**MINI-PROJECT TOPIC**

**Study and implementation of a filtering system**

**(Iptables under Linux, ASA under Packet Tracer)**

**Made by:**

**RHOZAN Hajar**

**BOUKEBI Hayat**

**Presented on 12/21/2024 before:**

**Prof. Mr. KARTIT Ali Professor at ENSAJ**

**Academic Year: 2024/2025**

**Table of contents**  
  
Introduction …………………………………………………………………………...…….. 05

Prerequisites…………………………………………………………………………………. 06

1. Definition of firewall…………………………………………………………………… 06

2. The different types of firewalls… ..… ...……………………………………………….06

3. The different types of filtering…………………………………………………………..07

Buts…………………………………………………………………………………………...09

Principle of operation of the filtering system………………………………………………... 09

Chapter 2: Iptables on Linux………………………………………………………………… 11

2.1 General Overview of Iptables……………………………………………...…………..11  
  
 2.2 Syntax and Configuration Rules/Usage Examples and Use Cases ……………………11

Chapter 3: ASA under Packet Tracer ………………………………………………………...15  
  
3.1 Presentation of the Adaptive Security Appliance (ASA) ….…………………………… 15  
  
3.2 Project Description……………………………………………………………………… 15

3.3 Analysis of the company's security needs……………………………………………….. 15  
  
3.4 Network Architecture Design…………………………………………………………… 16  
  
3.5 Firewall Configuration…………………………………………………………………... 16  
  
Comparison and selection of filtering systems……………………………………………… 22  
  
 1. Iptables (Linux)…………………………………………………………………..……… 22

2. Cisco ASA…………………………….………………………………………..………... 22  
  
 3. Criteria for Choice…………………………………….………………………………… 22  
  
Conclusion and Perspectives………………………………………………………………… 24  
  
References…………………………………………………………………………………… 25

**List of figures**

[Figure 1: Ensure the installation of iptables on Linux……………… .……………………...…11](#_Toc182235182)

[Figure 2: Display of current rules………………………. . …………………...………………11](#_Toc182235182)

[Figure 3: A rule to block incoming traffic to the server……….. ….……….…………...………11](#_Toc182235182)

[Figure 4: A rule to block crossing traffic…………………………….....………………………11](#_Toc182235182)

[Figure 5: A rule to block outgoing traffic……………………………...………………………12](#_Toc182235182)

[Figure 6: A rule for authorizing already established or related connections…………………12](#_Toc182235182)

[Figure 7: A rule for authorization…………………………..…………………………………...12](#_Toc182235182)

[Figure 8: A rule for SSH traffic authorization…………………………………………………12](#_Toc182235182)

[Figure 9: A rule for HTTP traffic authorization………………..….……………………………12](#_Toc182235182)

[Figure 10: A rule for HTTPS authorization…………………………….………………………12](#_Toc182235182)

[Figure 11: A rule for ICMP ping authorization……………………………………….………………………12](#_Toc182235182)

[Figure 12: A rule for blocking an address 10.0.2.15…………………..……………………... 13](#_Toc182235182)

[Figure 13: IP address of the blocked machine……………….……………………………….…13](#_Toc182235182)

[Figure 14: A test to test the blockage……………………………………………………………13](#_Toc182235182)

[Figure 15: Verification of access via ssh…………………..……………………………………14](#_Toc182235182)

[Figure 16: Limiting the rate of requests on SSH……………………………….……………………14](#_Toc182235182)

### [Figure 17: Blocking IPs after multiple unsuccessful login attempts ……………..……..…… 14](#_Toc182235182)

[Figure 18: Configuration of Logging Rules……………………………………….…….……… 14](#_Toc182235182)

[Figure 19: the network architecture………………….…………….…………………………….16](#_Toc182235182)

[Figure 20: Configuration basique…………………………………………………………………….17](#_Toc182235182)

[Figure 21: Configuration of the gig1/1 interface……………………..…….…………………..17](#_Toc182235182)

[Figure 22: Configuration of the gig1/2 interface……………………..……….……………….17](#_Toc182235182)

[Figure 23: Configuration of the gig1/3 interface……………………..……….………………..18](#_Toc182235182)

[Figure 24: The display of configurations …………………………………..…………………….18](#_Toc182235182)

[Figure 25: Configuration of DHCP……………...……………………………………………………19](#_Toc182235182)

[Figure 26: Configuration of DNS…………………………………...……………………..…………19](#_Toc182235182)

[Figure 27: Enabling DHCP on the interfaces………………………..…………………………19](#_Toc182235182)

[Figure 28: Enabling DHCP on the computers………………………………...…………….…19](#_Toc182235182)

[Figure 29: Connection Verification…………………………......………………………………20](#_Toc182235182)

[Figure 30: Activation of LOCAL aaa authentication…………………..………………………2](#_Toc182235182)0

[Figure 31: RSA Key Generation………………………………...………………………………2](#_Toc182235182)0

[Figure 32: Enable SSH………………………………..…………...………………………………2](#_Toc182235182)0

[Figure 33 : SSH timeout ……………………………………………...………………………………21](#_Toc182235182)

[Figure 34 : Test……………….……………………………..………...………………………………21](#_Toc182235182)

**Introduction**

**I**n the modern digital age, network security is paramount to protect sensitive information and ensure the reliability of communication systems. The constant evolution of cyber threats makes the implementation of robust and effective defense measures indispensable. In this context, filtering systems play a crucial role in controlling traffic flow and preventing unauthorized access. They constitute the first line of defense against intrusions, malicious attacks, and attempts to take control of systems, thereby ensuring data protection and operational-continuity.  
  
The threats to computer networks are diverse and constantly evolving, ranging from denial-of-service (DoS) attacks to sophisticated intrusion attempts orchestrated by malicious actors. In this threatening landscape, filtering systems do not merely block unwanted access; they analyze traffic in depth, apply complex security rules, and adapt security policies in real-time based on observed behaviors on the network. This ability to react proactively and adaptively is essential to maintaining a high level of security in the face of increasingly-complex-threats.  
  
**T**he importance of filtering systems also extends to the management of security policies within organizations. They allow for fine network segmentation, isolating different sections and thereby limiting the spread of attacks. This isolation contributes to the resilience of IT infrastructures, minimizing the potential impact of a security breach. Moreover, filtering systems play a key role in compliance with regulations and security standards, ensuring that data flows meet-legal-requirements-and-industry-best-practices.  
  
The implementation of effective filtering solutions relies on a thorough understanding of the available technologies and their respective capabilities. This project aims to explore two powerful and widely used filtering technologies: iptables on Linux and Cisco ASA in the Packet Tracer simulation environment.  
  
In summary, this project aims to provide a detailed and practical analysis of iptables filtering systems under Linux and Cisco ASA under Packet Tracer, highlighting their contributions to network security.

**Prerequisites**

To successfully carry out this project, several technical and conceptual prerequisites are necessary.

**1. Definition of firewall**  
  
A firewall is a software and/or hardware that controls internal and external traffic according to a certain security policy. Its main objective is to achieve security for the network we want to protect, and this is done through filtering. We therefore use the firewall as a filter that protects our network from another that may pose a threat, such as the Internet network, for example.  
  
**2. The different types of firewalls**  
  
**2.1 The firewall bridge**  
These are relatively widespread. They act like real network cables with the added function of filtering, hence their name firewall. Their interfaces do not have an IP address and only transfer packets from one interface to another while applying predefined rules. This absence is particularly useful, as it means that the firewall is undetectable to an average hacker. Indeed, when an ARP request is sent over the network cable, the firewall will never respond. Its MAC addresses will never circulate on the network, and since it only "forwards" the packets, it will be completely invisible on the network. This makes any attack directly targeting the firewall impossible, given that no packet will be processed by it as its own destination. So, the only way to bypass it is to override its drop rules. Any attack will therefore have to "deal" with its rules and try to circumvent them.  
  
In most cases, these have a separate configuration interface. A cable connects to a third interface, serial or even Ethernet, which should only be used occasionally and preferably in a secure environment.  
  
Bridged mode firewalls present both distinct advantages and disadvantages. On the advantage side, their positioning makes them indispensable: all packets must necessarily pass through their interfaces, thereby ensuring systematic control of network traffic. Moreover, this configuration is generally less expensive than other more complex solutions. However, bridge firewalls suffer from a few significant drawbacks. It is possible to bypass their security rules by circumventing the bridge itself, thus reducing their effectiveness. Moreover, the configuration of these firewalls is often considered cumbersome, requiring specific expertise to optimize their performance. Finally, their filtering features are often basic, mainly limited to control based on IP addresses and ports, and often stateless, which can limit their ability to manage complex network environments.

**2.2 Hardware Firewalls**  
  
They are often found on commercially purchased routers from major manufacturers like Cisco or Nortel. Integrated directly into the machine, they act as a "black box" and have perfect integration with the hardware. Their configuration is often relatively arduous, but their advantage is that their interaction with the other router features is simplified by their presence on the same network equipment. Often relatively inflexible in terms of configuration, they are also less vulnerable to attacks, as they are present in the "black box" that is the router. Moreover, being often very closely tied to the hardware, accessing their code is quite difficult, and the manufacturer has had complete freedom to produce "signed" code systems to authenticate the software (RSA system or similar). This system is only implemented in high-end firewalls, as it prevents the software from being replaced by another not produced by the manufacturer, or any modification of the latter, thus making the firewall very secure. Its administration is often easier than that of bridge firewalls, with major router brands using this argument as a selling point. Their level of security is also very good, except for the potential discovery of a flaw like any firewall. However, it should be noted that we are completely dependent on the hardware manufacturer for this update, which can be quite restrictive in some cases. Finally, only the specifications provided by the equipment manufacturer are implemented. This dependency means that if a feature interests us on a firewall from another brand, its use is impossible. It is therefore essential to clearly determine your needs in advance and carefully choose the router manufacturer.  
  
Hardware firewalls offer several distinct advantages. First of all, they are integrated into the network hardware, which simplifies their deployment and management within the existing infrastructure. Their administration is generally relatively simple, making them easy to use even for administrators with less expertise. In terms of security, these firewalls often ensure a good level of protection thanks to their advanced filtering and flow management capabilities.  
  
However, hardware firewalls also have significant drawbacks. They often depend on the manufacturer for software updates and security patches, which can lead to delays or difficulties in applying the latest protections against emerging threats. Moreover, these devices are sometimes less flexible than their software counterparts, which can limit their ability to quickly adapt to the evolving needs of modern networks and the specific requirements of end users.  
  
**2.3 Software Firewalls**  
  
Software firewalls are found in both servers and "homemade" routers, and can be classified into several distinct categories. First, personal firewalls, often commercial, aim to secure an individual computer rather than a group of computers. Although often paid and focused on ease of use, they can be limited in terms of security, sometimes making them easier to bypass. However, they offer end-point security for the client machine and can be relatively easily customized. On the other hand, their large number and variety sometimes make them difficult to evaluate and choose effectively. Next, more robust firewalls, often based on Linux like iptables, offer a high level of security and can be manually configured to mimic the behavior of hardware firewalls in routers. They are highly customizable and effective, but require additional system administration. However, a major flaw of these software firewalls is their inability to use the low network layer, potentially allowing packets to bypass the kernel and be retrieved by specific means, although this already requires an intrusion into the targeted computer, thus reducing their effectiveness under normal usage conditions.  
  
**3. The different types of filtering**  
  
A firewall allows defining trusted zones within a company and filtering access to these zones. We have three types of firewalls:  
  
**3.1 Stateless packet inspection firewalls**  
  
**3.2.1 The principle**  
  
A stateless firewall operates at the network and transport layers of the OSI model. It uses the information contained in the packet headers, such as source and destination addresses, source and destination port numbers, as well as other parameters, to assess potential threats. To configure it, it is necessary to use filtering rules called access control lists (ACLs), which de

Top of Form

**3.2 Stateful Packet Filtering  
  
3.2.1 The Principle**  
  
The improvement compared to simple filtering is the retention of session and connection traces in internal state tables within the Firewall. The Firewall then makes its decisions based on the states of connections and can react in the case of abnormal protocol situations. This filtering also helps protect against certain types of DoS attacks.  
  
**3.2.2 The limits**  
  
Stateful packet filtering, although effective for tracking and controlling the flow of network connections based on their state, has significant limitations. The management of the state table can become complex and resource-intensive, potentially affecting the firewall's performance during traffic spikes. Moreover, attacks aimed at saturating this state table, such as exhaustion attacks, can bypass the firewall's defenses. Although capable of blocking unauthorized traffic based on the connection state, stateful firewalls are not designed to inspect packet content in depth, which limits their ability to detect certain advanced threats.  
  
3.3 Application filtering (or proxy-type firewall or application proxying)  
  
**3.3.1  The principle**  
  
Application filtering, as its name suggests, is performed at the Application layer. For this, it is of course necessary to be able to extract the data from the level 7 protocol to study it. The requests are processed by dedicated processes, for example, an Http type request will be filtered by an Http proxy process. The firewall will reject all requests that do not comply with the protocol specifications. This implies that the proxy firewall must know all the protocol rules of the protocols it needs to filter.  
  
**3.3.2 The limits**  
  
The first problem that arises is the precision of the filtering performed by the proxy. It is extremely difficult to achieve filtering that lets nothing through, given the number of layer 7 protocols. Moreover, the requirement to know the protocol rules of each filtered protocol poses adaptability problems for new protocols or proprietary protocols.  
  
But it is undeniable that application filtering provides more security than stateful packet filtering, but it comes at the cost of performance. This excludes the use of a 100% proxy technology for high-traffic networks today. Nevertheless, in a few years, the technological problem will undoubtedly be resolved.

**Buts**  
  
The objectives of this project are varied and aim to provide an in-depth understanding and practical application of iptables filtering systems under Linux and Cisco ASA under Packet Tracer.  
  
**1. Main Objectives**  
- Understand in detail the filtering mechanisms of iptables and Cisco ASA (architecture, filtering tables, rule chains, security policies).  
  
- Implement practical configurations for iptables on Linux and Cisco ASA in Packet Tracer.  
  
- Compare the configuration and management between iptables and Cisco ASA.  
  
**2. Secondary Objectives**  
  
- Develop practical skills in configuring and managing firewalls under Linux and Cisco ASA.  
  
- Use Packet Tracer to simulate complex network environments.  
  
In summary, this project aims to provide a comprehensive and practical analysis of iptables filtering systems under Linux and Cisco ASA under Packet Tracer, while developing advanced technical skills and promoting effective network security practices. By achieving these objectives, we hope to contribute to the protection of digital infrastructures against current and future threats.

**Principle of operation of filtering systems**

The operating principle of iptables filtering systems under Linux and Cisco ASA under Packet Tracer relies on sophisticated mechanisms aimed at controlling network traffic flow and enhancing communication security. Here is how each of these technologies operates:  
  
**Iptables on Linux**  
  
Iptables is a component integrated into the Linux kernel, acting as a software firewall. Its operation is based on the manipulation of filtering rules applied to network packets that pass through the system. Here are the key principles of its operation:  
  
1. Filtering Tables: Iptables uses different tables (such as filter, nat, mangle) to organize filtering rules based on the types of traffic to be handled.  
  
2. Rule Chains: Each table contains chains where filtering rules are applied sequentially. The main chains include INPUT, OUTPUT, and FORWARD, which determine the direction of the traffic to be filtered.  
  
3. Filtering Criteria: The rules are defined based on criteria such as source/destination IP address, ports, protocols (TCP, UDP, ICMP), and other parameters of the network packet.  
  
4. Filtering Actions: Each rule specifies an action to take on packets that match the defined criteria: accept, reject, or redirect to another rule or a specific chain.  
  
5. NAT and Mangle Tables: In addition to the filter table, iptables uses the nat table to manage network address translation (NAT) and the mangle table to alter packet headers.  
  
Cisco ASA under Packet Tracer  
  
Cisco ASA is a hardware and software firewall developed by Cisco Systems. Under Packet Tracer, it is simulated as a virtual device, allowing its functionalities to be reproduced in a controlled environment. Here are the basic principles of its operation:  
  
1. Command Line Interface (CLI): The configuration of Cisco ASA is primarily done through a command line interface (CLI), similar to Cisco physical devices.  
  
2. Security Policies: Cisco ASA uses security policies to define how traffic should be filtered and allowed through the firewall.

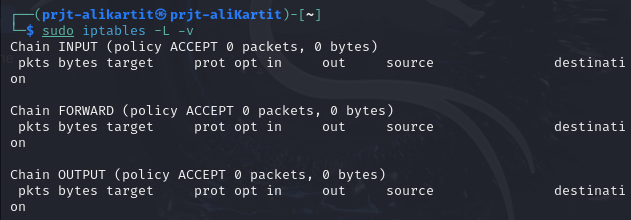
3. ACL (Access Control Lists): Access Control Lists (ACLs) are used to define filtering rules based on criteria similar to those of iptables: IP addresses, ports, protocols, etc.  
  
4. Packet Inspection: Cisco ASA includes advanced features such as packet inspection to identify and control communication sessions according to specific protocols.  
  
5. Advanced Security Features: In addition to basic filtering, Cisco ASA offers advanced features such as intrusion prevention (IPS), data loss detection and prevention, VPNs, and other integrated security services.  
  
In summary, both iptables under Linux and Cisco ASA under Packet Tracer operate by applying filtering rules based on specific criteria to network packets. Each technology has specific features and functionalities that make them suitable for different contexts and network security requirements.

**Chapter 2: Iptables on Linux**  
  
**2.1 General Overview of Iptables**  
Iptables is a packet filtering system integrated into the Linux kernel, which allows controlling the flow of incoming and outgoing network traffic. It is used to apply filtering rules, network address translation (NAT), and port redirection. Iptables works by inspecting the data packets that flow through the network, applying user-configured rules to decide how to handle them. The rules can be based on various criteria such as the source or destination address, the port, the protocol, etc. Iptables is a powerful and flexible tool, widely used to secure Linux systems and computer networks.  
  
**2.2 Syntax and Configuration Rules/Usage Examples and Use Cases**  
To ensure that the Linux system has iptables installed, if not, install it.

****

**Figure 1: ensure the installation of iptables on Linux**

To display the current rules if present.

****

**Figure 2: Display of current rules**

Refuse all incoming traffic to the server

****

**Figure 3: A rule to block incoming traffic to the server**

Refuse all traffic passing through the server

****

**Figure 4: A rule to block crossing traffic**

Allow all outgoing traffic

****

**Figure 5: A rule to block outgoing traffic**

Allow already established connections or relative connections

****

**Figure 6: A rule for authorizing already established or related connections**

****

**Figure 7: A rule for authorization**

Allow SSH traffic

****

**Figure 8: A rule for SSH traffic authorization**

Allow traffic HTTP

****

**Figure 9: A rule for HTTP traffic authorization**

Allow traffic HTTPS

****

**Figure 10: A rule for HTTPS authorization**

Allow ICMP ping

****

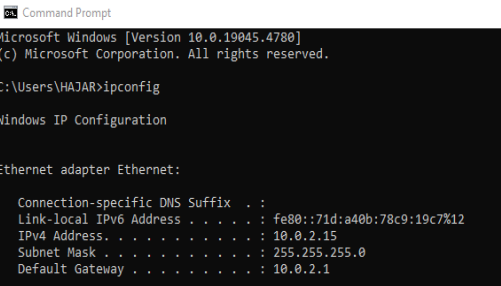
**Figure 11: A rule for ICMP ping authorization**

Block an IP address

****

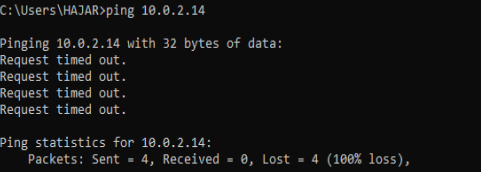
**Figure 12: A rule for blocking an address 10.0.2.15**

The machine that is stuck

****

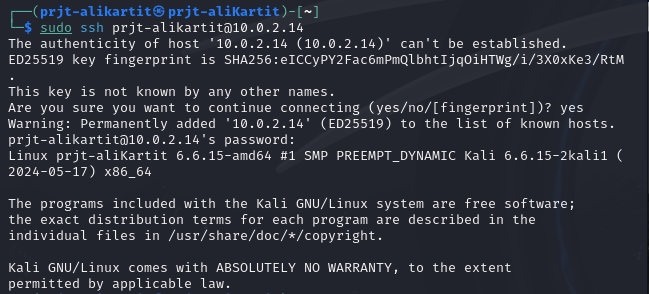
**Figure 13: IP address of the blocked machine**

The Testing Phase  
  
Test the IP blocking by attempting to connect from the blocked IP address.

****

**Figure 14: A test to test the blockage**

Access the server via SSH to verify that access is permitted.

****

**Figure 15: Verification of access via ssh**

### Contre-Mesures

### Iptables sous Linux

1. Limiting the rate of requests on SSH



**Figure 16: Limiting the rate of requests on SSH**

1. Use of fail2ban to block IPs after multiple unsuccessful login attempts

### 

**Figure 17: Blocking IPs after multiple unsuccessful login attempts**

2. It may be useful to configure logging rules to monitor blocked traffic. Adding a rule to log rejected packets

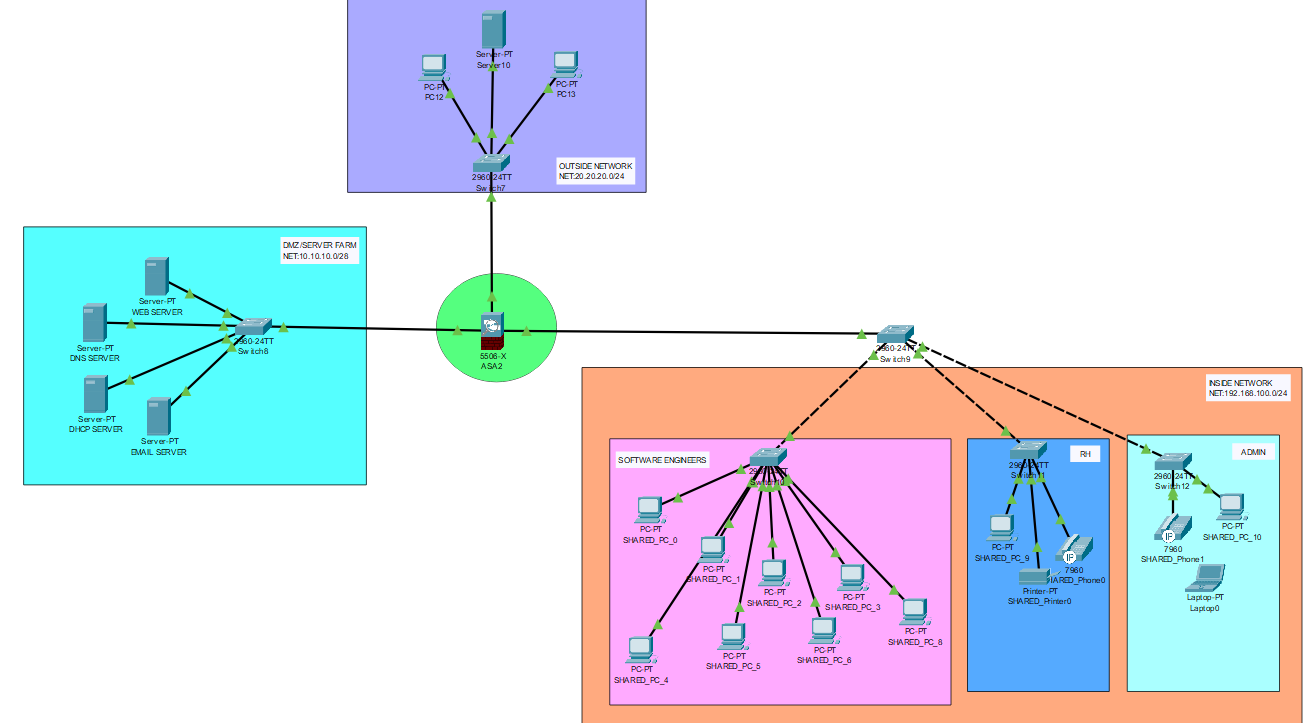
### 

**Figure 18: Configuration of Logging Rules**

**Chapter 3: ASA under Packet Tracer  
  
3.1 Presentation of the Adaptive Security Appliance (ASA)**  
  
The Adaptive Security Appliance (ASA) is a security device developed by Cisco, designed to protect networks against online threats. It offers a range of features such as intrusion detection and prevention, virtual private network (VPN) security, and protection against malware. ASA also uses advanced features such as Data Loss Prevention (DLP) and Advanced Application Security (ASA). It is widely used in enterprise environments to secure networks and protect sensitive data against attacks.  
  
**3.2 Project Description**  
  
The objective of the project is to develop and implement a comprehensive network security system for a fictitious small business. The objective is to establish a security infrastructure capable of protecting the company from external risks while ensuring a stable internal connection. To accomplish this, the project unfolds in several essential stages: assessment of security requirements, development of the network architecture including firewalls and secure subnets, configuration and deployment of firewalls on Linux servers, and finally, evaluation of resilience using simulated attack scenarios.  
  
**3.3 Analysis of the company's security needs**  
Network Segmentation: The different segments of the company's network include a DMZ zone, a zone dedicated to software engineers, as well as sections dedicated to the HR and Admin departments. It is essential to properly isolate each segment of the network in order to minimize the risks of unauthorized access. For example, it is recommended to separate essential servers in the DMZ, such as the DNS server, the web server, and the mail server, from the internal network to prevent any attempts of external intrusion.  
Access Control: For each segment of the network, it is essential to implement rigorous access control policies. It is preferable that software engineers only have access to the resources essential for their work. It is essential to strengthen access controls for the HR and Admin departments, which handle sensitive information, by using firewalls, VPNs, and multi-factor authentication (MFA) systems.  
Device Security: It is essential to ensure the security of all devices connected to the network, such as computers, printers, and phones. This includes regular maintenance of operating systems, installation of antivirus software, and the implementation of local firewalls. Shared devices, particularly in human resources and administrative services, must be carefully secured to prevent any unauthorized access to confidential information.  
Management of External Threats: It is essential to maintain constant monitoring of the network segment connected to the outside (Outside Network) in order to detect any suspicious activity. It is crucial to have a solid firewall and detection and prevention systems.

intrusion detection (IDS/IPS) to detect and prevent intrusion attempts. Furthermore, it is essential to strengthen the DMZ servers to protect them against DDoS (Distributed Denial of Service) attacks and other types of cyberattacks.

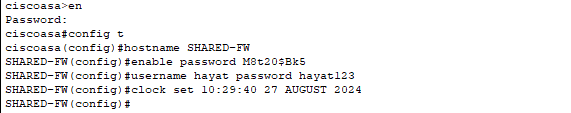
**3.4 Network Architecture Design**



**Figure 19: the network architecture**

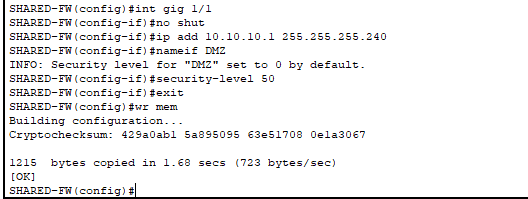
The detailed network diagram presented in this image highlights the organization and segmentation of an IT infrastructure. The network is subdivided into different segments, each with a different color corresponding to respective departments or functions. Let's take the example of the light blue segment, which represents a DMZ server farm, housing essential servers such as the web server, DNS, DHCP, and messaging, with an IP address range starting at 20.20.20.0/28. The HR and Admin departments are represented by the internal network, in orange, which uses the range 20.20.20.0/24, while the external network, in purple, uses the IP range of 292.268.200.0/24. In the center is a device identified as ASA2, probably a firewall or central router, which connects the different segments. The switches connect the various devices, such as computers, servers, phones, and printers, and the solid and dotted lines represent the physical and logical connections. This diagram provides an accurate overall view of the network organization and resource distribution.  
  
**3.5 Firewall Configuration**  
  
The implementation of firewall configurations is an essential step in creating the network security system. The implementation of specific rules on Linux servers allows for managing and filtering network traffic according to the particular requirements of the company. The goal is to establish a secure framework that prevents unauthorized intrusions while allowing legitimate exchanges.

It is necessary to start the firewall configuration by performing a simple initial setup to prepare the system for secure use. It is necessary to define a name for the firewall at this stage, which makes it easier to identify on the network. Subsequently, a username and password for administrative access are provided to secure the firewall settings against intrusions. Once these access credentials have been configured, it is necessary to set the clock and date to ensure that the event logs are correctly timestamped, which is essential for tracking activities and security incidents. Setting up these basic steps is crucial to create a solid foundation before performing more complex configurations.



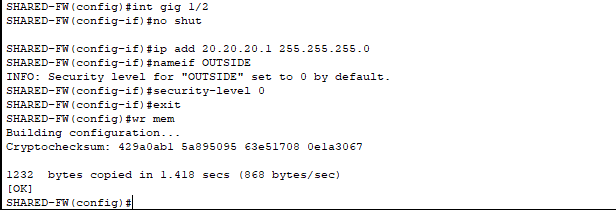
**Figure 20: Configuration basique**

Now, we configure the interfaces by giving the IP address, the name, and the security level, and finally, we save using 'WR MEM'.

La première interface représente la DMZ (zone démilitarisée) avec un niveau de sécurité de 50.

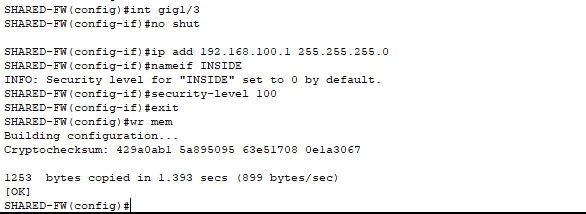
**Figure 21: Configuration of the gig1/1 interface**

The second interface represents the external zone with a security level of 0.



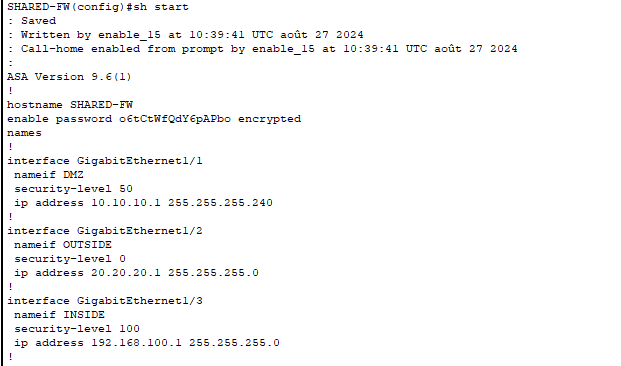
**Figure 22: Configuration of the gig1/2 interface**

The third interface represents the internal zone with a security level of 200.



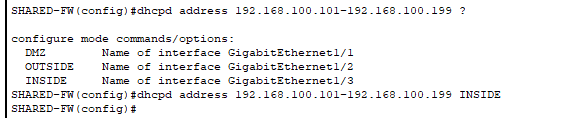
**Figure 23: Configuration of the gig1/3 interface**

The command show startup-config (or sh start in short) allows you to display the configurations of the firewall interfaces. The saved configuration parameters that will be loaded during the next system startup can be viewed using this command.



**Figure 24: The display of configurations**

In order to set up DHCP on the firewall, it is necessary to determine the number of IP addresses that will be assigned to the clients. Subsequently, the interface that will be linked to the DHCP service is chosen.



**Figure 25: Configuration of DHCP**

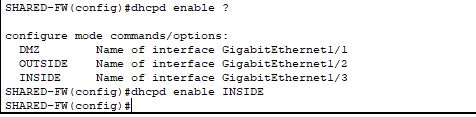
In order to set up the DNS (Domain Name System) service on the firewall, it is necessary to define the DNS servers to which client requests will be sent to resolve name issues.

.



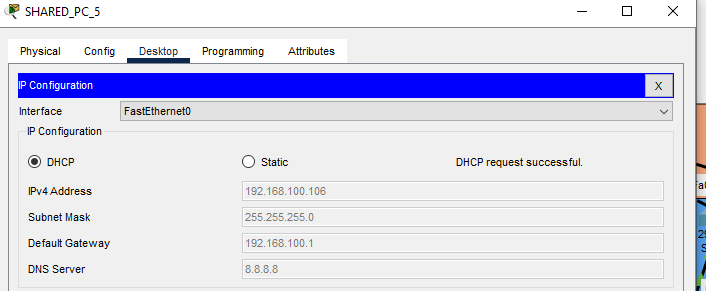
**Figure 26: Configuration of DNS**

It is essential to enable the DHCP service on a firewall's interfaces to facilitate IP address management in a network. Enabling DHCP allows the firewall to dynamically distribute IP addresses to devices that connect through the configured interfaces.

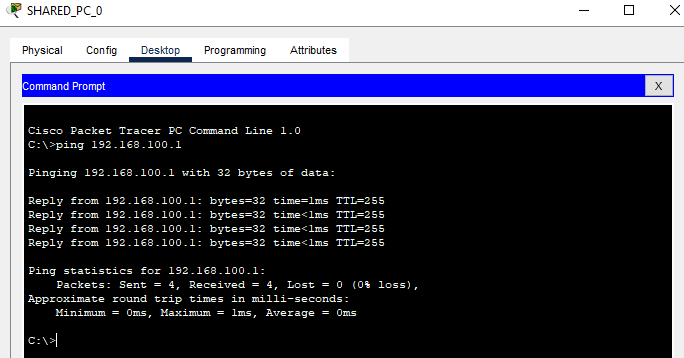


**Figure 27: Enabling DHCP on the interfaces**

Enable DHCP on the devices so they automatically obtain an IP address.

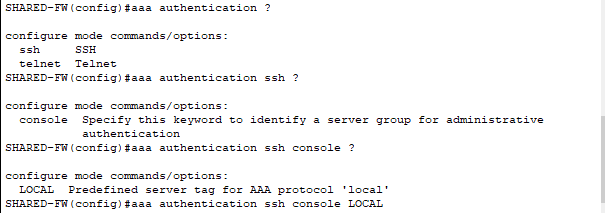
**Figure 28: Enabling DHCP on the computers**

We ensure that all devices can ping the firewall.



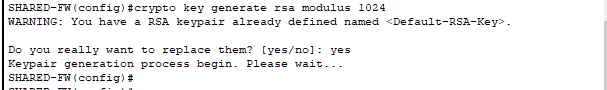
**Figure 29: Connection Verification**

Enable local AAA authentication to connect to the ASA firewall via SSH.



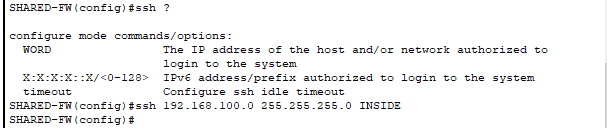
**Figure 30: Activation of LOCAL aaa authentication**

Generate an RSA key pair to secure SSH connections to the firewall.



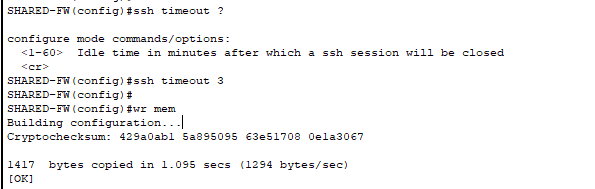
**Figure 31: RSA Key Generation**

Allow SSH on a Cisco ASA firewall.

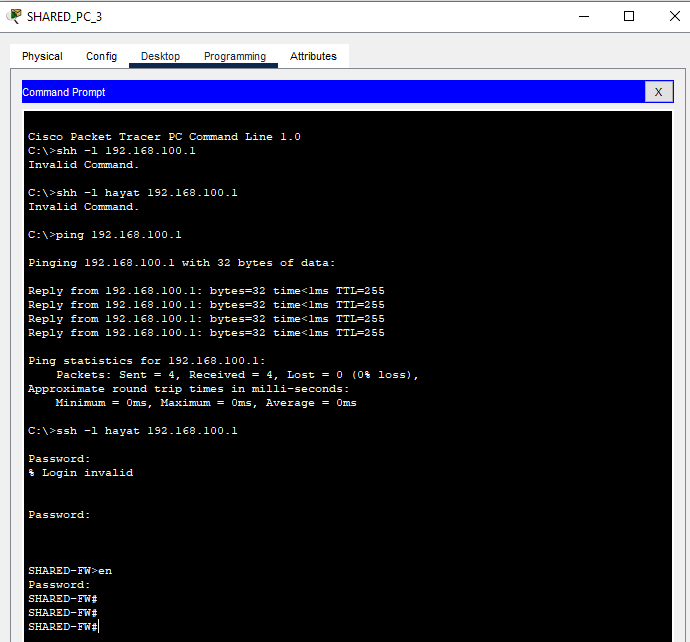


**Figure 32: Enable SSH**

Setting an expiration time allows for regulating the maximum duration during which an inactive SSH session remains open before being automatically disconnected.

**Figure 33 : SSH timeout**

Using an SSH client allows you to test the SSH connection to a Cisco ASA firewall using the appropriate credentials.



**Figure 34 : Test**

**Comparison and selection of filtering systems**  
  
In the context of this project, we studied two major filtering systems: Iptables under Linux and the Cisco ASA firewall. These two technologies offer robust solutions for managing and securing network traffic, but they have distinct characteristics that make them suitable for different environments and needs. Here is an in-depth comparison of these two solutions, followed by an analysis of the selection criteria to choose the most suitable system.  
  
**1. Iptables (Linux) Iptables**

Is an open-source packet filtering solution integrated into Linux systems. It is a flexible tool that allows configuring firewalls at a very granular level. Advantages:

- Cost: Free and open-source, making it an accessible solution for all businesses, big or small.

- Granular control: It offers detailed control over packets through tables and chains, allowing for fine manipulation of traffic.

- Community and support: A large community of Linux users and developers provides active support and numerous documentation resources.

- Disadvantages:

- Lack of native graphical interface: The absence of a simple-to-use graphical user interface makes managing large configurations more difficult for those who are not familiar with the command line.

- Decentralized administration: Unlike commercial solutions, managing security rules and configurations across multiple servers may require external tools to centralize administration.

**2.Cisco-ASA**  
  
The Cisco ASA is a hardware network security solution used by many companies to protect their networks against external and internal threats.  
  
**Advantages**  
  
- Centralized management: Cisco ASA offers centralized management tools through graphical interfaces like Cisco ASDM, allowing for simple and unified administration.

- Advanced features: In addition to packet filtering, Cisco ASA offers additional features such as VPNs, intrusion protection, and security policy management.

- Scalability: Cisco ASA is designed for enterprise environments with the ability to handle large amounts of network traffic while maintaining high performance.

**Disadvantages:**

- Cost: It is a relatively expensive solution, which includes hardware acquisition costs, software licenses, and support.- Hardware dependency: As a hardware solution, Cisco ASA is less flexible for companies looking for a virtual or cloud-based solution.

**3. Selection Criteria**

The choice between Iptables and Cisco ASA will mainly depend on the organization's needs in terms of costs, scalability, and the nature of the existing infrastructure. Here are some selection criteria to consider:

- Budget: If the company is looking for an economical solution, Iptables is the best option since it is free and open-source. Cisco ASA, on the other hand, involves significant costs, both in terms of hardware and maintenance.

- Ease of management: For companies without Linux expertise or those who prefer a user-friendly graphical interface, Cisco ASA is more suitable thanks to its centralized management tools. Iptables requires a certain level of command line expertise and is more difficult to manage without external tools.

- Size and complexity of the network: Cisco ASA is more suitable for large enterprises with advanced network security needs and centralized management. On the other hand, for small and medium-sized enterprises (SMEs) or environments requiring very fine packet control, Iptables can offer more flexibility at a lower cost.- Scalability and integration: Cisco ASA excels in environments where scalability and integration with other Cisco products are important. For a growing network with increasing performance and security needs, Cisco ASA offers a better long-term solution.So, the choice between these two solutions will depend on the specific objectives of the company. If the goal is to implement a low-cost network security solution with fine control and maximum flexibility, Iptables would be the preferred option. On the other hand, for a corporate infrastructure requiring centralized management and advanced features, Cisco ASA presents itself as a more robust and scalable solution.In conclusion, for a small or medium-sized enterprise (SME) with a primarily Linux infrastructure, Iptables would be a logical and effective choice. On the other hand, a company with complex needs, a large-scale infrastructure, and a sufficient budget would benefit from the advantages of a solution like Cisco ASA.

**Conclusion and Perspectives**  
  
This project allowed us to dive into the world of packet filtering, an essential component of network security. Through the use of Iptables on Linux and Cisco ASA in a simulation environment, we discovered the theoretical and practical foundations of these technologies. Packet filtering is a crucial first line of defense to protect systems against external threats, and the correct implementation of this technique is essential to ensure network security.  
  
The study of Iptables, a powerful and flexible tool, showed us how to define and manipulate filtering rules to effectively manage incoming and outgoing traffic. Thanks to its open-source nature, Iptables is particularly well-suited for Linux environments, offering a cost-effective solution while remaining highly efficient in network traffic control.  
  
On the other hand, exploring the Cisco ASA allowed us to approach packet filtering differently. This hardware firewall, used in enterprise network infrastructures, offers additional features such as centralized management, VPN, and seamless integration with other security solutions. Its ease of configuration and advanced options make it ideal for environments requiring a high level of security with granular policy control.  
  
The comparison between Iptables and Cisco ASA revealed distinct strengths of each technology. Iptables, although requiring some technical expertise for manual configuration, remains a lightweight and flexible solution for small infrastructures or those that prefer Linux systems. Cisco ASA, on the other hand, stands out in enterprise environments requiring centralized and scalable security features.  
  
By testing these two technologies, we not only learned to configure and adjust network security rules but also to validate our configurations through real-world scenarios. These steps allowed us to better understand how different types of traffic are handled by a firewall and how to strengthen security policies to minimize vulnerabilities.  
  
This project was an enriching practical exercise, allowing us to consolidate Thus, this mini-project represents a significant advancement in our cybersecurity training, preparing us to intervene effectively in various network environments, whether they are small infrastructures or large enterprises.

**References**  
  
ALICHE, R., & HENNI, D. (2016).Quality of Services in IPtables (Doctoral dissertation, Ibn Khaldoun University-Tiaret).  
  
Thileli, N., & Nassima, B. (2018).Filtering Network Traffic at Mouloud Mammeri University (Doctoral dissertation, Mouloud Mammeri University).  
  
RACHEDI, H. (2012).Implementation of a secure proxy firewall under Linux (Doctoral dissertation, Ibn Khaldoun University-Tiaret).  
  
Dawson, T., Bautts, T., & Purdy, G. N. (2005).Network Administration under Linux. O'Reilly Media, Inc.  
  
Kurniawan, D. E., Arif, H., Nelmiawati, N., Tohari, A. H., & Fani, M. (March 2019). Implementación y análisis de ipsec-vpn en el firewall Cisco ASA utilizando el simulador de red GNS3. En Journal of Physics: Conference Series (Vol. 1175, No. 1, p. 012031). IOP Publishing.  
  
Neumann, J. C. (2015).El libro de GNS3: construye laboratorios de redes virtuales usando Cisco, Juniper y más. No Starch Press.  
  
BENBRAHIM, K., & BEN SAADA, W. (2015).DESIGN OF LOCAL COMPUTER NETWORKS USING CISCO PACKET TRACER AND OPNET SOFTWARE (Doctoral dissertation).  
  
Ouazani Amel, B. H. (2021). Study and simulation of a LAN network developed using Cisco Packet Tracer.  
  
Diène, S. M. (2021). Design and implementation of a secure architecture for a multi-site enterprise network.  
  
Salim, A. (2016).Implementation of enterprise network security using ACLs (Doctoral dissertation, Mouloud Mammeri University).  
  
TRAORE, A. K., SOUABO, O., MarcNAKANNABO, M., & Sadouanouan, M. A. L. O. OF THE COMPUTER NETWORK OF THE SOFITEX HEADQUARTERS.  
  
RACHEDI, H. (2012).Implementation of a secure proxy firewall under Linux (Doctoral dissertation, Ibn Khaldoun University-Tiaret).