Relatório de ASIST

Grupo 25

Turma 3DE

Relatório realizado por:

* Francisco Tavares nº1181844
* José Cunha nº1181494
* César Ferreira nº1180811
* Sérgio Carreirinha nº1180800

Índice

[Caso de Uso 1 3](#_Toc57991428)

[Caso de Uso 2 4](#_Toc57991429)

[Caso de Uso 3 5](#_Toc57991430)

[Problemática em causa: 5](#_Toc57991431)

[Resolução proposta: 5](#_Toc57991432)

[Caso de Uso 4 8](#_Toc57991433)

[Problemática em causa: 8](#_Toc57991434)

[Proposta de Resolução: 8](#_Toc57991435)

[Caso de Uso 5 9](#_Toc57991436)

[Problemática em causa: 9](#_Toc57991437)

[Proposta de Resolução: 9](#_Toc57991438)

[Caso de Uso 6 10](#_Toc57991439)

[Problemática em causa: 10](#_Toc57991440)

[Resolução proposta: 10](#_Toc57991441)

[Caso de Uso 7 12](#_Toc57991442)

[Caso de Uso 8 13](#_Toc57991443)

# Caso de Uso 1

## Caso de Uso 2

## Caso de Uso 3

## Problemática em causa:

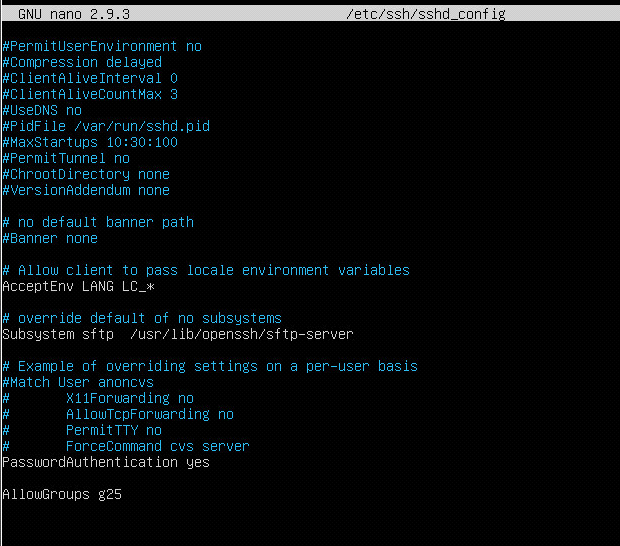
Como administrador da infraestrutura quero os servidores Windows e Linux estejam disponíveis apenas para pedidos HTTP e HTTPS. Tal não deve impedir o acesso por SSH ou RDP aos administradores (o grupo).

### Resolução proposta:

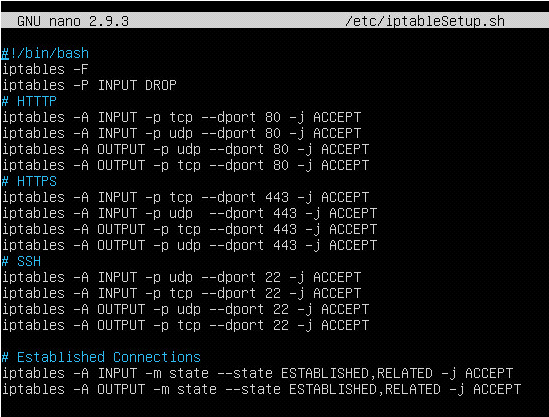
Começamos a nossa abordagem na implementação em linux. Numa fase inicial começamos por auturizar os admins a utilizar SSH, para isso, todos os utilizadores foram adicionados ao grupo criado por nós g25, com o comando "sudo addgroup g25", seguido dos comandos "sudo usermod -a -G g25 1181XXXX"



para cada membro. Depois de todos os membros estarem adicionados corretamente ao grupo foi necessário dar permissões de acesso ao SSH, com o comando "sudo nano /etc/ssh/sshd\_config" acedemos ao ficheiro de configuração e acrescentamos o nosso grupo como permitido com a linha "AllowGroups g25".



Terminada esta parte inicial temos apenas que permitir a passagem dos pacotes SSH nas nossas iptables.



Nesta imagem podemos ver todas as alterações feitas para permitir e bloquear tudo o que nos é pedido. São permitidos pacotes HTTP/HTTPS a todos os utilizadores e permitimos também pacotes SSH, que como sabemos vão estar disponiveis apenas para os elementos do grupo g25.

Certificamo-nos que o script é sempre corrido quando a máquina liga para garantir a mesma não fique desprotegida e que as iptables nunca ficam sem configuração. Como queremos esta funcionalidade criamos um service com a função de correr o script.

-G = To add a supplementary groups.

-a = To add anyone of the group to a secondary group.

## Caso de Uso 4

### Problemática em causa:

Como administrador da infraestrutura quero impedir o IP spoofing na minha rede.

### Proposta de Resolução:

Para esta problemática, o grupo decidiu fazer o IP Spoofing de acordo com o IP Tables, deixando apenas receber ou enviar pacotes com a source de uma das máquinas com IP’s atribuidos pelo DHCP. De acordo com isto, usamos dois comandos:

* sudo iptables -A OUTPUT -p all -s 192.168.125.0/24 -j ACCEPT
* sudo iptables -A INPUT-p all -s 192.168.125.0/24 -j ACCEPT

Estes comandos aceitam (-j ACCEPT) todos os tipos de pacotes (-p all), de uma origem dentro da gama de IP’s definido (-s 192.168.125.0/24).

Por defeito a tabela rejeita todos os tipos de pacotes de qualquer maquina com a origem ou destino, e vai apenas aceitar os pacotes com a origem definida entre a gama de IP’s.

## Caso de Uso 5

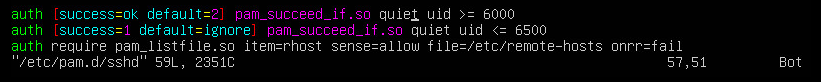
### Problemática em causa:

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.

### Proposta de Resolução:

Para esta problemática, foi necessário alterar o ficheiro que está no localizado no diretório /etc/pam.d/sshd, e inserir alguns comandos.

Os comandos são os seguintes:



Estes comandos verificam se UID está acima de 6000, caso não esteja o default vai saltar duas linhas e não é necessario verificar se o IP da maquina consta no /etc/remote-hosts.

Caso esteja acima, depois verifica se está abaixo dos 6500, caso esteja ele vai para a proxima linha e vai verificar se está entre os IP’s colocados no ficheiro.

Caso não esteja, não vai conseguir aceder via ssh.

## Caso de Uso 6

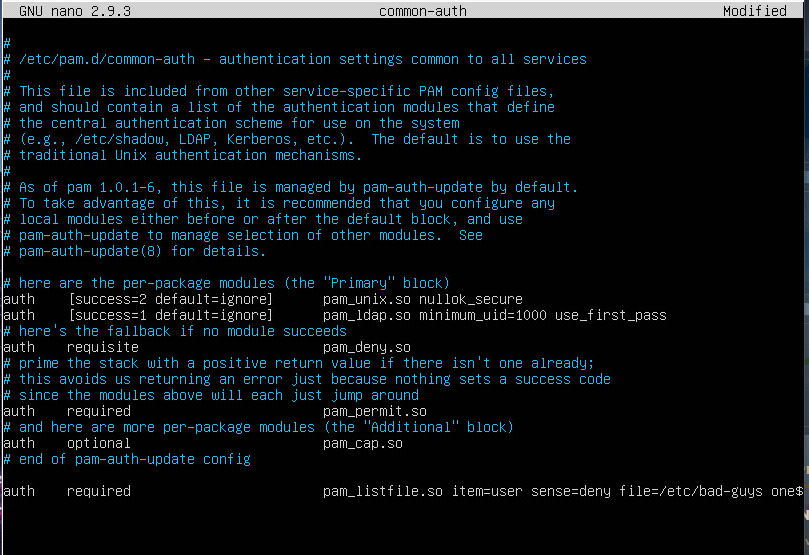
### Problemática em causa:

Como administrador da infraestrutura quero que o acesso ao sistema seja inibido aos utilizadores listados em /etc/bad-guys.

### Resolução proposta:

Começamos por abordar o problema em linux. Primeiro criamos o ficheiro bad-guys na pasta /etc e colocamos lá dentro 4 utilizadores definidos por nós em PLs passadas, luser1, luser2, luser3, luser4.

Depois da pasta estar criada e dos utilizadores já se encontrarem na mesma temos que lhes remover permissões, para isso temos que aceder à pasta /etc/pam.d/common-auth e introduzir a linha seguinte linha:



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

O modulo vai buscar o item do tipo especificado e garantir que este é denied (sense=deny) é também necessário indicar a localização do ficheiro (file=/etc/bad-guys ) e que a condição de sucesso é o err na autenticação do user (onerr=succeed).

Para que as alterações sejam aplicadas é necessário dar restart ao ssh e sshd com os comandos:

sudo service ssh restart

sudo service sshd restart

Depois disto os utilizadores estão bloqueados da máquina.

## Caso de Uso 7

## Caso de Uso 8