



TAREA 04

ADMINISTRACIÓN DE CREDENCIALES PARA EL ACCESO A SISTEMAS INFORMÁTICOS

BASTIONADO DE REDES Y SISTEMAS

ALBA MOREJÓN GARCÍA

2024/2025

CETI - Ciberseguridad en Entornos de las Tecnologías de la Información

Caso práctico

En la empresa de María, las cuentas de usuario consisten en un nombre de usuario y una contraseña (password). En los sistemas operativos estos dos elementos forman un conjunto de credenciales y sirven para identificar a una persona. Utilizar contraseñas es un método para autenticarse, pero no es el único, hay otros métodos como, por ejemplo, el uso de tarjetas inteligentes que tiene la identidad grabada. La administración de credenciales de acceso es algo fundamental debido a los numerosos ataques de contraseñas que hoy día pueden producirse.

De hecho, los sistemas de control de acceso protegidos con contraseña, suelen ser un punto crítico de la seguridad y por ello suelen recibir distintos tipos de ataques, siendo los más comunes los ataques de fuerza bruta y los ataques de diccionario.

Apartado 1: tarea de investigación e implementación

Para elaborar la práctica el alumno deberá investigar cómo llevar a cabo un despliegue de este tipo en un entorno doméstico. Existen numerosas fuentes en Internet que explican cómo llevarlo a cabo.

El alumno mostrará a través de capturas de pantalla el proceso que ha llevado a cabo.

En caso de no disponer de un router compatible, bastará con la explicación detallada del proceso en ese dispositivo.

Elementos necesarios:

- **Máquina virtual Ubuntu (u otro Linux)**
- **Aplicación FreeRADIUS para instalar en Ubuntu**
- **Router compatible con seguridad RADIUS. Habitualmente los routers actuales de los proveedores de Internet suelen disponer de él.**
- **Cliente wifi que puede ser un ordenador, móvil, etc.**

INTRODUCCIÓN

RADIUS (Remote Authentication Dial-In Service) es un protocolo estándar de red, diseñado para gestionar desde un punto central, tres funciones (AAA): autenticación, verificación de credenciales, certificados o tokens, autorización, definición de los recursos a los que puede acceder un usuario y contabilidad, registro de actividades hechas por el usuario. Este protocolo de Internet se utiliza en redes inalámbricas Wi-fi, VPS y otros servicios para proporcionar seguridad, control de accesos y aplicar políticas. RADIUS opera con tecnologías 802.1X para redes Wi-fi y emplea protocolos PAP, CHAP o EAP para la autenticación. Su comunicación con servidores NAS (de acceso) se realiza mediante UDP a través del puerto 1812.

Implementar este protocolo de seguridad en un servidor (servidor RADIUS) es útil para que una empresa con múltiples ubicaciones o redes complejas porque permite gestionar de manera centralizada el acceso seguro a su red, permitiendo autenticar y autorizar tanto empleados y dispositivos desde un único punto. Esto garantiza que únicamente las personas autorizadas tengan acceso a los recursos, fortaleciendo la seguridad y haciendo posible llevar un monitoreo y cumplimiento normativo.

FreeRADIUS es un servidor RADIUS de código abierto que centraliza la autenticación, autorización y contabilidad de usuarios en redes y servicios. Se utiliza para validar credenciales, gestionar permisos y registrar actividades, siendo clave en redes Wi-fi, VPS, empresas... Funciona integrándose con bases de datos como LDAP o Active Directory y dispositivos como puntos de acceso Wi-fi o Switches, garantizando seguridad y control de acceso. Su flexibilidad, escalabilidad y compatibilidad con protocolos como 802.1X o EAP lo hacen ideal para gestionar usuarios en redes modernas de forma eficiente y gratuita.

En mi caso el router del que dispongo, no tiene la opción de activar la seguridad WPA2- Enterprise para poder configurar RADIUS con todos sus requisitos, así que intentaré recrear esta práctica con:

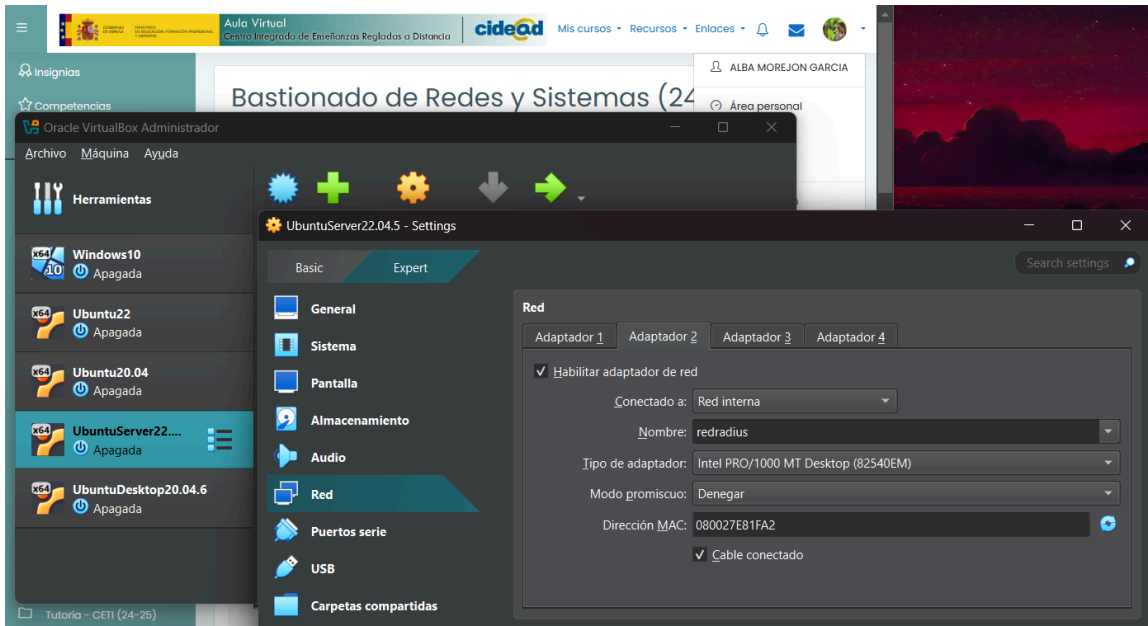
FUNCIÓN	VERSIÓN	IP
Servidor RADIUS	Ubuntu Server 22.04.5	192.168.50.1
Cliente Linux	Ubuntu Desktop 20.04.6	192.168.50.2

PROCESO DE DESPLIEGUE

1. CONFIGURAR RED

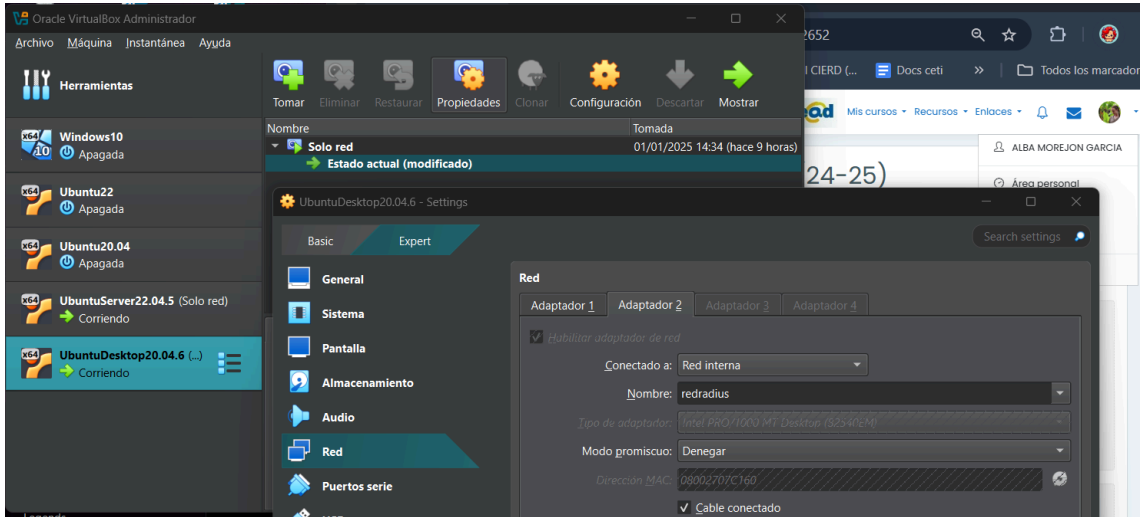
ADAPTADORES

En el Servidor RADIUS ponemos el adaptador 1 en la opción NAT y habilitamos el adaptador 2 en Red Interna eligiendo el nombre de “redradius”.



enp0s3 - nat - 080027CDF1B2 y enp0s8 - redinterna - 080027E81FA2

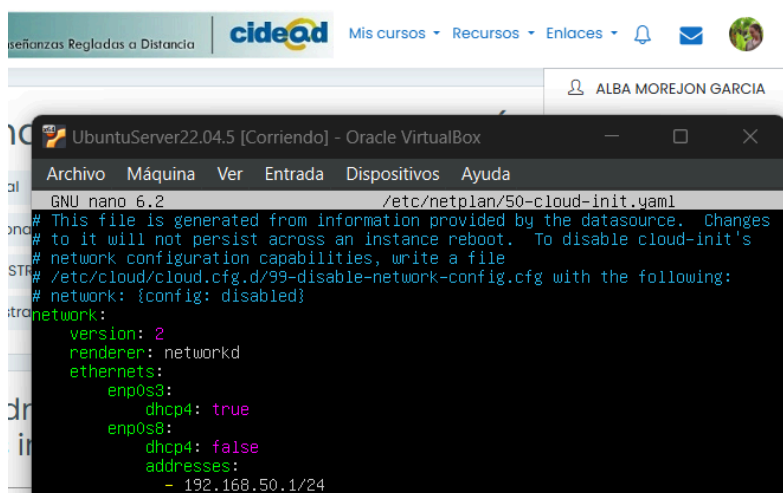
En el ClienteLinux elegimos la misma configuración que la máquina UbuntuServer22, Red interna: redradius *en principio la idea iba a ser que la máquina ubuntu server 22 fuera la única que saliese a internet pero tras encontrarnos posteriormente algún error en la resolución DNS optamos por la opción aplicada.



enp0s3 - nat- 08002707FEB0 y enp0s8 - redinterna - 08002707C160

IPs

Para la configuración de redes utilizaremos netplan, añadimos los datos necesarios en el siguiente fichero: “/etc/netplan/nom_fichero” y aplicamos los cambios con “sudo netplan apply”.
Con la ip correspondiente en cada máquina.



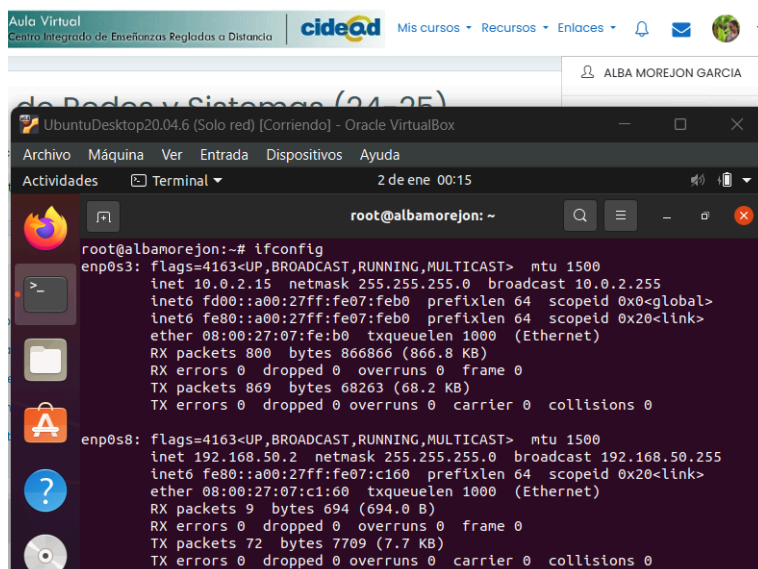
The screenshot shows a terminal window titled "UbuntuServer22.04.5 [Corriendo] - Oracle VirtualBox". The terminal is running the nano text editor, editing the file "/etc/netplan/50-cloud-init.yaml". The content of the file is as follows:

```
GNU nano 6.2 /etc/netplan/50-cloud-init.yaml
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: false
      addresses:
        - 192.168.50.1/24
```



The screenshot shows a terminal window titled "UbuntuServer22.04.5 [Corriendo] - Oracle VirtualBox". The terminal is running the command "sudo netplan apply" and then "ip a". The output of the "ip a" command is as follows:

```
albamorejon@UbuntuServer22:/etc/network$ sudo netplan apply
WARNING:root:Cannot call Open vSwitch: ovsdb-server.service is not running.
albamorejon@UbuntuServer22:/etc/network$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    link/ether 08:00:27:cd:f1:b2 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86389sec preferred_lft 86389sec
    inet6 fd00::a00:27ff:fe07:feb0/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 86390sec preferred_lft 14390sec
    inet6 fe80::a00:27ff:fe07:feb0/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    link/ether 08:00:27:e8:1f:a2 brd ff:ff:ff:ff:ff:ff
    inet 192.168.50.1/24 brd 192.168.50.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe07:feb0/64 scope link
        valid_lft forever preferred_lft forever
```



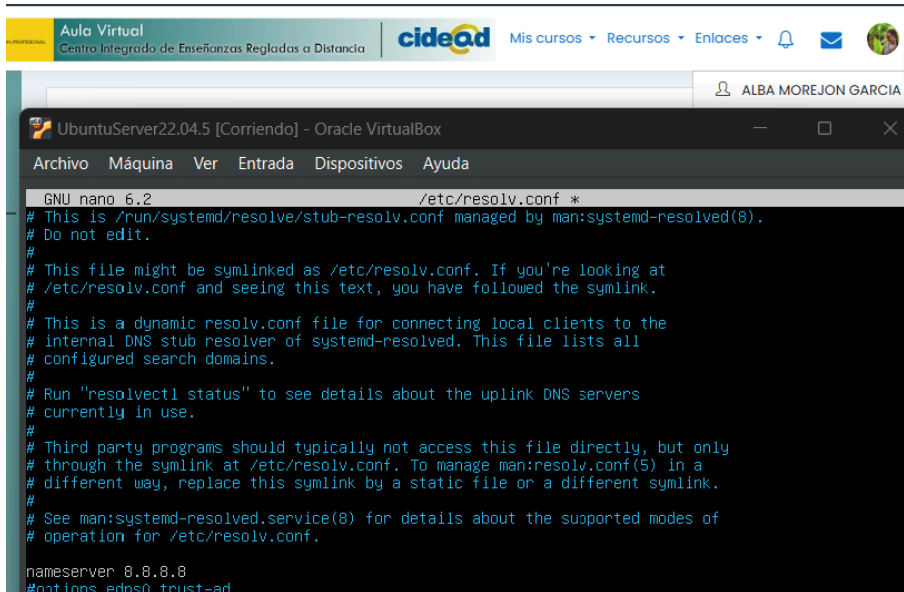
The screenshot shows a terminal window titled "UbuntuDesktop20.04.6 (Solo red) [Corriendo] - Oracle VirtualBox". The terminal is running the command "ifconfig". The output of the "ifconfig" command is as follows:

```
root@albamorejon:~# ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fd00::a00:27ff:fe07:feb0 prefixlen 64 scopeid 0x0<global>
    inet6 fe80::a00:27ff:fe07:feb0 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:07:fe:b0 txqueuelen 1000 (Ethernet)
    RX packets 800 bytes 866866 (866.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 869 bytes 68263 (68.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.50.2 netmask 255.255.255.0 broadcast 192.168.50.255
    inet6 fe80::a00:27ff:fe07:c160 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:07:c1:60 txqueuelen 1000 (Ethernet)
    RX packets 9 bytes 694 (694.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 72 bytes 7709 (7.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

DNS

En ambas máquinas agregamos los servidores dns de google (8.8.8.8) en el fichero “/etc/resolv.conf”



```
UbuntuServer22.04.5 [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 6.2 /etc/resolv.conf *
# This is /run/systemd/resolve/stub-resolv.conf managed by man:systemd-resolved(8).
# Do not edit.
#
# This file might be symlinked as /etc/resolv.conf. If you're looking at
# /etc/resolv.conf and seeing this text, you have followed the symlink.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs should typically not access this file directly, but only
# through the symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a
# different way, replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 8.8.8.8
#options edns0 trust-ad
```

Con esto, ambas máquinas tendrían salida a internet y se conectarán entre ellas.

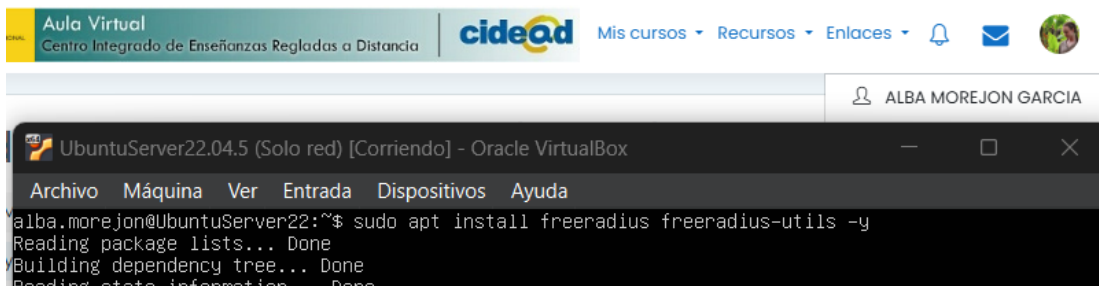
2. CONFIGURAR SERVIDOR RADIUS

Primero haremos una mejora y una actualización de las máquinas con los siguientes comandos:

“sudo apt update && sudo apt upgrade -y”

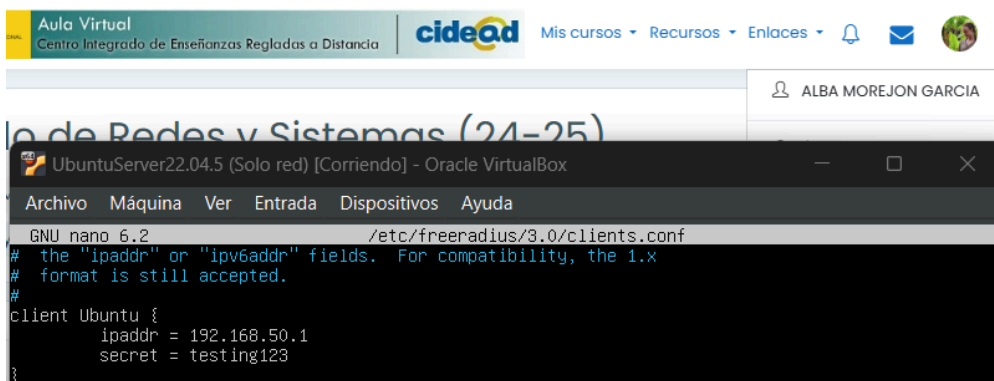
Instalamos el paquete principal del servidor y herramientas adicionales con el comando

“sudo apt install freeradius freeradius-utils -y” que sería



```
UbuntuServer22.04.5 (Solo red) [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
alba.morejon@UbuntuServer22:~$ sudo apt install freeradius freeradius-utils -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Modificamos el fichero de configuración para clientes “/etc/freeradius/3.0/clients.conf”, que enviará las solicitudes de autenticación al servidor radius (que sería en este caso el UbuntuServer22)



```
UbuntuServer22.04.5 (Solo red) [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
GNU nano 6.2 /etc/freeradius/3.0/clients.conf
# the "ipaddr" or "ipv6addr" fields. For compatibility, the 1.x
# format is still accepted.
#
client Ubuntu {
    ipaddr = 192.168.50.1
    secret = testing123
}
```

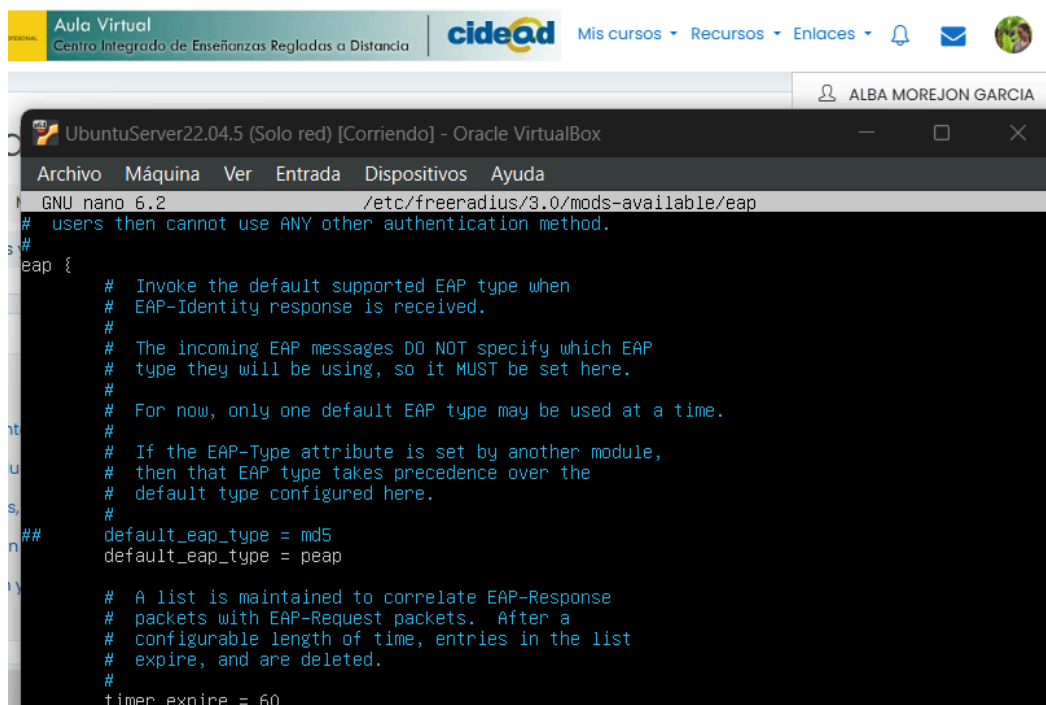
Los datos serían la ip del dispositivo que hará de router (la propia máquina), la contraseña entre el cliente y el servidor.

A continuación, editamos el archivo de usuarios para añadir el usuario con el que haremos las pruebas de autenticación, el fichero es:

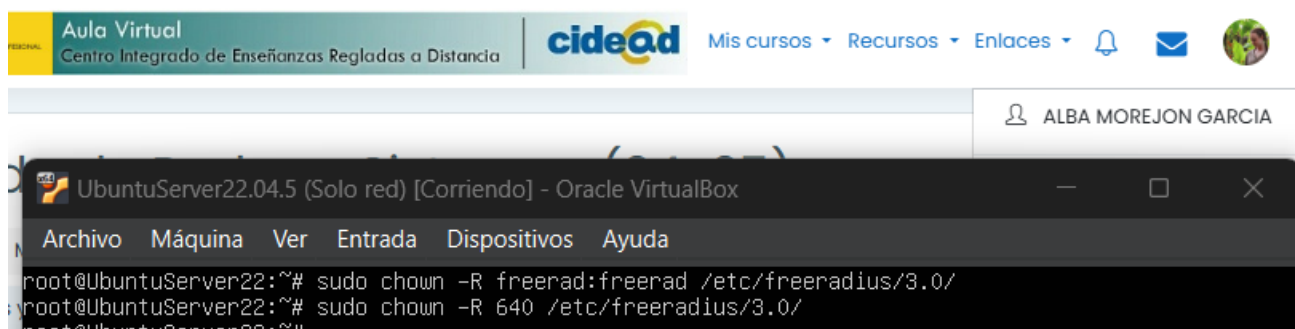
“/etc/freeradius/3.0/mods-config/files/authorize”



Modificamos el archivo eap “/etc/freeradius/3.0/mods-available/eap” para que la autenticación sea WPA2-Enterprise (PEAP) que es la más común en redes Wi-fi.



Aseguramos que el usuario/grupo freerad tengan propiedad y total sobre los archivos de configuración para freeradius, así como permisos de lectura y modificación.



Después de las modificaciones, reiniciamos el servicio.

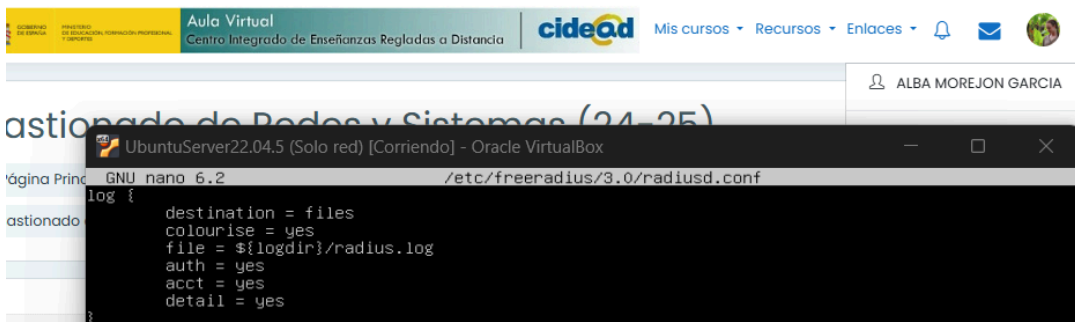


```
root@UbuntuServer22.04.5:~# sudo systemctl restart freeradius
root@UbuntuServer22.04.5:~# sudo systemctl status freeradius
● freeradius.service - FreeRADIUS multi-protocol policy server
   Loaded: loaded (/lib/systemd/system/freeradius.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2025-01-01 18:43:28 UTC; 7s ago
     Docs: man:radiusd(8)
           man:radiusd.conf(5)
           http://wiki.freeradius.org/
           http://networkradius.com/doc/
   Process: 4018 ExecStartPre=/usr/sbin/freeradius $FREERADIUS_OPTIONS -Cx -1stdout (code=exited,
   Main PID: 4019 (freeradius)
   Status: "Processing requests"
```

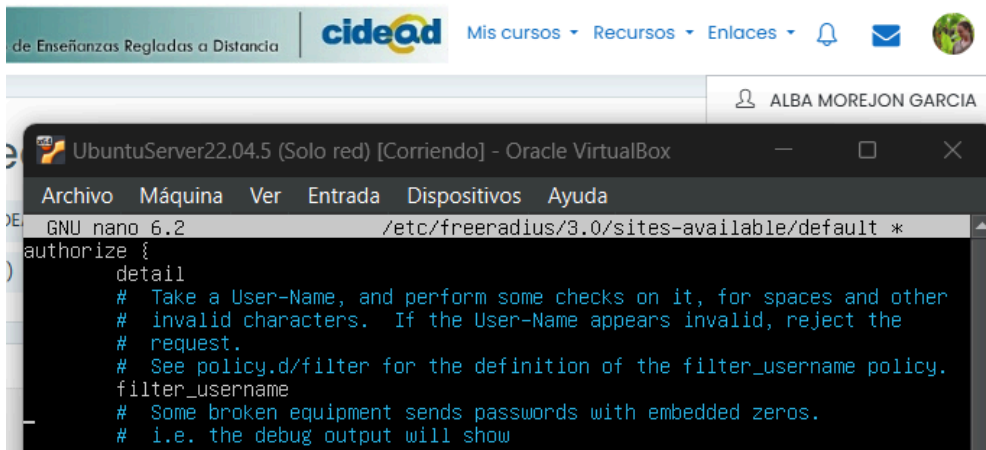
Monitorear los accesos en el servidor

Definimos el comportamiento de los log, para que se guarden en archivos en la ruta que le indicamos, que se registren las autenticaciones con los detalles y los inicios rechazados. En el fichero:

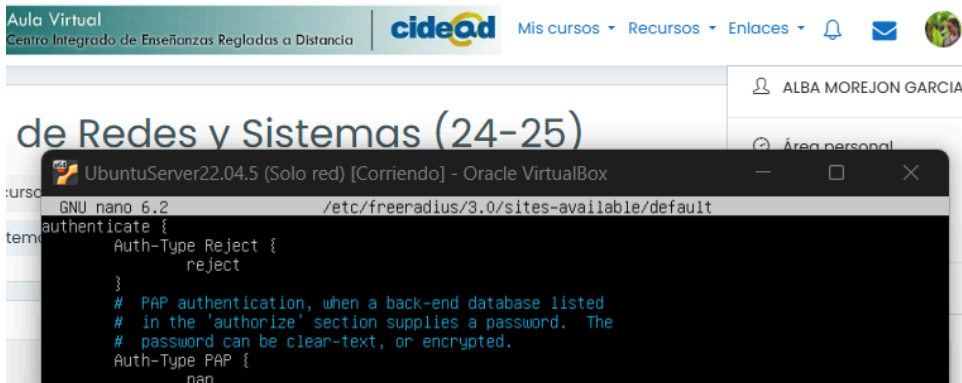
“/etc/freeradius/3.0/radiusd.conf” y “/etc/freeradius/3.0/sites-available/default”



```
GNU nano 6.2 /etc/freeradius/3.0/radiusd.conf
log {
    destination = files
    colourise = yes
    file = ${logdir}/radius.log
    auth = yes
    acct = yes
    detail = yes
}
```



```
GNU nano 6.2 /etc/freeradius/3.0/sites-available/default *
authorize {
    detail
    # Take a User-Name, and perform some checks on it, for spaces and other
    # invalid characters. If the User-Name appears invalid, reject the
    # request.
    # See policy.d/filter for the definition of the filter_username policy.
    filter_username
    # Some broken equipment sends passwords with embedded zeros.
    # i.e. the debug output will show
```



```
GNU nano 6.2 /etc/freeradius/3.0/sites-available/default
authenticate {
    Auth-Type Reject {
        reject
    }
    # PAP authentication, when a back-end database listed
    # in the 'authorize' section supplies a password. The
    # password can be clear-text, or encrypted.
    Auth-Type PAP {
        pap
    }
}
```

Reiniciamos el servicio con “sudo systemctl restart freeradius”

3. PRUEBAS

Recordemos:

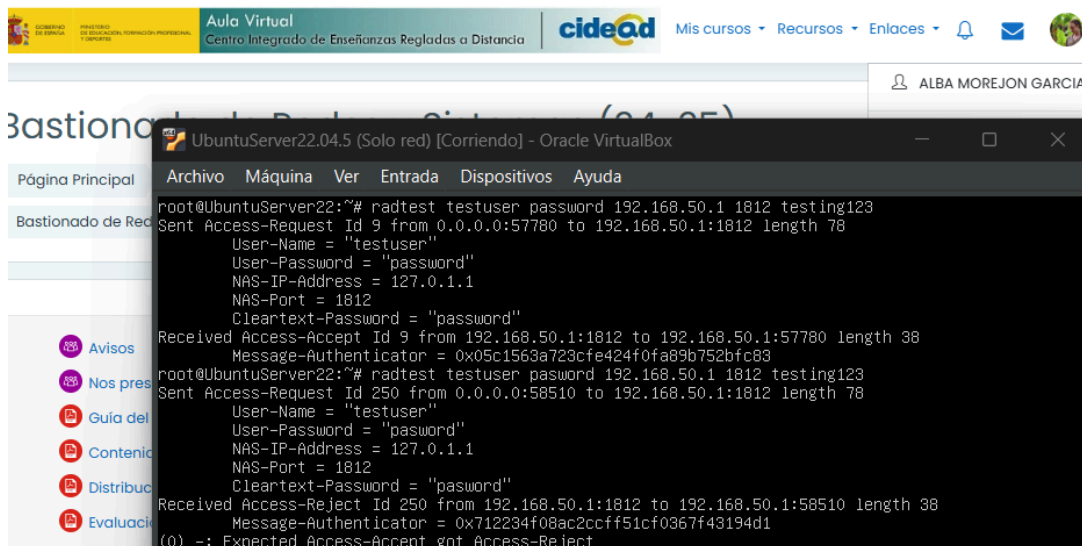
Client: 192.168.50.1 - testing123

User: testuser - password

Probamos la autenticación con usuario que habíamos creado, utilizando el comando:

“radtest testuser password 192.168.50.1 1812 testing123”

Hacemos dos intentos de inicio uno bien y el siguiente con un error

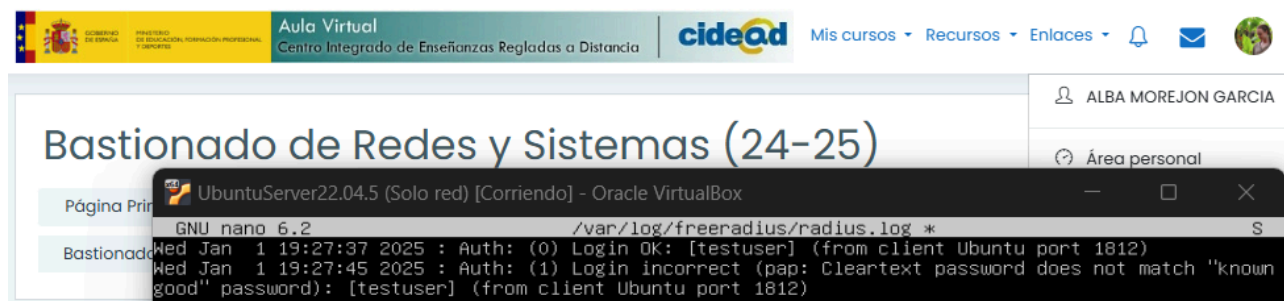


```
root@UbuntuServer22.04.5 (Solo red) [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
root@UbuntuServer22:~# radtest testuser password 192.168.50.1 1812 testing123
Sent Access-Request Id 9 from 0.0.0.0:57780 to 192.168.50.1:1812 length 78
  User-Name = "testuser"
  User-Password = "password"
  NAS-IP-Address = 127.0.1.1
  NAS-Port = 1812
  Cleartext-Password = "password"
Received Access-Accept Id 9 from 192.168.50.1:1812 to 192.168.50.1:57780 length 38
  Message-Authenticator = 0x05c1563a723cfe424f0fa89b752bfc83
root@UbuntuServer22:~# radtest testuser pasword 192.168.50.1 1812 testing123
Sent Access-Request Id 250 from 0.0.0.0:58510 to 192.168.50.1:1812 length 78
  User-Name = "testuser"
  User-Password = "pasword"
  NAS-IP-Address = 127.0.1.1
  NAS-Port = 1812
  Cleartext-Password = "pasword"
Received Access-Reject Id 250 from 192.168.50.1:1812 to 192.168.50.1:58510 length 38
  Message-Authenticator = 0x712234f08ac2c0ff51cf0367f43194d1
(0) -: Expected Access-Accept got Access-Reject
```

En la primera conexión, el servidor radius ha aceptado la solicitud de autenticación, el mensaje proviene de la dirección ip 192.168.50.1:1812, la respuesta fue enviada de vuelta al cliente por el puerto 42163 y el mensaje de autenticación es un código hash que garantiza la seguridad.

El segundo intento ha sido rechazado (Received Access-Reject)

Verificamos los logs en el fichero “/var/log/freeradius/radius.log” en el que se nos mostrarían todos los detalles como el día, la hora, la autenticación, etc.



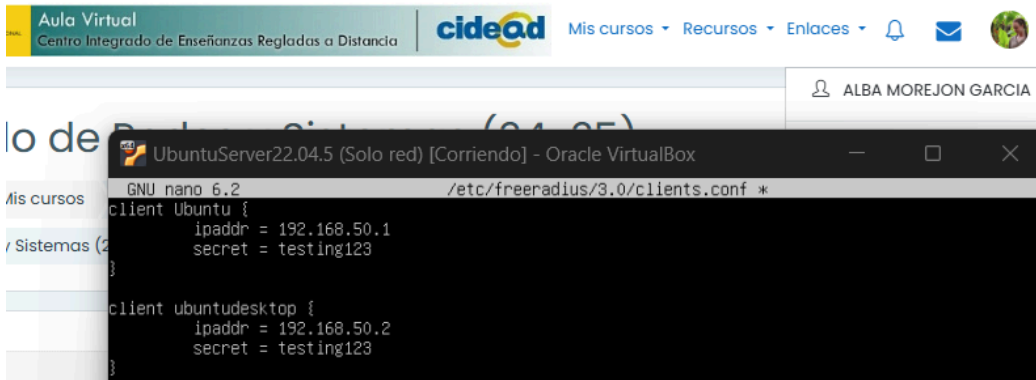
```
GNU nano 6.2 /var/log/freeradius/radius.log
Wed Jan 1 19:27:37 2025 : Auth: (0) Login OK: [testuser] (from client Ubuntu port 1812)
Wed Jan 1 19:27:45 2025 : Auth: (1) Login incorrect (pap: Cleartext password does not match "known good" password): [testuser] (from client Ubuntu port 1812)
```


4. CONFIGURAR CLIENTE

Configuramos la máquina UbuntuDesktop20 como cliente para que se pueda autenticar en el servidor freeradius (UbuntuServer22)

En el UbuntuServer22 modificamos el fichero de configuración del cliente para permitir la autenticación desde la otra máquina:

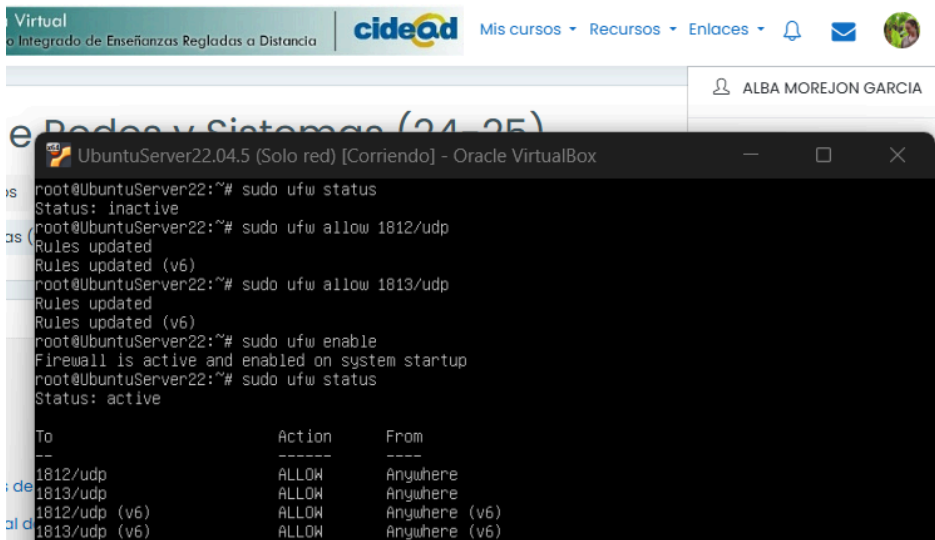
“/etc/freeradius/3.0/clients.conf”



```
GNU nano 6.2 /etc/freeradius/3.0/clients.conf *
client Ubuntu {
    ipaddr = 192.168.50.1
    secret = testing123
}

client ubuntudesktop {
    ipaddr = 192.168.50.2
    secret = testing123
}
```

(Opcional) Para evitar que nos de algún problema habilitamos el firewall y abrimos los puerto 1812 y 1813 que son los que utiliza el servicio radius

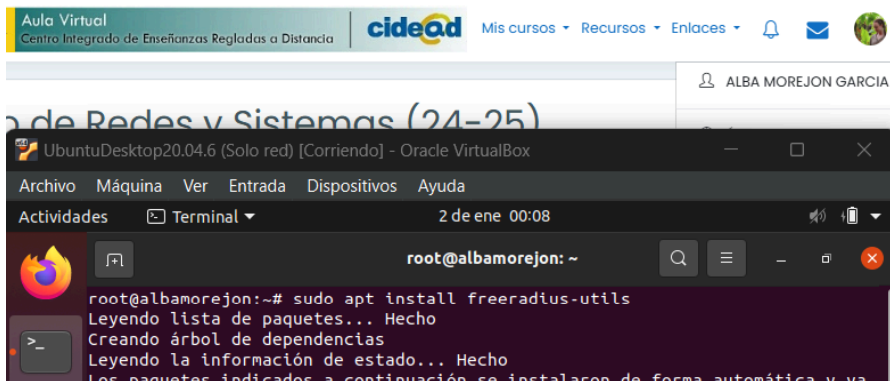


```
root@UbuntuServer22:~# sudo ufw status
Status: inactive
root@UbuntuServer22:~# sudo ufw allow 1812/udp
Rules updated
Rules updated (v6)
root@UbuntuServer22:~# sudo ufw allow 1813/udp
Rules updated
Rules updated (v6)
root@UbuntuServer22:~# sudo ufw enable
Firewall is active and enabled on system startup
root@UbuntuServer22:~# sudo ufw status
Status: active

To Action From
--
1812/udp ALLOW Anywhere
1813/udp ALLOW Anywhere
1812/udp (v6) ALLOW Anywhere (v6)
1813/udp (v6) ALLOW Anywhere (v6)
```

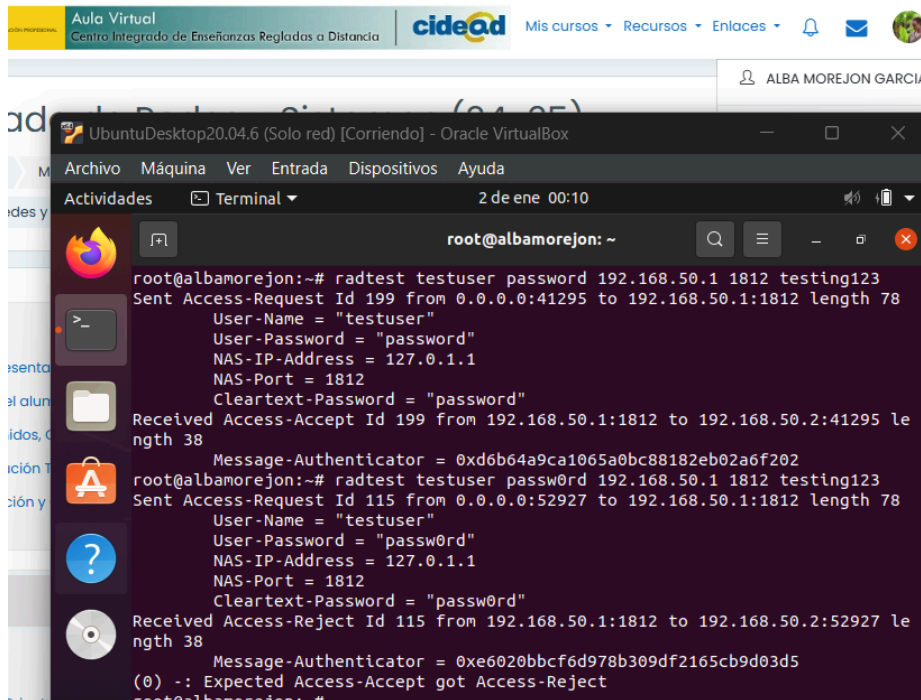
Ahora configuramos el cliente UbuntuDesktop20

Instalamos las herramienta adicionales con el comando “sudo apt install freeradius-utils”



```
root@albamorejon:~# sudo apt install freeradius-utils
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya
```

Hacemos dos pruebas de acceso, una con los datos bien y otra con los datos mal.

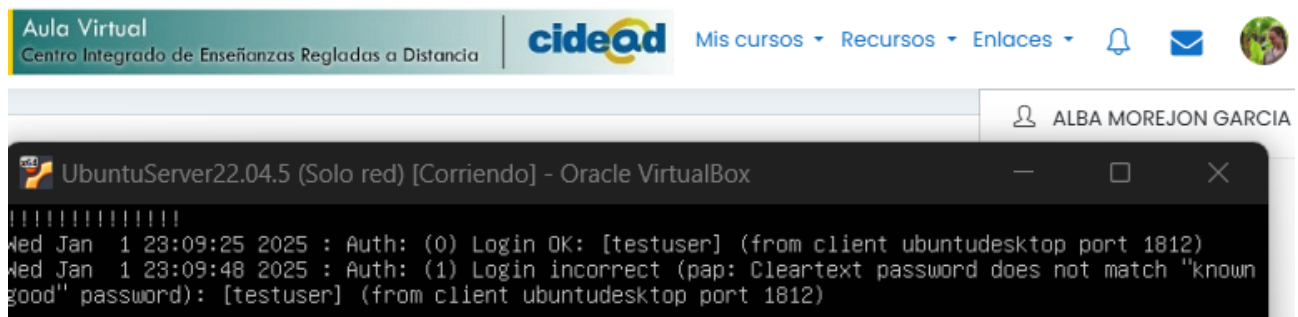


```
root@albamorejon:~# radtest testuser password 192.168.50.1 1812 testing123
Sent Access-Request Id 199 from 0.0.0.0:41295 to 192.168.50.1:1812 length 78
  User-Name = "testuser"
  User-Password = "password"
  NAS-IP-Address = 127.0.1.1
  NAS-Port = 1812
  Cleartext-Password = "password"
Received Access-Accept Id 199 from 192.168.50.1:1812 to 192.168.50.2:41295 length 38
  Message-Authenticator = 0xd6b64a9ca1065a0bc88182eb02a6f202
root@albamorejon:~# radtest testuser passwd 192.168.50.1 1812 testing123
Sent Access-Request Id 115 from 0.0.0.0:52927 to 192.168.50.1:1812 length 78
  User-Name = "testuser"
  User-Password = "passwd"
  NAS-IP-Address = 127.0.1.1
  NAS-Port = 1812
  Cleartext-Password = "passwd"
Received Access-Reject Id 115 from 192.168.50.1:1812 to 192.168.50.2:52927 length 38
  Message-Authenticator = 0xe6020bbcf6d978b309df2165cb9d03d5
(0) -: Expected Access-Accept got Access-Reject
root@albamorejon:~#
```

En la primera conexión vemos que ha sido aceptada (Received Access-Accept) y la segunda rechazada (Received Access-Reject)

Para mostrar los logs a tiempo real, utilizamos el comando:

“tail -f /var/log/freeradius/radius.log”



```
Wed Jan 1 23:09:25 2025 : Auth: (0) Login OK: [testuser] (from client ubuntu desktop port 1812)
Wed Jan 1 23:09:48 2025 : Auth: (1) Login incorrect (pap: Cleartext password does not match "known good" password): [testuser] (from client ubuntu desktop port 1812)
```

Adicional:

En caso de querer borrar todo lo instalado para el servidor radius los comandos serían:

```
sudo rm -rf /etc/freeradius
```

```
sudo rm -rf /var/log/freeradius
```

```
sudo rm -rf /usr/lib/freeradius
```

```
sudo apt purge freeradius freeradius-utils freeradius-config -y
```

El único router que he encontrado con la posibilidad de activar la seguridad RADIUS es:

TL-WA830RE, V1, Versión de firmware:111108

Así se haría la configuración en caso de tener un router compatible y después solo haría falta loguearse con los usuarios creados, en la red Wi-fi

The screenshot shows the configuration interface of a TP-Link TL-WA830RE 300Mbps Wireless Range Extender. The page is titled "emulator.tp-link.com/TL_WA830(UN)1.0/index.htm". The left sidebar contains navigation options: Status, Quick Setup, QSS, Network, Wireless (selected), Wireless Settings, Wireless Security, Wireless MAC Filtering, Wireless Advanced, Throughput Monitor, Wireless Statistics, DHCP, and System Tools. The main content area is for "Wireless Security" and shows the "WPA/WPA2 - Enterprise" configuration. The "Key 2", "Key 3", and "Key 4" fields are all set to "Disabled". The "WPA/WPA2 - Enterprise" section is selected, showing fields for "Version" (WPA2), "Encryption" (AES), "Radius Server IP" (192.168.50.1), "Radius Port" (1812), "Radius Password" (testing123), and "Group Key Update Period" (30). The "WPA/WPA2 - Personal(Recommended)" section is also visible, with "Version" set to "Automatic(Recommended)". The right sidebar contains additional information: "disabled even if you have selected Shared Key as Authentication Type.", "WPA/WPA2", "Version" (Automatic, WPA, WPA2), "Encryption" (Automatic, TKIP, AES), "Radius Server IP", "Radius Port", "Radius Password", "Group Key Update Period", "WPA-PSK/WPA2-PSK", and "Version" (Automatic, WPA, WPA2).

TP-LINK® 300Mbps Wireless Range Extender Model No. TL-WA830RE

Key 2: ☐ Disabled
Key 3: ☐ Disabled
Key 4: ☐ Disabled

☒ WPA/WPA2 - Enterprise
Version: WPA2
Encryption: AES
Radius Server IP: 192.168.50.1
Radius Port: 1812 (1-65535, 0 stands for default port 1812)
Radius Password: testing123
Group Key Update Period: 30 (in second, minimum is 30, 0 means no update)

☐ WPA/WPA2 - Personal(Recommended)
Version: Automatic(Recommended)

disabled even if you have selected Shared Key as Authentication Type.

WPA/WPA2
Version - You can select one of following versions:
• Automatic - Select WPA or WPA2 automatically based on the wireless station's capability and request.
• WPA - Wi-Fi Protected Access.
• WPA2 - WPA version 2.

Encryption - You can select either Automatic, or TKIP or AES.

Radius Server IP - Enter the IP address of the Radius Server.

Radius Port - Enter the port that radius service uses.

Radius Password - Enter the password for the Radius Server.

Group Key Update Period - Specify the group key update interval in seconds. The value can be either 0 or at least 30. Enter 0 to disable the update.

WPA-PSK/WPA2-PSK
Version - You can select one of following versions: