1 2 9 0 FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE D COIMBRA

Trabalho Prático#2 – Security Assessment and Improvement

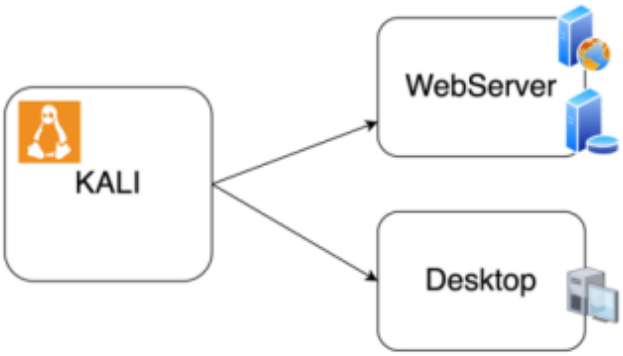


Figure 1 - Assessment Scenario

Mestrado de Segurança Informática

Avaliação e Gestão de Cibersegurança

André Ferreira - 2023188951

Leonardo Pereira – 2020239125

Dezembro 2024

Índice

Introdução

Este trabalho tem como objetivo explorar e aplicar práticas de avaliação e melhoria de segurança em sistemas computacionais, utilizando ferramentas automatizadas e benchmarks reconhecidos. Através do uso de máquinas virtuais configuradas para simular cenários reais, como servidores web e estações de trabalho com vulnerabilidades, é possível entender a dinâmica de identificação de riscos, desenvolvimento de medidas mitigadoras e reavaliação de segurança após implementações corretivas.

O trabalho está dividido em três fases principais: avaliação de segurança, que envolve a análise inicial do sistema para identificar vulnerabilidades e medir a superfície de ataque; melhoria de segurança, onde são propostas e implementadas alterações no sistema visando mitigar os problemas encontrados; e reavaliação de segurança, que analisa as mudanças realizadas para verificar sua eficácia. Este processo busca não apenas reforçar os conhecimentos teóricos sobre cibersegurança, mas também capacitar os participantes para utilizar ferramentas como o framework Metasploit, benchmarks CIS, e outras tecnologias relevantes.

A abordagem prática e a divisão em etapas refletem uma metodologia estruturada e alinhada com as melhores práticas da indústria, garantindo que os objetivos de segurança sejam alcançados sem comprometer a funcionalidade do sistema.

1) Security Assessment

Para a avaliação de segurança, assumiu-se não ter informações adicionais, além do endereço IP e do nome do utilizador na máquina. O **Webserver** está a correr no endereço IP 192.168.56.101, e o **Desktop** está a correr no endereço IP 192.168.56.103, tal como se pode verificar nas imagens a seguir. De forma a evitar duplicação nos comandos efetuados nas diferentes máquinas virtuais, o **x.x.x.x** representa os endereços IP de cada uma delas.

Webserver

```
nuno@localhost:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
\    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
en 1000
    link/ether 08:00:27:fd:52:28 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global eth0
    inet6 fe80::a00:27ff:fefd:5228/64 scope link
       valid_lft forever preferred_lft forever
nuno@localhost:~$
```

Desktop

```
nuno@localhost:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast sta
en 1000
    link/ether 08:00:27:5f:cd:4e brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.103/24 brd 192.168.56.255 scope global eth0
    inet6 fe80::a00:27ff:fe5f:cd4e/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
    link/ether 08:00:27:db:1a:fb brd ff:ff:ff:ff:ff:ff
```

2)Measurement of simplified attack surface

Acesso sem SSH/ Remote Assessment

Open TCP ports:

Para iniciar a avaliação, foi necessário recolher mais informações sobre a máquina. Assim, começamos a análise de segurança executando um comando **nmap**. Este comando foi realizado através do console do **Metasploit** para manter um espaço de trabalho mais organizado e ter toda a informação disponível ao tentar explorar o sistema vulnerável.

Comando utilizado: **db_nmap -v -T4 -PA -sV --version-all -osscan-guess -A -sS -p 1-65535 x.x.x.x**

Este comando utiliza a versão do nmap integrada na base de dados do Metasploit, uma ferramenta de análise de redes, para realizar um scan TCP SYN de todas as portas da máquina-alvo, representada por **x.x.x.x**, que pode ser o endereço IP do Webserver ou do Desktop. O comando utiliza várias opções para ativar a saída detalhada (verbose output), ARP ping, deteção de serviços e versões, identificação do sistema operativo (OS fingerprinting) e traceroute. O resultado fornece informações como as portas TCP abertas, os serviços e as versões em execução nessas portas. Estes dados são relevantes para a avaliação e melhoria de segurança porque revelam partes da superfície de ataque da máquina-alvo e possíveis vulnerabilidades dos serviços.

Webserver

```
port   proto  name     state  info
────   ─────  ────     ─────  ────
21     tcp    ftp      open   vsftpd 2.0.8 or later
22     tcp    ssh      open   OpenSSH 6.0p1 Debian 4+deb7u3 protocol 2.0
80     tcp    http     open   Apache httpd 2.2.22 (Debian)
111    tcp    rpcbind  open   2-4 RPC #100000
40324  tcp    status   open   1 RPC #100024
46740  tcp             open
```

Desktop

```
21     tcp    ftp      open   vsftpd 2.0.8 or later
22     tcp    ssh      open   OpenSSH 6.0p1 Debian 4+deb7u3 protocol 2.0
110    tcp    pop3     open   Openwall popa3d
111    tcp    rpcbind  open   2-4 RPC #100000
3632   tcp    distccd  open   distccd v1 (Debian 4.7.2-5) 4.7.2
33804  tcp    status   open   1 RPC #100024
```

Com estes resultados, já descobrimos alguns pontos de entrada que podem ser explorados para aceder ao sistema, como, por exemplo, ataques de força bruta ao SSH. No entanto, decidimos recolher o máximo de informações possível sobre a máquina antes de tentar aceder a ela. Por isso, executámos outro script do nmap, desta vez para as portas UDP.

Open UDP ports:

Comando utilizado: **db_nmap -v -sU -T5 -p- --max-retries 3 --max-rtt-timeout 100ms --max-scan-delay 10ms -host-timeout 4h x.x.x.x -oA nmap_sU**

Webserver

```
[*] Nmap: PORT       STATE SERVICE
[*] Nmap: 111/udp    open  rpcbind
[*] Nmap: 123/udp    open  ntp
[*] Nmap: 57864/udp  open  unknown
```

Desktop

```
[*] Nmap: PORT       STATE SERVICE
[*] Nmap: 111/udp    open  rpcbind
[*] Nmap: 123/udp    open  ntp
[*] Nmap: 47475/udp  open  unknown
```

Em ambos os scans identificou-se a existência de alguns endpoints **RPC** em ambas as máquinas e portanto decidiu-se aprofundar a análise para obter mais informações sobre estes pontos.

Open RPC endpoints:

Comando utilizado: **rpcinfo -p x.x.x.x**

O comando **rpcinfo** lista todos os serviços RPC registados no **rpcbind** de um host. Se nenhum host for especificado, o host local é utilizado como padrão. Com a opção "-p", é exibida uma lista de todos os programas RPC registados.

Webserver

```
 program vers proto   port  service
  100000    4   tcp    111  portmapper
  100000    3   tcp    111  portmapper
  100000    2   tcp    111  portmapper
  100000    4   udp    111  portmapper
  100000    3   udp    111  portmapper
  100000    2   udp    111  portmapper
  100024    1   udp  43108  status
  100024    1   tcp  49341  status
```

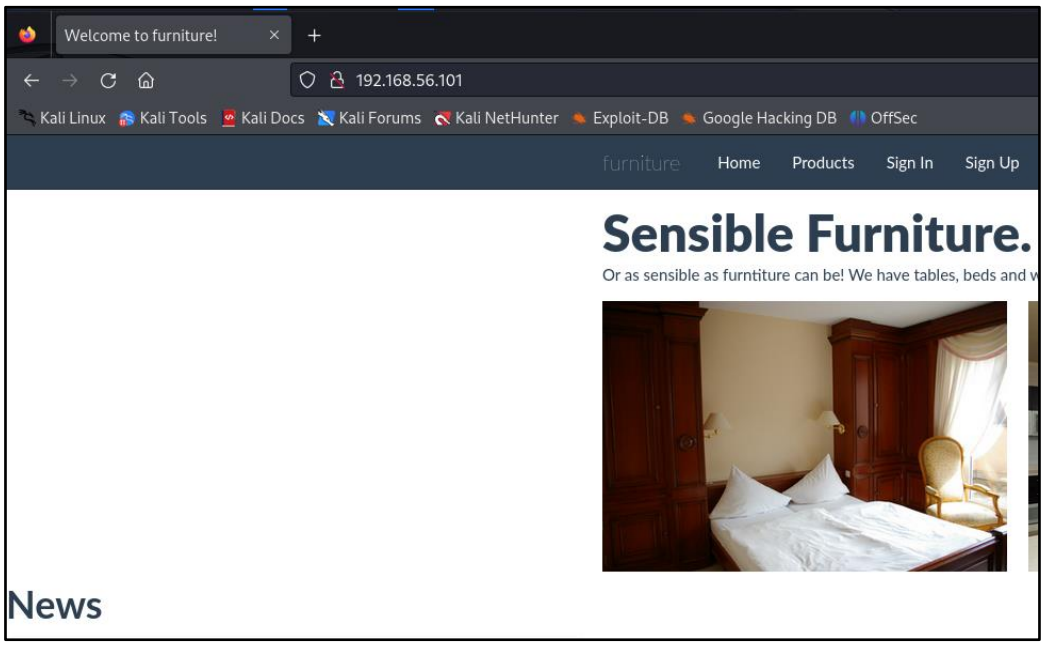
Desktop

```
program vers proto   port  service
 100000    4   tcp    111  portmapper
 100000    3   tcp    111  portmapper
 100000    2   tcp    111  portmapper
 100000    4   udp    111  portmapper
 100000    3   udp    111  portmapper
 100000    2   udp    111  portmapper
 100024    1   udp  48047  status
 100024    1   tcp  33804  status
```

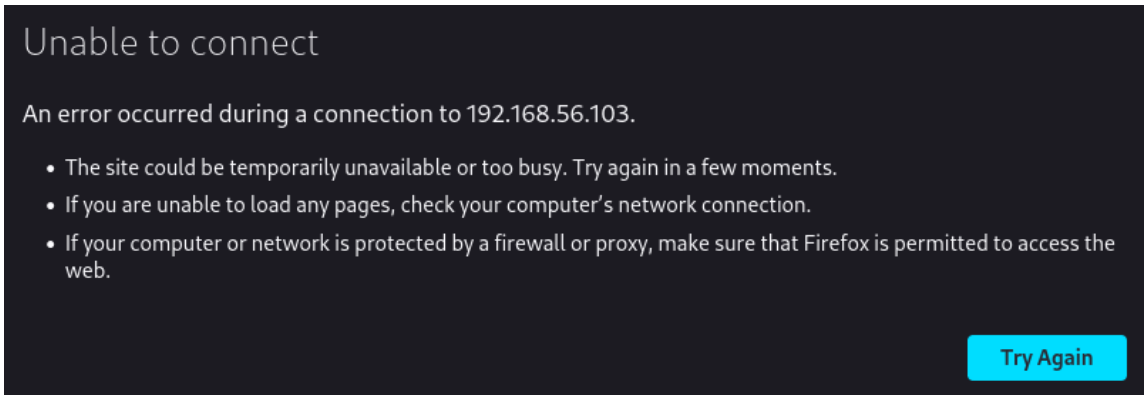
Verificar se o serviço web está a correr no servidor:

Para verificar se os servidores estavam a correr, acedeu-se ao browser no porto 80 e simplesmente introduziu-se o ip de cada uma das máquinas virtuais (webserver e desktop) e verificou-se se corria alguma aplicação web ou não.

Webserver



Desktop



Esta mensagem significa que o **Firefox** (browser utilizado) não conseguiu estabelecer uma conexão com o servidor do Desktop, o que era esperado uma vez que esta máquina só é utilizada para aceder à aplicação Web, enviar e-mails e armazenar ficheiros. Possíveis causas incluem:

1. **Servidor indisponível**: O serviço ou servidor no endereço especificado pode estar desligado ou sobrecarregado.
2. **Problemas de rede**: O computador pode não estar conectado corretamente à rede que contém esse endereço IP.
3. **Configuração de firewall/proxy**: Um firewall ou proxy pode estar a bloquear a tentativa de conexão do Firefox.
4. **Endereço incorreto**: O IP pode estar errado ou não corresponder a nenhum serviço ativo.

Pelos resultados obtidos verificou-se que não havia nenhum servidor a funcionar na máquina virtual Desktop.

Até este ponto, já foi descoberto tudo o que era possível a partir do exterior. Chegou o momento de descobrir a palavra-passe de cada máquina virtual, de forma a medir a superfície de ataque restante a partir do interior.

Descobrir a password:

Após este ponto, decidiu-se tentar obter acesso a cada sistema via SSH. Para tal, utilizou-se o **hydra** que é uma ferramenta de força bruta para testar credenciais de login em serviços remotos.

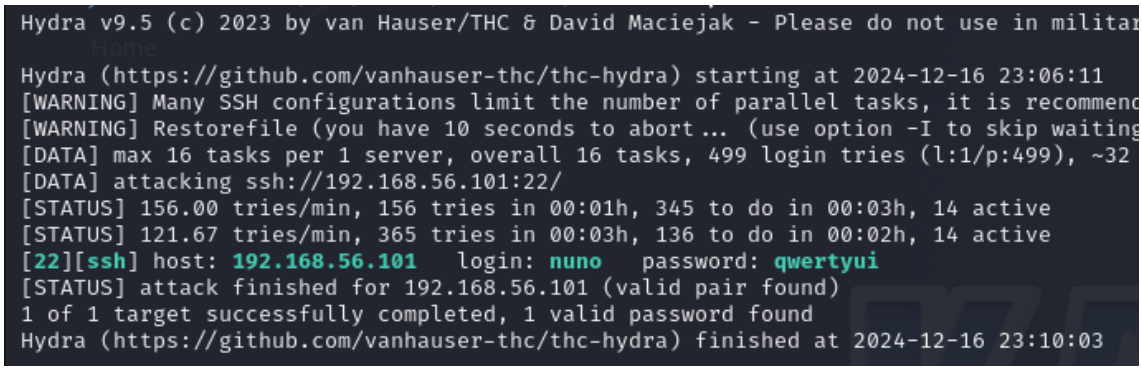
Como já era conhecido o nome de utilizador, tentou-se descobrir ambas as palavras-passe utilizando uma lista com as 500 (quinhentas) palavras-passe mais comuns [1].

Comando utilizado: **hydra -l nuno -P /usr/share/wordlists/500-worst-passwords.txt -f x.x.x.x ssh**

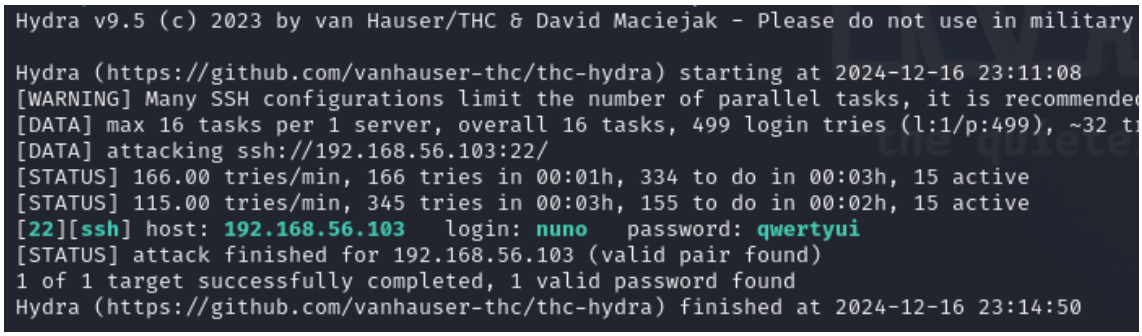
O comando começa por estabelecer uma ligação ao serviço SSH no sistema identificado pelo endereço IP **x.x.x.x**. Ele tenta fazer login repetidamente utilizando o nome de utilizador **nuno** e, para cada tentativa, usa uma palavra-passe retirada da wordlist **500-worst-passwords.txt.** Se encontrar uma combinação correta de nome de utilizador e

palavra-passe, o Hydra exibe essa combinação no terminal e pára imediatamente, graças à flag **-f**. Se não encontrar nenhuma combinação válida, continua até testar todas as palavras-passe disponíveis na lista.

Webserver



Desktop



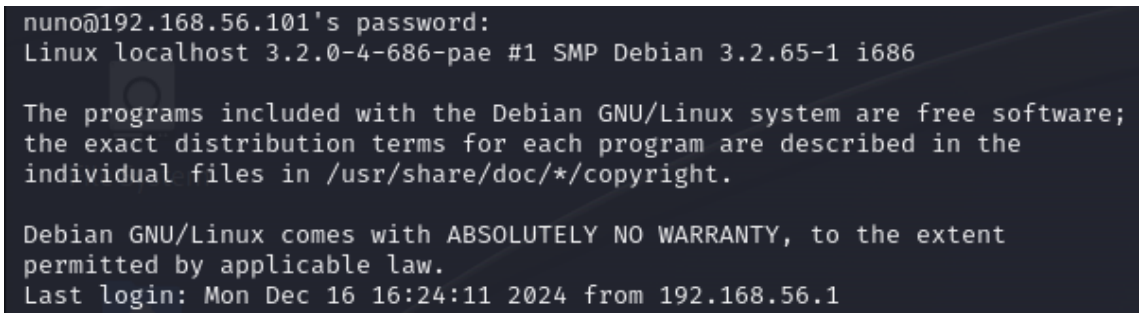
Conseguiu-se obter com sucesso o par de credenciais (**nuno-qwertyui**) para aceder às máquinas virtuais **Webserver e Desktop** via SSH.

Acesso com SSH/ Local Assessment

Chegou o momento de penetrar o Webserver e o Desktop utilizando o serviço ssh.

Comando utilizado: **ssh nuno@x.x.x.x**

Webserver



Desktop

```
The authenticity of host '192.168.56.103 (192.168.56.103)' can't be established.
ECDSA key fingerprint is SHA256:UA7V/8AO88CO6VDqVyoxlE2+Qy62biDkZQXYHKmr7rs.
This host key is known by the following other names/addresses:
    ~/.ssh/known_hosts:1: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.103' (ECDSA) to the list of known hosts.
nuno@192.168.56.103's password:
Linux localhost 3.2.0-4-686-pae #1 SMP Debian 3.2.65-1 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Dec 16 08:03:10 2024
```

Como se pode verificar pelas imagens anteriores, o acesso ao Webserver e ao Desktop utilizando o serviço **ssh** foi bem sucedido. A partir deste momento já é possível executar comandos nas próprias máquinas alvo e obter informação mais detalhada

Open Sockets:

O comando **ss** não é considerada uma ferramenta de attack surface, porque é executada a partir da própria máquina alvo do ataque, para conseguir executar este comando era necessário já estar dentro da própria máquina, o propósito deste comando é apenas para ver os sockets abertos para as comunicações internas, conforme é demonstrado na seguinte imagem.

Comando utilizado: **ss -s**

Webserver

```
root@localhost:/home/nuno# ss -s
Total: 83 (kernel 87)
TCP:   15 (estab 1, closed 1, orphaned 0,

Transport Total     IP        IPv6
*         87        -         -
RAW       0         0         0
UDP       16        9         7
TCP       14        8         6
INET      30        17        13
FRAG      0         0         0
```

Desktop

```
root@localhost:/home/nuno# ss -s
Total: 226 (kernel 228)
TCP:   13 (estab 1, closed 0, orpha

Transport Total     IP        IPv6
*         228       -         -
RAW       0         0         0
UDP       16        9         7
TCP       13        9         4
INET      29        18        11
FRAG      0         0         0
```

Open Named Pipes:

Lsof (List Open Files) é utilizado para listar todos os ficheiros abertos no sistema. Este comando gera uma lista com informações sobre os ficheiros abertos, incluindo o processo que os está a utilizar, o tipo de ficheiro, e outros detalhes.

Depois, o output do **lsof** é enviado para o comando grep 'FIFO'. O comando **grep** filtra os resultados do **lsof** e exibe apenas as linhas que contêm a palavra "FIFO", ou seja, apenas os ficheiros abertos que são pipes FIFO. Por fim, o comando **wc -l** que vai contar o número de linhas no output do grep. Cada linha corresponde a um pipe FIFO aberto que foi identificado pelo **lsof**. Assim, o número final que é apresentado no terminal representa quantos named pipes estão atualmente abertos no sistema.

Comando utilizado: **lsof | grep 'FIFO' | wc -l**

Webserver

```
root@localhost:/home/nuno# lsof | grep 'FIFO' | wc -l
60
```

Desktop

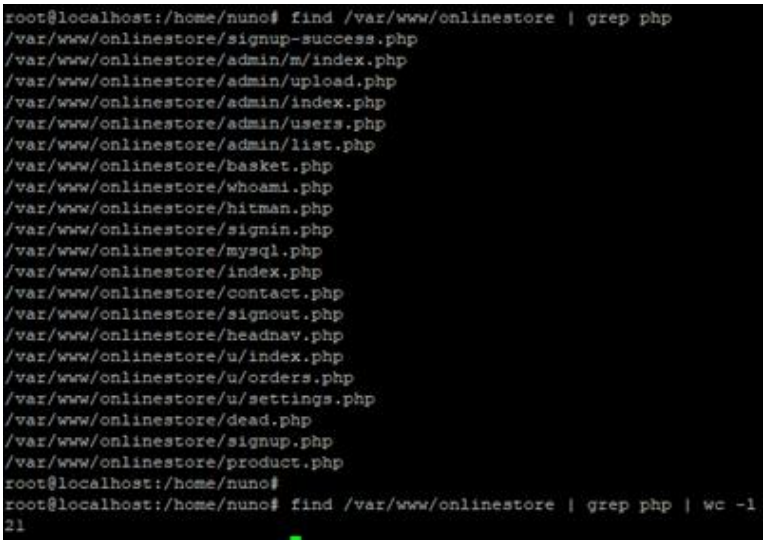
```
root@localhost:/home/nuno# lsof | grep 'FIFO' | wc -l
198
```

Dynamic web pages:

Primeiro, o **find /var/www/onlinestore** vai listar todos os ficheiros e pastas dentro desse caminho. O resultado é passado para o comando **grep**, que filtra apenas os ficheiros cujo nome contém "php". Por fim, o **wc -l** conta o número de linhas no output do **grep**, que corresponde à quantidade de ficheiros PHP encontrados.

Comando utilizado: **find /var/www/onlinestore | grep php | wc -l**

Webserver



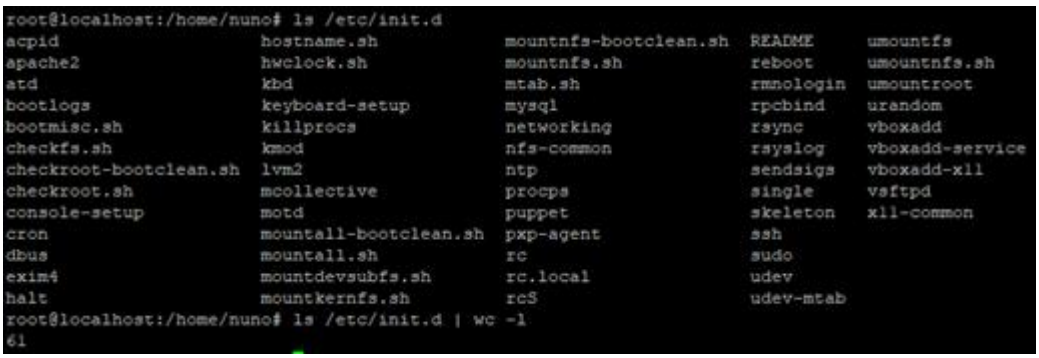
Como o **Desktop** não tem serviço web, não foi possível encontrar ficheiros correspondentes.

Services:

Em primeiro, o **ls /etc/init.d** vai listar todos os ficheiros dentro de /etc/init.d, que é o local onde estão os scripts usados para gerir serviços no sistema. Depois, o output é enviado para o **wc -l**, que conta o número de linhas, representando assim a quantidade total de serviços.

Comando utilizado: **ls /etc/init.d | wc -l**

Webserver



Desktop

```
root@localhost:/home/nuno# ls /etc/init.d
acpid          checkroot-bootclean.sh  distcc
atd            checkroot.sh            exim4
bootlogs       console-setup           halt
bootmisc.sh    cron                    hdparm
checkfs.sh     dbus                    hostname.sh
root@localhost:/home/nuno# ls /etc/init.d | wc -l
66
```

Services running by default:

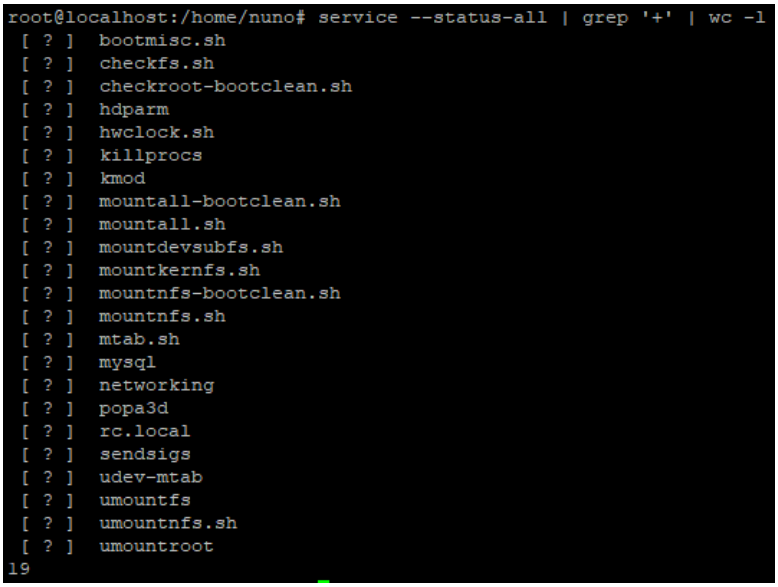
O comando **service --status-all** vai listar todos os serviços disponíveis no sistema, indicando o estado de cada um. O comando **grep** '+' filtra apenas as linhas que representam serviços ativos. Por fim, o **wc -l** conta o número dessas linhas, que corresponde ao número total de serviços em execução.

Comando utilizado: **service --status-all | grep '+' | wc -l**

Webserver

```
root@localhost:/home/nuno# service --status-all | grep '+' | wc -l
 [ ? ]  bootmisc.sh
 [ ? ]  checkfs.sh
 [ ? ]  checkroot-bootclean.sh
 [ ? ]  hwclock.sh
 [ ? ]  killprocs
 [ ? ]  kmod
 [ ? ]  mountall-bootclean.sh
 [ ? ]  mountall.sh
 [ ? ]  mountdevsubfs.sh
 [ ? ]  mountkernfs.sh
 [ ? ]  mountnfs-bootclean.sh
 [ ? ]  mountnfs.sh
 [ ? ]  mtab.sh
 [ ? ]  mysql
 [ ? ]  networking
 [ ? ]  rc.local
 [ ? ]  sendsigs
 [ ? ]  udev-mtab
 [ ? ]  umountfs
 [ ? ]  umountnfs.sh
 [ ? ]  umountroot
17
```

Desktop

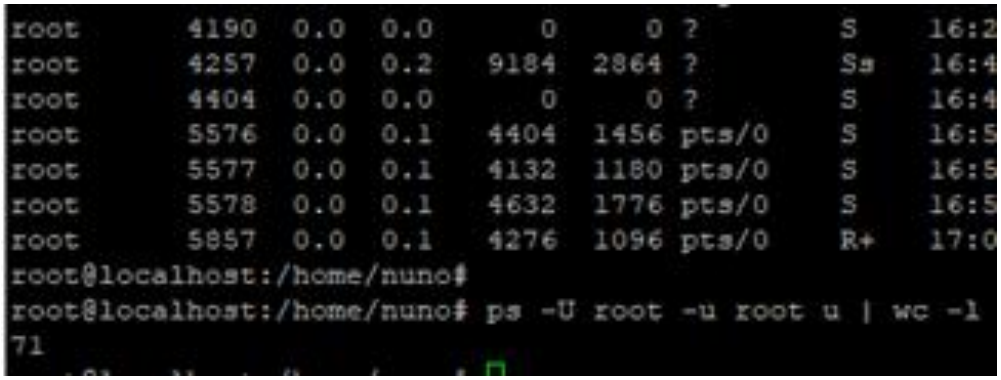


Services running as root:

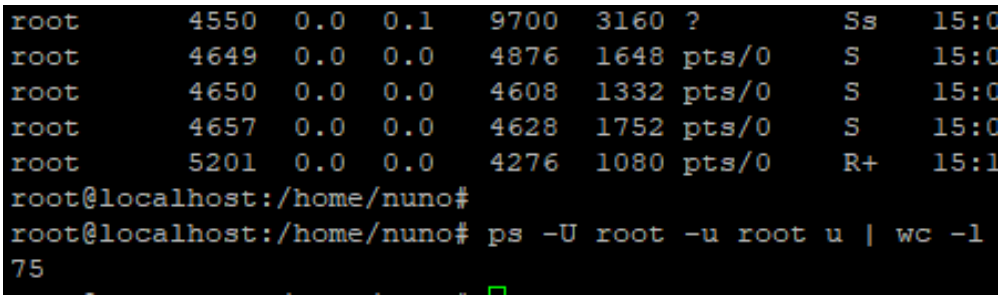
Este comando indica quantos processos estão a correr sob o utilizador root no sistema.

Comando utilizado: **ps -U root -u root u | wc -l**

Webserver



Desktop

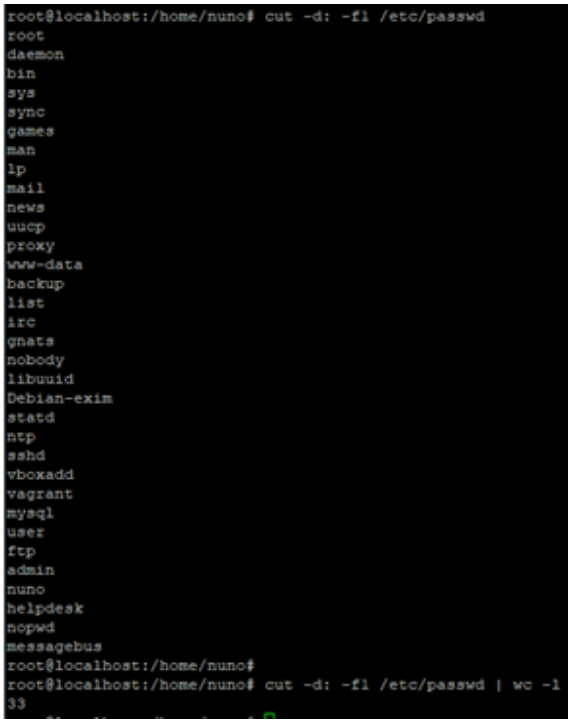


Enabled accounts:

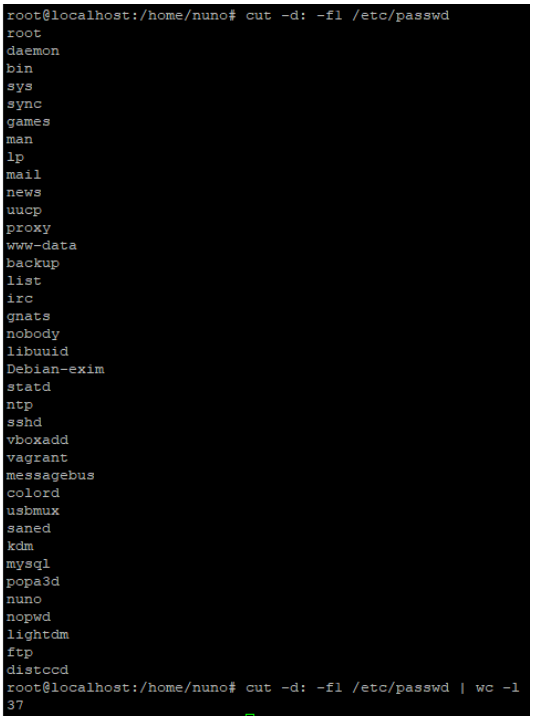
O comando **cut -d: -f1 /etc/passwd** extrai apenas o primeiro campo do ficheiro /etc/passwd, que contém a lista de utilizadores do sistema. O **-d:** define o delimitador como dois pontos (:), e o **-f**1 seleciona o primeiro campo, ou seja, os nomes dos utilizadores. Por fim, o **wc -l** conta o número de linhas no output, o que corresponde ao número total de utilizadores listados no ficheiro /etc/passwd

Comando utilizado: **cut -d: -f1 /etc/passwd | wc -l**

Webserver



Desktop



Enabled accounts in root group:

Em primeiro, o **cut -d: -f1,4 /etc/passwd** extrai o primeiro e o quarto campos do ficheiro /etc/passwd. O primeiro campo representa os nomes dos utilizadores, e o quarto campo é o ID do grupo principal associado a cada utilizador. Depois, o **grep -w 0** filtra as linhas

em que o ID do grupo principal é exatamente 0, e não 10 ou 100 por exemplo. Encontrado assim, quais os utilizadores que pertencem ao grupo de root.

Comando utilizado: **cut -d: -f1,4 /etc/passwd | grep -w 0**

Webserver

```
root@localhost:/home/nuno# cut -d: -f1,4 /etc/passwd | grep -w 0
root:0
```

Desktop

```
root@localhost:/home/nuno# cut -d: -f1,4 /etc/passwd | grep -w 0
root:0
ftp:0
```

Guest accounts:

Este comando vai contar quantos utilizadores com o termo "guest" no nome têm o seu grupo principal especificado no ficheiro

Comando utilizado: **cut -d: -f1,4 /etc/passwd | grep 'guest' | wc -l**

Webserver

```
root@localhost:/home/nuno# cut -d: -f1,4 /etc/passwd | grep 'guest'
root@localhost:/home/nuno# cut -d: -f1,4 /etc/passwd | grep 'guest' | wc -l
0
```

Desktop

```
root@localhost:/home/nuno# cut -d: -f1,4 /etc/passwd | grep 'guest' | wc -l
0
```

Resumo do Attack Surface

Webserver

Avenues of Attack	Number Avenues	Bias	Effective Attack Surface	Area
Open sockets	83	1	83	Channels
Open TCP ports	6	1	6	Channels
Open UDP ports	4	1	4	Channels
Open RPC endpoints	8	0,9	7,2	Channels
Open named pipes	60	0,8	48	Channels
Dynamic web pages	21	0,6	12,6	Process Targets
Services	61	0,2	12,2	Process Targets
Services running by default	17	0,8	13,6	Process Targets
Services running as root	71	0,6	42,6	Process Targets
Enabled accounts	33	0,5	16,5	Access Rights
Enabled accounts in root group	1	0,9	0,9	Access Rights
Guest accounts enabled	0	0,9	0	Access Rights
		RASQ:	246,6	

Desktop

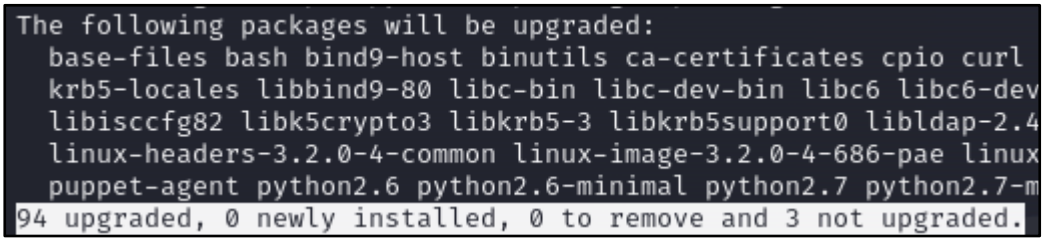
Avenues of Attack	Number Avenues	Bias	Effective Attack Surface	Area
Open sockets	226	1	226	Channels
Open TCP ports	6	1	6	Channels
Open UDP ports	3	1	3	Channels
Open RPC endpoints	8	0,9	7,2	Channels
Open named pipes	198	0,8	158,4	Channels
Dynamic web pages	0	0,6	0	Process Targets
Services	66	0,2	13,2	Process Targets
Services running by default	19	0,8	15,2	Process Targets
Services running as root	75	0,6	45	Process Targets
Enabled accounts	37	0,5	18,5	Access Rights
Enabled accounts in root group	2	0,9	1,8	Access Rights
Guest accounts enabled	0	0,9	0	Access Rights
		RASQ:	494,3	

O Relative Attack Surface Quotient (**RASQ**) é uma métrica desenvolvida para avaliar a superfície de ataque a um sistema, ajudando a medir o quão exposto ele está a possíveis ataques. Esta métrica foi criada para comparar diferentes versões de software ou configurações de sistemas, indicando qual é a mais segura ou mais vulnerável.

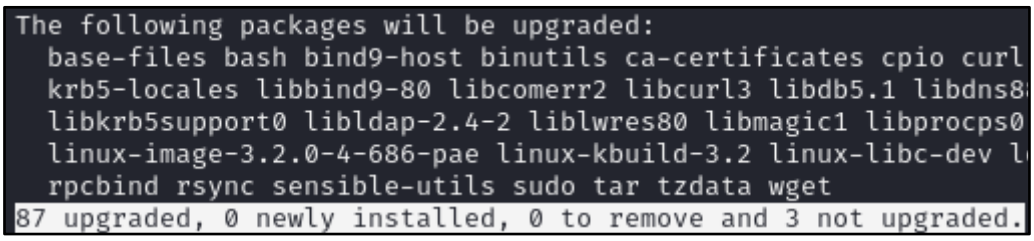
a) Check the system for outdated software and famous vulnerabilities

Para detectar o número de pacotes nas máquinas alvo de investigação, foi efetuado o comando **apt-get -s upgrade**, de forma a ver quais os pacotes que é preciso atualizar, concluindo que, para o Webserver ficar atualizado, necessitava de atualizar **94** pacotes e para o Desktop ficar atualizado, precisa de atualizar **87** pacotes. As seguintes imagens provam o que anteriormente foi dito.

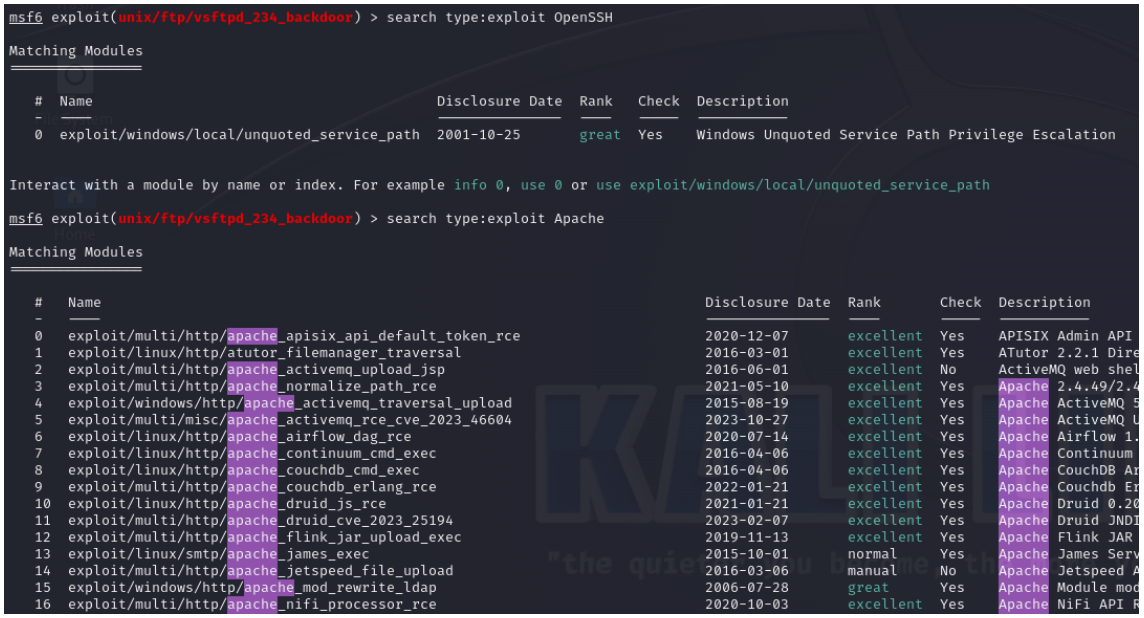
Webserver



Desktop



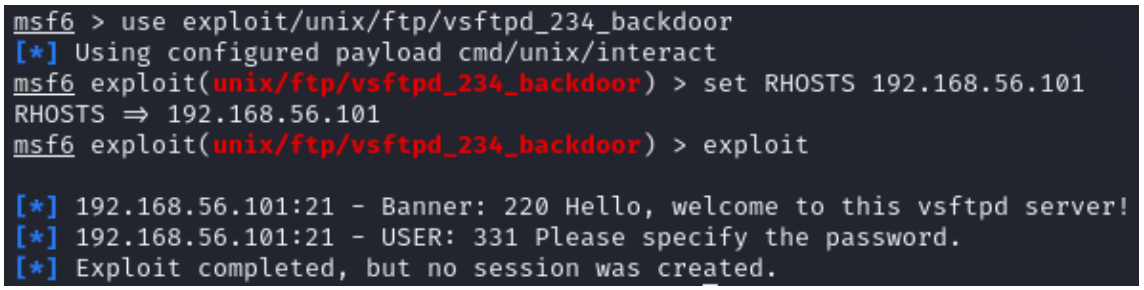
Com os seguintes resultados, demonstrados nas imagens a seguir, já foi possível descobrir alguns pontos de entrada que podemos explorar nomeadamente acerca do serviço FTP, SSH e do APACHE, nos dois primeiros serviços só encontrámos uma vulnerabilidade conhecida para cada uma, mas para o terceiro serviço, há várias vulnerabilidades conhecidas que podemos explorar. Na seguinte secção iremos dar exploit à vulnerabilidade do FTP.



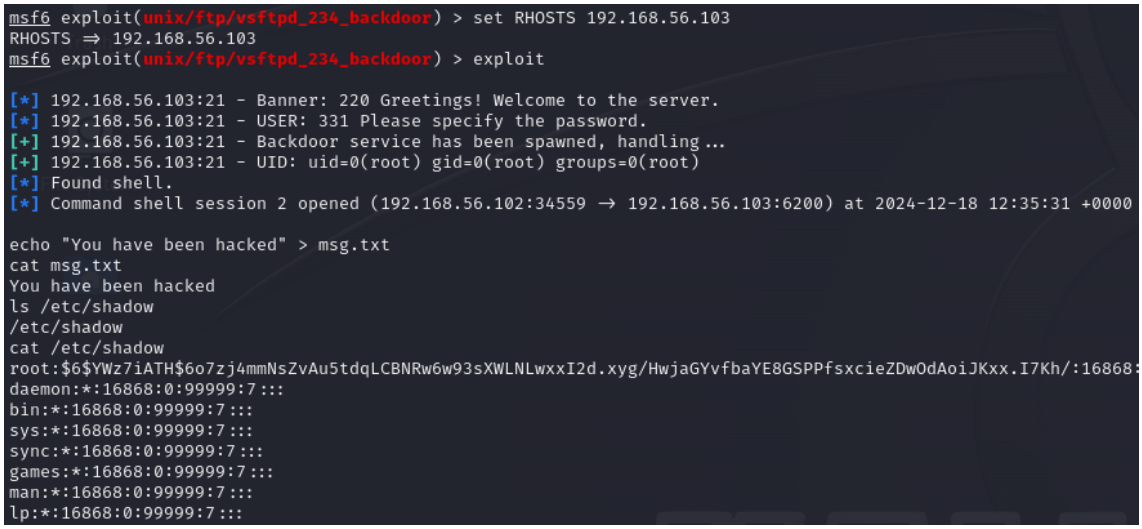
b) Creation of an exploit for one of the vulnerabilities found in **b)**

Após a listagem anterior, decidiu-se escolher um dos exploits à qual o escolhido foi o "**vsftpd_234_backdor**". O mesmo exploit foi aplicado nas duas máquinas alvo.

Webserver



Desktop



Como é possível verificar dos resultados obtidos, o exploit apenas funcionou na máquina **Desktop**. Isto consegue-se justificar devido às versões do **VSFTPD** serem diferentes em cada uma das máquinas. O **Desktop** está a executar a versão vulnerável 2.3.4, enquanto o **Webserver** está a executar a versão 2.3.5, que inclui uma correção desta vulnerabilidade.

c) Analysis according to CIS Benchmarks

Nesta secção irá ser feita uma análise de segurança no Webserver de forma a verificar o nível de conformidade do sistema com o **CIS Benchmark**.

Em primeiro, vão ser aplicados no Webserver os controlos do ponto 3 ao 5 e posteriormente do ponto 8 ao 10 no Desktop. O sucesso do mesmo está presente na seguinte tabela, podendo analisar assim, se o sistema está ou não em conformidade com cada controlo aplicado.

Webserver

	Controlo	**Conformidade**	
3	**Secure Boost Settings**	**SIM**	**NÃO**
3.1	Set User/Group Owner on bootloader config (Scored)	**X**	
3.2	Set Permissions on bootloader config (Scored)		**X**
3.3	Set Boot Loader Password (Scored)		**X**

3.4	Require Authentication for Single-User Mode (Scored)	**X**	
4	**Additional Process Hardening**		
4.1	Restrict Core Dumps (Scored)		**X**
4.2	Enable XD/NX Support on 32-bit x86 Systems (Not Scored)	**X**	
4.3	Enable Randomized Virtual Memory Region Placement (Scored)	**X**	
4.4	Disable Prelink (Scored)	**X**	
4.5	Activate AppArmor (Scored)		**X**
5	**OS Services**		
5.1.1	Ensure NIS is not installed (Scored)	**X**	
5.1.2	Ensure rsh server is not enabled (Scored)	**X**	
5.1.3	Ensure rsh client is not installed (Scored)	**X**	
5.1.4	Ensure talk server is not enabled (Scored)	**X**	
5.1.5	Ensure talk client is not installed (Scored)	**X**	
5.1.6	Ensure telnet server is not enabled (Scored)	**X**	
5.1.7	Ensure tftp-server is not enabled (Scored)	**X**	
5.1.8	Ensure xinetd is not enabled (Scored)	**X**	
5.2	Ensure chargen is not enabled (Scored)	**X**	
5.3	Ensure daytime is not enabled (Scored)	**X**	
5.4	Ensure echo is not enabled (Scored)	**X**	
5.5	Ensure discard is not enabled (Scored)	**X**	
5.6	Ensure time is not enabled (Scored)	**X**	

Desktop

	Controlo	Conformidade	
8	**Logging and Auditing**	**SIM**	**NÃO**

8.1.1.1	Configure System Accounting (auditd)		**X**
8.1.1.2	Disable System on Audit Log Full (Not Scored)		**X**
8.1.1.3	Keep All Auditing Information (Scored)		**X**
8.1.2	Install and Enable auditd Service (Scored)		**X**
8.1.3	Enable Auditing for Processes That Start Prior to auditd (Scored)		**X**
8.1.4	Record Events That Modify Date and Time Information (Scored)		**X**
8.1.5	Record Events That Modify User/Group Information (Scored)		**X**
8.1.6	Record Events That Modify the System's Network Environment (Scored)		**X**
8.1.7	Record Events That Modify the System's Mandatory Access Controls (Scored)		**X**
8.1.8	Collect Login and Logout Events (Scored)		**X**
8.1.9	Collect Session Initiation Information (Scored)		**X**
8.1.10	Collect Discretionary Access Control Permission Modification Events (Scored)		**X**
8.1.11	Collect Unsuccessful Unauthorized Access Attempts to Files (Scored)		**X**
8.1.12	Collect Use of Privileged Commands (Scored)		**X**
8.1.13	Collect Successful File System Mounts (Scored)		**X**
8.1.14	Collect File Deletion Events by User (Scored)		**X**
8.1.15	Collect Changes to System Administration Scope (sudoers) (Scored)		**X**
8.1.16	Collect System Administrator Actions (sudolog) (Scored)		**X**
8.1.17	Collect Kernel Module Loading and Unloading (Scored)		**X**
8.1.18	Make the Audit Configuration Immutable (Scored)		**X**
8.2.1	Install the rsyslog package (Scored)	**X**	
8.2.2	Ensure the rsyslog Service is activated (Scored)	**X**	
8.2.3	Configure /etc/rsyslog.conf (Not Scored)	**X**	
8.2.4	Create and Set Permissions on rsyslog Log Files (Scored)	**X**	

8.2.5	Configure rsyslog to Send Logs to a Remote Log Host (Scored)		**X**
8.2.6	Accept Remote rsyslog Messages Only on Designated Log Hosts (Not Scored)		**X**
8.3.1	Install AIDE (Scored)		**X**
8.3.2	Implement Periodic Execution of File Integrity (Scored)		**X**
8.4	Configure logrotate (Not Scored)	**X**	
9	**System Access, Authentication and Authorization**		
9.1.1	Enable cron Daemon (Scored)		**X**
9.1.2	Set User/Group Owner and Permission on /etc/crontab (Scored)		**X**
9.1.3	Set User/Group Owner and Permission on /etc/cron.hourly (Scored)		**X**
9.1.4	Set User/Group Owner and Permission on /etc/cron.daily (Scored)		**X**
9.1.5	Set User/Group Owner and Permission on /etc/cron.weekly (Scored)		**X**
9.1.6	Set User/Group Owner and Permission on /etc/cron.monthly (Scored)		**X**
9.1.7	Set User/Group Owner and Permission on /etc/cron.d (Scored)		**X**
9.1.8	Restrict at/cron to Authorized Users (Scored)		**X**
9.2.1	Set Password Creation Requirement Parameters Using pam_cracklib (Scored)		**X**
9.2.2	Set Lockout for Failed Password Attempts (Not Scored)		**X**
9.2.3	Limit Password Reuse (Scored)		**X**
9.3.1	Set SSH Protocol to 2 (Scored)		**X**
9.3.2	Set LogLevel to INFO (Scored)		**X**
9.3.3	Set Permissions on /etc/ssh/sshd_config (Scored)		**X**
9.3.4	Disable SSH X11 Forwarding (Scored)		**X**
9.3.5	Set SSH MaxAuthTries to 4 or Less (Scored)		**X**
9.3.6	Set SSH IgnoreRhosts to Yes (Scored)		**X**

9.3.7	Set SSH HostbasedAuthentication to No (Scored)		X
9.3.8	Disable SSH Root Login (Scored)		X
9.3.9	Set SSH PermitEmptyPasswords to No (Scored)		X
9.3.10	Do Not Allow Users to Set Environment Options (Scored)		X
9.3.11	Use Only Approved Cipher in Counter Mode (Scored)		X
9.3.12	Set Idle Timeout Interval for User Login (Scored)		X
9.3.13	Limit Access via SSH (Scored)		X
9.3.14	Set SSH Banner (Scored)		X
9.4	Restrict root Login to System Console (Not Scored)	X	
9.5	Restrict Access to the su Command (Scored)		X
10	**User Accounts and Environment**		
10.1.1	Set Password Expiration Days (Scored)		X
10.1.2	Set Password Change Minimum Number of Days (Scored)		X
10.1.3	Set Password Expiring Warning Days (Scored)	X	
10.2	Disable System Accounts (Scored)		X
10.3	Set Default Group for root Account (Scored)	X	
10.4	Set Default umask for Users (Scored)		X
10.5	Lock Inactive User Accounts (Scored)		X

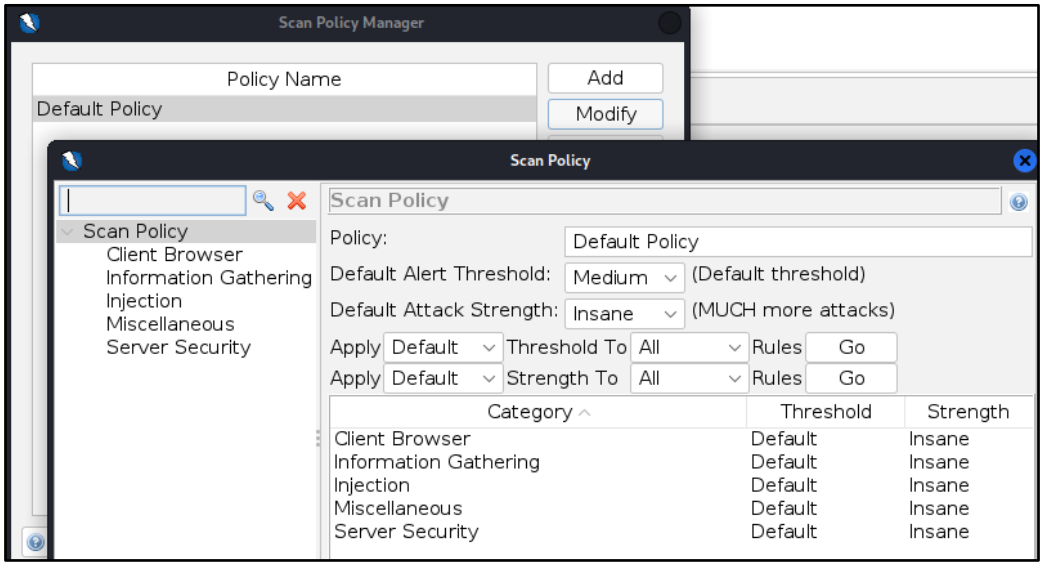
d) External analysis of the web application

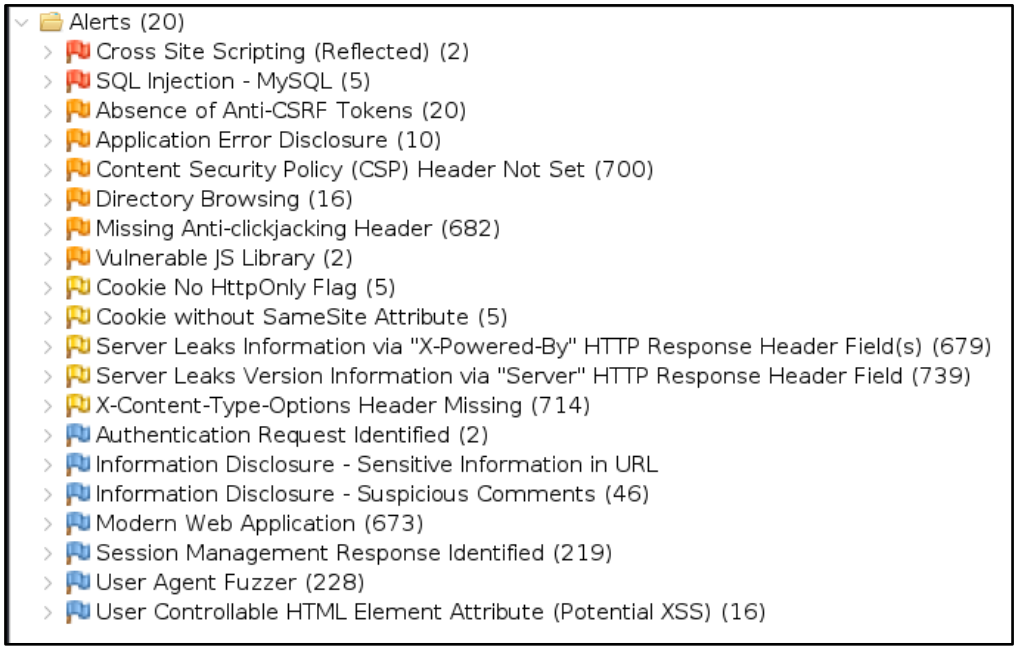
Foi feita uma análise recorrendo à aplicação **OWASP ZAP**, com a policy (Strength = Insane e Threshold = Medium) definida na seguinte imagem.

Recorrendo a mais uma ferramenta para descobrir vulnerabilidades em aplicações web, foi escolhido o **Wapiti**, pois consideramos ser uma aplicação interessante, para dar nos informação de vulnerabilidades que o **OWASP Zap** já nos deu, bem como de outras que podem ainda não ter sido descobertas por essa mesma ferramenta.

As próximas imagens vão mostrar os resultados dos scans feitos das diferentes ferramentas utilizadas.

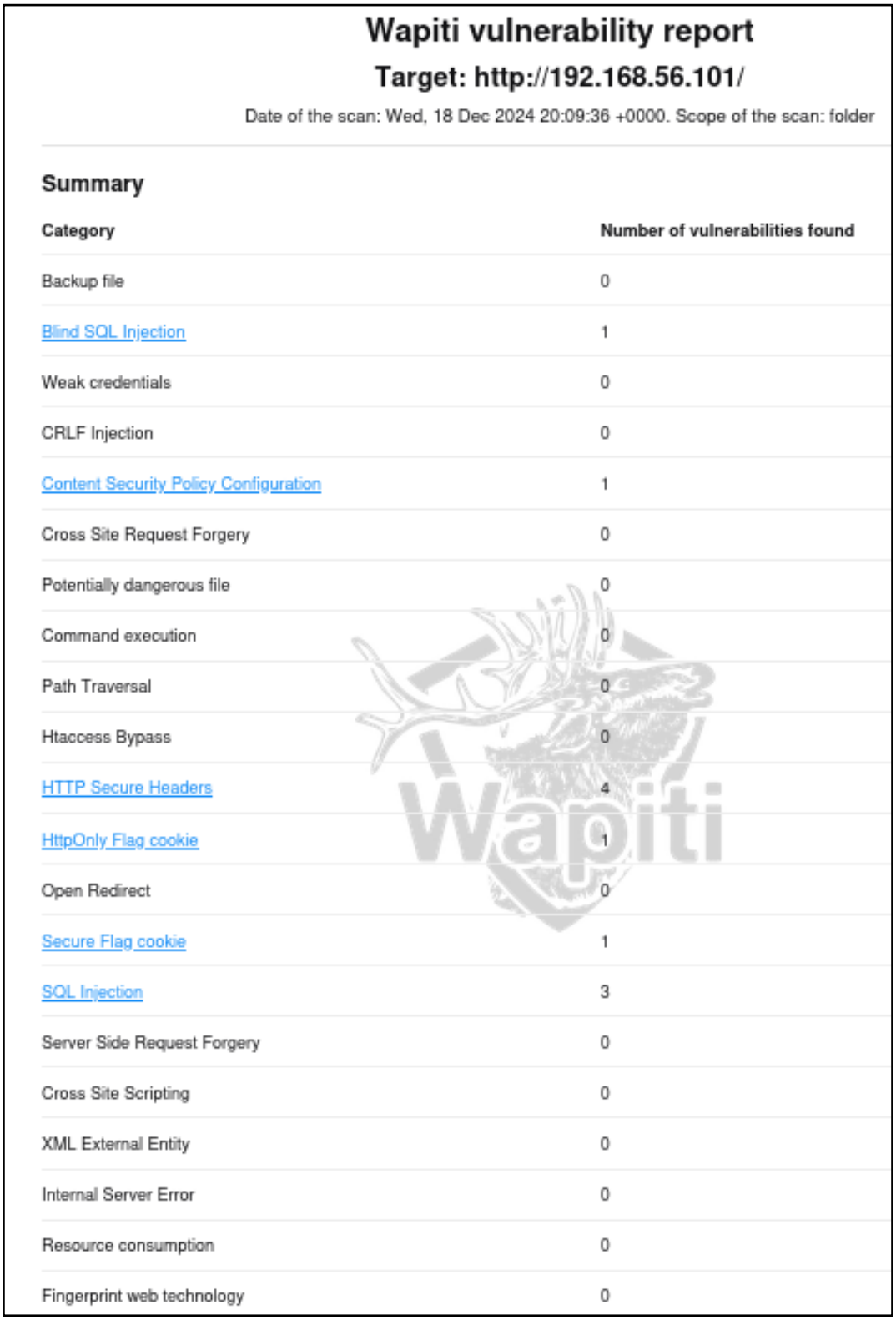
Zap





Após a análise, com o OWASP Zap, é possível constatar vulnerabilidades de, SQL Injection, Cross Site Scripting (XSS), CSRF, etc.

Wapiti



Wapiti vulnerability report

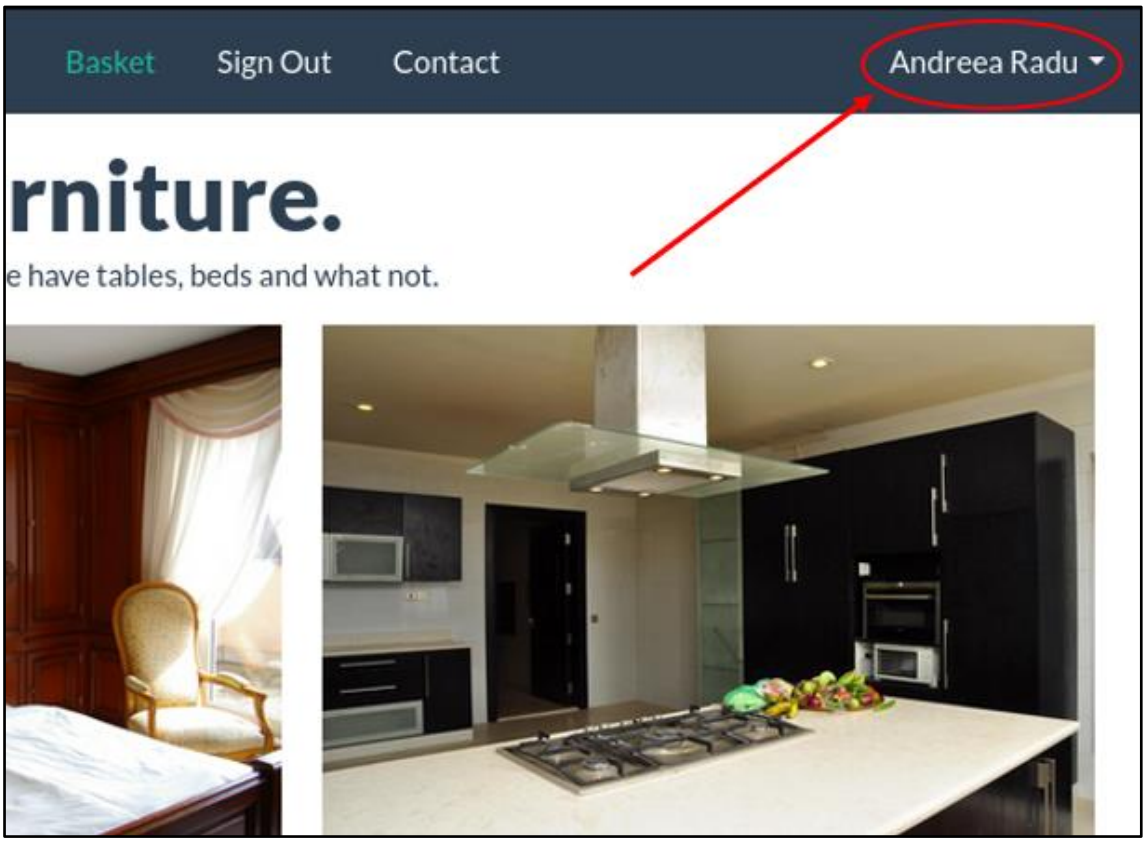
Target: http://192.168.56.101/

Date of the scan: Wed, 18 Dec 2024 20:09:36 +0000. Scope of the scan: folder

Summary

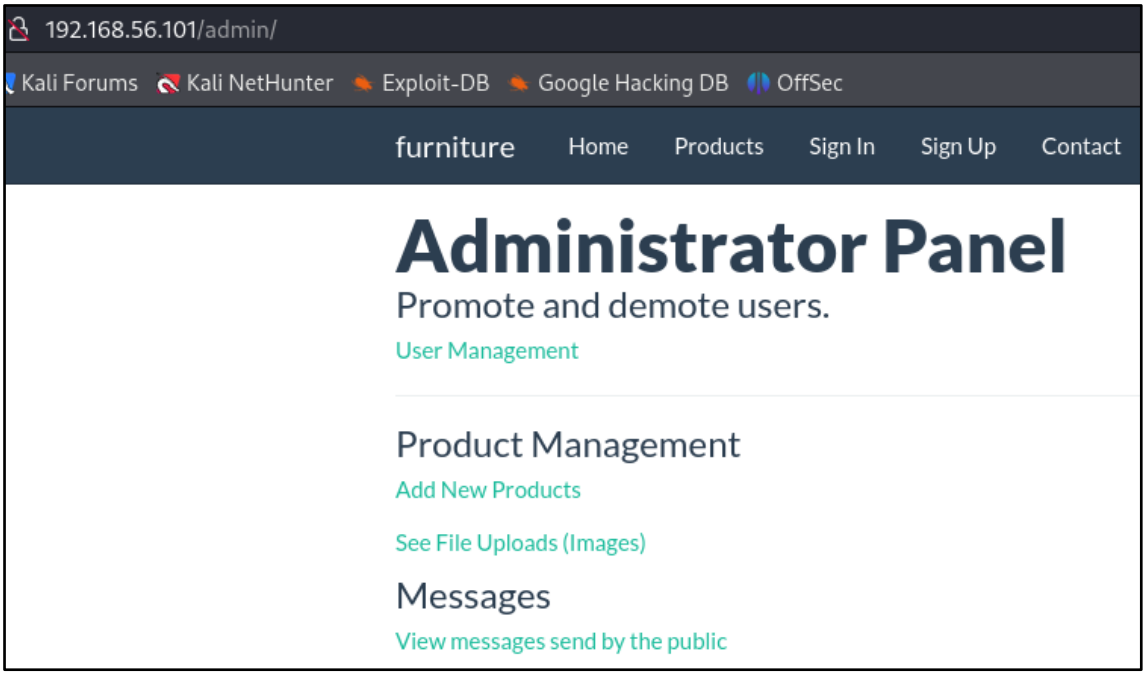
Category	Number of vulnerabilities found
Backup file	0
Blind SQL Injection	1
Weak credentials	0
CRLF Injection	0
Content Security Policy Configuration	1
Cross Site Request Forgery	0
Potentially dangerous file	0
Command execution	0
Path Traversal	0
Htaccess Bypass	0
HTTP Secure Headers	4
HttpOnly Flag cookie	1
Open Redirect	0
Secure Flag cookie	1
SQL Injection	3
Server Side Request Forgery	0
Cross Site Scripting	0
XML External Entity	0
Internal Server Error	0
Resource consumption	0
Fingerprint web technology	0

Mais à frente irá ser explorada a vulnerabilidade de **SQL Injection** na página de login, no qual irá ser introduzido o seguinte input no campo do username, **‘ or ‘1’=’1**.



Como se pode verificar na imagem anterior, após ser feito o ataque de SQL Injection na página Web, conseguiu-se entrar numa conta existente, neste caso "**Andrea Radu**", sem qualquer conhecimento das credenciais do utilizador.

Outra vulnerabilidade que se verificou nos resultados obtidos dos scans que foram feitos, foi o acesso livre ao diretório do painel de controlo de administrador em que qualquer utilizador pode aceder sem qualquer restrição, bastando apenas acrescentar o endpoint **/admin/.** Esta falha expõe a aplicação e o servidor a um risco significativo, podendo levar a perda total do controlo, roubo de dados, carregar ficheiros maliciosos (scripts PHP, ASP, etc.) disfarçados como imagens ou mesmo a interrupção do serviço.



3) Security Improvement

a) Configuration changes

Nesta secção foram parados e removidos alguns serviços de forma a fechar algumas portas de acesso às mesmas máquinas, mantendo assim as portas indispensáveis abertas, bem como os seus serviços.

Foram **atualizadas as máquinas**, garantindo assim os pacotes com as últimas atualizações disponíveis, contendo assim algumas correções a nível de segurança.

Webserver

Foram parados e removidos os serviços **VSFTPD** e **Rpcbind** porque considera-se que são serviços que não fazem falta para o sistema funcionar bem, tal como o mesmo conseguir satisfazer os pedidos que receba. Foram mantidos ligados os serviços **OpenSSH**, **Apache**.

Desktop

Foram parados e removidos os serviços **VSFTPD** e **Rpcbind** e **Distcc** porque considerámos não fazerem falta para o sistema funcionar bem como o mesmo conseguir satisfazer os pedidos que receba. Foram mantidos ligados os serviços **OpenSSH** e **Pop3**.

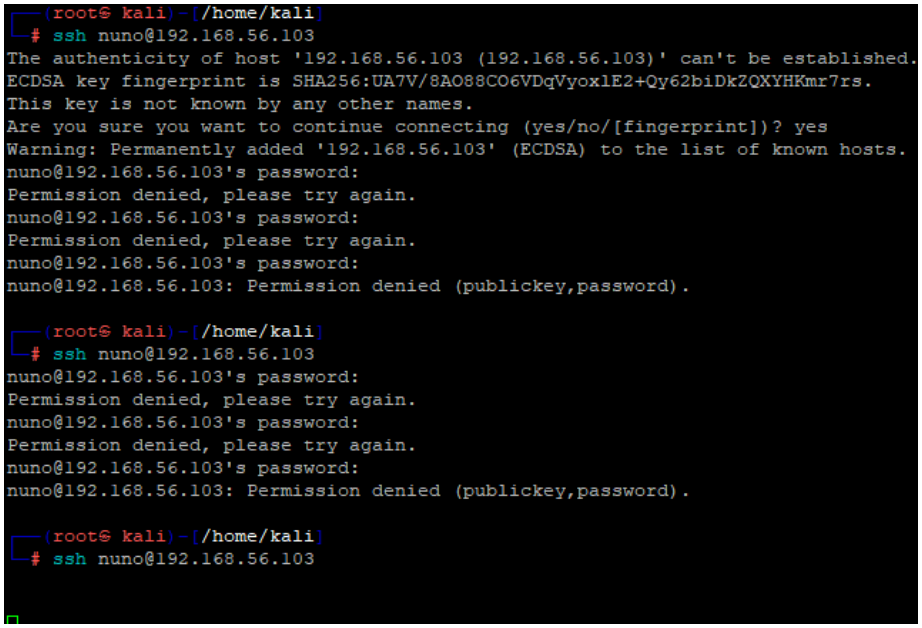
Após estas alterações conseguiu-se fechar portas que estavam abertas desnecessariamente, evitando assim possíveis acessos às mesmas.

Adicionalmente foram **aplicadas** as recomendações dos **CIS Controls**, de forma a meter o sistema mais resiliente e protegido.

b) Security Controls

De forma a bloquear possíveis ataques de força bruta através de SSH, foi instalado o serviço **Fail2Ban**, para monitorizar e bloquear os mesmos. Após 6 tentativas de meter a password mal, ele vai bloquear, necessitando depois de ser removido da "jail", caso seja do interesse de quem gere o sistema. Este serviço é bastante leve e prático de se usar, foi notado alguma lentidão no momento em que se removeu um IP da lista de bloqueio e só passado uns 2 minutos é que teve efeito.

Nas duas imagens seguintes, é possível ver uma tentativa de bruteforce no login de forma manual, na máquina Webserver. Em determinado momento é notório que a shell, já não nos permite introduzir a password, isto é, porque a outra máquina bloqueou.



Na seguinte imagem e através do comando presente na mesma é possível ver a lista de ip's bloqueados, bem como o número de tentativas erradas de introdução de password.

Foi usado o **Snort** nas duas máquinas para **analisar** o tráfego que recebe e ficar guardado de forma a **monitorizar** e a **prevenir** futuras ameaças. Esta ferramenta não bloqueia ameaças, mas regista-as e as guarda num ficheiro. Para bloquear seria necessário interligar com **iptables,** de forma a caso o sistema deteta-se um tipo de linha no ficheiro de logs semelhante a um do seu script, iria bloquear de imediato o IP.

Para verificar a eficácia do snort na máquina de destino, usou-se o seguinte comando: **tail -f /var/log/snort/alert**

Após a verificação dos logs do **Snort**, é possível ver que está a haver um **Nmap** por parte de outra máquina.

```
[**] [1:1228:7] SCAN nmap XMAS [**]
[Classification: Attempted Information Leak] [Priority: 2]
12/18-16:44:01.381444 192.168.56.102:45366 -> 192.168.56.103:1
TCP TTL:57 TOS:0x0 ID:9475 IpLen:20 DgmLen:60
**U*P**F Seq: 0x3C0E5BBD  Ack: 0x258E869A  Win: 0xFFFF  TcpLen: 40  UrgPtr: 0x0
TCP Options (5) => WS: 15 NOP MSS: 265 TS: 4294967295 0 SackOK
[Xref => http://www.whitehats.com/info/IDS30]

[**] [1:402:7] ICMP Destination Unreachable Port Unreachable [**]
[Classification: Misc activity] [Priority: 3]
12/18-16:44:01.815174 192.168.56.103 -> 192.168.56.102
ICMP TTL:64 TOS:0xC0 ID:13245 IpLen:20 DgmLen:57
Type:3  Code:3  DESTINATION UNREACHABLE: PORT UNREACHABLE
** ORIGINAL DATAGRAM DUMP:
192.168.56.102:41773 -> 192.168.56.103:1434
UDP TTL:64 TOS:0x0 ID:53776 IpLen:20 DgmLen:29 DF
Len: 1  Csum: 25078
(1 more bytes of original packet)
** END OF DUMP
```

Aplicação da firewall IPtables para o **Webserver** de forma a garantir apenas os portos 22 e 80 abertos, limitando o número de portas para os atacantes.

Configuração da Iptable para o Webserver:

```
root@localhost:/home/nuno# iptables -L -v
Chain INPUT (policy DROP 36 packets, 14088 bytes)
 pkts bytes target     prot opt in     out     source               destination
  342 25136 fail2ban-ssh  tcp  --  any    any     anywhere             anywhere             multiport dports ssh
  311 23224 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:ssh state NEW,ESTABLISHED
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:http state NEW,ESTABLISHED

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 271 packets, 32008 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain fail2ban-ssh (1 references)
 pkts bytes target     prot opt in     out     source               destination
  342 25136 RETURN     all  --  any    any     anywhere             anywhere
```

Aplicação da firewall IPtables para o **Desktop** de forma a garantir apenas os portos 22 e 110 abertos, limitando o número de portas para os atacantes.

Configuração da Iptable para o Desktop:

```
Chain INPUT (policy DROP 706 packets, 33842 bytes)
 pkts bytes target     prot opt in     out     source               destination
   15   984 fail2ban-ssh  tcp  --  any    any     anywhere             anywhere             multiport dports ssh
  118  8016 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:ssh state NEW,ESTABLISHED
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere             tcp dpt:pop3 state NEW,ESTABLISHED

Chain FORWARD (policy DROP 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 16 packets, 1692 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain fail2ban-ssh (1 references)
 pkts bytes target     prot opt in     out     source               destination
   15   984 RETURN     all  --  any    any     anywhere             anywhere
```

4) Security Reassessment

Acesso sem SSH/ Remote Assessment

Open TCP ports:

Comando utilizado: **db_nmap -v -T4 -PA -sV --version-all -osscan-guess -A -sS -p 1-65535 x.x.x.x**

Webserver

```
Services
========

host            port  proto  name  state  info
----            ----  -----  ----  -----  ----
192.168.56.103  22    tcp    ssh   open   OpenSSH 6.0p1 Debian 4+deb7u7 protocol 2.0
192.168.56.103  80    tcp    http  open   Apache httpd 2.2.22 (Debian)
```

Desktop

```
Services
========

host            port  proto  name  state  info
----            ----  -----  ----  -----  ----
192.168.56.101  22    tcp    ssh   open   OpenSSH 6.0p1 Debian 4+deb7u7 protocol 2.0
192.168.56.101  110   tcp    pop3  open   Openwall popa3d
```

Open UDP ports:

Utilizando o primeiro comando dentro do metasploit, **db_nmap -v -sU -T5 -p- --max-retries 3 --max-rtt-timeout 100ms --max-scan-delay 10ms -host-timeout 4h x.x.x.x -oA nmap_sU,** verificou-se que não havia portas UDP abertas. Para confirmar este resultado, decidiu-se fazer mais um scan, sendo este fora do metasploit, utilizando o **nmap.**

Comando utilizado: **nmap -sU x.x.x.x**

Webserver

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-18 23:20 E
Nmap scan report for 192.168.56.103
Host is up (0.00043s latency).
All 1000 scanned ports on 192.168.56.103 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
MAC Address: 08:00:27:C2:1B:0D (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 36.41 seconds
```

Desktop

```
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-18 23:23 E
Nmap scan report for 192.168.56.101
Host is up (0.00044s latency).
All 1000 scanned ports on 192.168.56.101 are in ignored states.
Not shown: 1000 open|filtered udp ports (no-response)
MAC Address: 08:00:27:0E:A2:94 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 36.44 seconds
```

Open RPC endpoints:

Comando utilizado: **rpcinfo -p x.x.x.x**

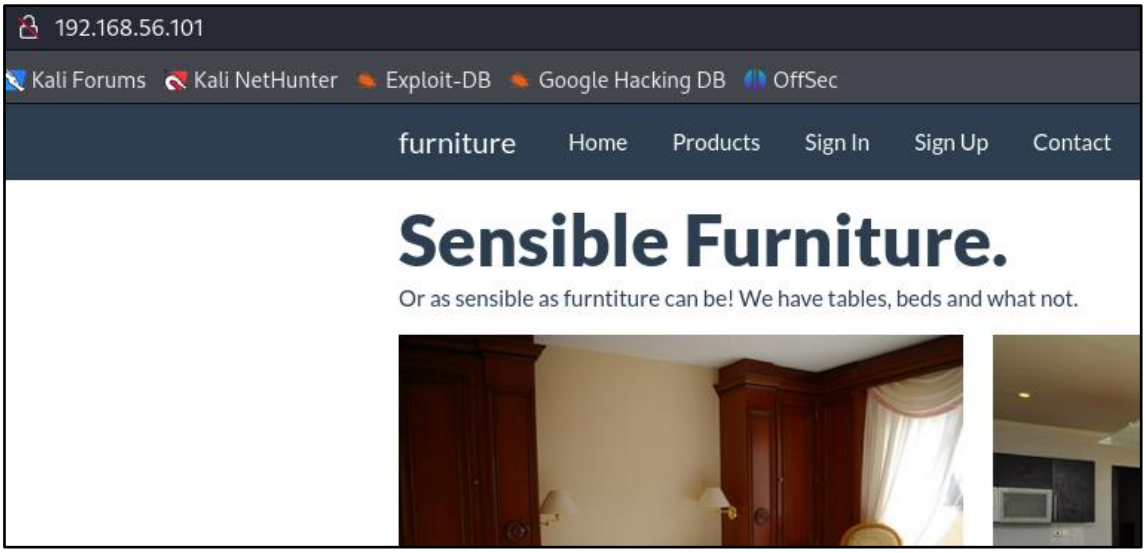
```
rpcinfo: can't contact portmapper: RPC: Remote system error - Connection refused
```

Message: *rpcinfo: can't contact portmapper: RPC: Remote system error - Connection refused*

Tal como seria de esperar, o comando não conseguiu ser executado em ambas as máquinas uma vez que o serviço rpc foi removido em cada uma delas.

Verificar se o serviço web está a correr no servidor:

Webserver



Mesmo após todas as melhorias de segurança do sistema, é possível verificar que a aplicação web do **Webserver** continua a correr.

Acesso com SSH/ Local Assessment

Open Sockets:

Comando utilizado: **ss -s**

Webserver: 68

Desktop: 257

Open Named Pipes:

Comando utilizado: **lsof | grep 'FIFO' | wc -l**

Webserver: 53

Desktop: 219

Dynamic web pages:

Comando utilizado: **find /var/www/onlinestore | grep php | wc -l**

Webserver: 21

Desktop: 0

Services:

Comando utilizado: **ls /etc/init.d | wc -l**

Webserver: 62

Desktop: 63

Services running by default:

Comando utilizado: **service --status-all | grep '+' | wc -l**

Webserver: 17

Desktop: 15

Services running as root:

Comando utilizado: **ps -U root -u root u | wc -l**

Webserver: 69

Desktop: 72

Enabled accounts:

Comando utilizado: **cut -d: -f1 /etc/passwd | wc -l**

Webserver: 32

Desktop: 36

Enabled accounts in root group:

Comando utilizado: **cut -d: -f1,4 /etc/passwd | grep -w 0**

Webserver: 1

Desktop: 2

Guest accounts:

Comando utilizado: **cut -d: -f1,4 /etc/passwd | grep 'guest' | wc -l**

Webserver: 0

Desktop: 0

Resumo do Attack Surface

Webserver

Avenues of Attack	Number Avenues	Bias	Effective Attack Surface	Area
Open sockets	68	1	68	Channels
Open TCP ports	2	1	2	Channels
Open UDP ports	0	1	0	Channels
Open RPC endpoints	0	0,9	0	Channels
Open named pipes	53	0,8	42,4	Channels
Dynamic web pages	21	0,6	12,6	Process Targets
Services	62	0,2	12,4	Process Targets
Services running by default	17	0,8	13,6	Process Targets
Services running as root	69	0,6	41,4	Process Targets
Enabled accounts	32	0,5	16	Access Rights
Enabled accounts in root group	1	0,9	0,9	Access Rights
Guest accounts enabled	0	0,9	0	Access Rights
		RASQ:	209,3	

RASQ anterior: 246,6 (+37,3)

Desktop

Avenues of Attack	Number Avenues	Bias	Effective Attack Surface	Area
Open sockets	257	1	257	Channels
Open TCP ports	2	1	2	Channels
Open UDP ports	0	1	0	Channels
Open RPC endpoints	0	0,9	0	Channels
Open named pipes	219	0,8	175,2	Channels
Dynamic web pages	0	0,6	0	Process Targets
Services	63	0,2	12,6	Process Targets
Services running by default	15	0,8	12	Process Targets
Services running as root	72	0,6	43,2	Process Targets
Enabled accounts	36	0,5	18	Access Rights
Enabled accounts in root group	2	0,9	1,8	Access Rights
Guest accounts enabled	0	0,9	0	Access Rights
		RASQ:	521,8	

RASQ anterior: 494,3 (-27,5)

Conclusão

Após a análise realizada neste trabalho, foi possível identificar vulnerabilidades significativas nas máquinas alvo, bem como implementar medidas eficazes para mitigar os riscos identificados. Durante a avaliação inicial, verificou-se a presença de serviços desnecessários em execução, como o **rpcbind** e o **vsftpd**, bem como várias portas abertas que deixavam expostos os sistemas a potenciais ataques. Essas vulnerabilidades, aliadas a versões desatualizadas de sistemas operativos e serviços, representavam um risco elevado, como demonstrado pela exploração bem-sucedida da backdoor na versão 2.3.4 do **vsftpd**.

A fase de melhoria envolveu a desativação de serviços não essenciais, a aplicação de atualizações críticas e a implementação de ferramentas de segurança, como **iptables**, para restringir o acesso a portas críticas, e **Fail2Ban**, para proteger contra ataques de força bruta. Além disso, o uso do **Snort** permitiu monitorizar o tráfego e identificar possíveis tentativas de intrusão. A aplicação de controles recomendados pelos benchmarks CIS assegurou a conformidade com as melhores práticas de segurança, resultando em sistemas mais resilientes e protegidos. Por último, após a análise e correção das melhorias recomendadas pelo CIS Benchmarks, sentiu-se que as máquinas ficaram em conformidade com as boas práticas de forma a estarmos mais seguros.

Após a reavaliação, constatou-se uma redução significativa da superfície de ataque. As máquinas passaram a apresentar apenas os serviços essenciais em execução, como o **SSH** e o **HTTP**, enquanto vulnerabilidades previamente exploradas, como a associada ao **vsftpd**, foram eliminadas. A aplicação de atualizações corrigiu falhas conhecidas, reforçando a integridade dos sistemas.

Este trabalho demonstrou a importância de um ciclo contínuo de avaliação e melhoria de segurança, evidenciando a necessidade de atualizações regulares, monitorização constante e revisão periódica das configurações de segurança. A implementação das

medidas recomendadas reduziu significativamente os riscos e garantiu que os sistemas estivessem mais preparados para enfrentar ameaças futuras, cumprindo os objetivos estabelecidos no início do projeto.

Referências

[1] Daniel Miessler, “Password List.” Accessed: Dec. 16, 2024. [Online]. Available: https://github.com/danielmiessler/SecLists/blob/master/Passwords/Common-Credentials/500-worst-passwords.txt

[2] Igor_sec, “Snorting | Installing, Configuring, & Exploring Snort” Accessed: Dec.18, 2024. [Online]. Available: https://igorsec.blog/2023/07/31/snorting-installing-configuring-exploring-snort/