FACULTAD DE
INGENERÍAS

Clasificación de Malware mediante Análisis de Imágenes

Sistemas Inteligentes II
Proyecto Final: Deep Learning para Detección de Familias de Malware

Over Haider Castrillon Valencia

Juan David Diaz Castaño

Juan Carlos Saldarriaga Urrea

Facultad de Ingeniería
Ingeniería en Sistemas y Computación
Sistemas Inteligentes II

Manizales Caldas, Noviembre 18 de 2025

This page intentionally left blank.

FACULTAD DE
INGENERÍAS

Clasificación de Malware mediante Análisis de Imágenes

Over Haider Castrillon Valencia

Juan David Diaz Castaño

Juan Carlos Saldarriaga Urrea

Docente: Jorge Alberto Jaramillo Garzón

Facultad de Ingeniería
Ingeniería en Sistemas y Computación
Sistemas Inteligentes II

Manizales Caldas, Noviembre 18 de 2025

This page intentionally left blank.

Índice general

This page intentionally left blank.

1

Instalación de Hotspot Inalámbrico con Portal Cautivo

1.1. Introducción

El propósito de este documento es demostrar cómo convertir un computador o laptop basado en Linux en un hotspot inalámbrico donde los usuarios pueden autenticarse mediante una página de portal cautivo. Para esta tarea, el software controlador principal será **CoovaChilli**. Este software es una solución ideal de gestión de hotspots para hoteles, restaurantes, supermercados, parques y cualquier lugar que ofrezca Internet WiFi.

1.2. Prerrequisitos

- Una distribución Linux. En este artículo se utilizará Fedora 20. Las versiones posteriores 21/22 deberían funcionar bien.
- Bibliotecas de desarrollo necesarias para compilaciones de paquetes fuente.
- Una instalación funcional del servidor MySQL.
- Un dispositivo de red cableado que se conecte a Internet.
- Capacidad para ejecutar comandos sudo.
- Un dispositivo de red inalámbrico que soporte el modo Access Point (AP).

Para verificar si su dispositivo inalámbrico soporta el modo AP, ejecute:

```
sudo iw phy |grep -A 5 -i 'Supported interface modes' | grep '*'
```

1.3. Instalación de Dependencias de CoovaChilli

```
yum install libnl3-devel libtalloc-devel iptables
```

1.4. Instalación de hostapd

1.4.1. Descripción

Hostapd permite que su computadora funcione como un Punto de Acceso (AP) y Autenticador WPA/WPA2. Otras funcionalidades incluyen servicios de autenticación Radius, aunque no las usaremos aquí.
La mayoría de distribuciones Linux (incluyendo Fedora) tienen versiones pre-empaquetadas de hostapd que pueden instalarse usando el software de gestión de paquetes. Por ejemplo, en Fedora, CentOS y otras distribuciones Linux basadas en Red Hat, un comando simple instalará este paquete:

```
yum install hostapd
```

Sin embargo, para instalar la versión más reciente de hostapd, necesitaremos descargar y compilar las fuentes:

```
cd /usr/src
sudo git clone git://w1.fi/hostap.git
```

Esto descargará tanto hostapd (el daemon del servidor) como las fuentes de wpa_supplicant. Nos interesa el primero, así que cambiaremos a las fuentes de hostapd:

```
cd hostap/hostapd
```

Hostapd no tiene un comando 'configure', así que antes de compilar hostapd, necesitamos cambiar el prefijo de instalación. Una forma rápida y simple de cambiar el directorio de instalación predeterminado es usando sed:

```
sed -i "s:export BINDIR ?= /usr/local/bin/:export BINDIR ?=
   /usr/sbin:g" Makefile
```

A continuación, copie el archivo de configuración predeterminado:

```
cp -v defconfig .config
```

Necesitaremos cambiar algunos valores predeterminados en el archivo de configuración:

```
vim .config
```

Descomente las siguientes opciones:

```
CONFIG_LIBNL32=y            # Use libnl 3.2 libraries
CONFIG_IEEE80211N=y         # Enables IEEE 802.11n support
CONFIG_WNM=y                # Enables Network Management support
CONFIG_IEEE80211AC=y        # Enables IEEE 802.11ac support
CONFIG_DEBUG_FILE=y         # Support for writing debug log to file
```

Ejecute make e instale:

```
make
sudo make install
```

Para verificar si hostapd está correctamente instalado, ejecute:

```
hostapd -v
```

El comando anterior mostrará la versión y los créditos.

1.4.2. Configuración de hostapd

Cree el archivo de configuración de hostapd usando el archivo de ejemplo:

```
sudo mkdir /etc/hostapd
sudo cp -v /usr/src/hostap/hostapd/hostapd.conf /etc/hostapd/
sudo vim /etc/hostapd/hostapd.conf
```

Cambie los siguientes parámetros en el archivo hostapd.conf:

```
driver=nl80211
interface=wlan0                    # Change this to your wireless
    device
ssid=KAMPALA-3                     # Change this to your SSID
hw_mode=g
channel=6                          # Enter your desired channel
ieee80211n=1                       # Enable IEEE 802.11n
wpa=1
wpa_passphrase=myverysecretpassword
wpa_pairwise=TKIP CCMP
rsn_pairwise=CCMP
```

Cree el directorio para los sockets de hostapd:

```
sudo mkdir /var/run/hostapd
```

Configure el estado de la interfaz WiFi en 'UP' y desbloquee WiFi si el interruptor suave está activado:

```
sudo rfkill unblock wifi
sudo ip link set dev wlan0 up
```

Pruebe e inicie hostapd:

```
sudo hostapd -d /etc/hostapd/hostapd.conf
```

Si todo va bien, el daemon hostapd debería iniciarse y no cerrarse.
A continuación, cree un archivo de servicio systemd. La mayoría de distribuciones Linux ahora usan systemd para controlar servicios:

```
sudo vim /etc/systemd/system/hostapd.service
```

```
[Unit]
Description=Hostapd IEEE 802.11 AP, IEEE
    802.1X/WPA/WPA2/EAP/RADIUS Authenticator
After=dnsmasq.service
```

```
4
5 [Service]
6 Type=forking
7 PIDFile=/var/run/hostapd.pid
8 ExecStartPre=/bin/mkdir -p /var/run/hostapd
9 ExecStart=/usr/sbin/hostapd /etc/hostapd/hostapd.conf -P
     /var/run/hostapd.pid -B
10
11 [Install]
12 WantedBy=multi-user.target
```

Habilite el servicio para que se inicie automáticamente al arrancar:

```
1 sudo systemctl enable hostapd.service
```

1.5. Instalación de FreeRadius

Dado que requeriremos que los clientes se autentiquen antes de acceder a Internet, se necesita la instalación de un servidor radius. FreeRadius es un servidor radius de código abierto. También puede instalarse usando su gestor de paquetes Linux favorito como yum o apt. Pero dado que queremos instalar la versión más reciente, compilaremos desde las fuentes.

1.5.1. Descarga de FreeRadius

```
1 cd /tmp/
2 wget -c ftp://ftp.freeradius.org
3 /pub/freeradius/freeradius-server-3.0.9.tar.bz2
```

Desempaquete las fuentes y cambie a la ubicación de instalación:

```
1 sudo tar jxvf freeradius-server-3.0.9.tar.bz2 -C /usr/src/
2 cd /usr/src/freeradius-server-3.0.9
```

Ejecute el script configure asegurándose de usar el prefijo y la ruta de biblioteca correctos para su configuración:

```
1 sudo ./configure --prefix=/usr --libdir=/usr/lib64
     --sysconfdir=/etc \
2   --localstatedir=/var/ --enable-fast-install=no
```

Proceda a compilar e instalar:

```
1 sudo make
2 sudo make install
```

Si encuentra el siguiente error:

```
mkdir: cannot create directory '/etc/raddb/': File exists
make: *** [/etc/raddb/] Error 1
```

Ejecute lo siguiente para solucionarlo:

```
rmdir /etc/raddb
make install && make install
```

Agregue el grupo y usuario radiusd:

```
sudo groupadd -r radiusd
sudo useradd -r -M -c "Radius Server User" -g radiusd radiusd -s
   /sbin/nologin
```

1.5.2. Configuración de Tablas MySQL de FreeRadius

Inicie el servidor MySQL si no está en ejecución. Como se mencionó anteriormente, el proceso de inicialización es vía systemd, así que:

```
sudo systemctl -q is-active mysqld.service || sudo systemctl
   start mysqld.service
```

Asegúrese de que el servicio se inicie incluso al arrancar:

```
sudo systemctl enable mysqld.service
```

Cree la base de datos radius:

```
mysqladmin -u root -p[MYSQL_ROOT_PASSWORD] create radius
```

Genere las tablas de base de datos usando el esquema MySQL:

```
sudo cat /etc/raddb/mods-config/sql/main/mysql/schema.sql | \
  mysql -u root -p[MYSQL_ROOT_PASSWORD] radius
```

Cree el usuario MySQL radius y establezca privilegios en la base de datos radius:

```
mysql -u root -p[MYSQL_ROOT_PASSWORD] radius

GRANT ALL PRIVILEGES ON radius.* to
   [FREERADIUS_DB_USER]@localhost \
  IDENTIFIED by '[FREERADIUS_DB_PASS]';
```

1.5.3. Configuración del Módulo SQL de Radius

```
sudo vim /etc/raddb/mods-available/sql
```

Descomente y/o cambie los siguientes parámetros:

```
driver = "rlm_sql_mysql"
dialect = "mysql"
server = "localhost"
port = 3306
login = "FREERADIUS_DB_USER"
password = "FREERADIUS_DB_PASS"
read_clients = yes
```

Agregue contadores SQL de chillispot:

```
sudo vim /etc/raddb/mods-available/sqlcounter
```

Agregue esta línea al final del archivo anterior:

```
$INCLUDE ${modconfdir}/sql/counter/
${modules.sql.dialect}/chillispot.conf
```

A continuación, enlace sql, sqlcounter a módulos disponibles:

```
sudo ln -s /etc/raddb/mods-available/sql
   /etc/raddb/mods-enabled/sql
sudo ln -s /etc/raddb/mods-available/sqlcounter
   /etc/raddb/mods-enabled/sqlcounter
```

1.5.4. Configuración de Clientes Radius

```
sudo vim /etc/raddb/clients.conf
```

Cambie la contraseña a la contraseña usada anteriormente para la base de datos MySQL de FreeRadius:

```
secret = [FREERADIUS_DB_PASS]
```

1.5.5. Configuración del Servidor Radius

```
sudo vim /etc/raddb/radiusd.conf
```

En la sección de seguridad, cambie el usuario y grupo al nombre creado durante la instalación:

```
user = radiusd
group = radiusd
allow_vulnerable_openssl = yes
```

IMPORTANTE: No haga esto. Realmente debería actualizar a versiones recientes de OpenSSL.
En la sección instantiate (cerca de la línea 728), agregue los siguientes módulos de contador:

```
chillispot_max_bytes
noresetcounter
```

1.5.6. Configuración del Servidor Virtual Predeterminado

Configure el servidor virtual predeterminado bajo sites-available:

```
sudo vim /etc/raddb/sites-available/default
```

En la sección authorize:
Comente lo siguiente:

```
#filter_username
```

Descomente lo siguiente:

```
auth_log
unix
```

Cambie lo siguiente:

```
'-sql' to sql
```

Agregue lo siguiente al final de la sección authorize:

```
chillispot_max_bytes
noresetcounter
```

A continuación, en la sección accounting, descomente lo siguiente:

```
radutmp
```

Cambie lo siguiente:

```
'-sql' to sql
```

A continuación, en la sección session, descomente lo siguiente:

```
radutmp
sql
```

A continuación, en la sección post-auth, descomente lo siguiente:

```
reply_log
```

Cambie lo siguiente:

```
'-sql' to sql
```

1.5.7. Configuración de Inner Tunnel

Configure el servidor virtual de solicitudes de túnel interno bajo sites-available:

```
sudo vim /etc/raddb/sites-available/inner-tunnel
```

En la sección authorize, cambie lo siguiente:

```
'-sql' to sql
```

Agregue lo siguiente al final de la sección authorize:

```
chillispot_max_bytes
noresetcounter
```

A continuación, en la sección session, descomente lo siguiente:

```
sql
```

A continuación, en la sección post-auth, descomente lo siguiente:

```
reply_log
```

Cambie lo siguiente:

```
'-sql' to sql
```

1.5.8. Contadores MySQL para Chillispot

Agregue los siguientes contadores MySQL para Chillispot:

```
sudo vim /etc/raddb/mods-config/sql/counter/mysql/chillispot.conf
```

```
sqlcounter chillispot_max_bytes {
    counter_name = Max-Total-Octets
    check_name = ChilliSpot-Max-Total-Octets
    reply_name = ChilliSpot-Max-Total-Octets
    reply_message = "You have reached your bandwidth limit"
    sql_module_instance = sql
    key = User-Name
    reset = never
    query = "SELECT IFNULL((SUM(AcctInputOctets +
       AcctOutputOctets)),0) \
            FROM radacct WHERE username = '%{${key}}' \
            AND UNIX_TIMESTAMP(AcctStartTime) + AcctSessionTime
               > '%%b'"
}
```

Cambie la propiedad de los directorios de configuración y registro:

```
sudo touch /var/log/radius/radutmp
sudo chown -R radiusd:radiusd /etc/raddb
sudo chown -R radiusd:radiusd /var/log/radius
```

1.5.9. Creación de Usuario Admin

Cree un usuario Admin en la base de datos MySQL de radius:

```
echo "INSERT INTO radcheck (UserName, Attribute, Value, Op) \
      VALUES ('[ADMIN_USER]', 'Cleartext-Password',
         '[ADMIN_PASSWORD]', ':=');" | \
      mysql -u radius -p[FREERADIUS_DB_PASS] radius
```

1.5.10. Prueba de Radius

Inicie radius para propósitos de inicialización y prueba:

```
sudo /usr/sbin/radiusd -X
```

Abra una nueva ventana de terminal para probar conexiones:

```
radtest [ADMIN_USER] [ADMIN_PASSWORD] 127.0.0.1 0
   [FREERADIUS_DB_PASS]
```

Si obtiene un mensaje como este, entonces ha terminado con la configuración mínima y requerida de radius para los siguientes pasos:

```
Received Access-Accept Id 174 from 127.0.0.1:1812 to 0.0.0.0:0
   length 20
```

1.5.11. Servicio Systemd para Radius

Antes de dejar radius de lado, cree un archivo de servicio systemd para su servidor radius:

```
sudo vim /etc/systemd/system/radiusd.service
```

```
[Unit]
Description=FreeRADIUS high performance RADIUS server.
After=mysqld.service syslog.target network.target

[Service]
Type=forking
ExecStartPre=-/bin/mkdir /var/log/radius
ExecStartPre=-/bin/mkdir /var/run/radiusd
ExecStartPre=-/bin/chown -R radiusd.radiusd /var/log/radius
ExecStartPre=-/bin/chown -R radiusd.radiusd /var/run/radiusd
ExecStartPre=/usr/sbin/radiusd -C
ExecStart=/usr/sbin/radiusd -d /etc/raddb
ExecReload=/usr/sbin/radiusd -C
ExecReload=/bin/kill -HUP $MAINPID

[Install]
WantedBy=multi-user.target
```

Habilite el servicio para que se inicie automáticamente al arrancar:

```
sudo systemctl enable radiusd.service
```

1.6. Instalación de Haserl

Haserl es necesario para el miniportal embebido incluido en CoovaChilli.
Descargue haserl:

```
cd /tmp
wget -c http://superb-dca2.dl.sourceforge.net/project/haserl
/haserl-devel/haserl-0.9.35.tar.gz
```

Desempaquete el tarball:

```
sudo tar zxvf haserl-0.9.35.tar.gz -C /usr/src/
cd /usr/src/haserl-0.9.35/
```

Compile e instale:

```
./configure --prefix=/usr --libdir=/usr/lib64
make
sudo make install
```

(Asegúrese de cambiar a la biblioteca correcta o prefijo deseado)

1.7. Instalación de CoovaChilli

CoovaChilli es un software de portal cautivo de código abierto. Comenzó a partir del proyecto chilli obsoleto. Después de la instalación y configuración de coovachilli, podrá redirigir a los clientes de su hotspot WiFi a una página de inicio de sesión, es decir, portal cautivo donde pueden iniciar sesión y acceder a Internet.
Descargue las últimas fuentes de coovachilli:

```
cd /usr/src
sudo git clone https://github.com/coova/coova-chilli.git
```

Configure y compile coova:

```
cd /usr/src/coova-chilli
sh bootstrap
./configure --prefix=/usr --libdir=/usr/lib64
   --localstatedir=/var \
  --sysconfdir=/etc --enable-miniportal --with-openssl
     --enable-libjson \
  --enable-useragent --enable-sessionstate --enable-sessionid \
  --enable-chilliredir --enable-binstatusfile --enable-statusfile
     \
  --disable-static --enable-shared --enable-largelimits \
```

```
  --enable-proxyvsa --enable-chilliproxy --enable-chilliradsec
    --with-poll
```

(Asegúrese de cambiar a la biblioteca correcta o prefijo deseado)

```
make
sudo make install
```

1.7.1. Configuración de CoovaChilli

Todos los archivos de configuración se encuentran en: /etc/chilli. Necesitará crear un archivo de configuración con las modificaciones de su sitio de la siguiente manera:

```
sudo cp -v /etc/chilli/defaults /etc/chilli/config
sudo vim /etc/chilli/config
```

Cambie los siguientes parámetros para que coincidan con su entorno:

```
HS_WANIF=eth0                    # WAN Interface toward the
   Internet
HS_LANIF=wlan0                   # Subscriber Interface for
   client devices
HS_NETWORK=10.1.0.0              # HotSpot Network (must
   include HS_UAMLISTEN)
HS_NETMASK=255.255.255.0         # HotSpot Network Netmask
HS_UAMLISTEN=10.1.0.1            # HotSpot IP Address (on
   subscriber network)
HS_RADSECRET=[FREERADIUS_DB_PASS]  # Set to be your RADIUS shared
   secret
HS_UAMSECRET=[FREERADIUS_DB_PASS]  # Set to be your UAM secret
HS_ADMUSR=[ADMIN_USER]
HS_ADMPWD=[ADMIN_PASSWORD]
```

1.7.2. Script ipup.sh

Agregue el script chilli ipup.sh. El propósito de este script es preparar el sistema para actuar como enrutador. También puede desear agregar otros comandos, por ejemplo, configurar el gateway.

```
sudo vim /etc/chilli/ipup.sh
```

```
#!/bin/sh
#
# Allow IP masquerading through this box
/usr/sbin/iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

IMPORTANTE: Cambie el dispositivo de Internet al correcto.
Haga el script ejecutable:

```
sudo chmod 755 /etc/chilli/ipup.sh
```

Habilite coovachilli para que se inicie al arrancar:

```
sudo systemctl enable chilli
```

Inicie coovachilli:

```
sudo systemctl start chilli
```

1.8. Prueba del Portal Cautivo

Antes de comenzar las pruebas, asegúrese de que primero pueda acceder a Internet localmente.
Luego, usando un cliente inalámbrico como un smartphone o laptop, abra su navegador web favorito. Vaya a cualquier URL/sitio web de su elección. Debería ser redirigido automáticamente a la página de portal cautivo donde puede iniciar sesión con las credenciales configuradas.

1.9. Gestión de Usuarios

Para la gestión de usuarios, puede:

- Agregar usuarios directamente a la base de datos MySQL de radius
- Implementar una interfaz web de administración
- Usar herramientas de terceros para gestión de usuarios radius
- Integrar con sistemas de autenticación existentes (LDAP, Active Directory)

Para agregar un nuevo usuario manualmente:

```
echo "INSERT INTO radcheck (UserName, Attribute, Value, Op) \
      VALUES ('username', 'Cleartext-Password', 'password',
          ':=');" | \
      mysql -u radius -p[FREERADIUS_DB_PASS] radius
```

1.10. Conclusiones

Este tutorial ha demostrado cómo configurar un hotspot inalámbrico completo con portal cautivo usando CoovaChilli, FreeRadius y hostapd en Linux. La solución proporciona:

- Control de acceso mediante autenticación
- Gestión de usuarios mediante base de datos
- Portal cautivo profesional para inicio de sesión
- Contabilidad y límites de ancho de banda
- Infraestructura escalable para entornos empresariales