11.0.0.83 (unknown)
11.0.0.103 (unknown)
11.0.0.108 (unknown)
11.0.0.136 (unknown)
11.0.0.153 (unknown)
11.0.0.202 (unknown)
11.0.0.205 (unknown)
11.0.1.28 (unknown)
11.0.1.69 (unknown)
11.0.1.75 (unknown)
11.0.1.114 (unknown)
11.0.1.153 (unknown)
11.0.1.155 (unknown)
11.0.1.157 (unknown)
11.0.1.184 (unknown)
11.0.1.221 (unknown)
11.0.1.250 (unknown)
11.0.2.18 (unknown)
Services Scripts Information CVEs Notes screenshot (80/tcp) screenshot (443/tcp) ftp-default (21/tcp) screenshot (80/tcp) screenshot (443/tcp) ftp-default (21/tcp) screenshot (
Port Protocol State Name Version
21 tcp open ftp vsftpd 3.0.5
22 tcp open ssh OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
23 tcp open telnet Linux telnetd
80 tcp open http Apache httpd 2.4.52 ((Ubuntu))
443 tcp open http Apache httpd 2.4.52
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.69 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.75 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.75 Finished
70.43s 0.00s 1387323 nmap (stage 3) 11.0.1.0/24 Finished
0.00s 0.00s 1387412 nmap (stage 3) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.114 Finished


File Help
Scan Brute
Hosts Services Tools
OS Host
11.0.0.17 (unknown)
11.0.0.45 (unknown)
11.0.0.49 (unknown)
11.0.0.83 (unknown)
11.0.0.103 (unknown)
11.0.0.108 (unknown)
11.0.0.136 (unknown)
11.0.0.153 (unknown)
11.0.0.202 (unknown)
11.0.0.205 (unknown)
11.0.1.28 (unknown)
11.0.1.69 (unknown)
11.0.1.75 (unknown)
11.0.1.114 (unknown)
11.0.1.153 (unknown)
11.0.1.155 (unknown)
11.0.1.157 (unknown)
11.0.1.184 (unknown)
11.0.1.221 (unknown)
11.0.1.250 (unknown)
11.0.2.18 (unknown)
Services Scripts Information CVEs Notes screenshot (80/tcp) screenshot (443/tcp) ftp-default (21/tcp) screenshot (80/tcp) screenshot (443/tcp) ftp-default (21/tcp) screenshot (
Script Port
vulners 80/tcp
http-server... 80/tcp
vulners 443/tcp
cpe:/a:apache:http_server:2.4.52:
CVE-2023-25690 7.5 https://vulners.com/cve/CVE-2023-25690
CVE-2022-31813 7.5 https://vulners.com/cve/CVE-2022-31813
CVE-2022-23943 7.5 https://vulners.com/cve/CVE-2022-23943
CVE-2022-22720 7.5 https://vulners.com/cve/CVE-2022-22720
CNVD-2022-73123 7.5 https://vulners.com/cnvd/CNVD-2022-73123
5C1BB960-90C1-5EBF-9BEF-F58BFFDFEED9 7.5 https://vulners.com/githubexploit/5C1BB960-90C1-5EBF-9BEF-F58BFFDFEED9 *EXPLOIT*
CVE-2022-28615 6.4 https://vulners.com/cve/CVE-2022-28615
CVE-2021-44224 6.4 https://vulners.com/cve/CVE-2021-44224
CVE-2022-22721 5.8 https://vulners.com/cve/CVE-2022-22721
CVE-2022-36760 5.1 https://vulners.com/cve/CVE-2022-36760
CVE-2023-27522 5.0 https://vulners.com/cve/CVE-2023-27522
CVE-2022-37436 5.0 https://vulners.com/cve/CVE-2022-37436
CVE-2022-30556 5.0 https://vulners.com/cve/CVE-2022-30556
CVE-2022-29404 5.0 https://vulners.com/cve/CVE-2022-29404
CVE-2022-28614 5.0 https://vulners.com/cve/CVE-2022-28614
CVE-2022-26377 5.0 https://vulners.com/cve/CVE-2022-26377
CVE-2022-22719 5.0 https://vulners.com/cve/CVE-2022-22719
CVE-2006-20001 5.0 https://vulners.com/cve/CVE-2006-20001
CNVD-2022-73122 5.0 https://vulners.com/cnvd/CNVD-2022-73122
CNVD-2022-53584 5.0 https://vulners.com/cnvd/CNVD-2022-53584
CNVD-2022-53582 5.0 https://vulners.com/cnvd/CNVD-2022-53582
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.69 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.75 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.75 Finished
70.43s 0.00s 1387323 nmap (stage 3) 11.0.1.0/24 Finished
0.00s 0.00s 1387412 nmap (stage 3) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.114 Finished

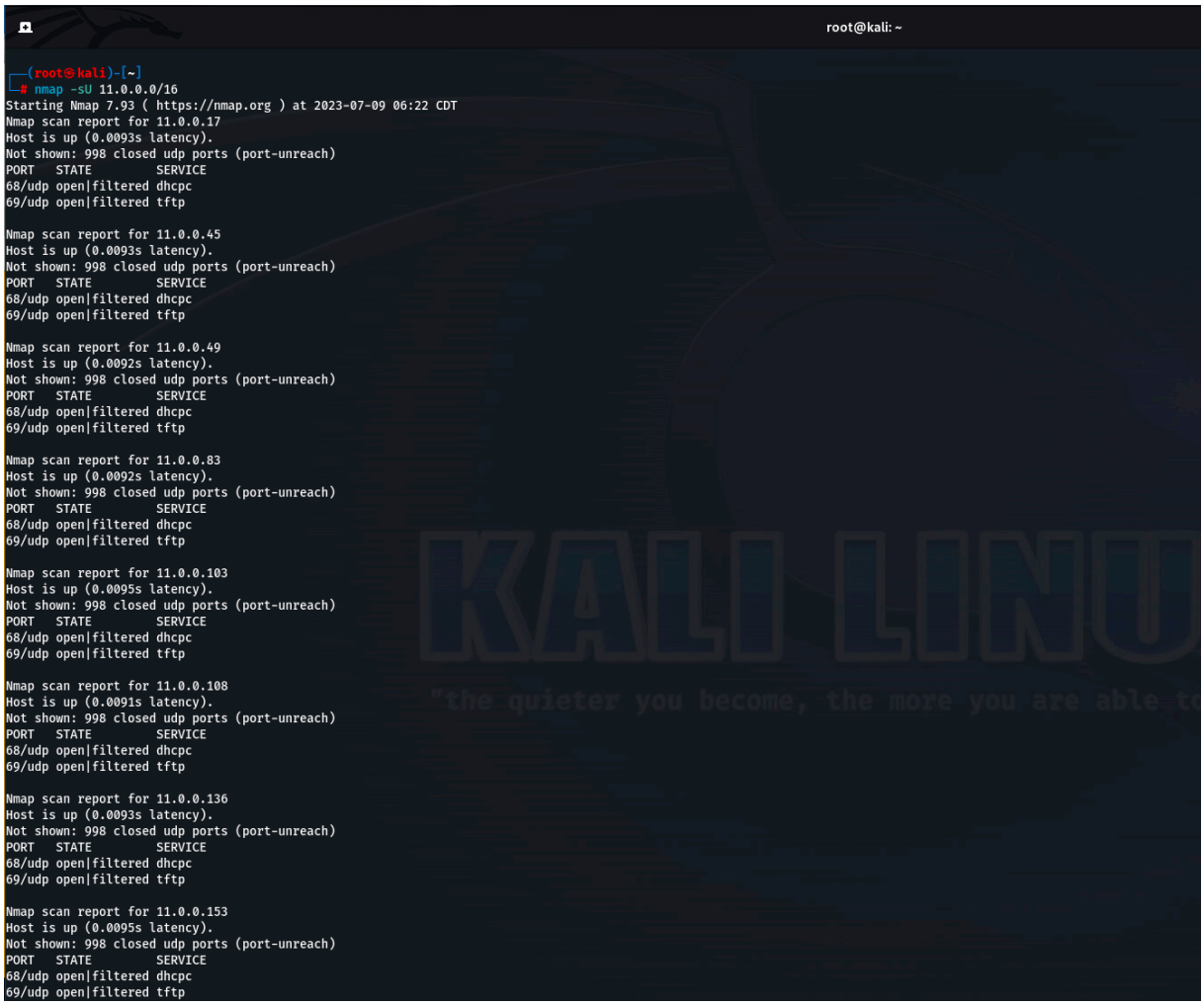

File Help
Scan Brute
Hosts Services Tools
Name
ftp
http
ssh
telnet
Services
Host Port Protocol State Version
11.0.0.17 21 tcp open vsftpd 3.0.5
11.0.0.45 21 tcp open vsftpd 3.0.5
11.0.0.49 21 tcp open vsftpd 3.0.5
11.0.0.83 21 tcp open vsftpd 3.0.5
11.0.0.103 21 tcp open vsftpd 3.0.5
11.0.0.108 21 tcp open vsftpd 3.0.5
11.0.0.136 21 tcp open vsftpd 3.0.5
11.0.0.153 21 tcp open vsftpd 3.0.5
11.0.0.202 21 tcp open vsftpd 3.0.5
11.0.0.205 21 tcp open vsftpd 3.0.5
11.0.1.28 21 tcp open vsftpd 3.0.5
11.0.1.69 21 tcp open vsftpd 3.0.5
11.0.1.75 21 tcp open vsftpd 3.0.5
11.0.1.114 21 tcp open vsftpd 3.0.5
11.0.1.153 21 tcp open vsftpd 3.0.5
11.0.1.155 21 tcp open vsftpd 3.0.5
11.0.1.157 21 tcp open vsftpd 3.0.5
11.0.1.184 21 tcp open vsftpd 3.0.5
11.0.1.221 21 tcp open vsftpd 3.0.5
11.0.1.250 21 tcp open vsftpd 3.0.5
11.0.2.18 21 tcp open vsftpd 3.0.5
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.69 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.75 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.75 Finished
70.43s 0.00s 1387323 nmap (stage 3) 11.0.1.0/24 Finished
0.00s 0.00s 1387412 nmap (stage 3) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.114 Finished

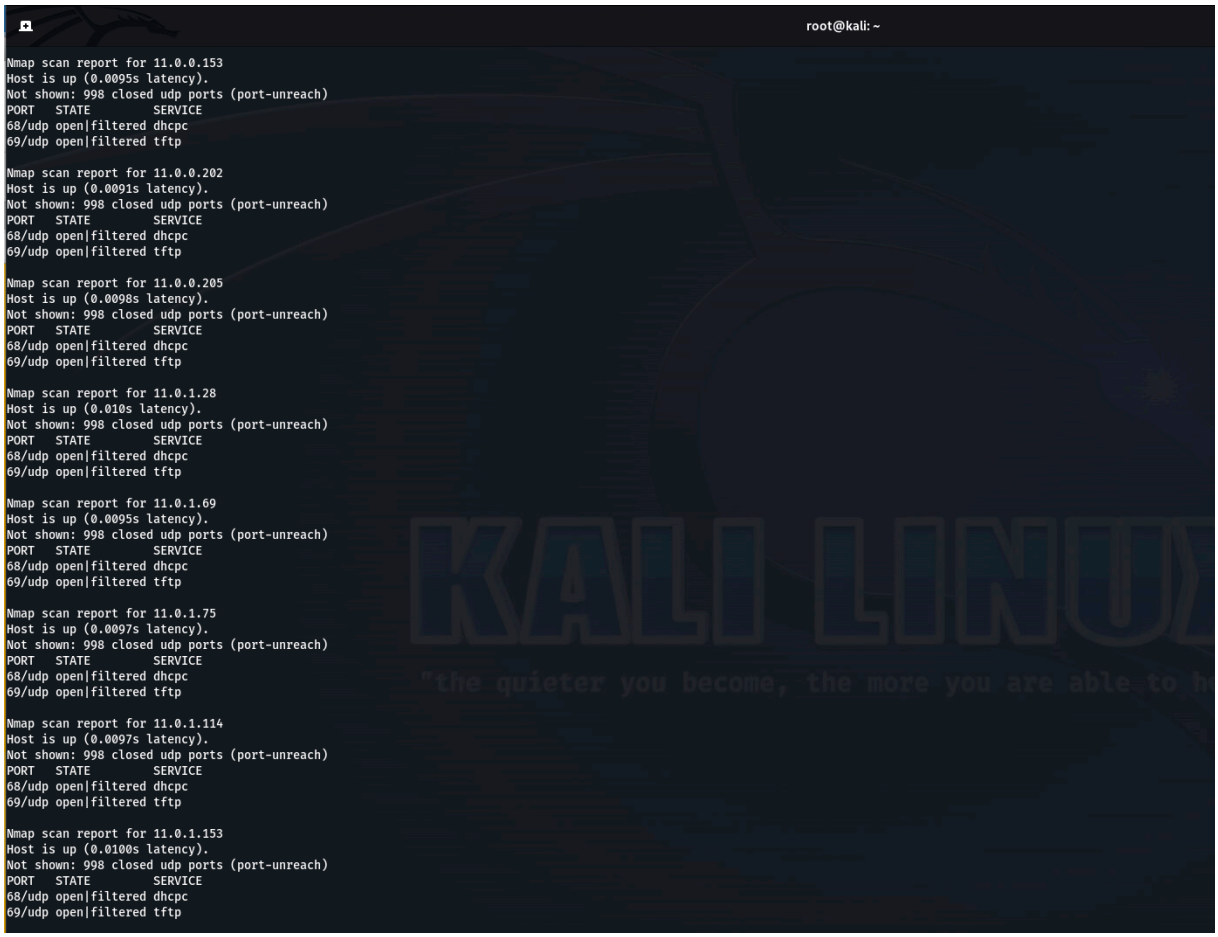

File Help
Scan Brute
Hosts Services Tools
Name
ftp
http
ssh
telnet
Services
Host Port Protocol State Version
11.0.0.17 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.17 443 tcp open Apache httpd 2.4.52
11.0.0.45 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.45 443 tcp open Apache httpd 2.4.52
11.0.0.49 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.49 443 tcp open Apache httpd 2.4.52
11.0.0.83 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.83 443 tcp open Apache httpd 2.4.52
11.0.0.103 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.103 443 tcp open Apache httpd 2.4.52
11.0.0.108 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.108 443 tcp open Apache httpd 2.4.52
11.0.0.136 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.136 443 tcp open Apache httpd 2.4.52
11.0.0.153 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.153 443 tcp open Apache httpd 2.4.52
11.0.0.202 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.202 443 tcp open Apache httpd 2.4.52
11.0.0.205 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.0.205 443 tcp open Apache httpd 2.4.52
11.0.1.28 80 tcp open Apache httpd 2.4.52 ((Ubuntu))
11.0.1.28 443 tcp open Apache httpd 2.4.52
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.69 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.75 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.75 Finished
70.43s 0.00s 1387323 nmap (stage 3) 11.0.1.0/24 Finished
0.00s 0.00s 1387412 nmap (stage 3) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.114 Finished

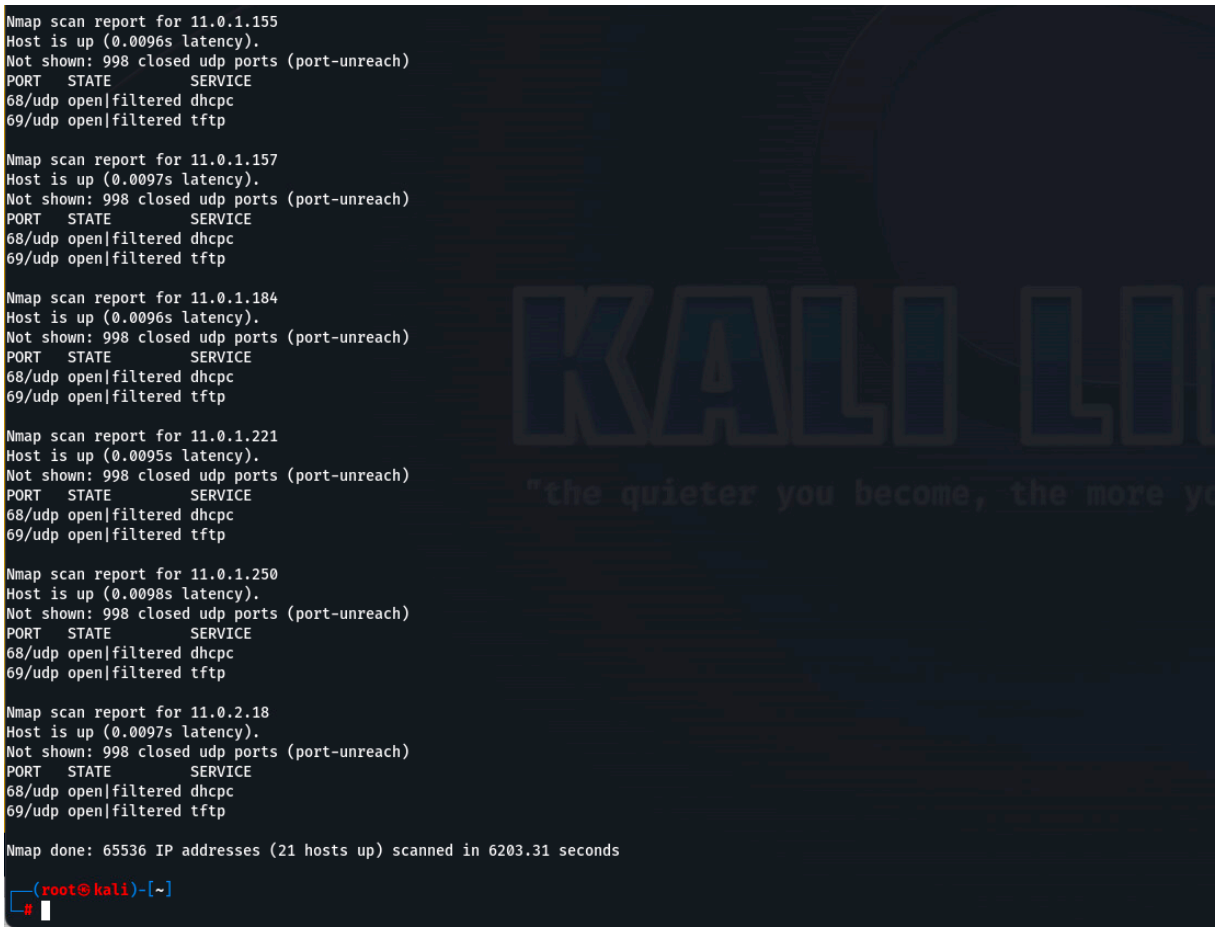

File Help
Scan Brute
Hosts Services Tools
Name
ftp
http
ssh
telnet
Services
Host Port Protocol State Version
11.0.0.17 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.45 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.49 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.83 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.103 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.108 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.136 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.153 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.202 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.205 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.28 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.69 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.75 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.114 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.153 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.155 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.157 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.184 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.221 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.250 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.2.18 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.69 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.75 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.75 Finished
70.43s 0.00s 1387323 nmap (stage 3) 11.0.1.0/24 Finished
0.00s 0.00s 1387412 nmap (stage 3) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.114 Finished


File Help
Scan Brute
Hosts Services Tools
Name
ftp
http
ssh
telnet
Services
Host Port Protocol State Version
11.0.0.17 23 tcp open Linux telnetd
11.0.0.45 23 tcp open Linux telnetd
11.0.0.49 23 tcp open Linux telnetd
11.0.0.83 23 tcp open Linux telnetd
11.0.0.103 23 tcp open Linux telnetd
11.0.0.108 23 tcp open Linux telnetd
11.0.0.136 23 tcp open Linux telnetd
11.0.0.153 23 tcp open Linux telnetd
11.0.0.202 23 tcp open Linux telnetd
11.0.0.205 23 tcp open Linux telnetd
11.0.1.28 23 tcp open Linux telnetd
11.0.1.69 23 tcp open Linux telnetd
11.0.1.75 23 tcp open Linux telnetd
11.0.1.114 23 tcp open Linux telnetd
11.0.1.153 23 tcp open Linux telnetd
11.0.1.155 23 tcp open Linux telnetd
11.0.1.157 23 tcp open Linux telnetd
11.0.1.184 23 tcp open Linux telnetd
11.0.1.221 23 tcp open Linux telnetd
11.0.1.250 23 tcp open Linux telnetd
11.0.2.18 23 tcp open Linux telnetd
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.69 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.75 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.1.75 Finished
70.43s 0.00s 1387323 nmap (stage 3) 11.0.1.0/24 Finished
0.00s 0.00s 1387412 nmap (stage 3) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (80/tcp) 11.0.1.114 Finished

Nmap scan for UDP ports for the same baseline scan.

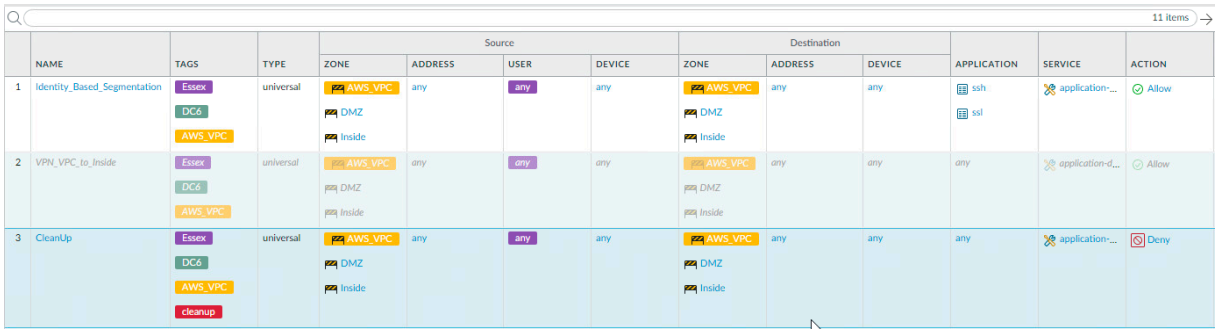



root@kali: ~
Nmap scan report for 11.0.0.153
Host is up (0.0095s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.0.202
Host is up (0.0091s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.0.205
Host is up (0.0098s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.28
Host is up (0.010s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.69
Host is up (0.0095s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.75
Host is up (0.0097s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.114
Host is up (0.0097s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.153
Host is up (0.0100s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp


Nmap scan report for 11.0.1.155
Host is up (0.0096s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.157
Host is up (0.0097s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.184
Host is up (0.0096s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.221
Host is up (0.0095s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.1.250
Host is up (0.0098s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap scan report for 11.0.2.18
Host is up (0.0097s latency).
Not shown: 998 closed udp ports (port-unreach)
PORT STATE SERVICE
68/udp open|filtered dhcpc
69/udp open|filtered tftp
Nmap done: 65536 IP addresses (21 hosts up) scanned in 6203.31 seconds
(root㉿kali)-[~]
#

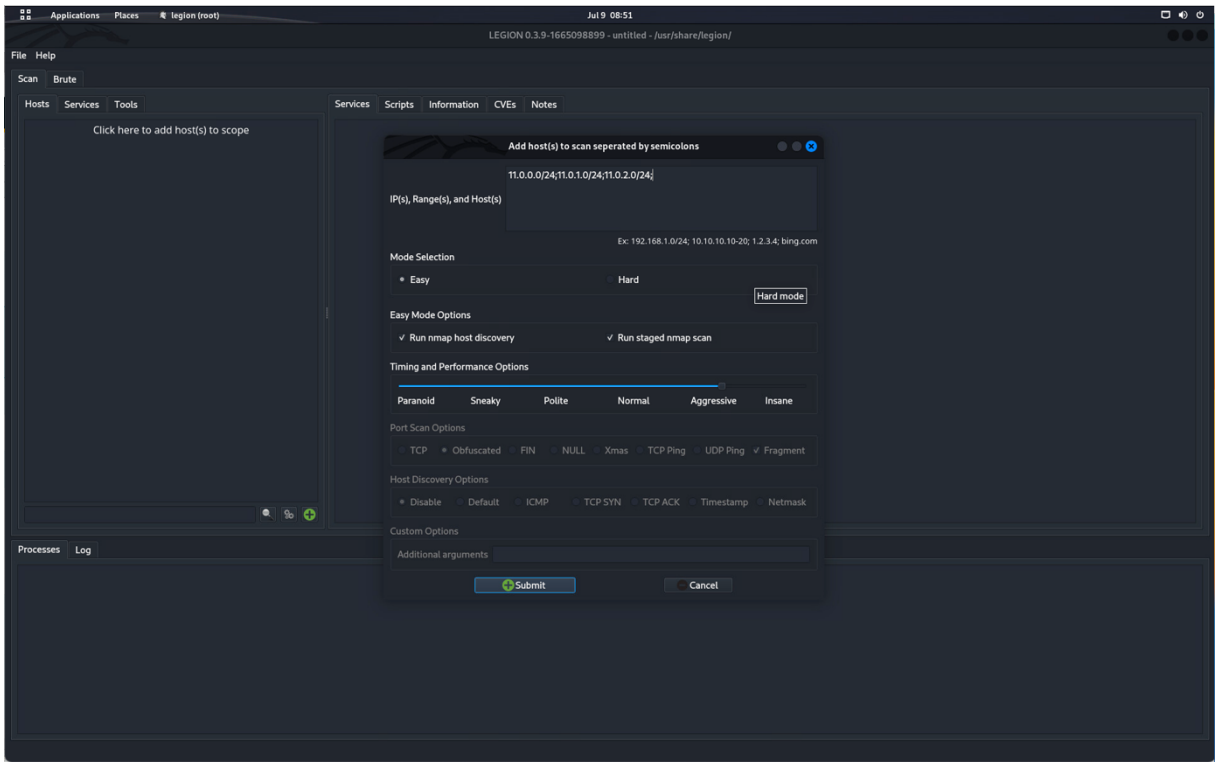
b. Identity Based Segmentation Scanning

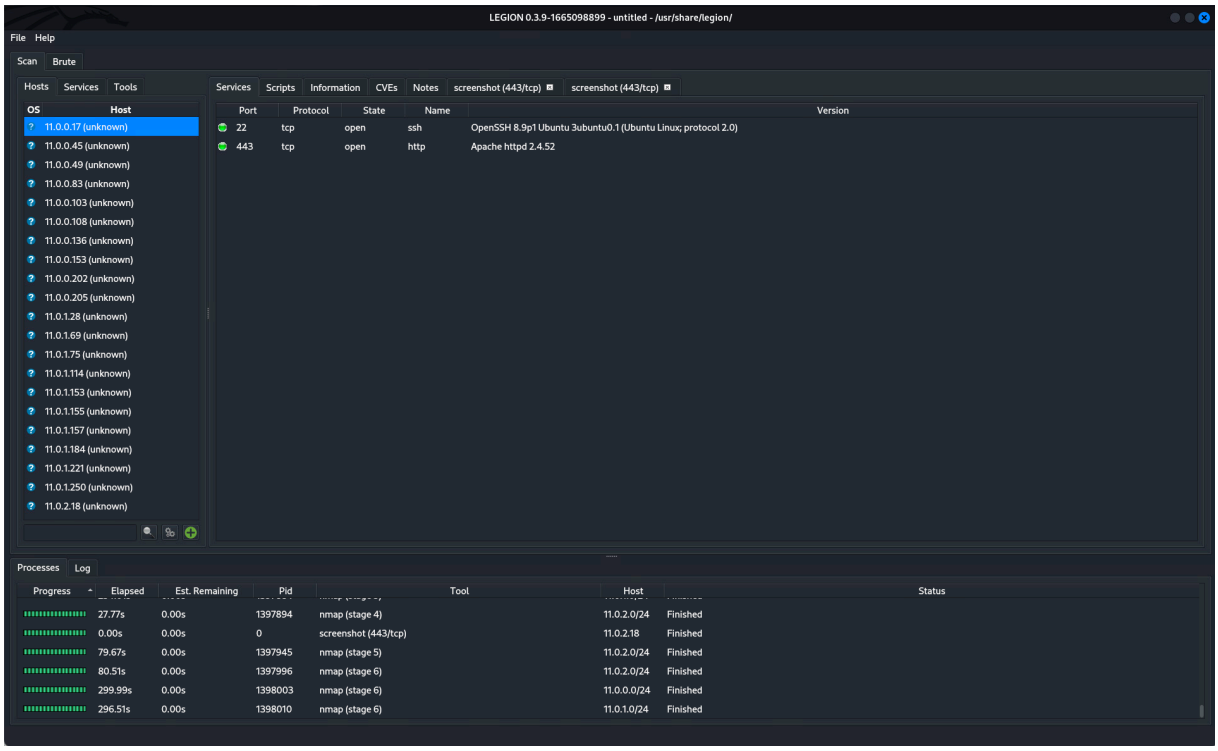
Firewall rulesets defined for Identity-based segmentation. The application section of the firewall ruleset contains the identity of the applications used to permit traffic.

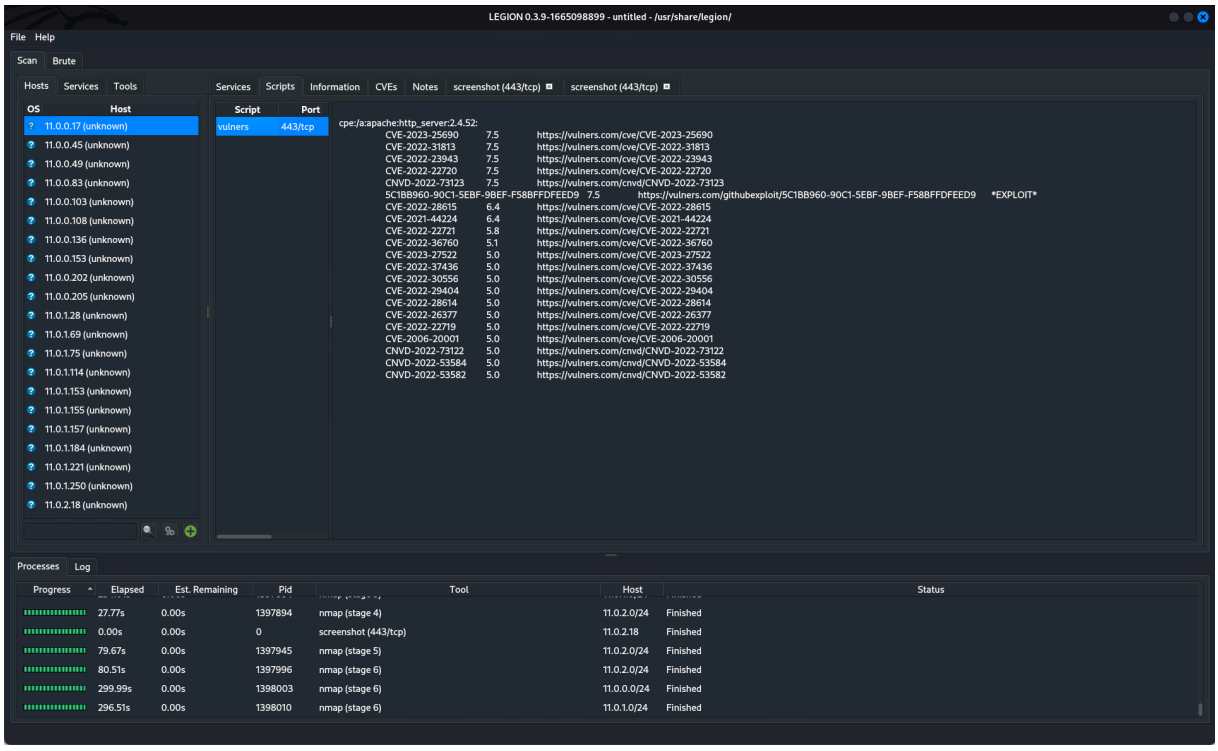


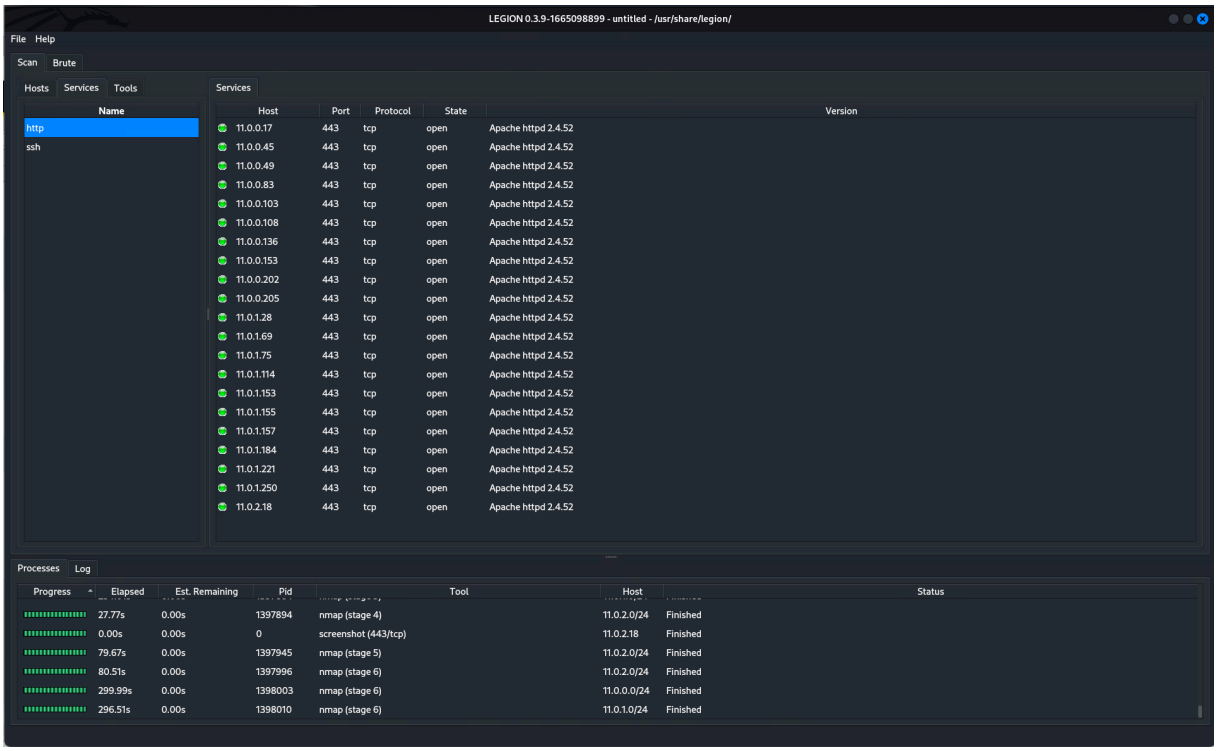
	NAME	TAGS	TYPE	Source ZONE	Source ADDRESS	Source USER	Source DEVICE	Destination ZONE	Destination ADDRESS	Destination DEVICE	APPLICATION	SERVICE	ACTION
1	Identity_Based_Segmentation	Essex, DC6, AWS_VPC	universal	AWS_VPC, DMZ, Inside	any	any	any	AWS_VPC, DMZ, Inside	any	any	ssh, ssl	application-...	Allow
2	VPN_VPC_to_Inside	Essex, DC6, AWS_VPC	universal	AWS_VPC, DMZ, Inside	any	any	any	AWS_VPC, DMZ, Inside	any	any	any	application-d...	Allow
3	CleanUp	Essex, DC6, AWS_VPC, cleanup	universal	AWS_VPC, DMZ, Inside	any	any	any	AWS_VPC, DMZ, Inside	any	any	any	application-...	Deny

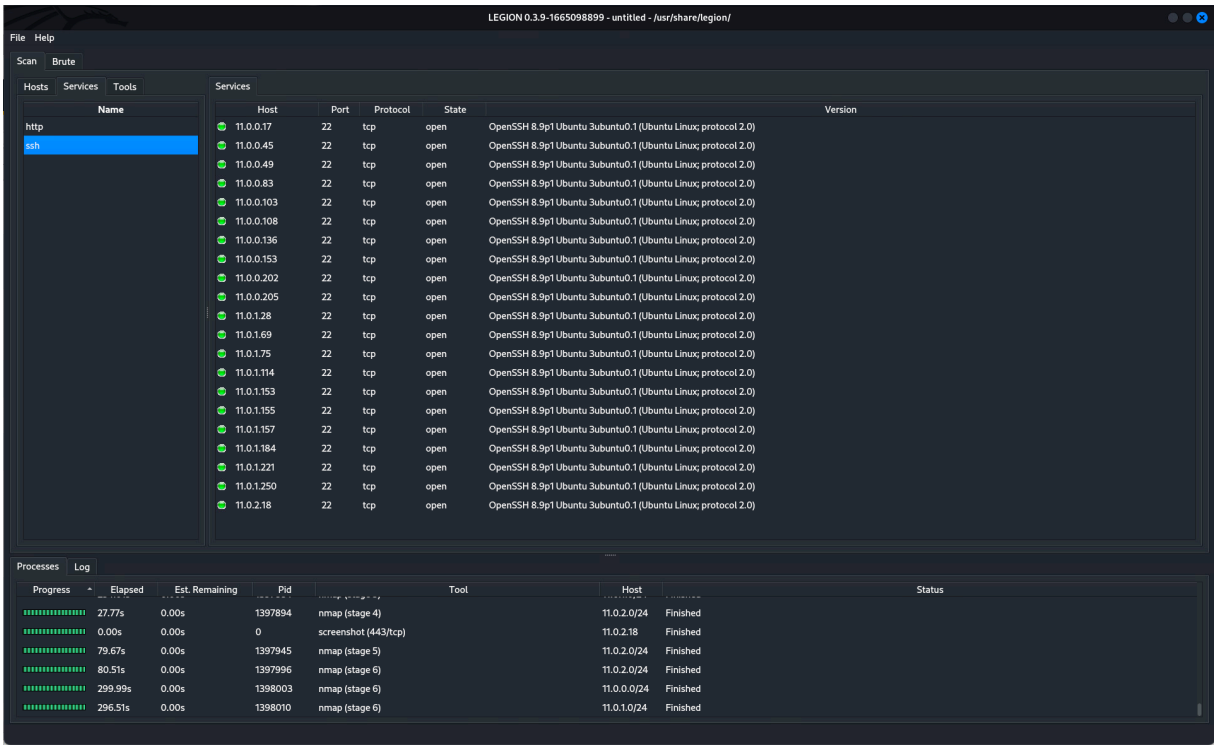
Defining and initiating the TCP Scan using the Legion application part of Kali Linux:



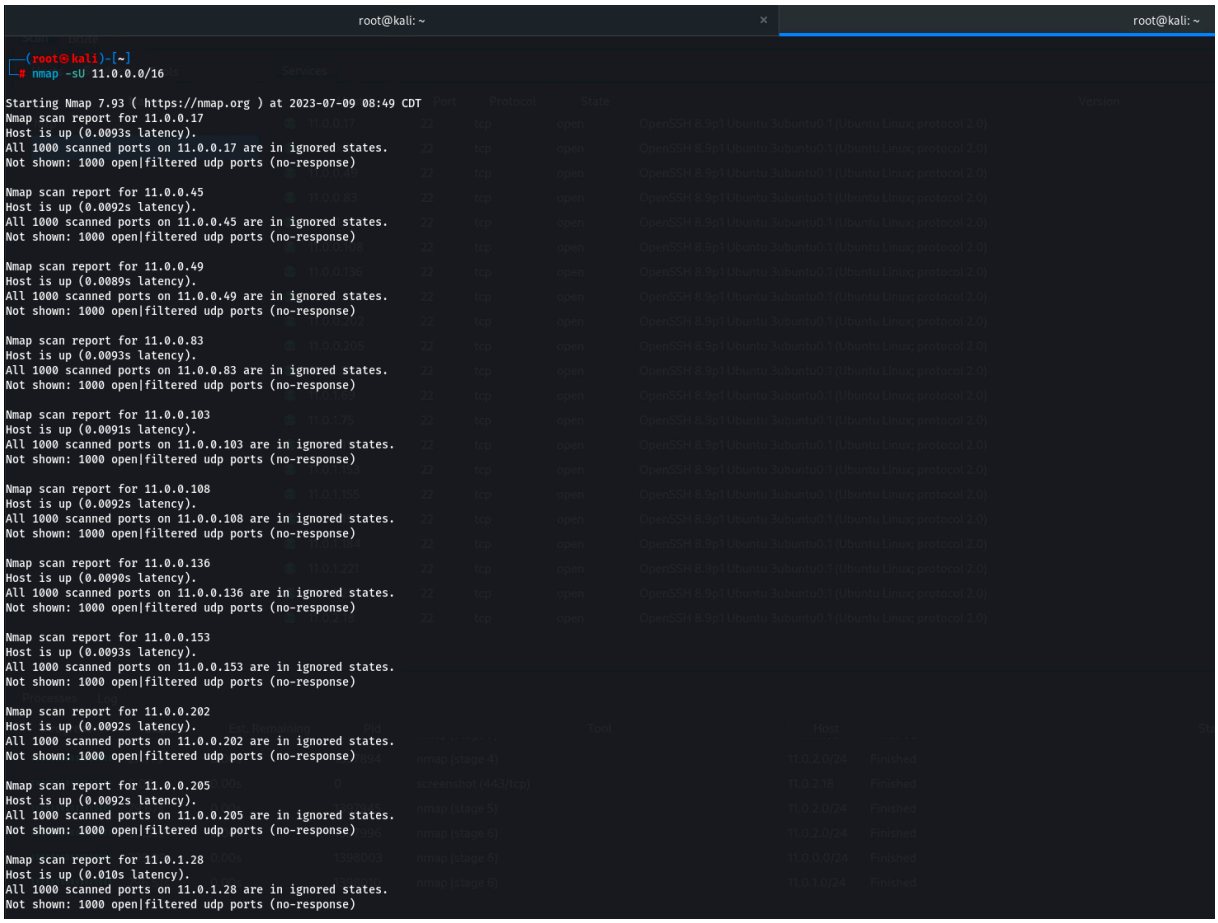

LEGION 0.3.9-1665098899 - untitled - /usr/share/legion/
File Help
Scan Brute
Hosts Services Tools
OS Host
11.0.0.17 (unknown)
11.0.0.45 (unknown)
11.0.0.49 (unknown)
11.0.0.83 (unknown)
11.0.0.103 (unknown)
11.0.0.108 (unknown)
11.0.0.136 (unknown)
11.0.0.153 (unknown)
11.0.0.202 (unknown)
11.0.0.205 (unknown)
11.0.1.28 (unknown)
11.0.1.69 (unknown)
11.0.1.75 (unknown)
11.0.1.114 (unknown)
11.0.1.153 (unknown)
11.0.1.155 (unknown)
11.0.1.157 (unknown)
11.0.1.184 (unknown)
11.0.1.221 (unknown)
11.0.1.250 (unknown)
11.0.2.18 (unknown)
Services Scripts Information CVEs Notes screenshot (443/tcp) screenshot (443/tcp)
Port Protocol State Name Version
22 tcp open ssh OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
443 tcp open http Apache httpd 2.4.52
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
27.77s 0.00s 1397894 nmap (stage 4) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.2.18 Finished
79.67s 0.00s 1397945 nmap (stage 5) 11.0.2.0/24 Finished
80.51s 0.00s 1397996 nmap (stage 6) 11.0.2.0/24 Finished
299.99s 0.00s 1398003 nmap (stage 6) 11.0.0.0/24 Finished
296.51s 0.00s 1398010 nmap (stage 6) 11.0.1.0/24 Finished

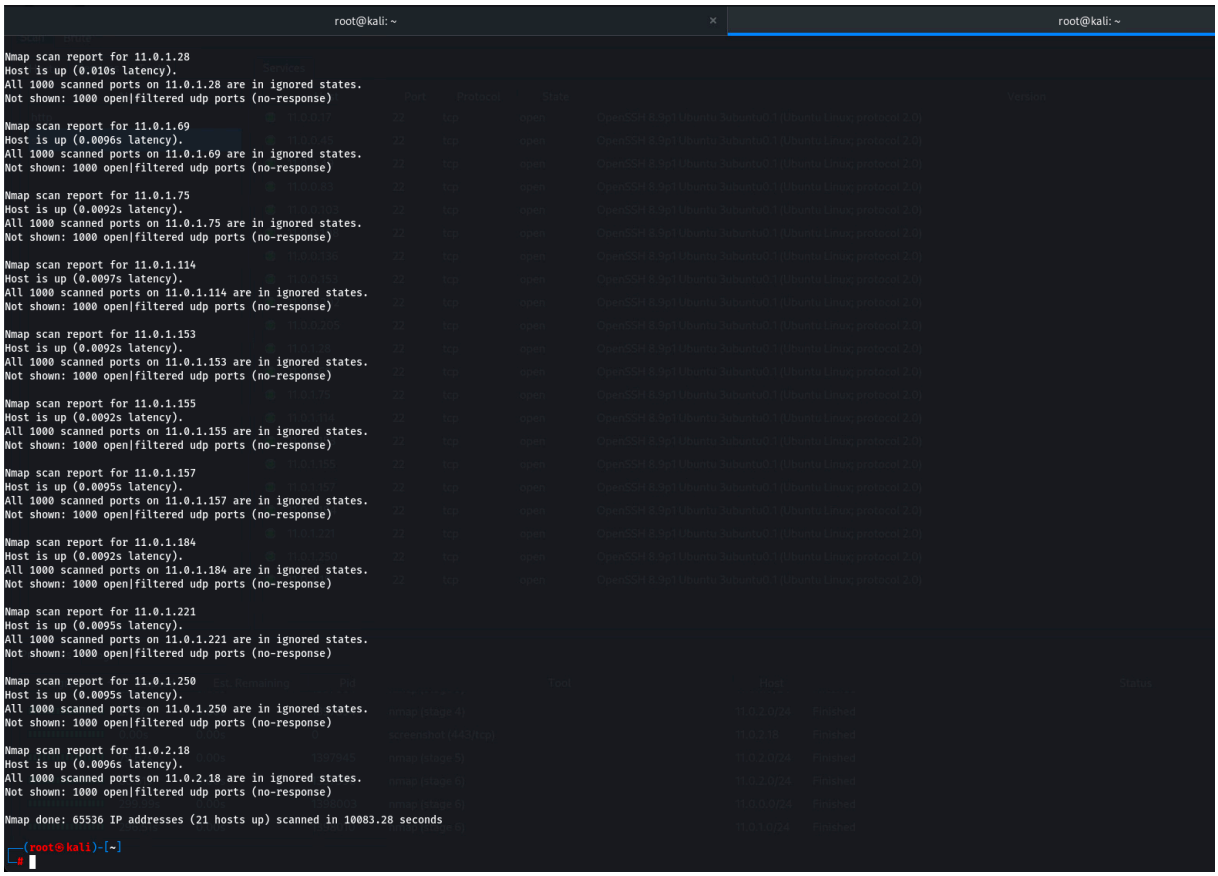

LEGION 0.3.9-1665098899 - untitled - /usr/share/legion/
File Help
Scan Brute
Hosts Services Tools
OS Host
11.0.0.17 (unknown)
11.0.0.45 (unknown)
11.0.0.49 (unknown)
11.0.0.83 (unknown)
11.0.0.103 (unknown)
11.0.0.108 (unknown)
11.0.0.136 (unknown)
11.0.0.153 (unknown)
11.0.0.202 (unknown)
11.0.0.205 (unknown)
11.0.1.28 (unknown)
11.0.1.69 (unknown)
11.0.1.75 (unknown)
11.0.1.114 (unknown)
11.0.1.153 (unknown)
11.0.1.155 (unknown)
11.0.1.157 (unknown)
11.0.1.184 (unknown)
11.0.1.221 (unknown)
11.0.1.250 (unknown)
11.0.2.18 (unknown)
Services Scripts Information CVEs Notes screenshot (443/tcp) screenshot (443/tcp)
Script Port
vulners 443/tcp
cpe:/a:apache:http_server:2.4.52:
CVE-2023-25690 7.5 https://vulners.com/cve/CVE-2023-25690
CVE-2022-31813 7.5 https://vulners.com/cve/CVE-2022-31813
CVE-2022-23943 7.5 https://vulners.com/cve/CVE-2022-23943
CVE-2022-22720 7.5 https://vulners.com/cve/CVE-2022-22720
CNVD-2022-73123 7.5 https://vulners.com/cnvd/CNVD-2022-73123
5C1BB960-90C1-5EBF-9BEF-F58BFFDFEED9 7.5 https://vulners.com/githubexploit/5C1BB960-90C1-5EBF-9BEF-F58BFFDFEED9 *EXPLOIT*
CVE-2022-28615 6.4 https://vulners.com/cve/CVE-2022-28615
CVE-2021-44224 6.4 https://vulners.com/cve/CVE-2021-44224
CVE-2022-22721 5.8 https://vulners.com/cve/CVE-2022-22721
CVE-2022-36760 5.1 https://vulners.com/cve/CVE-2022-36760
CVE-2023-27522 5.0 https://vulners.com/cve/CVE-2023-27522
CVE-2022-37436 5.0 https://vulners.com/cve/CVE-2022-37436
CVE-2022-30556 5.0 https://vulners.com/cve/CVE-2022-30556
CVE-2022-29404 5.0 https://vulners.com/cve/CVE-2022-29404
CVE-2022-28614 5.0 https://vulners.com/cve/CVE-2022-28614
CVE-2022-26377 5.0 https://vulners.com/cve/CVE-2022-26377
CVE-2022-22719 5.0 https://vulners.com/cve/CVE-2022-22719
CVE-2006-20001 5.0 https://vulners.com/cve/CVE-2006-20001
CNVD-2022-73122 5.0 https://vulners.com/cnvd/CNVD-2022-73122
CNVD-2022-53584 5.0 https://vulners.com/cnvd/CNVD-2022-53584
CNVD-2022-53582 5.0 https://vulners.com/cnvd/CNVD-2022-53582
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
27.77s 0.00s 1397894 nmap (stage 4) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.2.18 Finished
79.67s 0.00s 1397945 nmap (stage 5) 11.0.2.0/24 Finished
80.51s 0.00s 1397996 nmap (stage 6) 11.0.2.0/24 Finished
299.99s 0.00s 1398003 nmap (stage 6) 11.0.0.0/24 Finished
296.51s 0.00s 1398010 nmap (stage 6) 11.0.1.0/24 Finished


LEGION 0.3.9-1665098899 - untitled - /usr/share/legion/
File Help
Scan Brute
Hosts Services Tools
Name
http
ssh
Services
Host Port Protocol State Version
11.0.0.17 443 tcp open Apache httpd 2.4.52
11.0.0.45 443 tcp open Apache httpd 2.4.52
11.0.0.49 443 tcp open Apache httpd 2.4.52
11.0.0.83 443 tcp open Apache httpd 2.4.52
11.0.0.103 443 tcp open Apache httpd 2.4.52
11.0.0.108 443 tcp open Apache httpd 2.4.52
11.0.0.136 443 tcp open Apache httpd 2.4.52
11.0.0.153 443 tcp open Apache httpd 2.4.52
11.0.0.202 443 tcp open Apache httpd 2.4.52
11.0.0.205 443 tcp open Apache httpd 2.4.52
11.0.1.28 443 tcp open Apache httpd 2.4.52
11.0.1.69 443 tcp open Apache httpd 2.4.52
11.0.1.75 443 tcp open Apache httpd 2.4.52
11.0.1.114 443 tcp open Apache httpd 2.4.52
11.0.1.153 443 tcp open Apache httpd 2.4.52
11.0.1.155 443 tcp open Apache httpd 2.4.52
11.0.1.157 443 tcp open Apache httpd 2.4.52
11.0.1.184 443 tcp open Apache httpd 2.4.52
11.0.1.221 443 tcp open Apache httpd 2.4.52
11.0.1.250 443 tcp open Apache httpd 2.4.52
11.0.2.18 443 tcp open Apache httpd 2.4.52
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
27.77s 0.00s 1397894 nmap (stage 4) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.2.18 Finished
79.67s 0.00s 1397945 nmap (stage 5) 11.0.2.0/24 Finished
80.51s 0.00s 1397996 nmap (stage 6) 11.0.2.0/24 Finished
299.99s 0.00s 1398003 nmap (stage 6) 11.0.0.0/24 Finished
296.51s 0.00s 1398010 nmap (stage 6) 11.0.1.0/24 Finished


LEGION 0.3.9-1665098899 - untitled - /usr/share/legion/
File Help
Scan Brute
Hosts Services Tools
Name
http
ssh
Services
Host Port Protocol State Version
11.0.0.17 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.45 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.49 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.83 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.103 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.108 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.136 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.153 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.202 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.0.205 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.28 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.69 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.75 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.114 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.153 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.155 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.157 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.184 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.221 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.1.250 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
11.0.2.18 22 tcp open OpenSSH 8.9p1 Ubuntu 3ubuntu0.1 (Ubuntu Linux; protocol 2.0)
Processes Log
Progress Elapsed Est. Remaining Pid Tool Host Status
27.77s 0.00s 1397894 nmap (stage 4) 11.0.2.0/24 Finished
0.00s 0.00s 0 screenshot (443/tcp) 11.0.2.18 Finished
79.67s 0.00s 1397945 nmap (stage 5) 11.0.2.0/24 Finished
80.51s 0.00s 1397996 nmap (stage 6) 11.0.2.0/24 Finished
299.99s 0.00s 1398003 nmap (stage 6) 11.0.0.0/24 Finished
296.51s 0.00s 1398010 nmap (stage 6) 11.0.1.0/24 Finished

Running the UDP scan using Nmap for identity-based segmentation:

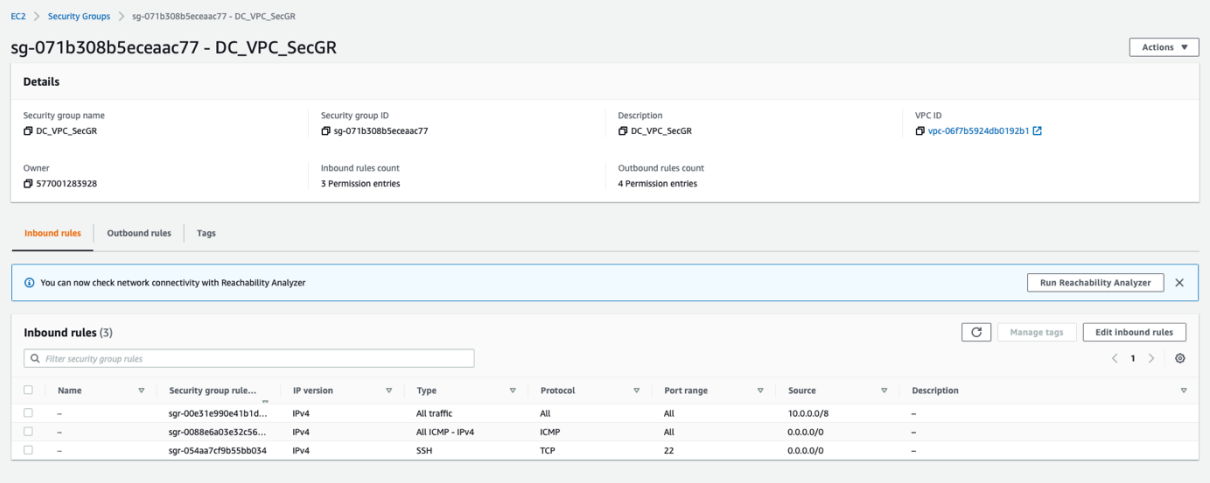



root@kali: ~
root@kali: ~
Nmap scan report for 11.0.1.28
Host is up (0.010s latency).
All 1000 scanned ports on 11.0.1.28 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.69
Host is up (0.0096s latency).
All 1000 scanned ports on 11.0.1.69 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.75
Host is up (0.0092s latency).
All 1000 scanned ports on 11.0.1.75 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.114
Host is up (0.0097s latency).
All 1000 scanned ports on 11.0.1.114 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.153
Host is up (0.0092s latency).
All 1000 scanned ports on 11.0.1.153 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.155
Host is up (0.0092s latency).
All 1000 scanned ports on 11.0.1.155 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.157
Host is up (0.0095s latency).
All 1000 scanned ports on 11.0.1.157 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.184
Host is up (0.0092s latency).
All 1000 scanned ports on 11.0.1.184 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.221
Host is up (0.0095s latency).
All 1000 scanned ports on 11.0.1.221 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.1.250
Host is up (0.0095s latency).
All 1000 scanned ports on 11.0.1.250 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap scan report for 11.0.2.18
Host is up (0.0096s latency).
All 1000 scanned ports on 11.0.2.18 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
Nmap done: 65536 IP addresses (21 hosts up) scanned in 10083.28 seconds
┌──(root㉿kali)-[~]
└─#

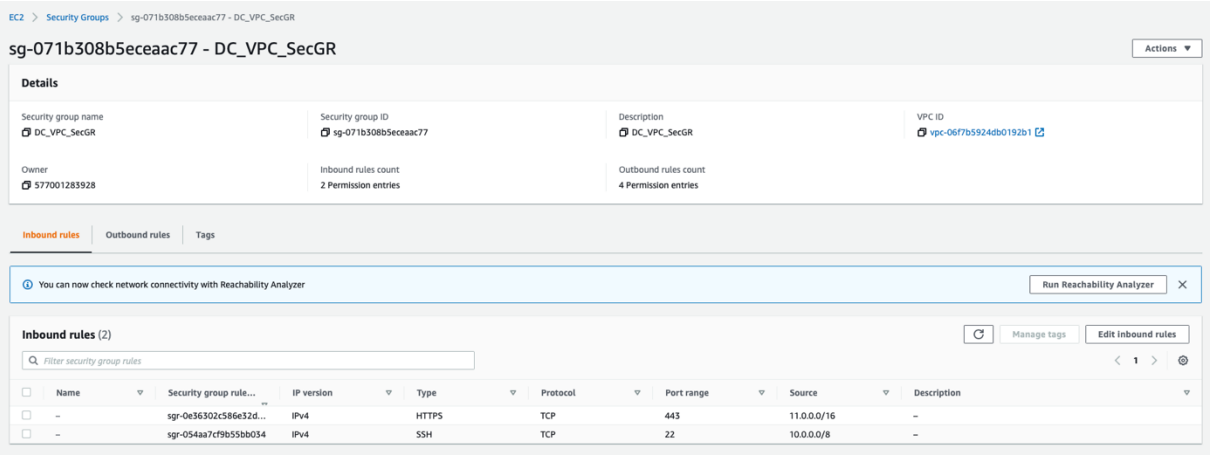
c. Micro-Segmentation Scanning

Defining the new security group for the purpose of micro-segmentation:

Before:



After:

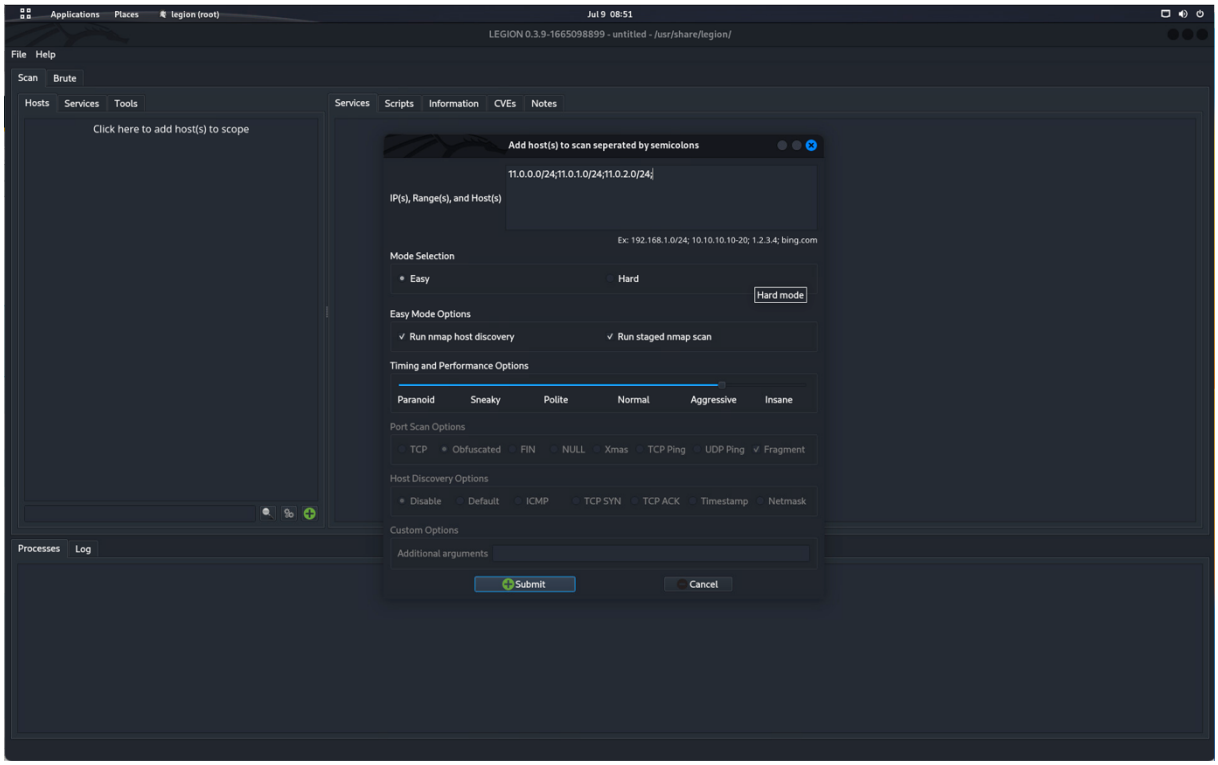


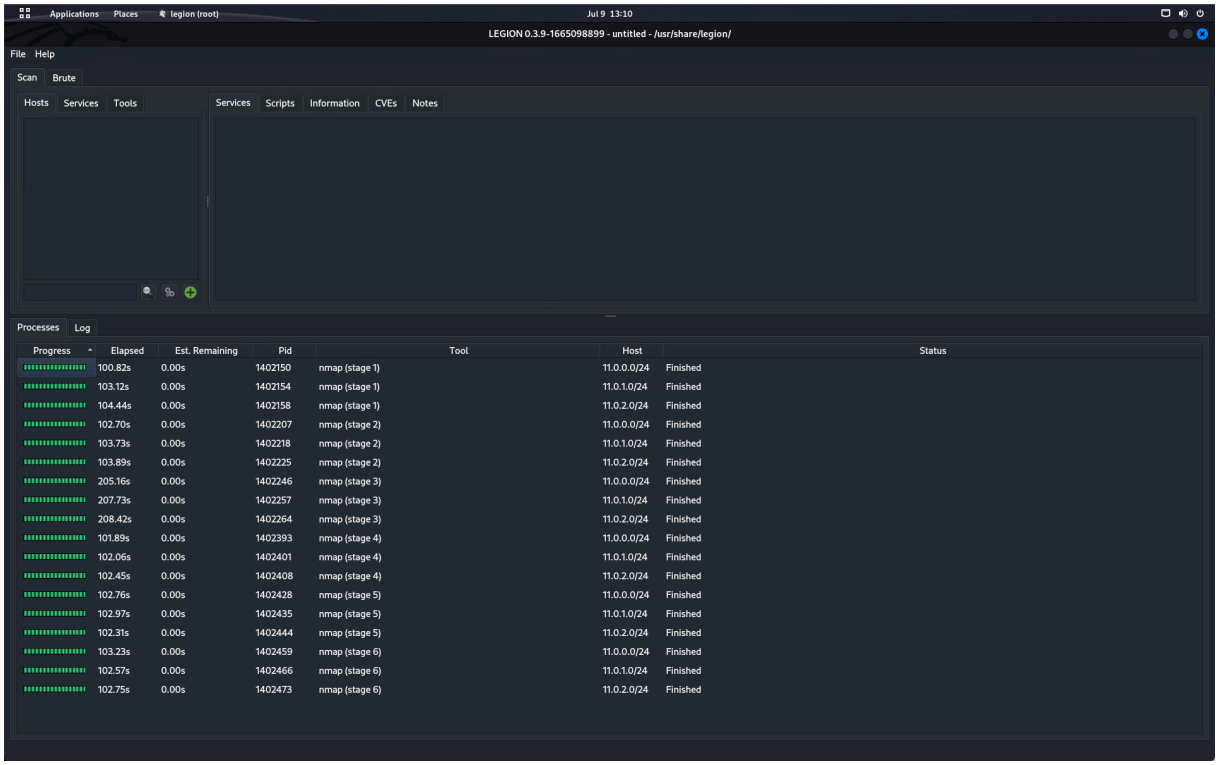
Adjusting the firewall ruleset by disabling the identity base implementation and reverting to baseline configuration:

11 items

	NAME	TAGS	TYPE	Source				Destination			APPLICATION	SERVICE	ACTION
				ZONE	ADDRESS	USER	DEVICE	ZONE	ADDRESS	DEVICE			
1	Identity_Based_Segmentation	Essex DC6 AWS_VPC	universal	AWS_VPC DMZ Inside	any	any	any	AWS_VPC DMZ Inside	any	any	ssh ssl	application-d...	Allow
2	VPN_VPC_to_Inside	Essex DC6 AWS_VPC	universal	AWS_VPC DMZ Inside	any	any	any	AWS_VPC DMZ Inside	any	any	any	application-...	Allow
3	CleanUp	Essex DC6 AWS_VPC cleanup	universal	AWS_VPC DMZ Inside	any	any	any	AWS_VPC DMZ Inside	any	any	any	application-...	Deny

Defining and running the scan for micro-segmentation:





Running Metasloit for a secondary scan: