WEBSERVER SECURITY IMPLEMENTATION WITH DEMILITARIZED ZONE USING IAC IN VAGRANT

Group 4:

Kheyral Sutan Dumas (2340040301)

Faculty:

Mr. Ivan Firdaus S.T

Class:

3CS1

CEP CCIT FACULTY OF ENGINEERING

UNIVERSITY OF INDONESIA

2024

PROJECT INFORMATION

Project Title **:** Webserver Security Implementation with Demilitarized Zone Using IaC in Vagrant

Batch Code **:** 3CS1

Start Date **:** October 15, 2024

End Date **:** October 20, 2024

Name of Faculty **:** Mr. Ivan Firdaus S.T

Names of Administrator :

1. Kheyral Sutan Dumas
2. Charisma Bayu Majestyno

ACKNOWLEDGEMENT

The author would like to acknowledge the completion of the insightful paper entitled **"Webserver Security Implementation with Demilitarized Zone Using IaC In Vagrant."** This paper comprehensively discusses the development and configuration of a secure webserver environment utilizing a Demilitarized Zone (DMZ) and Vagrant, focusing on key aspects such as isolating internal networks, mitigating external threats, and automating environment setup.

However, the project is still far from perfect, as it contains several configuration issues, inconsistencies in firewall rules, and areas where security policies can be further refined. These imperfections reflect the iterative nature of system administration and the challenges faced during the implementation of complex network security structures like a DMZ.

Overall, the paper serves as a significant contribution to the growing body of knowledge on webserver security implementations using DMZ architecture with Vagrant, while recognizing the need for continued development and refinement.

Depok, 15 October 2024

SYSTEM ANALYSIS

The primary objective is to create a functional system that allows for the implementation of a secure webserver environment using a Demilitarized Zone (DMZ) and Vagrant. The system is designed to simplify webserver security by utilizing Vagrant for automated environment setup and configuration, ensuring a clear separation between internal and external networks. Users can configure firewall rules, isolate sensitive resources, and implement security policies based on predefined parameters like network segments, IP restrictions, and access control rules.

To enhance the system's functionality, several tools and Vagrant plugins have been incorporated, such as virtual networking configurations and automated provisioning scripts. These components help streamline various aspects of the deployment process while also supporting better error handling for security configurations.

However, as with any system, the implementation is not without flaws. There may be bugs or unforeseen configuration errors due to inconsistent rule sets or unhandled security exceptions. The current setup may also lack full robustness, requiring further debugging and testing to ensure that it operates effectively across different network environments.

PREPARATION

Before commencing with the development process, it is essential to prepare the necessary requirements:

1. Install Vagrant
https://developer.hashicorp.com/vagrant/install?product_intent=vagrant

2. Install Virtualbox 7.0.8
https://download.virtualbox.org/virtualbox/7.0.8/VirtualBox-7.0.8-156879-Win.exe

3. Install Visual Studio Codes
https://code.visualstudio.com/download

Keep in mind in this paper the authors using Windows 10 as the Operating System to develop the project, Readers can adjust the preparation based on their own Operating System.

CODES

```
Vagrant.configure("2") do |config|
  config.vm.box = "ubuntu/bionic64"

  config.vm.provider "virtualbox" do |vb|
    vb.memory = "1024"
  end

  # Restart network services to ensure correct configuration
  config.vm.provision "shell", inline: <<-SHELL
    sudo systemctl restart systemd-networkd
  SHELL

  # Firewall VM
  config.vm.define "firewall" do |firewall|
    firewall.vm.hostname = "router"

    # Internet Gateway (using correct bridge interface)
    firewall.vm.network "public_network", bridge: "Intel(R) Wireless-AC 9560"

    # Expose Port 80 to Public
    firewall.vm.network "forwarded_port", guest: 80, host: 8080, host_ip: "0.0.0.0"

    # DMZ network
    firewall.vm.network "private_network", ip: "192.168.20.21", virtualbox__intnet: "dmz_net"

    # Internal network
    firewall.vm.network "private_network", ip: "192.168.30.31", virtualbox__intnet: "int_net"

    # Provision firewall with iptables rules and routing setup
    firewall.vm.provision "shell", inline: <<-SHELL
      sudo -i

      # Preconfigure iptables-persistent to automatically save rules
      echo "iptables-persistent iptables-persistent/autosave_v4 boolean true" | sudo debconf-set-selections
      echo "iptables-persistent iptables-persistent/autosave_v6 boolean true" | sudo debconf-set-selections

      apt-get update
      apt-get full-upgrade -y
      apt-get install -y iptables-persistent

      # Enable IP forwarding
      sysctl -w net.ipv4.ip_forward=1
      echo "net.ipv4.ip_forward = 1" >> /etc/sysctl.conf

      # Clear existing rules
      iptables -F
      iptables -t nat -F
```

CODES

```
    # Set up NAT and forwarding rules
    iptables -t nat -A POSTROUTING -o enp0s8 -j MASQUERADE
    iptables -A FORWARD -i enp0s8 -o enp0s9 -m state --state RELATED,ESTABLISHED -j ACCEPT
    iptables -A FORWARD -i enp0s9 -o enp0s8 -j ACCEPT
    iptables -A FORWARD -i enp0s8 -o enp0s10 -m state --state RELATED,ESTABLISHED -j ACCEPT
    iptables -A FORWARD -i enp0s10 -o enp0s8 -j ACCEPT

    # Allow HTTP to DMZ
    iptables -A FORWARD -i enp0s8 -o enp0s9 -p tcp --dport 80 -j ACCEPT
    iptables -t nat -A PREROUTING -i enp0s8 -p tcp --dport 80 -j DNAT --to-destination 192.168.20.22:80

    # Save iptables rules
    netfilter-persistent save

    # Add Default Routing for Internet Gateway
    while ! ip -4 addr show enp0s8 | grep -q "inet "; do
      echo "Waiting for enp0s8 to be initialized..."
      sleep 1
    done

    # Delete the current default route
    ip route del default

    # Dynamically fetch the default gateway from the routing table
    GATEWAY_IP=$(ip route | grep default | grep enp0s8 | awk '{print $3}')

    # Check if a gateway IP was successfully extracted
    if [ -n "$GATEWAY_IP" ]; then
      echo "Gateway IP set to: $GATEWAY_IP"
      # Add the new default route through the extracted gateway IP
      ip route add default via $GATEWAY_IP dev enp0s8
    else
      echo "Error: Failed to extract gateway IP."
      exit 1
    fi
  SHELL
end
```

CODES

```
  # DMZ Web Server VM
  config.vm.define "dmz" do |dmz|
    dmz.vm.hostname = "dmz"
    dmz.vm.network "private_network", ip: "192.168.20.22", virtualbox__intnet: "dmz_net"

    dmz.vm.provision "shell", inline: <<-SHELL
      sudo -i
      apt-get update
      apt-get full-upgrade -y
      apt-get install -y apache2
      echo "<h1>Welcome to the DMZ Web Server</h1>" > /var/www/html/index.html
      systemctl start apache2 && systemctl enable apache2
      ip route del default
      ip route add default via 192.168.20.21 dev enp0s8
    SHELL
  end

  # Internal Server VM
  config.vm.define "internal" do |internal|
    internal.vm.hostname = "internal"
    internal.vm.network "private_network", ip: "192.168.30.32", virtualbox__intnet: "int_net"

    internal.vm.provision "shell", inline: <<-SHELL
      sudo -i
      apt-get update
      apt-get full-upgrade -y
      ip route del default
      ip route add default via 192.168.30.31 dev enp0s8
    SHELL
  end
end
```

SIMULATION

Accessing DMZ Webserver from Host, Internal, and Firewall network with curl command.

```
vagrant@dmz:~$ sudo tail -f /var/log/apache2/access.log
192.168.20.21 - - [22/Oct/2024:03:19:46 +0000] "HEAD / HTTP/1.1" 200 227 "-" "curl/7.58.0"
192.168.30.32 - - [22/Oct/2024:03:22:23 +0000] "HEAD / HTTP/1.1" 200 227 "-" "curl/7.58.0"
192.168.18.5 - - [22/Oct/2024:03:23:23 +0000] "HEAD / HTTP/1.1" 200 227 "-" "curl/7.83.1"
```

192.168.20.21 Firewall Interface to DMZ

192.168.30.32 Internal Virtual Machine

192.168.18.5 Host OS

REQUIREMENTS

Hardware :

1. Lenovo V14 G2

Operating System :

1. Windows 10 64-bit

Software :

1. Vagrant Latest Version
2. Virtualbox 7.0.8
3. VSCode Latest Version

PROJECT FILE DETAILS

No	Filename	Remarks
1	3CS1 Project 2.pdf	Microsoft Words contain documentation paper about the project
2	Vagrantfile	Files contains the source codes
3	Project 2 Presentation.pptx	Presentation file