

ADMINISTRAÇÃO DE SISTEMAS

SPRINT B

TURMA 3NA

GRUPO 74

1140858	Carlos Moutinho
1171602	Rui Marinho
1181882	Rafael Soares
1181892	Sara S. Silva
1181895	Fábio Silva

INTRODUÇÃO

O presente trabalho foi desenvolvido no âmbito da disciplina de Administração de Sistemas (ASIST) no âmbito do projeto integrador de LAPR5.

Neste relatório será efetuada o estudo das seguintes UC's:

- > Como administrador da infraestrutura quero que o servidor Windows e Linux forneçam endereços IP (na segunda placa de rede) da família 192.168.X.0/24 aos postos clientes, onde X é obtido por 100 + número_do_grupo (exemplo, para o grupo 99, X=199); para o efeito devo alterar o endereço dessa placa assignado nas aulas PL.
- > Como administrador da infraestrutura quero que os serviços acima referidos funcionem em failover, com um deles a facultar endereços de 192.168.X.50 a 192.168.X.150 e o outro de 192.168.X.151 a 192.168.X.200.
- > Como administrador da infraestrutura quero os servidores Windows e Linux estejam disponíveis apenas para pedidos HTTP e HTTPS. Tal não deve impedir o acesso por SSH ou RDP aos administradores (o grupo).
- > Como administrador da infraestrutura quero impedir o IP spoofing na minha rede.
- > Como administrador da infraestrutura quero que os utilizadores registados no Linux com UID entre 6000 e 6500 só consigam aceder via SSH se esse acesso for a partir de uma máquina listada em /etc/remote-hosts.
- > Como administrador da infraestrutura quero que o acesso ao sistema seja inibido aos utilizadores listados em /etc/bad-guys.
- > Como administrador da infraestrutura quero que as mensagens pré-login e pós-login bem-sucedido sejam dinâmicas (por exemplo, “[Bom dia] | [Boa tarde] username”, etc.).
- > Como administrador da infraestrutura quero que o servidor Linux responda e envie pedidos ICMP para teste de conectividade apenas e só aos computadores dos elementos do grupo.

21 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO QUE O SERVIDOR WINDOWS E LINUX FORNEÇAM ENDEREÇOS IP (NA SEGUNDA PLACA DE REDE) DA FAMÍLIA 192.168.X.0/24 AOS POSTOS CLIENTES, ONDE X É OBTIDO POR 100 + NÚMERO_DO_GRUPO (EXEMPLO, PARA O GRUPO 99, X=199); PARA O EFEITO DEVO ALTERAR O ENDEREÇO DESSA PLACA ASSIGNADO NAS AULAS PL

CONFIGURAÇÕES:

- > servidor **Linux** com ip fixo **192.168.174.1**
- > servidor **Windows** com ip fixo **192.168.174.2**.

LINUX:

Editamos o ficheiro **/etc/network/interfaces** com o comando **nano /etc/network/interfaces** e alteramos as linhas correspondentes à segunda placa - no nosso caso, a placa **ens33**. Configuramos com o ip **192.168.174.1** e a respetiva máscara **255.255.255.0** como representado na imagem abaixo.

```
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

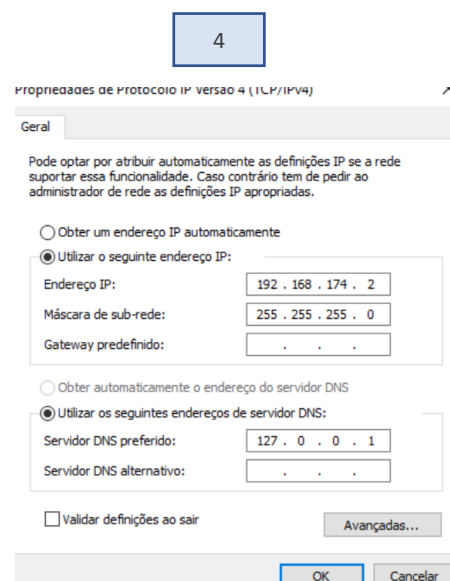
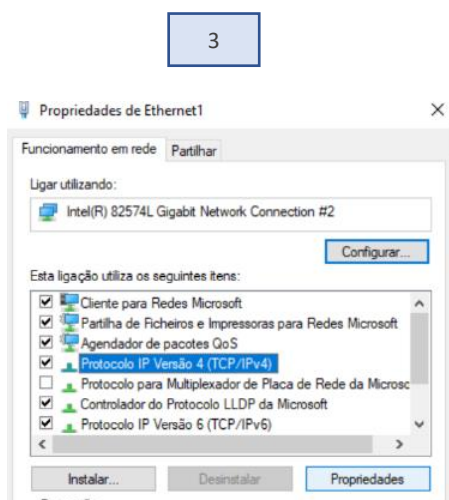
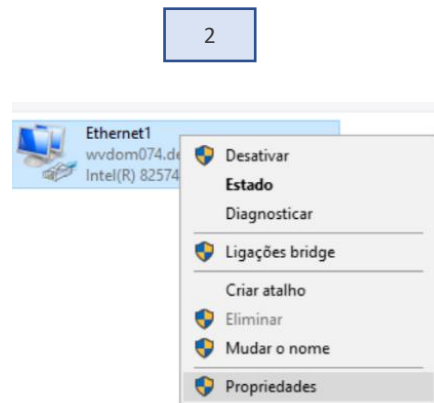
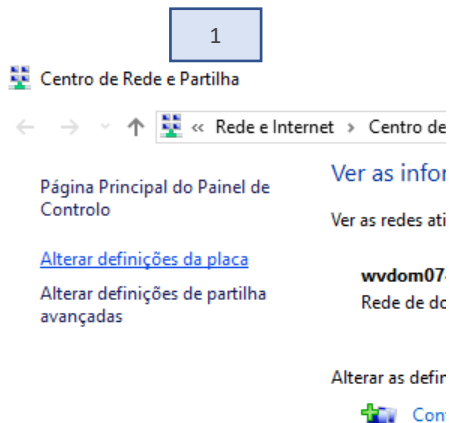
# The primary network interface
auto ens32
iface ens32 inet static
    address 10.9.10.74
    netmask 255.255.0.0
    gateway 10.9.0.1
    dns-nameserver 192.168.62.32 192.168.62.8

# The second network interface
auto ens33
allow-hotplug ens33
iface ens33 inet static
    address 192.168.174.1
    netmask 255.255.255.0

# This is an autoconfigured IPv6 interface
#iface ens33 inet6 dhcp
# address fdle:2bae:c::10:4a
# netmask 64
```

WINDOWS:

Para o Windows, fomos às definições de rede, às propriedades da placa pretendida (segunda placa: **Ethernet1**), selecionamos a ligação que pretendemos, neste caso IPv4, e alteramos o ip para o ip pretendido (**192.168.174.2**).



22 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO QUE OS SERVIÇOS ACIMA REFERIDOS FUNCIONEM EM FAILOVER, COM UM DELES A FACULTAR ENDEREÇOS DE 192.168.X.50 A 192.168.X.150 E O OUTRO DE 192.168.X.151 A 192.168.X.200

Para os sistemas funcionarem em failover, configuramos:

- > servidor **Linux** para atribuir ips entre **192.168.174.50 e 192.168.174.150**
- > servidor **Windows** entre **192.168.174.151 e 192.168.174.200**.

LINUX:

Para configurar o DHCP no servidor Linux, foi necessário instalar o serviço **isc-dhcp-server** com o comando **sudo apt install isc-dhcp-server**.

Uma vez instalado definimos a subnet e o range como solicitado e para tal, foi necessário editar o ficheiro **dhcpd.conf** através do comando **nano /etc/dhcp/dhcpd.conf** e inserir os ips e a máscara pretendida.

```
subnet 192.168.174.0 netmask 255.255.255.0 {  
    range 192.168.174.50 192.168.174.150;  
}
```

Uma vez configurado, definimos a segunda placa, ens33, como sendo a interface com o DHCP, editando o ficheiro **isc-dhcp-server** com o comando **nano /etc/default/isc-dhcp-server**.

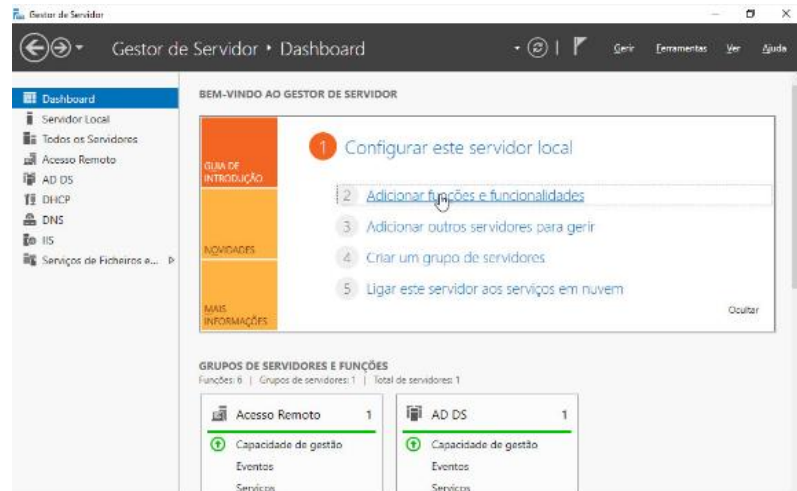
```
# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?  
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".  
INTERFACESv4="ens33"  
INTERFACESv6=""
```

Finalmente, arrancamos com o respetivo serviço com o comando **Systemctl start isc-dhcp-server**.

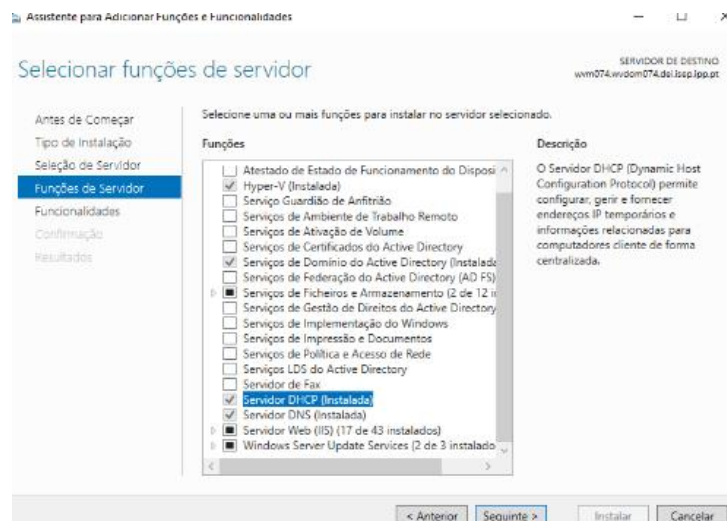
WINDOWS:

Em primeiro lugar, instalamos o servidor DHCP através da opção **adicionar funções e funcionalidades** no gestor de servidor.

1

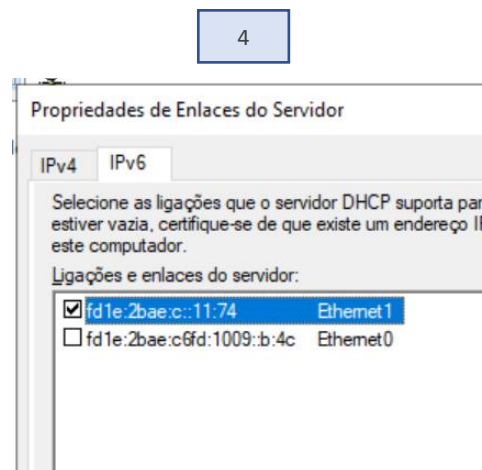
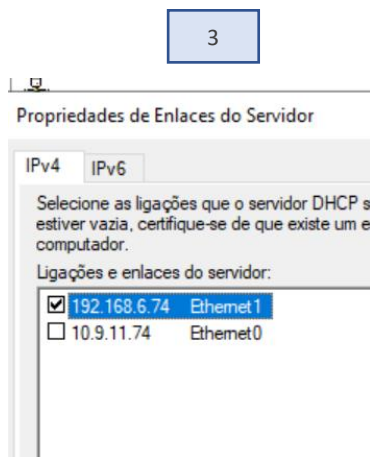
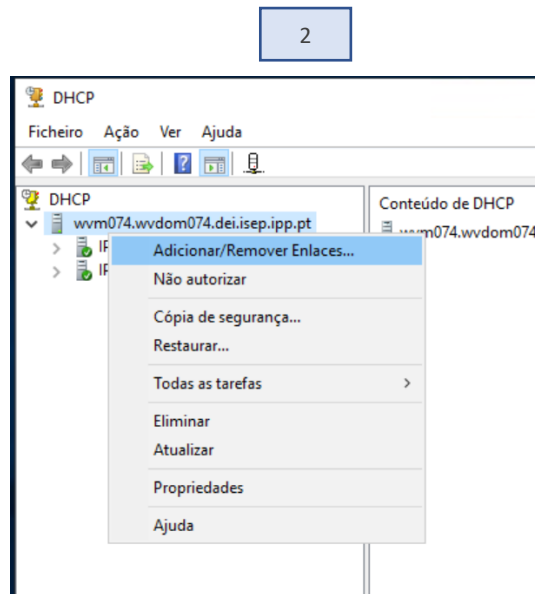


2



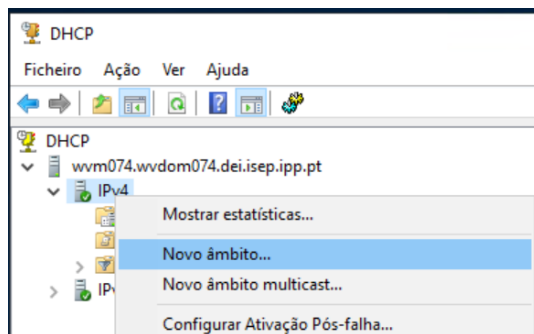
Depois de instalado, na pesquisa apareceu a aplicação **DHCP**. Selecionando-a, a janela mostrou os servidores presentes no sistema.

Para adicionar o range na placa que pretendemos, tivemos que seleccionar a opção **Adicionar/Remover Enlaces** e de seguida seleccionar a placa Ethernet1 (segunda placa), tanto para Ipv4 como para Ipv6.



Para adicionar/configurar o range de atribuição de ips, tivemos que selecionar a versão de ip (IPv4) e selecionar **Novo Âmbito**. Para finalizar a configuração, seguimos os passos e definimos as configurações necessárias, tais como, **Nome**, **Descrição**, **Range** de ips e respetiva **máscara**, **duração** da concessão e ips a excluir (no nosso caso não foi necessário especificar).

1



2

Assistente de novo âmbito

Nome do âmbito

Tem de fornecer um nome de âmbito de multicast identificativo. Tem também a opção de fornecer uma descrição.



Escreva um nome e uma descrição para este âmbito. Estas informações ajudam a identificar rapidamente como o âmbito deve ser utilizado na rede.

Nome:

Descrição:

< Anterior **Seguinte >** Cancelar

3

Assistente de Novo Âmbito

Intervalo do endereço IP

O intervalo de endereços do âmbito é definido através da identificação de um conjunto de endereços IP consecutivos.



Definições de configuração para Servidor DHCP

Introduza o intervalo de endereços distribuído pelo âmbito.

Endereço IP inicial:

Endereço IP final:

Definições de configuração propagadas ao Cliente DHCP

Comprimento:

Máscara de sub-rede:

< Anterior **Seguinte >** Cancelar

4

Assistente de novo âmbito

Duração da concessão

A duração da concessão especifica o período de tempo durante o qual um cliente pode utilizar um endereço IP deste âmbito.



Regra geral, as durações das concessões devem ser iguais ao tempo médio de ligação do computador à mesma rede física. Para redes móveis que consistam essencialmente em computadores portáteis ou clientes de acesso telefónico, podem ser úteis concessões mais curtas.

Da mesma forma, para uma rede estável que consista essencialmente em computadores de secretária em localizações fixas, são mais apropriadas concessões de maior duração.

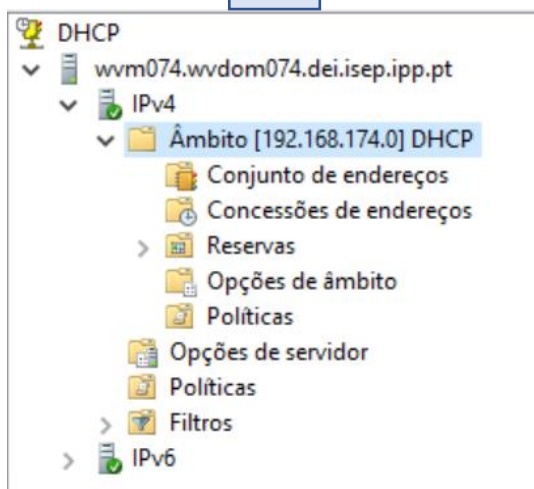
Defina a duração de concessões de âmbito quando distribuídas por este servidor.

Limitadas a:

Dias: Horas: Minutos:

< Anterior **Seguinte >** Cancelar

5



23 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO OS SERVIDORES WINDOWS E LINUX ESTEJAM DISPONÍVEIS APENAS PARA PEDIDOS HTTP E HTTPS. TAL NÃO DEVE IMPEDIR O ACESSO POR SSH OU RDP AOS ADMINISTRADORES (O GRUPO).

LINUX:

A nossa solução passou por bloquear todas as portas e apenas libertar as que pretendíamos. Para isso, as regras de permissão foram inseridas antes da regra de bloqueio, de forma a analisar o tráfego primeiro e só depois bloquear. Para isso foi criado um ficheiro para as iptables com o nome **iptables.sh** criado em **/etc/save** com o comando nano **/etc/save/iptables.sh**.

```
#!/bin/bash

# Setting default policies
iptables -F
iptables -P INPUT DROP

# HTTP
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p udp --dport 80 -j ACCEPT

# HTTPS
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
iptables -A INPUT -p udp --dport 443 -j ACCEPT

# SSH
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A INPUT -p udp --dport 22 -j ACCEPT

# Established Connections
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Para memorizar permanentemente as configurações das iptables, foi executado o comando **iptables-save > /etc/save/iptables.sh**.

```
root@uvm074:~# iptables-save > /etc/save/iptables.sh
```

De forma a carregar as iptables quando o servidor é desligado ou reiniciado, foi necessário adicionar o comando **iptables-restore > /etc/save/iptables.sh** ao ficheiro **crontab**.

O ficheiro **crontab** no Linux é um daemon que executa tarefas editadas pelo usuário em horários e eventos específicos. Neste caso, não temos horário definido é apenas no arranque da máquina. Alteramos o ficheiro e colocamos o comando.

```

GNU nano 5.4
Edit this file to introduce tasks to be run by cron.

Each task to run has to be defined through a single line
indicating with different fields when the task will be run
and what command to run for the task

To define the time you can provide concrete values for
minute (m), hour (h), day of month (dom), month (mon),
and day of week (dow) or use '*' in these fields (for 'any').

Notice that tasks will be started based on the cron's system
daemon's notion of time and timezones.

Output of the crontab jobs (including errors) is sent through
email to the user the crontab file belongs to (unless redirected).

For example, you can run a backup of all your user accounts
at 5 a.m every week with:
0 5 * * 1 tar -zcf /var/backups/home.tgz /home/

For more information see the manual pages of crontab(5) and cron(8)

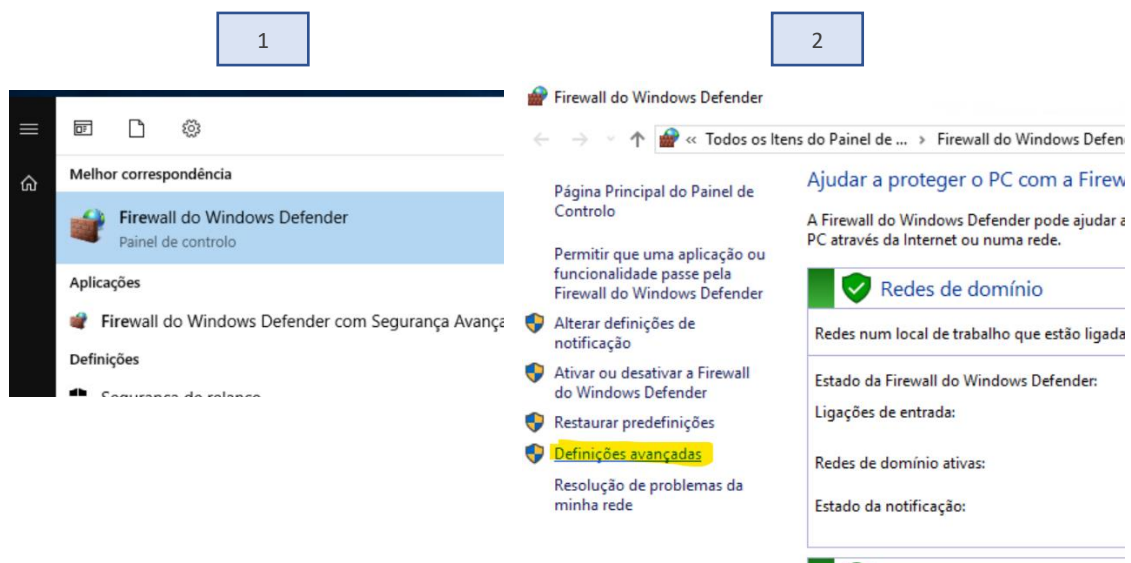
m h dom mon dow   command

Comando para restarurar as iptables no arranque
reboot iptables-restore>/etc/save/iptables.sh

```

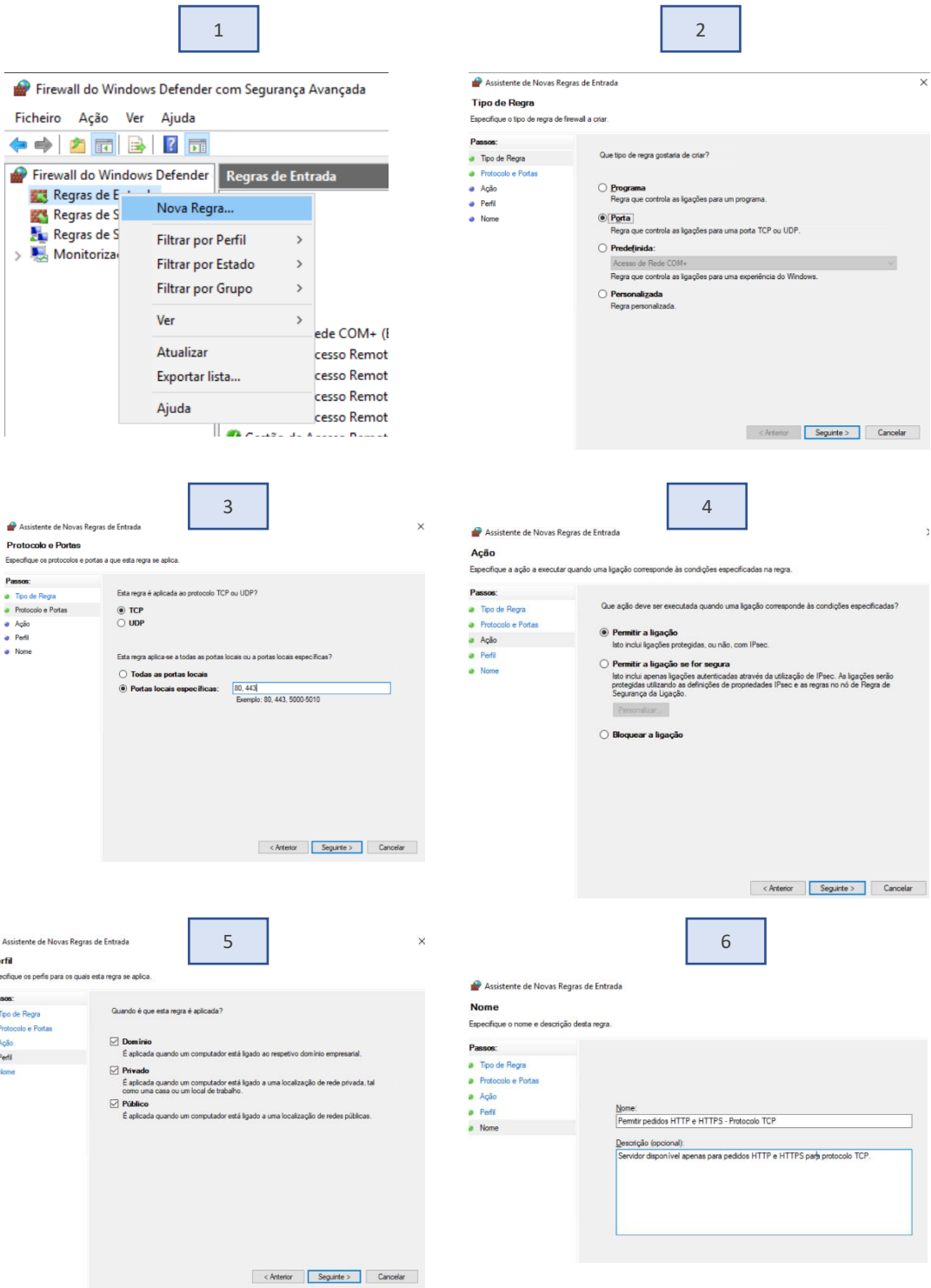
WINDOWS:

No sistema Windows não há ordem de execução de regras na firewall, ou seja, estando a Firewall ativa, é permitido apenas o tráfego indicado nas regras sendo o restante bloqueado por defeito. Para o servidor ficar apenas disponível para pedidos HTTP e HTTPS, foi necessário aceder às definições avançadas da Firewall.

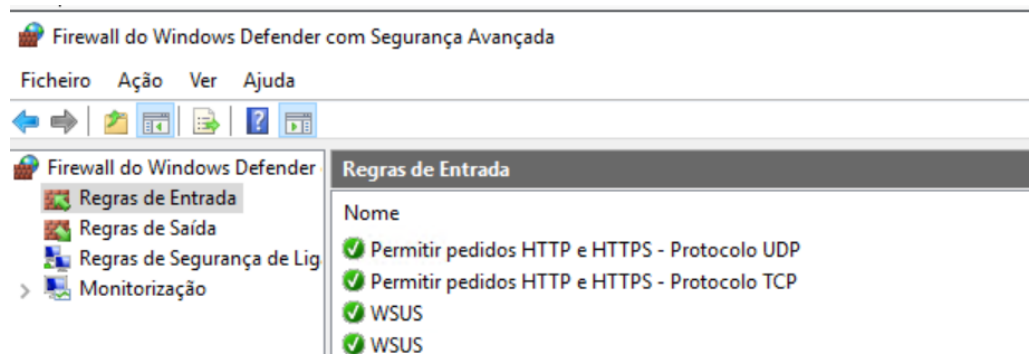


Dentro da Firewall foi necessário criar as regras de entrada. Criamos duas regras, uma para permitir pedidos HTTP e HTTPs para o protocolo TCP, nas portas correspondentes 80 e 443, e uma segunda com as mesmas configurações para o protocolo UDP.

Exemplo para protocolo TCP:



Por fim, as regras criadas:



24 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO IMPEDIR O IP SPOOFING NA MINHA REDE

LINUX:

Para evitar o *spoofing* na nossa rede, que pode ser definida pela apropriação de uma identidade numa rede de computadores, bloqueamos os pacotes das redes privadas. Isto que significa que as solicitações enviadas com estes endereços de ips seriam impossíveis, pois indicariam que existe uma rede privada dentro de nossa rede. Também bloqueamos o intervalo de ips na nossa rede.

```
root@uvvm074:~# nano anti_spoofing.sh
```

```
#!/bin/bash

INT_IF="ens32" # connected to internet
SERVER_IP="202.54.10.20" # server IP
LAN_RANGE="192.168.1.0/24" # your LAN IP range

# Add your spoofed IP range/IPs here
SPOOF_IPS="0.0.0.0/8 127.0.0.0/8 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16 224.0.0.0/3 192.168.174.1/24"

IPT="/sbin/iptables" # path to iptables

# default action, can be DROP or REJECT
ACTION="DROP"

# Drop packet that claiming from our own server on WAN port
$IPT -A INPUT -i $INT_IF -s $SERVER_IP -j $ACTION
$IPT -A OUTPUT -o $INT_IF -s $SERVER_IP -j $ACTION

# Drop packet that claiming from our own internal LAN on WAN port
$IPT -A INPUT -i $INT_IF -s $LAN_RANGE -j $ACTION
$IPT -A OUTPUT -o $INT_IF -s $LAN_RANGE -j $ACTION

## Drop all spoofed
for ip in $SPOOF_IPS
do
    $IPT -A INPUT -i $INT_IF -s $ip -j $ACTION
    $IPT -A OUTPUT -o $INT_IF -s $ip -j $ACTION
done
## add or call your rest of script below to customize iptables ##

iptables -A OUTPUT -p all -s 192.168.174.0/24 -j ACCEPT
iptables -A INPUT -p all -s 192.168.174.0/24 -j ACCEPT
```

A entrada `net.ipv4.conf.all.rp_filter = 1` ativa a verificação do endereço de origem, que é embutida no próprio kernel do Linux e as duas últimas linhas registam todos esses pacotes falsificados no arquivo de log.

```
uvrn074 - Root
GNU nano 5.4 /e
# Do not accept ICMP redirects (prevent MITM attacks)
#net.ipv4.conf.all.accept_redirects = 0
#net.ipv6.conf.all.accept_redirects = 0
# _or_
# Accept ICMP redirects only for gateways listed in our default
# gateway list (enabled by default)
# net.ipv4.conf.all.secure_redirects = 1
#
# Do not send ICMP redirects (we are not a router)
#net.ipv4.conf.all.send_redirects = 0
#
# Do not accept IP source route packets (we are not a router)
#net.ipv4.conf.all.accept_source_route = 0
#net.ipv6.conf.all.accept_source_route = 0
#
# Log Martian Packets
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.log_martians = 1
#
#####
# Magic system request Key
# 0=disable, 1=enable all, >1 bitmask of sysrq functions
# See https://www.kernel.org/doc/html/latest/admin-guide/sysrq.html
# for what other values do
#kernel.sysrq=438
```

WINDOWS:

Para impedir o IP spoofing no servidor Windows, foi necessário criar regras de entrada e de saída.

Para as regras de entrada, foi necessário criar uma para aceitar entrada de IPs na nossa gama de IPs **192.168.174.0/24** e uma para bloquear as várias gamas, tais como: **0.0.0.0/8** - Rede atual (válido apenas como endereço de origem), **127.0.0.0/8** – ip privado, **10.0.0.0/8** – ip privado,

- > **172.16.0.0/12** – ip privado, **192.168.0.0/16**, **224.0.0.0/3**, **192.168.174.2/24** (IP do próprio servidor).

Exemplo de criação de regra de entrada para impedir acesso:

1

Firewall do Windows Defender com Segurança Avançada

Ficheiro Ação Ver Ajuda

Regras de Segurança de Ligação

Regras de Entrada

Regras de Saída

Regras de Segurança de Lig

Monitorização

Firewall

Regras de Segurança de

Associações de seguran

Nome

✓ Permitir pedidos HTTP e HTTPS - Proto

✓ Permitir pedidos HTTP e HTTPS - Proto

✓ WSUS

✓ WSUS

✓ A sua conta

✓ A sua conta

✓ A sua conta

Acesso de Rede COM+ (Entrada de DC

✓ Gestão de Acesso Remoto (Entrada de

✓ Gestão de Acesso Remoto (Entrada de

2

Assistente de Novas Regras de Entrada

Tipo de Regra

Especifique o tipo de regra de firewall a criar.

Passos:

Tipo de Regra

Programa

Protocolo e Portas

Âmbito

Ação

Perfil

Nome

Que tipo de regra gostaria de criar?

☐ Programa

Regra que controla as ligações para um programa.

☐ Porta

Regra que controla as ligações para uma porta TCP ou UDP.

☐ Predefinida:

Acesso de Rede COM+.

Regra que controla as ligações para uma experiência do Windows.

☒ Personalizada

Regra personalizada.

3

Assistente de Novas Regras de Entrada

Programa

Especifique o caminho de programa completo e o nome executável do programa ao qual esta regra corresponde.

Passos:

Tipo de Regra

Programa

Protocolo e Portas

Âmbito

Ação

Perfil

Nome

Esta regra é aplicável a todos os programas ou a um programa específico?

☒ Todos os programas

A regra é aplicável a todas as ligações no computador que correspondam a outras propriedades da regra.

☐ Este caminho de programa:

Procurar...

Exemplo: c:\path\program.exe
%ProgramFiles%\browser\browser.exe

Serviços

Especificar os serviços a que esta regra se aplica.

Personalizar...

4

Assistente de Novas Regras de Entrada

Protocolo e Portas

Especifique os protocolos e portas a que esta regra se aplica.

Passos:

Tipo de Regra

Programa

Protocolo e Portas

Âmbito

Ação

Perfil

Nome

A que portas e protocolos se aplica esta regra?

Tipo de protocolo:

Qualquer

Número de protocolo:

0

Porta local:

Todas as portas

Exemplo: 80, 443, 5000-5010

Porta remota:

Todas as portas

Exemplo: 80, 443, 5000-5010

Definições de protocolo ICMP:

Personalizar

5

Endereço IP local

☐ Qualquer endereço IP

☒ Estes endereços IP:

192.168.174.2

Adicionar...

Editar...

Remover

Endereço IP remoto

☐ Qualquer endereço IP

☒ Estes endereços IP:

127.0.0.0/8
10.0.0.0/8
172.16.0.0/12
192.168.0.0/16
224.0.0.0/3
0.0.0.0/8

Adicionar...

Editar...

Remover

6

Assistente de Novas Regras de Entrada

Ação

Especifique a ação a executar quando uma ligação corresponde às condições especificadas na regra.

Passos:

Tipo de Regra

Programa

Protocolo e Portas

Âmbito

Ação

Perfil

Nome

Que ação deve ser executada quando uma ligação corresponde às condições especificadas?

☐ Permitir a ligação

Isto inclui ligações protegidas, ou não, com IPsec.

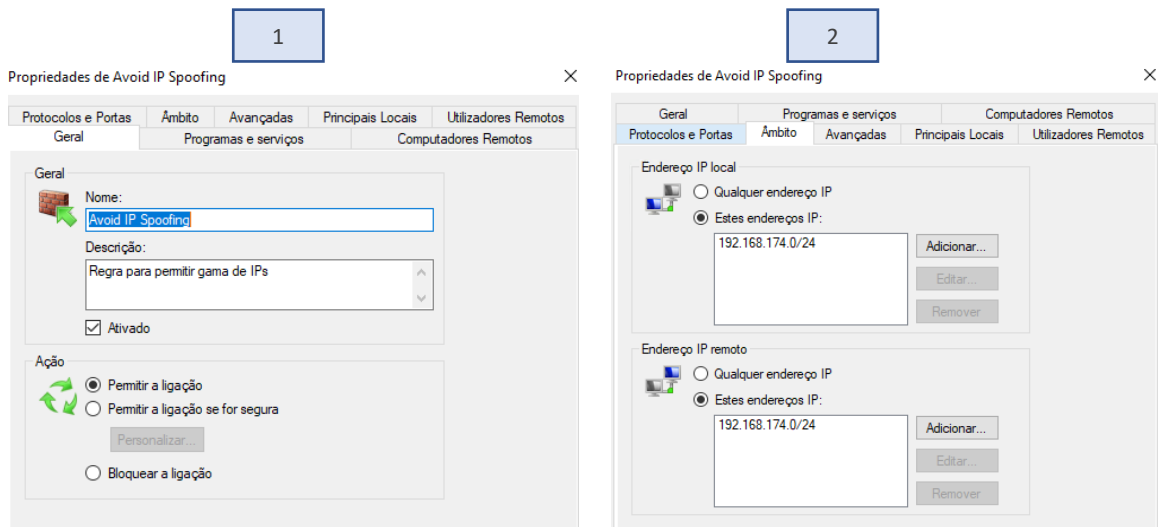
☐ Permitir a ligação se for segura

Isto inclui apenas ligações autenticadas através da utilização de IPsec. As ligações serão protegidas utilizando as definições de propriedades IPsec e as regras no nó de Regra de Segurança da Ligação.

Personalizar...

☒ Bloquear a ligação

Regra de entrada para permitir acesso a gama de IPs **192.168.174.0/24**:



Resultado das regras de entrada:

Firewall do Windows Defender com Segurança Avançada

Ficheiro Ação Ver Ajuda

Regras de Entrada									
Nome	Grupo	Perfil	Ativado	Ação	Contornar	Programa	Endereço local	Endereço remoto	
Avoid IP Spoofing		Tudo	Sim	Permi...	Não	Qualquer	192.168.174.0/24	192.168.174.0/24	
Avoid IP Spoofing		Tudo	Sim	Block	Não	Qualquer	192.168.174.2	127.0.0.0/8, 10.0.0.0/8	
Permitir pedidos HTTP e HTTPS - Protocolo TCP		Tudo	Sim	Permi...	Não	Qualquer	Qualquer	Qualquer	
Permitir pedidos HTTP e HTTPS - Protocolo UDP		Tudo	Sim	Permi...	Não	Qualquer	Qualquer	Qualquer	
WSUS		Tudo	Sim	Permi...	Não	Qualquer	Qualquer	Qualquer	

Para as regras de saída, foram criadas as regras com as mesmas configurações:

Firewall do Windows Defender com Segurança Avançada

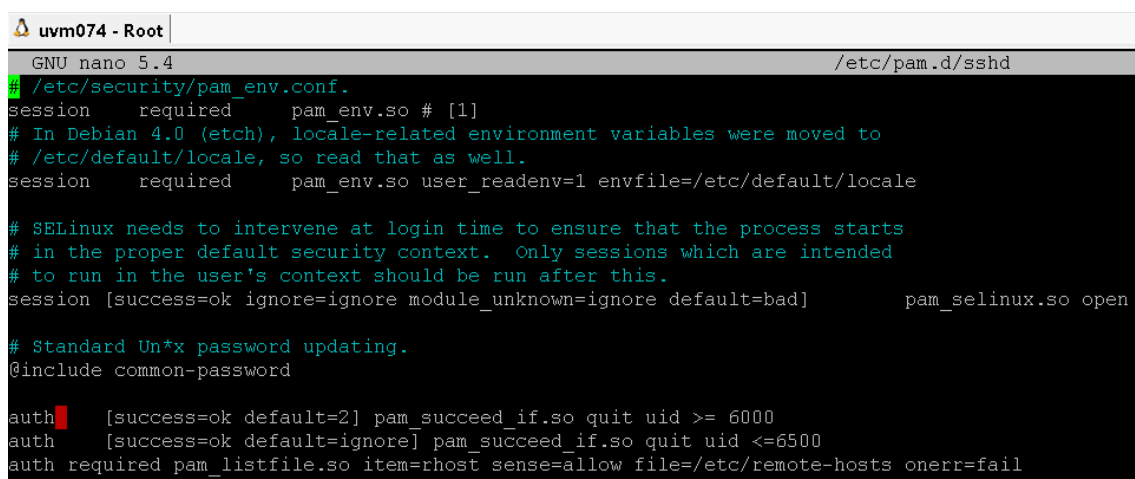
Ficheiro Ação Ver Ajuda

Firewall do Windows Defender com Segurança Avançada em Computador local

Nome	Grupo	Perfil	Ativado	Ação	Contornar	Programa	Endereço local	Endereço remoto	Protocolo
Avoid IP Spoofing		Tudo	Sim	Block	Não	Qualquer	192.168.174.0/24	0.0.0.0/8, 127.0.0.0/8	Qualquer
Avoid IP Spoofing		Tudo	Sim	Permi...	Não	Qualquer	192.168.174.0/24	192.168.174.0/24	Qualquer
@FirewallAPI.dll - 5022	Gestão do Servidor DHCP	Tudo	Sim	Permi...	Não	%system...	Qualquer	Qualquer	TCP
A sua conta	A sua conta	Tudo	Sim	Permi...	Não	Qualquer	Qualquer	Qualquer	Qualquer
A sua conta	A sua conta	Tudo	Sim	Permi...	Não	Qualquer	Qualquer	Qualquer	Qualquer
A sua conta	A sua conta	Tudo	Sim	Permi...	Não	Qualquer	Qualquer	Qualquer	Qualquer

25 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO QUE OS UTILIZADORES REGISTADOS NO LINUX COM UID ENTRE 6000 E 6500 SÓ CONSIGAM ACEDER VIA SSH SE ESSE ACESSO FOR A PARTIR DE UMA MÁQUINA LISTADA EM /ETC/REMOTE-HOSTS

O módulo PAM, `pam_limits`, define os limites dos recursos do sistema que podem ser obtidos numa sessão do utilizador ou seja, PAM é o modulo que nega sempre os acessos, salvo indicação do contrário. Tendo isto em mente, abrimos o ficheiro `/etc/pam.d/sshd` e inserimos no final três linhas de código (ver imagem abaixo). A primeira linha irá testar se o UID do user em questão é igual ou superior a 6000, caso não seja, por defeito, irá saltar as duas linhas, descartando assim a necessidade de verificar se o IP está na lista de IPs autorizados. Caso seja superior a 6000, irá verificar na segunda linha se o UID é igual ou inferior a 6500 o que, caso não seja, irá ignorar e tentar a terceira linha. A terceira linha irá verificar o ficheiro `/etc/remote-hosts` para ver se o IP da sessão que está a ser testada pertence à lista de IPs aceites.



```
uvm074 - Root |
GNU nano 5.4 /etc/pam.d/sshd
# /etc/security/pam_env.conf.
session required pam_env.so # [1]
# In Debian 4.0 (etch), locale-related environment variables were moved to
# /etc/default/locale, so read that as well.
session required pam_env.so user_readenv=1 envfile=/etc/default/locale

# SELinux needs to intervene at login time to ensure that the process starts
# in the proper default security context. Only sessions which are intended
# to run in the user's context should be run after this.
session [success=ok ignore=ignore module_unknown=ignore default=bad] pam_selinux.so open

# Standard Unix password updating.
@include common-password

auth [success=ok default=2] pam_succeed_if.so quit uid >= 6000
auth [success=ok default=ignore] pam_succeed_if.so quit uid <=6500
auth required pam_listfile.so item=rhost sense=allow file=/etc/remote-hosts onerr=fail
```

Após definir este ficheiro, criamos então o ficheiro `/etc/remote-hosts` e inserimos um IP de uma das nossas máquinas para testarmos.



```
root@uvm074:~# nano /etc/remote-hosts
GNU nano 5.4
10.8.1.1
```

26 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO QUE O ACESSO AO SISTEMA SEJA INIBIDO AOS UTILIZADORES LISTADOS EM /ETC/BAD-GUYS

Iniciamos o processo criando o ficheiro bad-guys com os users que pretendíamos inibir o acesso.

1

```
root@uvm074:~# nano /etc/bad-guys
root@uvm074:~#
```

2

```
GNU nano 5.4
# Os users listados neste ficheiro não podem aceder ao servidor

# Users bloqueados:
luser1
luser2
```

Para configurar e inibir o respetivo acesso, foi necessário editar o ficheiro **common-auth** com o comando **nano /etc/pam.d/common-auth** e adicionar a seguinte linha:

auth required pam_listfile.so item=user sense=deny file=/etc/bad-guys onerr=succeed

```
# here are the per-package modules (the "Primary" block)
auth [success=2 default=ignore] pam_unix.so nullok
auth [success=1 default=ignore] pam_ldap.so minimum_uid=1000 use_first_pass
# here's the fallback if no module succeeds
auth requisite pam_deny.so
# prime the stack with a positive return value if there isn't one already;
# this avoids us returning an error just because nothing sets a success code
# since the modules above will each just jump around
auth required pam_permit.so
# and here are more per-package modules (the "Additional" block)
# end of pam-auth-update config

#A linha abaixo serve para retirar acessos aos utilizadores no ficheiro /etc/bad-guys
auth required pam_listfile.so item=user sense=deny file=/etc/bad-guys onerr=succeed
```

O **pam_listfile** é um módulo **PAM** que dá a possibilidade de negar ou permitir serviços baseados num ficheiro arbitrário. Nessa mesma configuração, o argumento **item = user** indica a **pam_listfile** que este deve encontrar nomes de utilizadores em **//etc/bad-guys**.

O argumento **sense = deny** indica a **pam_listfile** que o ficheiro **bad-guys**, em **file=//etc/bad-guys**, é uma lista de negação, ou seja, qualquer utilizador nesse ficheiro faz com que o **pam_listfile** falhe, e por consequente, fazendo com que a autenticação falhe.

O argumento **onerr = succeed** indica que a condição de sucesso é erro.

Por último, foi necessário reiniciar os serviços através dos comandos **sudo service ssh restart** e **sudo service sshd restart**.

```
root@uvm074:~# service ssh restart
root@uvm074:~# service sshd restart
```

Como podemos ver, ao utilizador luser2 que estava presente na lista, não é permitido iniciar sessão.

```
Using username "luser2".
Pre-authentication banner message from server:

#####
=====
      Group 79
=====
#####
End of banner message from server
Access denied
luser2@uvm074.dei.isep.ipp.pt's password: █
```

27 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO QUE AS MENSAGENS PRÉ-LOGIN E PÓS-LOGIN BEM-SUCEDIDO SEJAM DINÂMICAS (POR EXEMPLO, “[BOM DIA] | [BOA TARDE] USERNAME”, ETC.)

LINUX:

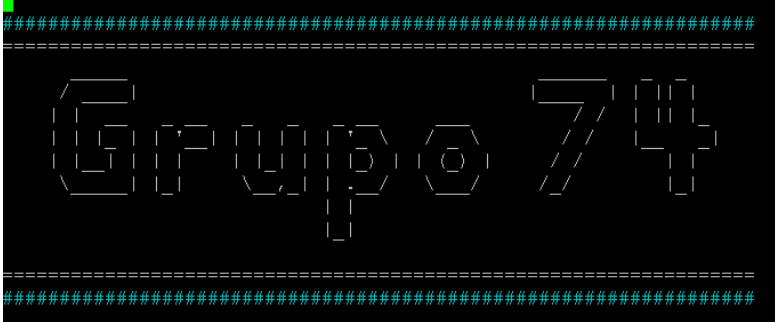
Para resolvermos esta UC, dividimos a questão em duas partes.

A primeira parte consistiu em definir a mensagem de pré login (mensagem apresentada aos utilizadores que estão prestes a entrar e a efetuar o login). Para isso criamos um ficheiro no diretório **/etc/ssh/** com o nome “ssh_banner_pre_login”.

```
root@uvm074:/# nano /etc/ssh/ssh_banner_pre_login
```

Neste ficheiro colocamos a mensagem que gostaríamos de ver ao inicializarmos a máquina.

```
#####  
=====
```



```
=====
```

Depois da criação e edição do ficheiro, vamos ao ficheiro **sshd_config**, localizado no diretório **/etc/ssh/**, e descomentamos a linha que indica o banner (posteriormente estava a banner none) e inserimos o banner definido anteriormente.

```
root@uvm074:~# nano /etc/ssh/sshd_config
```

```

GNU nano 5.4 /etc/ssh/sshd_config
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
PrintMotd no
#PrintLastLog yes
#TCPKeepAlive yes
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
#UseDNS no
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
Banner /etc/ssh/ssh_banner_pre_login

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

# override default of no subsystems
Subsystem sftp /usr/lib/openssh/sftp-server

# Example of overriding settings on a per-user basis
#Match User anoncvs
#    X11Forwarding no
#    AllowTcpForwarding no
#    PermitTTY no
#    ForceCommand cvs server

```

Com disto, damos como concluído a criação do banner de pré-login. Para testarmos, reiniciamos a máquina com o comando **init 6** e comprovamos o resultado.

```
root@uvm074:/# init 6
```

```

login as: root
Pre-authentication banner message from server:

#####
=====
|
|  074
|
|
|=====
|#####
| End of banner message from server
| root@uvm074.dei.isep.ipp.pt's password:

```


28 - COMO ADMINISTRADOR DA INFRAESTRUTURA QUERO QUE O SERVIDOR LINUX RESPONDA E ENVIE PEDIDOS ICMP PARA TESTE DE CONECTIVIDADE APENAS E SÓ AOS COMPUTADORES DOS ELEMENTOS DO GRUPO

Para permitirmos envios de ping de determinadas máquinas podemos usar o comando **iptables**. Este comando funciona de forma ordenada, ou seja, o primeiro comando tem prioridade sobre os demais. Com isto em mente, colocamos a aceitação dos IPs das nossas máquinas nas primeiras linhas, tanto para input como para output. Nas linhas imediatamente a seguir colocamos o *drop* de qualquer IP tanto para input como para output.

Ao fazermos um ping à nossa máquina em Linux, através de uma das nossas máquinas dos elementos do grupo, iremos conseguir obter uma resposta do ping uma vez que os nossos IPs estão na primeira linha da iptables. Caso não seja o IP autorizado, automaticamente passaria para a segunda linha.

```
root@uvm074:~# iptables -A INPUT -s 10.8.1.1 -p ICMP --icmp-type 8 -j ACCEPT
iptables -A INPUT -p ICMP --icmp-type 8 -j DROP

iptables -A OUTPUT -s 10.8.1.1 -p ICMP --icmp-type 8 -j ACCEPT
iptables -A OUTPUT -p ICMP --icmp-type 8 -j DROP
```

```
C:\Users\Rafael>ping -a 10.9.10.74

Pinging uvm074.dei.isep.ipp.pt [10.9.10.74] with 32 bytes of data:
Reply from 10.9.10.74: bytes=32 time=33ms TTL=62
Reply from 10.9.10.74: bytes=32 time=35ms TTL=62
Reply from 10.9.10.74: bytes=32 time=33ms TTL=62
Reply from 10.9.10.74: bytes=32 time=33ms TTL=62

Ping statistics for 10.9.10.74:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 33ms, Maximum = 35ms, Average = 33ms
```