

### Pergunta 1:

```
root@AAA-1: ~  
AAA-1 console is now available... Press RETURN to get started.  
root@AAA-1:~# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.200.3 netmask 255.255.255.0 broadcast 0.0.0.0  
    inet6 fe80::8cb7:86ff:fedf:2cf8 prefixlen 64 scopeid 0x20<link>  
    ether 06:23:41:10:bb:03 txqueuelen 1000 (Ethernet)  
    RX packets 19  bytes 1366 (1.3 KB)  
    RX errors 0  dropped 2  overruns 0  frame 0  
    TX packets 9  bytes 726 (726.0 B)  
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 0  bytes 0 (0.0 B)  
    RX errors 0  dropped 0  overruns 0  frame 0  
    TX packets 0  bytes 0 (0.0 B)  
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

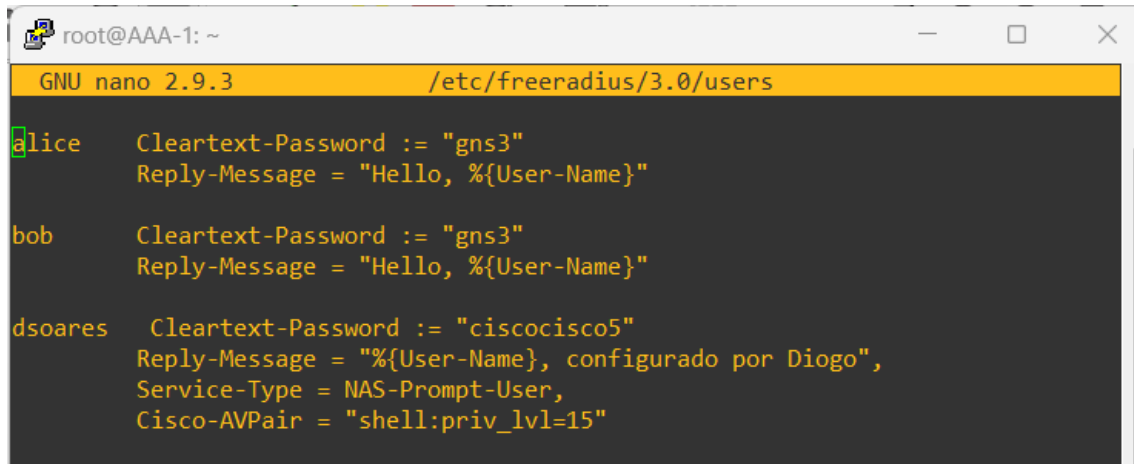
### Pergunta 2:

```
root@AAA-1:~# netstat -anu | grep -E '1812|1813'  
udp        0      0 127.0.0.1:18120      0.0.0.0:*  
udp        0      0 0.0.0.0:1812        0.0.0.0:*  
udp        0      0 0.0.0.0:1813        0.0.0.0:*  
udp6       0      0 :::1812             :::*  
udp6       0      0 :::1813             :::*  
root@AAA-1:~# netstat -anu | grep -E '1812|1813'  
udp        0      0 127.0.0.1:18120      0.0.0.0:*  
udp        0      0 0.0.0.0:1812        0.0.0.0:*  
udp        0      0 0.0.0.0:1813        0.0.0.0:*  
udp6       0      0 :::1812             :::*  
udp6       0      0 :::1813             :::*
```

### Pergunta 3:

```
root@AAA-1: /etc/freeradius/3.0/users  
GNU nano 2.9.3  
alice Cleartext-Password := "gns3"  
    Reply-Message = "Hello, %(User-Name)"  
bob Cleartext-Password := "gns3"  
    Reply-Message = "Hello, %(User-Name)"  
  
Configuration file for the rlm_files module.  
Please see rlm_files(5) manpage for more information.  
  
This file contains authentication security and configuration  
information for each user. Accounting requests are NOT processed  
through this file. Instead, see 'accounting', in this directory.  
  
The first field is the user's name and can be up to  
253 characters in length. This is followed (on the same line) with  
the list of authentication requirements for that user. This can  
include password, conn server name, conn server port number, protocol  
type (perhaps set by the "hints" file), and huntgroup name (set by  
the "huntgroups" file).  
  
If you are not sure why a particular reply is being sent by the  
server, then run the server in debugging mode (radiusd -X), and  
you will see which entries in this file are matched.  
  
When an authentication request is received from the conn server,  
these values are tested. Only the first match is used unless the  
"Fall-Through" variable is set to "Yes".  
  
A special user named "DEFAULT" matches on all usernames.  
You can have several DEFAULT entries. All entries are processed  
in the order they appear in this file. The first entry that  
matches the login-request will stop processing unless you use  
the Fall-Through variable.  
  
Indented (with the tab character) lines following the first  
line indicate the configuration values to be passed back to  
the conn server to allow the initiation of a user session.  
This can include things like the ppp configuration values  
or the host to log the user onto.  
  
You can include another 'users' file with '$INCLUDE users.other'
```

Comando: nano /etc/freeradius/3.0/users



```
root@AAA-1: ~
GNU nano 2.9.3 /etc/freeradius/3.0/users

alice  Cleartext-Password := "gns3"
      Reply-Message = "Hello, %{User-Name}"

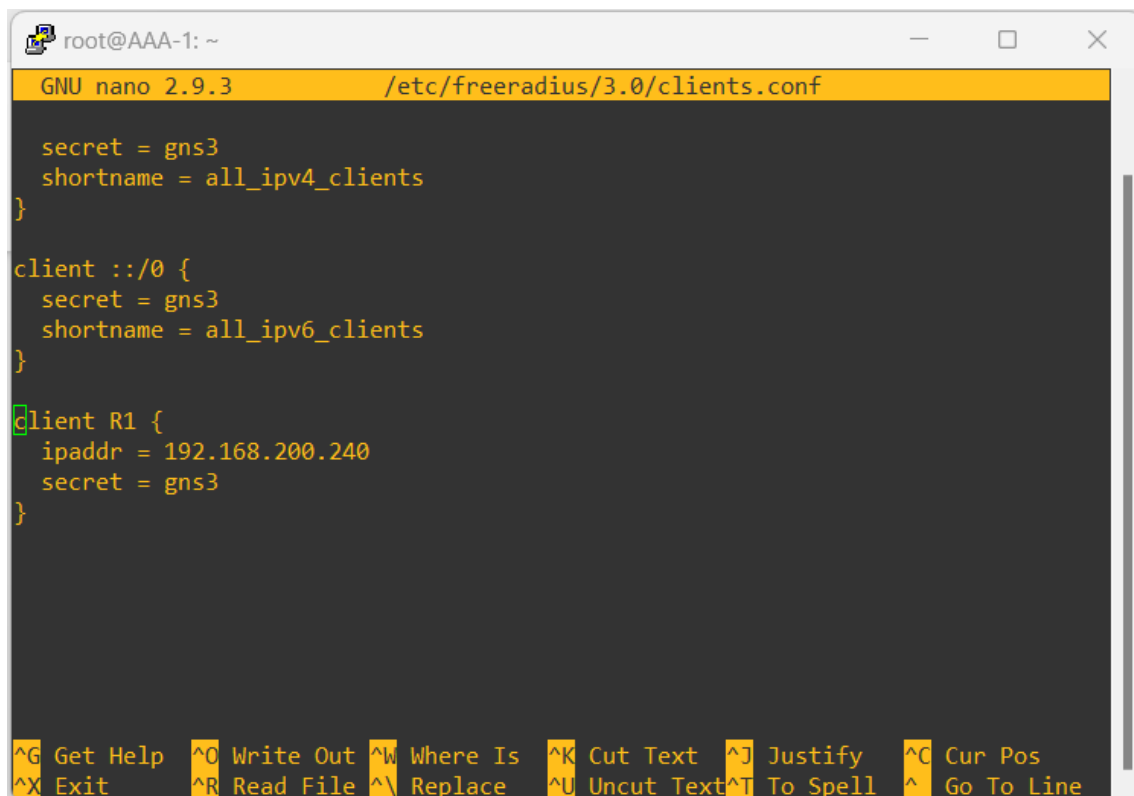
bob    Cleartext-Password := "gns3"
      Reply-Message = "Hello, %{User-Name}"

dsoares Cleartext-Password := "ciscocisco5"
      Reply-Message = "%{User-Name}, configurado por Diogo",
      Service-Type = NAS-Prompt-User,
      Cisco-AVPair = "shell:priv_lvl=15"
```

Vai so ate a segunda linha

Pergunta 4:

Comando: nano /etc/freeradius/3.0/clients.conf



```
root@AAA-1: ~
GNU nano 2.9.3 /etc/freeradius/3.0/clients.conf

secret = gns3
shortname = all_ipv4_clients
}

client :::/0 {
  secret = gns3
  shortname = all_ipv6_clients
}

client R1 {
  ipaddr = 192.168.200.240
  secret = gns3
}

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

Pergunta 5:

```
R1 - PuTTY
Username:
Username: admin
Password:

R1-4110>en
Password:
R1-4110#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1-4110(config)#aaa new-model
R1-4110(config)#radius-
R1-4110(config)#radius server SERVER
R1-4110(config-radius-server)#add ipv4 192.168.200.2 aut
R1-4110(config-radius-server)#add ipv4 192.168.200.2 auth-port 1812
R1-4110(config-radius-server)#$2.168.200.2 auth-port 1812 acct-port 1813
R1-4110(config-radius-server)#key gns3
R1-4110(config-radius-server)#exit
R1-4110(config)#aaa authe
R1-4110(config)#aaa authentication login default group radius none
R1-4110(config)#aaa authe
R1-4110(config)#aaa authentication login SSH group radius none
R1-4110(config)#line vty 0 4
R1-4110(config-line)#login authe
R1-4110(config-line)#login authentication SSH
R1-4110(config-line)#
```

aaa authentication login default group radius none

aaa authentication login ssh group radius none

Pergunta 6:

```
root@AAA-1:~# service freeradius reload
* Checking FreeRADIUS daemon configuration... [ OK ]
* FreeRADIUS daemon is running
* Reloading FreeRADIUS daemon freeradius [ OK ]
root@AAA-1:~#
```

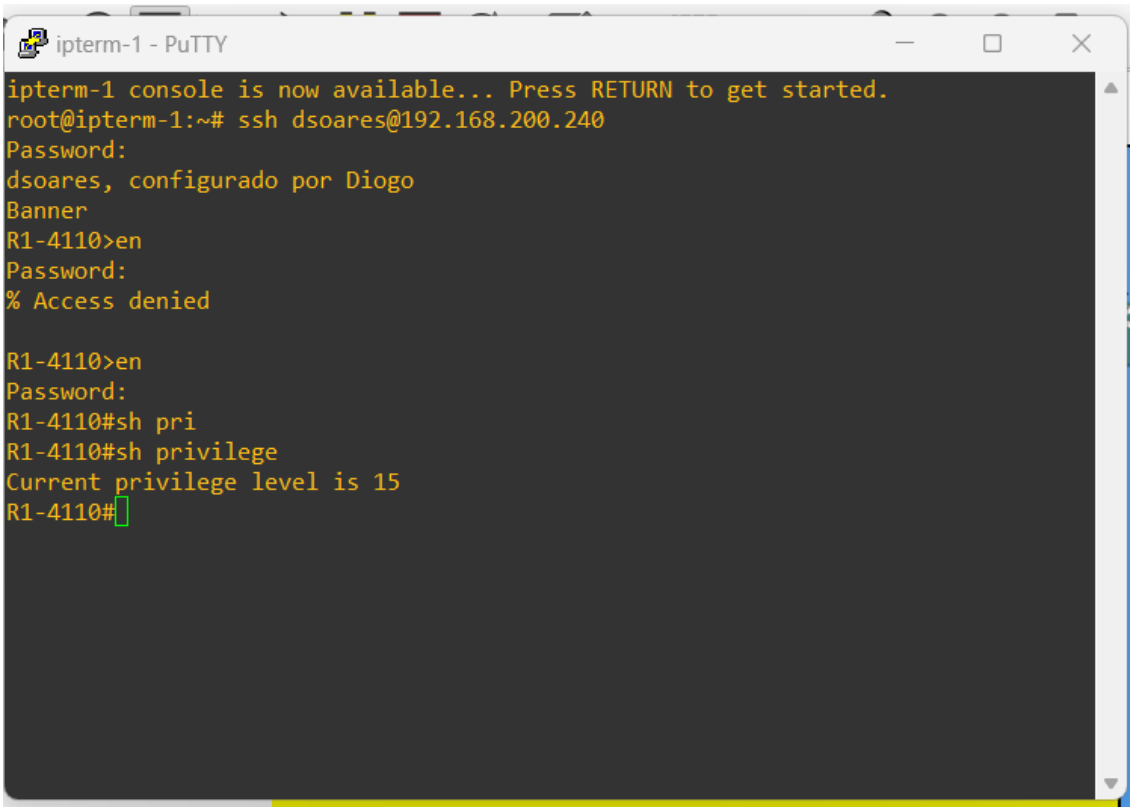
O PDU Access-Request contém o nome de usuário (Username) do cliente que está tentando se autenticar. Já o PDU Access-Accept pode incluir uma mensagem de resposta (Reply-Message) que pode conter informações adicionais, como uma saudação personalizada ou um aviso.

Os autores da RFC 2865 foram inspirados no livro "Applied Cryptography" de Bruce Schneier, que é uma referência na área de segurança da informação e criptografia.

é ofuscada utilizando uma técnica de criptografia de senha chamada de "Challenge-Response Authentication". Esse processo envolve a geração de um desafio (Challenge) pela AAA-1, que é enviado ao cliente. O cliente então aplica uma função hash à senha do usuário, juntamente com o desafio, para gerar uma resposta (Response). Essa

resposta é enviada de volta à AAA-1, que verifica se ela é válida e, em caso afirmativo, concede o acesso.

#### Pergunta 7



```
ipterm-1 console is now available... Press RETURN to get started.  
root@ipterm-1:~# ssh dsoares@192.168.200.240  
Password:  
dsoares, configurado por Diogo  
Banner  
R1-4110>en  
Password:  
% Access denied  
  
R1-4110>en  
Password:  
R1-4110#sh pri  
R1-4110#sh privilege  
Current privilege level is 15  
R1-4110#
```

```
root@AAA-1: ~
GNU nano 2.9.3 /etc/freeradius/3.0/users Modified

alice    Cleartext-Password := "gns3"
         Reply-Message = "Hello, %{User-Name}"

bob      Cleartext-Password := "gns3"
         Reply-Message = "Hello, %{User-Name}"

dsoares  Cleartext-Password := "ciscocisco5"
         Reply-Message = "%{User-Name}, configurado por Diogo",
         Service-Type = NAS-Prompt-User,
         Cisco-AVPair = "shell:priv_lvl=15"

#
# Configuration file for the rlm_files module.
# Please see rlm_files(5) manpage for more information.
#
# This file contains authentication security and configuration
# information for each user. Accounting requests are NOT processed
# through this file. Instead, see 'accounting', in this directory.
#
Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No      ^C Cancel
```

Pergunta 8

```
R1 - PuTTY

Banner

User Access Verification

Username: dsoares
Password:
dsoares, configurado por Diogo

R1-4110>en
Password:
R1-4110#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1-4110(config)#aaa accounting exec default start-stop
% Incomplete command.

R1-4110(config)#aaa acc
R1-4110(config)#aaa accounting exec de
R1-4110(config)#aaa accounting exec default s
R1-4110(config)#aaa accounting exec default star
R1-4110(config)#aaa accounting exec default start-stop group radius
R1-4110(config)#

4 7.960832 - 192.168.200.240 192.168.200.2 RADIUS 129 Accounting-Request id=3
5 7.960832 - 192.168.200.2 192.168.200.240 RADIUS 62 Accounting-Response id=3
```

Pergunta 9

Parser view projeto4

```
R1 - PuTTY
config_app_monitor      Configure application monitoring
config_app_session      Define script processes
config_voice            Define application services, modules,
                        groups
config_voice_app        Define application parameters
configure               Global configuration mode
congestion              Frame Relay congestion configuration mode
conn                    Connection configuration mode
control-class-map       control-classmap config mode
controller              Controller configuration mode
cpf-classmap            Class-map configuration mode
cpf-policyclass         Class-in-Policy configuration mode
cpf-policymap           Policy-map configuration mode
credentials             credentials configuration mode
crypto-identity         Crypto identity config mode
crypto-ipsec-profile    IPSec policy profile mode
crypto-keyring          Crypto Keyring command mode
crypto-map              Crypto map config mode

R1-4110(config-view)#command exec include show all
```

```
root@AAA-1: ~
GNU nano 2.9.3 /etc/freeradius/3.0/users

alice    Cleartext-Password := "gns3"
         Reply-Message = "Hello, %{User-Name}"

bob      Cleartext-Password := "gns3"
         Reply-Message = "Hello, %{User-Name}"

dsoares  Cleartext-Password := "ciscocisco5"
         Reply-Message = "%{User-Name}, configurado por Diogo",
         Service-Type = NAS-Prompt-User,
         Cisco-AVPair = "shell:priv_lvl=15"

dsoares2 Cleartext-Password := "ciscocisco6"
         Reply-Message = "User2 %{User-Name}",
         Service-Type = NAS-Prompt-User,
         Cisco-AVPair = "shell:cli-view-name=projeto4"

#
# Configuration file for the rlm_files module.
# Please see rlm_files(5) manpage for more information.
[ Read 236 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

```
ipterm-1 - PuTTY
R1-4110#sh privilege
^
% Invalid input detected at '^' marker.

R1-4110#sh privilege
Current privilege level is 15
R1-4110#sh parser-view
^
% Invalid input detected at '^' marker.

R1-4110#sh parser view
No view is active ! Currently in Privilege Level Context

R1-4110#logout
Connection to 192.168.200.240 closed by remote host.
Connection to 192.168.200.240 closed.
root@ipterm-1:~# ssh dsoares2@192.168.200.240
Password:
User2 dsoares2
Banner
R1-4110>sh parser view
Current view is 'projeto4'

R1-4110>
```

Pergunta 10

```
root@AAA-1: /etc/tacacs+
freeradius/      mime.types      selinux/
freetds/         mke2fs.conf    services
fstab            mtab            shadow
gai.conf         nanorc          shadow-
group            network/        shells
group-           networks        skel/
gshadow          nsswitch.conf  ssh/
gshadow-         opt/            ssl/
root@AAA-1:/etc# cd tacacs+/
root@AAA-1:/etc/tacacs+# ls
tac_plus.conf
root@AAA-1:/etc/tacacs+# nano tac_plus.conf
GNU nano 2.9.3      tac_plus.conf

# Created by Henry-Nicolas Tourneur(henry.nicolas@tourneur.be)
# See man(5) tac_plus.conf for more details

# Define where to log accounting data, this is the default.
accounting file = /var/log/tac_plus.acct

# This is the key that clients have to use to access Tacacs+
key = gns3
```

```
root@AAA-1: /etc/tacacs+
GNU nano 2.9.3 /etc/tacacs+/tac_plus.conf
#       key = test
#       type = cisco
#       enable = <des|cleartext> enablepass
#       prompt = "Welcome XXX ISP Access Router \n\nUsername:"
#}

# We also can define local users and specify a file where data is stored.
# That file may be filled using tac_pwd
#user = test1 {
#   name = "Test User"
#   member = staff
#   login = file /etc/tacacs/tacacs_passwords
#}

user = dsoares{
    name = "Diogo Soares"
    member = admin
    login = cleartext ciscocisco5
}

group = permissoes_lc{
    service exec {
        priv.lvl = 2
    }
    cmd = show {
        permit .*
    }
}

# We can also specify rules valid per group of users.
#group = group1 {
#   cmd = conf {
#       deny
#   }
#}

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo        M-A Mark Text
^X Exit          ^R Read File    ^_ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line   M-E Redo        M-G Copy Text
```

aaa authentication login default group tacacs+ none

aaa authentication login ssh group tacacs+ none

aaa authorization exec default group tacacs+ if-authenticated

aaa accounting exec default start-stop group tacacs+



Ao capturar o tráfego TACACS+, é possível identificar as etapas específicas do processo de autenticação, autorização e contabilização. Algumas características importantes dos pacotes capturados incluem:

- Pacotes de autenticação: Esses pacotes são usados para validar as credenciais do usuário. Eles podem incluir o nome de usuário e a senha criptografada, além de outras informações de autenticação. Em geral, os pacotes de autenticação são enviados do cliente para o servidor.
- Pacotes de autorização: Uma vez que o usuário é autenticado com sucesso, o servidor TACACS+ envia pacotes de autorização ao cliente para conceder ou negar o acesso a determinados recursos na rede. Esses pacotes podem incluir informações como os comandos que o usuário está autorizado a executar e as interfaces de rede que ele pode acessar.
- Pacotes de contabilização: O TACACS+ também suporta a contabilização de recursos da rede. Isso envolve a coleta de informações sobre o uso da rede por parte dos usuários, como os comandos que foram executados e o tempo de conexão. Os pacotes de contabilização são enviados do cliente para o servidor TACACS+ e podem ser usados para gerar relatórios de uso da rede.

Ao analisar o tráfego TACACS+ capturado, é possível identificar os tipos de pacotes mencionados acima e entender como eles são usados no processo de autenticação, autorização e contabilização. É importante notar que, para proteger as informações de autenticação e autorização, a senha é criptografada nos pacotes de autenticação usando um algoritmo de hash seguro.