Universidad San Carlos de Guatemala
Centro Universitario de Occidente

Sistema de Base de Datos 2

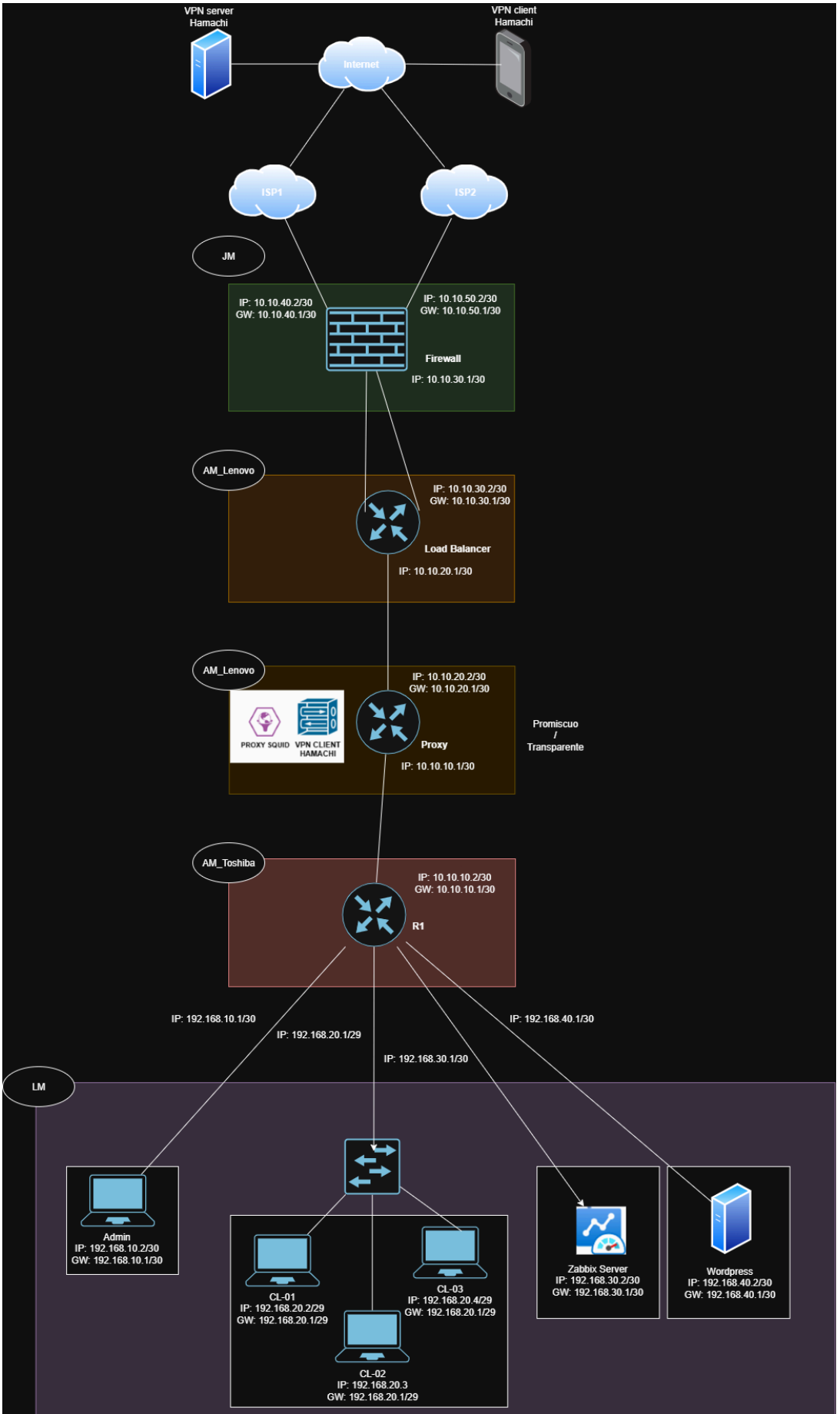
Aux. Manuel Rojas
Ing. Francisco Rojas

Documentación Arquitectura de la Solución

Carné	Nombre
202031794	Luis Antonio Monterroso Guzman
201930967	William Alexander Miranda Santos
201930643	Jeffrey Kenneth Menendez Castillo
201731874	Sofía Alejandra Quintana Gutiérrez
202131284	José Roberto Bautista Rojas

Arquitectura de la solución

Redes22025-Diseño de Arquitectura de Red.png



Router 1

Pasos para iniciar el servicio

Al iniciar la máquina, se deben ejecutar los siguientes scripts en el orden indicado:

Ejecutar el script de control de acceso:

```
./access-control.sh
```

1. Este script aplica las políticas de control de acceso basadas en las listas de IPs y MACs permitidas.

Ejecutar el script de inicialización del router:

```
./initr.sh
```

2. Este script configura las interfaces, VLANs y rutas internas según los archivos de configuración definidos.

Modificación de configuraciones

Para realizar cambios en los archivos de entrada o en la configuración del router, debe navegar hacia el siguiente directorio:

```
cd /root/r1
```

En esta carpeta se encuentran los archivos principales de configuración:

- VLANs.conf — Definición de VLANs e interfaces LAN.
- access.mac — Lista de direcciones MAC permitidas.
- access.ip — Lista de direcciones IP autorizadas.

Resolvedor de DNS

La resolución de nombres de dominio dentro de la red se realiza a través del router principal (**R1**), utilizando la herramienta **dnsmasq** como resolvedor de DNS local.

Para ello, se instaló el paquete correspondiente y se configuró adecuadamente el archivo de configuración ubicado en:
/etc/dnsmasq.conf

```
# Escuchar solo en las interfaces Vlans
interface=enp1s0.10
interface=enp1s0.20
interface=enp1s0.30
interface=enp1s0.40

# IPs del router que atenderán peticiones DNS
listen-address=127.0.0.1,192.168.10.1,192.168.20.1,192.168.30.1,192.168.40.1

# Define los DNS externos manualmente
no-resolv
server=8.8.8.8
server=1.1.1.1
```

Configuración del Router 1 cómo resolvedor DNS para los clientes

Para establecer el Router 1 como servidor DNS de los equipos cliente, se deben seguir los siguientes pasos:

En cada cliente (máquina donde se desee configurar), abrir el archivo de configuración de resolución DNS

```
sudo nano /etc/resolv.conf
```

Eliminar todo su contenido y agregar la siguiente línea:

```
nameserver 192.168.20.1
```

1. *(donde 192.168.20.1 corresponde a la dirección IP del Router 1).*
2. Guardar los cambios y cerrar el archivo.

Para evitar que este archivo sea sobrescrito automáticamente por otros servicios del sistema (como dhclient o NetworkManager), se recomienda bloquearlo con el siguiente comando

```
sudo chattr +i /etc/resolv.conf
```

Con esto, el cliente utilizará de forma permanente al Router 1 cómo resolvedor DNS.

Máquina Proxy

Squid

Para poder filtrar las peticiones HTTP y HTTPS, se instaló y configuró el proxy Squid en modo transparente.

La instalación se realizó mediante el siguiente comando:

```
sudo apt install squid -y
```

Configuración de HTTP

La configuración del servicio HTTP en modo transparente se basa en definir el puerto de escucha y los parámetros necesarios dentro del archivo principal de Squid:

```
/etc/squid/squid.conf
```

En este archivo se agregan las siguientes líneas:

```
http_access deny all
http_port 3128 intercept
```

Configuración de HTTPS

El filtrado de tráfico HTTPS se implementó utilizando la funcionalidad SSL Bump de Squid. Este mecanismo permite inspeccionar y filtrar conexiones cifradas para aplicar las mismas políticas de control que en el tráfico HTTP.

Instalar soporte SSL para Squid

```
sudo apt install squid-openssl -y
```

Generar el certificado SSL
Crear una autoridad certificadora (CA) local para que Squid pueda generar certificados dinámicamente:

```
sudo mkdir -p /etc/squid/ssl_cert
sudo openssl req -new -newkey rsa:4096 -days 1095 -nodes -x509 \
```

```
  -keyout /etc/squid/ssl_cert/myCA.key \
  -out /etc/squid/ssl_cert/myCA.crt \
  -subj "/C=GT/ST=Quetzaltenango/L=Xela/O=Speedwagon
Foundation/OU=IT/CN=Speedwagon Root CA"
```

Configurar el archivo /etc/squid/squid.conf

Agregar las siguientes líneas para habilitar el filtrado HTTPS y definir las reglas SSL Bump:

```
 # --- Puerto intercept HTTPS ---
https_port 3129 intercept ssl-bump \
   cert=/etc/squid/ssl_cert/myCA.crt key=/etc/squid/ssl_cert/myCA.key

# --- Reglas SSL Bump ---
acl step1 at_step SslBump1
ssl_bump peek step1
ssl_bump splice all

# --- Seguridad ---
sslcrtd_program /usr/lib/squid/security_file_certgen -s /var/lib/ssl_db -M 4MB
sslcrtd_children 5

# --- ACLs de acceso ---
acl localnet src 192.168.20.0/24
http_access allow localnet
http_access allow localhost
http_access deny all
```

Inicializar la base de datos de certificados

Si la base de datos /var/lib/ssl_db no existe o no ha sido inicializada, crearla con:

```
sudo /usr/lib/squid/security_file_certgen -c -s /var/lib/ssl_db -M 4MB
```

Otorgar permisos al usuario de Squid

```
sudo chown -R proxy:proxy /var/lib/ssl_db
```

Reiniciar el servicio de Squid

```
sudo systemctl restart squid
```

Instalar el certificado en los clientes

El archivo generado /etc/squid/ssl_cert/myCA.crt debe instalarse manualmente en las computadoras cliente para evitar advertencias de seguridad.

En Firefox, se realiza desde:

```
Configuración → Privacidad y seguridad → Certificados → Ver certificados → Importar
```

Router

El Router 1 se configuró para detectar automáticamente las conexiones salientes desde la red de clientes hacia los puertos 80 (HTTP) y 443 (HTTPS).
Cuando se identifica un paquete con alguno de estos destinos, el router redirige el tráfico hacia los puertos donde Squid está escuchando, permitiendo así el funcionamiento del proxy transparente.

Las reglas implementadas en `iptables` son las siguientes:

```
# --- Redirigir solo tráfico HTTP hacia el Proxy ---
iptables -t nat -A PREROUTING -i enp1s0.20 -p tcp --dport 80 -j DNAT
--to-destination 10.10.10.1:3128
iptables -t nat -A PREROUTING -i enp1s0.20 -p tcp --dport 443 -j DNAT
--to-destination 10.10.10.1:3129
```

Bloqueo de dominios

sudo mkdir -p /etc/squid/lists

Instalación de Wordpress

Preparación del entorno

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y curl wget unzip tar vim ufw
```

Instalar servidor web Apache

```
sudo apt install -y apache2
```

Habilitar el servicio

```
sudo systemctl enable apache2
sudo systemctl start apache2
```

Verificar que funcione

```
sudo systemctl status apache2
```

Instalar MariaDB

```
sudo apt install -y mariadb-server mariadb-client
```

Iniciar y habilitar el servicio

```
sudo systemctl enable mariadb
sudo systemctl start mariadb
```

Crear base de datos y usuario para WordPress

Ingresar a MariaDB

```
sudo mysql -u root -p
```

Dentro del prompt de MySQL

```
CREATE DATABASE wordpress;
CREATE USER 'wp_user'@'localhost' IDENTIFIED BY "123";
GRANT ALL PRIVILEGES ON wordpress.* TO 'wp_user'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

Instalar PHP y extensiones necesarias

```
sudo apt install -y php php-mysql php-curl php-xml php-gd php-mbstring php-zip
libapache2-mod-php
```

Verificar versión

```
php -v
```

Reiniciar Apache

```
sudo systemctl restart apache2
```

Descargar y configurar WordPress

Descargar la última versión

```
cd /tmp
wget https://wordpress.org/latest.tar.gz
tar -xvzf latest.tar.gz
```

Mover los archivos al directorio web

```
sudo mv wordpress /var/www/html/
```

Asignar permisos adecuados

```
sudo chown -R www-data:www-data /var/www/html/wordpress
sudo chmod -R 755 /var/www/html/wordpress
```

Configurar archivo wp-config.php

Copiar el archivo de ejemplo

```
cd /var/www/html/wordpress
sudo cp wp-config-sample.php wp-config.php
```

Editar el archivo

```
sudo nano wp-config.php
```

Modificar las siguientes líneas

```
define( 'DB_NAME', 'wordpress' );
define( 'DB_USER', 'wp_user' );
define( 'DB_PASSWORD', 'TuContraseñaSegura' );
define( 'DB_HOST', 'localhost' );
```

Configurar Apache para WordPress

Crear un VirtualHost

```
sudo nano /etc/apache2/sites-available/wordpress.conf
```

Agregar el siguiente contenido

```
<VirtualHost *:80>
    ServerAdmin admin@example.com
    DocumentRoot /var/www/html/wordpress
    ServerName wordpress.local

    <Directory /var/www/html/wordpress>
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/wordpress_error.log
    CustomLog ${APACHE_LOG_DIR}/wordpress_access.log combined
</VirtualHost>
```

Habilitar el sitio y módulos necesarios

```
sudo a2ensite wordpress.conf
sudo a2enmod rewrite
sudo systemctl reload apache2
```

Acceder a la instalación web

En el navegador

http://<IP_del_servidor>/wordpress

Para administración

http://<IP_del_servidor>/wordpress/wp-admin

Explicación del Funcionamiento Solución

sudo mkdir -p /etc/squid/lists