

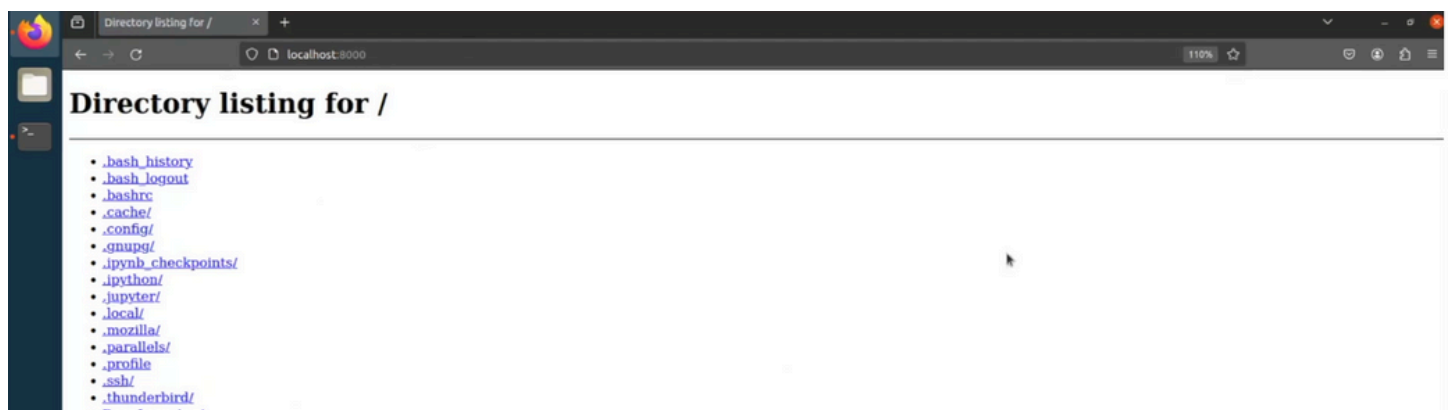
maquina de ataque:

```
user@singular1:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:1c:42:d0:02:5f brd ff:ff:ff:ff:ff:ff
    inet 10.211.55.5/24 brd 10.211.55.255 scope global dynamic eth0
        valid_lft 1515sec preferred_lft 1515sec
    inet6 fdb2:2c26:f4e4:0:21c:42ff:fed0:25f/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 2591979sec preferred_lft 604779sec
    inet6 fe80::21c:42ff:fed0:25f/64 scope link
        valid_lft forever preferred_lft forever
user@singular1:~$
```

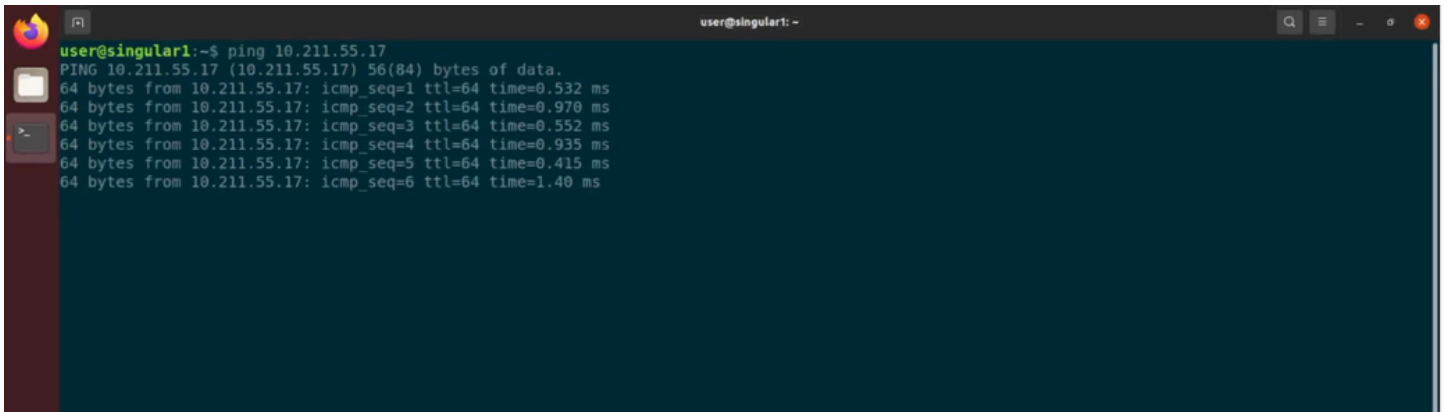
maquina de defensa, vemos que tenemos un directorio internet activo y que levantamos con python para ver si podemos defenderlo de la maquina de ataque:

```
user@ubuntu:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:1c:42:84:e7:e2 brd ff:ff:ff:ff:ff:ff
    inet 10.211.55.17/24 brd 10.211.55.255 scope global dynamic eth0
        valid_lft 1409sec preferred_lft 1409sec
    inet6 fdb2:2c26:f4e4:0:21c:42ff:fe84:e7e2/64 scope global dynamic mngtmpaddr noprefixroute
        valid_lft 2591954sec preferred_lft 604754sec
    inet6 fe80::21c:42ff:fe84:e7e2/64 scope link
        valid_lft forever preferred_lft forever
user@ubuntu:~$
```

```
user@ubuntu:~$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```



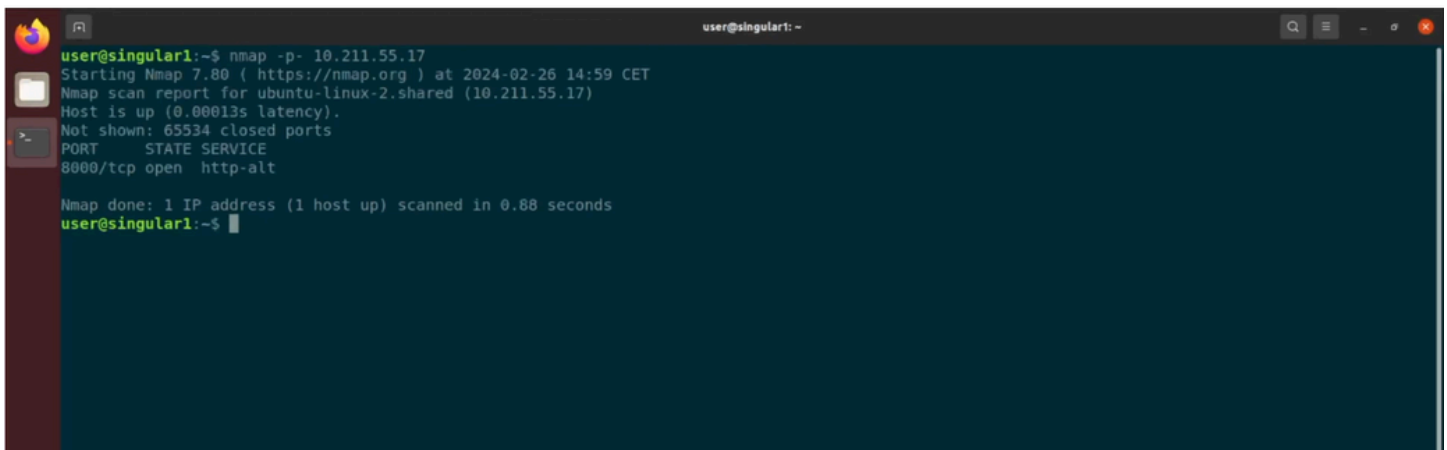
preparamos la maquina de ataque (hemos instalado nmap, curl, ssh y hydra), comprobamos si hay ping también:

A terminal window titled 'user@singular1: ~' showing the output of a 'ping' command. The command is 'ping 10.211.55.17'. The output shows six successful pings, each with 64 bytes of data, an ICMP sequence number from 1 to 6, a TTL of 64, and various response times ranging from 0.415 ms to 1.40 ms.

```
user@singular1:~$ ping 10.211.55.17
PING 10.211.55.17 (10.211.55.17) 56(84) bytes of data:
64 bytes from 10.211.55.17: icmp_seq=1 ttl=64 time=0.532 ms
64 bytes from 10.211.55.17: icmp_seq=2 ttl=64 time=0.970 ms
64 bytes from 10.211.55.17: icmp_seq=3 ttl=64 time=0.552 ms
64 bytes from 10.211.55.17: icmp_seq=4 ttl=64 time=0.935 ms
64 bytes from 10.211.55.17: icmp_seq=5 ttl=64 time=0.415 ms
64 bytes from 10.211.55.17: icmp_seq=6 ttl=64 time=1.40 ms
```

Ahora probamos las herramientas instaladas sin protección previa de la máquina de defensa para comparar resultados tras poner todas las defensas luego:

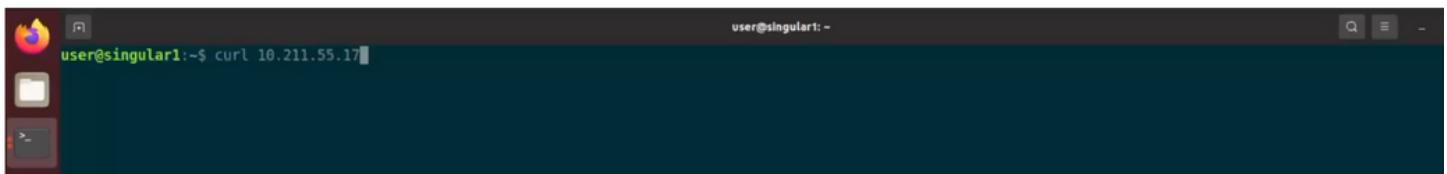
nmap nos dice que hay un puerto abierto, el 8000 que es el que hemos levantado con python y hay un listado de directorios:

A terminal window titled 'user@singular1: ~' showing the output of an 'nmap' scan. The command is 'nmap -p- 10.211.55.17'. The output shows the scan starting at 2024-02-26 14:59 CET, reporting for 'ubuntu-linux-2.shared (10.211.55.17)', and finding one open port: 8000/tcp with service 'http-alt'. It also shows 65534 closed ports and a scan time of 0.88 seconds.

```
user@singular1:~$ nmap -p- 10.211.55.17
Starting Nmap 7.80 ( https://nmap.org ) at 2024-02-26 14:59 CET
Nmap scan report for ubuntu-linux-2.shared (10.211.55.17)
Host is up (0.00013s latency).
Not shown: 65534 closed ports
PORT      STATE SERVICE
8000/tcp  open  http-alt

Nmap done: 1 IP address (1 host up) scanned in 0.88 seconds
user@singular1:~$
```

probamos curl tras ver que el puerto 8000 esta activo y nos da el listado de directorios de ese puerto con una http request:

A terminal window titled 'user@singular1: ~' showing the command 'curl 10.211.55.17' being entered. The output is not yet visible.

```
user@singular1:~$ curl 10.211.55.17
```

```
user@singular1: ~  
<ul>  
<li><a href=".bash_history">.bash_history</a></li>  
<li><a href=".bash_logout">.bash_logout</a></li>  
<li><a href=".bashrc">.bashrc</a></li>  
<li><a href=".cache">.cache</a></li>  
<li><a href=".config">.config</a></li>  
<li><a href=".gnupg">.gnupg</a></li>  
<li><a href=".ipynb_checkpoints">.ipynb_checkpoints</a></li>  
<li><a href=".ipython">.ipython</a></li>  
<li><a href=".jupyter">.jupyter</a></li>  
<li><a href=".local">.local</a></li>  
<li><a href=".mozilla">.mozilla</a></li>  
<li><a href=".parallels">.parallels</a></li>  
<li><a href=".profile">.profile</a></li>  
<li><a href=".ssh">.ssh</a></li>  
<li><a href=".thunderbird">.thunderbird</a></li>  
<li><a href="DeepLearning">DeepLearning</a></li>  
<li><a href="Desktop">Desktop</a></li>  
<li><a href="Documents">Documents</a></li>  
<li><a href="Downloads">Downloads</a></li>  
<li><a href="Firefox_wallpaper.png">Firefox_wallpaper.png</a></li>  
<li><a href="ml">ml</a></li>  
<li><a href="Music">Music</a></li>  
<li><a href="nltk_data">nltk_data</a></li>  
<li><a href="NoSupervisado">NoSupervisado</a></li>  
<li><a href="OpenCv">OpenCv</a></li>  
<li><a href="Pictures">Pictures</a></li>  
<li><a href="planets.csv">planets.csv</a></li>  
<li><a href="Public">Public</a></li>  
<li><a href="RedesNeuronales">RedesNeuronales</a></li>  
<li><a href="Supervisado">Supervisado</a></li>  
<li><a href="tecnico_ml">tecnico_ml</a></li>  
<li><a href="Templates">Templates</a></li>  
<li><a href="UE Ejemplos.ipynb">UE Ejemplos.ipynb</a></li>  
<li><a href="Untitled.ipynb">Untitled.ipynb</a></li>  
<li><a href="Videos">Videos</a></li>  
</ul>  
<hr>  
</body>  
</html>  
user@singular1:~$
```

probamos con ataque ssh (sabiendo el usuario): vemos que hay respuesta, que tenemos acceso.

```
user@singular1: ~  
user@singular1:~$ ssh user@10.211.55.17  
The authenticity of host '10.211.55.17 (10.211.55.17)' can't be established.  
ECDSA key fingerprint is SHA256:8Iiv/8aBCprl53ohWLFH9ii0faIpwXFjxaRKAT6pZvc.  
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

vamos a quitar el limite de intentos ssh como experimento a la maquina de defensa, aunque no es nada recomendable.

```
user@ubuntu: ~  
user@ubuntu:~$ sudo nano /etc/ssh/sshd_config
```

```
#LoginGraceTime 2m
#PermitRootLogin prohibit-password
#StrictModes yes
#MaxAuthTries 6
MaxSessions 0
```

reseteamos el servidor tras hacerlo:

```
user@ubuntu:~$ sudo systemctl restart ssh
user@ubuntu:~$
```

ahora probamos un ataque con hydra en la maquina de ataque, tenemos un diccionario con 50 contraseñas en un archivo llamado password.txt, es un exito:

```
user@singular1:~$ hydra -l user -P passwords.txt ssh://10.211.55.17
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-26 15:21:06
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 51 login tries (l:l/p:p:51), ~4 tries per task
[DATA] attacking ssh://10.211.55.17:22/
[22][ssh] host: 10.211.55.17 login: user password: batman123
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-26 15:21:14
user@singular1:~$
```

Vemos que es muy facil encontrar una contraseña usando fuerza bruta sino hay limitación de intentos. Los cuatro intentos han sido exitosos, tenemos que preparar la maquina de defensa para evitarlos.

preparamos la maquina de defensa.

Empezamos con algo muy importante, preparando las reglas de por defecto par asegurarnos que el comportamiento general del sistema es el que queremos.

```
user@singular2:~$ sudo iptables -P INPUT DROP
[sudo] password for user:
user@singular2:~$ sudo iptables -P OUTPUT ACCEPT
user@singular2:~$
```

permitimos el trafico local:

```
user@singular2:~$ sudo iptables -A INPUT -i lo -j ACCEPT
user@singular2:~$ sudo iptables -A OUTPUT -o lo -j ACCEPT
user@singular2:~$
```

importante para no bloquear las conexiones que ya estan esablecidas o relacionadas:

```
user@singular2:~$ sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
user@singular2:~$
```

Vamos a ocuparnos de crear las reglas para bloquear a nmap, primero bloqueamos el escaneo de puertos:

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --tcp-flags SYN,ACK,FIN,RST RST -m limit --limit 1/s -j ACCEPT
user@singular2:~$
```

luego descartamos cualquier paquete TCP que tenga el flag RST, que serian los paquetes enviados por un atacante que está escaneando para ver que puertos están abiertos, para evitar que nmap funcione. Importante entender que petitiones lanza un escaneo de puertos, son paquetes tcp con flag RST, por éso los bloqueamos.

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --tcp-flags SYN,ACK,FIN,RST RST -j DROP
```

Ahora vamos a crear las reglas en iptables para bloquear el acceso http al directorio online que hemos levantado antes.

Hasta ahora con curl podiamos acceder al directorio web, vamos a limitar el acceso, limitamos el acceso al puerto 8000 y 443 (https): Http debería ser el puerto 80 pero bueno no pasa nada.

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 443 -j DROP
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 8000 -j DROP
user@singular2:~$
```

Ahora vamos a bloquear el ssh, su puerto es el 22 y podemos especificar las ip que estan permitidas con el -s y ACCEPT:

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 22 -s 10.211.55.10 -j ACCEPT
user@singular2:~$
```

Las reglas se aplicarán sólo a conexiones nuevas iniciadas en el puerto 22 para no cortar las conexiones legítimas. También vamos a poner límite en la tasa de paquetes que coinciden con la regla para proteger contra ataques DOS para evitar sobrecargar el servidor (limitando el número de paquetes por minuto), eliminando también así o reduciendo la efectividad de hydra y los ataques de fuerza bruta que pasan por ssh:

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 22 -m conntrack --ctstate NEW -m limit --limit 1/minute --limit-burst 3 -j ACCEPT
user@singular2:~$
```

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 22 -j DROP
user@singular2:~$
```

Vale la defensa ya está hecha, ahora vamos a probar los ataques de nuevo tras establecer éstas defensas.

Para almacenar los logs de registro, para un futuro análisis.

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 22 -j LOG --log-prefix "SSH attempt: "
user@singular2:~$
```

Hemos puesto muchas reglas así que ahora vamos a guardarlas en un documento:

```
user@singular2:~$ sudo iptables-save > /etc/iptables/rules
user@singular2:~$
```

Volvemos a la máquina de ataque y vamos a volver a atacar usando ping, nmap, curl, ssh y hydra, haber si hemos logrado crear unas buenas defensas.

El ping no funciona. Bien.

```
user@singular1:~$ ping 10.211.55.17
PING 10.211.55.17 (10.211.55.17) 56(84) bytes of data.
```

NMAP y escaneo de puertos, no funciona. Bien.

```
user@singular1:~$ nmap -p- 10.211.55.17
Starting Nmap 7.80 ( https://nmap.org ) at 2024-02-26 16:14 CET
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.03 seconds
user@singular1:~$
```

Petición HTTP con curl, no funciona. Bien.

```
user@singular1:~$ curl http://10.211.55.17
^C
user@singular1:~$ curl http://10.211.55.17:8000
```

Establecer una conexión SSH, no funciona. Bien.

```
user@singular1:~$ ssh user@10.211.55.17
ssh: connect to host 10.211.55.17 port 22: Connection refused
user@singular1:~$
```

Probemos con Hydra ahora para ver si con fuerza bruta podemos probar las contraseñas. No funciona. Bien.

```
user@singular1:~$ hydra -l user -P passwords.txt ssh://10.211.55.17
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-26 16:17:54
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 52 login tries (l:1/p:52), ~4 tries per task
[DATA] attacking ssh://10.211.55.17:22/
[ERROR] could not connect to ssh://10.211.55.17:22 - Connection refused
user@singular1:~$
```

Probemos ahora habilitando la ip del atacante para permitir el acceso ssh (antes le hemos denegado el acceso habilitando todas excepto una al azar). Para hacerlo tenemos que volver a la máquina de defensa y habilitar la ip de la máquina de ataque.

```
user@singular2:~$ sudo iptables -A INPUT -p tcp --dport 22 -s 10.211.55.5 -j ACCEPT
user@singular2:~$
```

Probamos primero establecer una conexión ssh, funciona,

```
user@singular1:~$ ssh user@10.211.55.17
The authenticity of host '10.211.55.17 (10.211.55.17)' can't be established.
ECDSA key fingerprint is SHA256:8Iiv/8aBCprLS3ohWLFH9iiofaIpwXFjxaRKAT6pZvc.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

Probemos ahora con hydra. Termina el proceso de hydra porque no le deja seguir con las pruebas, incluso con la ip permitida, pero no consigue probar todas las contraseñas.

```
user@singular1:~$ hydra -l user -P passwords.txt ssh://10.211.55.17
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-02-26 16:28:31
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 52 login tries (l:1/p:52), ~4 tries per task
[DATA] attacking ssh://10.211.55.17:22/
1 of 1 target completed, 0 valid passwords found
[WARNING] Writing restore file because 2 final worker threads did not complete until end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-02-26 16:28:42
user@singular1:~$
```

Así tenemos una primera capa de protección.