



PRACTICA

BLUE TEAM





CONTENIDO:

PFSENSE.....
Instalación del Pfsense.....
Creación de las redes en Virtual Box.....
Configuración Pfsense.....
Reglas de los Firewall.....
Creación y configuración de la VPN.....
ELASTIC CLOUD.....
Creación de la cuenta de Elastic Cloud.....
Configuración Elastic.....
Integración de los logs de Windows.....
Integración de los logs del Honeypot en Elastic.....



1. PFSENSE:

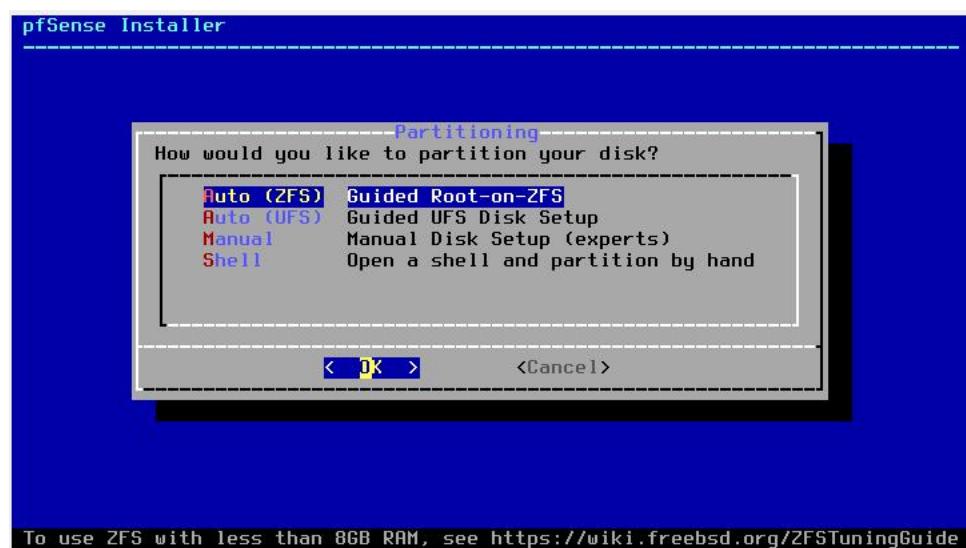
- SO: free BSD
- Memoria: 1024MB
- CPUs:1
- Almacenamiento: 20GB

1.1 INSTALACION DEL PFSENSE:

- Instalamos:

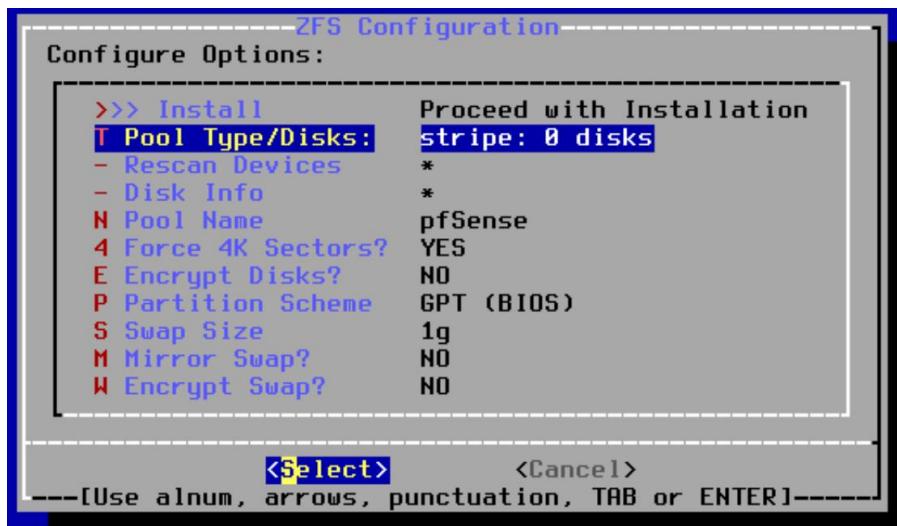


- Le damos a Guided Root-on-ZFS

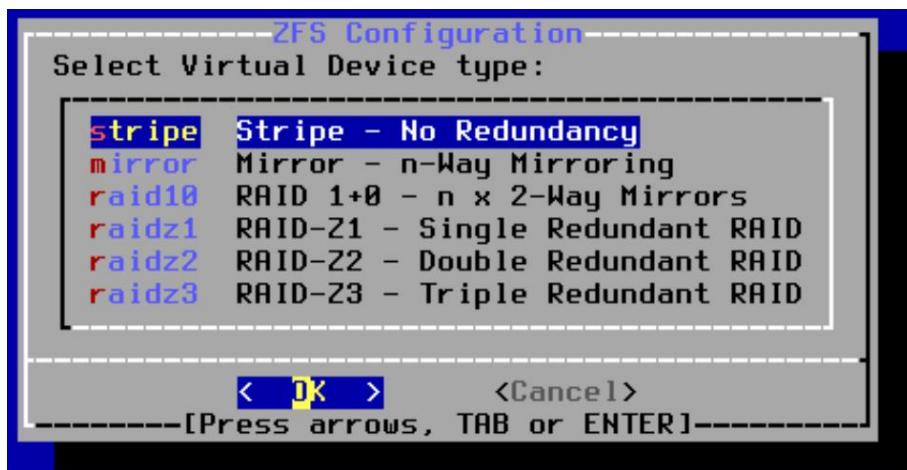




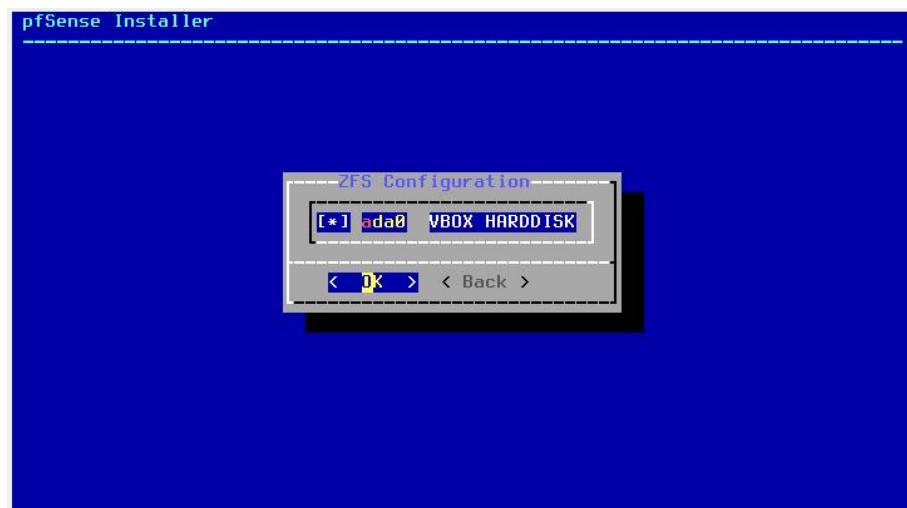
- Marcamos Pool Type/Disks



- Ahora vamos a Stripe Stripe – No Redundancy

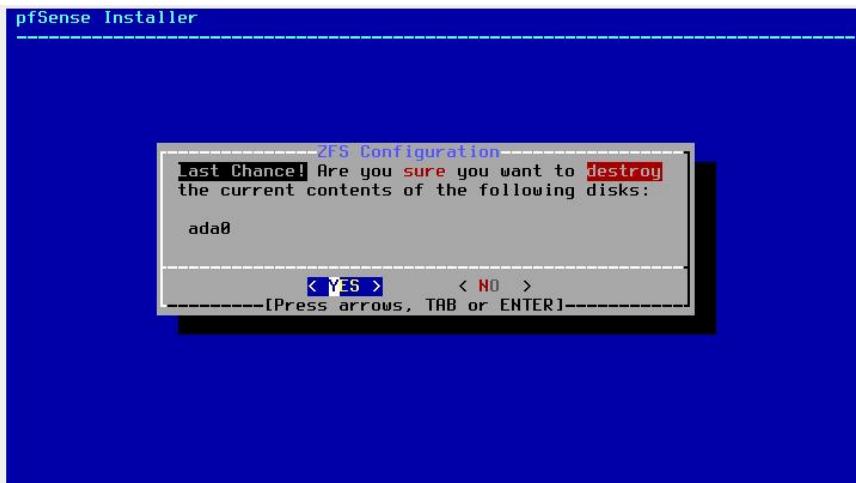


- Y marcamos con la barra espaciadora ada0 VBOX HARDDISK

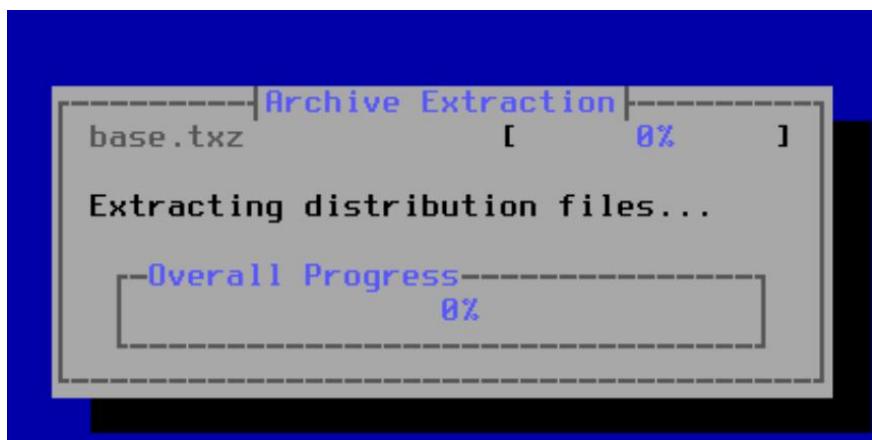




- Le damos a instalar y aquí marcamos Destroy data



- Comienza la instalación



- Una vez instalada, le damos a reboot





1.2CREACIÓN DE LAS REDES EN VIRTUALBOX

Lo primero que vamos a hacer es la creacion de las redes de nuestro UTM.

Para ello, en Virtual Box, vamos a la configuración de nuestro UTM y lo configuraremos de la siguiente manera:

Adaptador 1:

Adaptador 1	Adaptador 2	Adaptador 3	Adaptador 4
<input checked="" type="checkbox"/> Habilitar adaptador de red			
Conectado a:	Adaptador puente		
Nombre:	Intel(R) Wireless-AC 9560 160MHz		

Adaptador 2-LAN:

Adaptador 1	Adaptador 2	Adaptador 3	Adaptador 4
<input checked="" type="checkbox"/> Habilitar adaptador de red			
Conectado a:	Red interna		
Nombre:	LAN		

Adaptador 3-DMZ:

Adaptador 1	Adaptador 2	Adaptador 3	Adaptador 4
<input checked="" type="checkbox"/> Habilitar adaptador de red			
Conectado a:	Red interna		
Nombre:	DMZ		

Adaptador 4-DMZ_2

Adaptador 1	Adaptador 2	Adaptador 3	Adaptador 4
<input checked="" type="checkbox"/> Habilitar adaptador de red			
Conectado a:	Red interna		
Nombre:	DMZ_2		



Si ahora arrancamos el UTM deberíamos ver esto:

```
done.
Starting CRON... done.
pfSense 2.7.2-RELEASE amd64 20231206-2010
Bootup complete

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)

VirtualBox Virtual Machine - Netgate Device ID: 914eb5a50d5d03d866ca

*** Welcome to pfSense 2.7.2-RELEASE (amd64) on pfSense ***

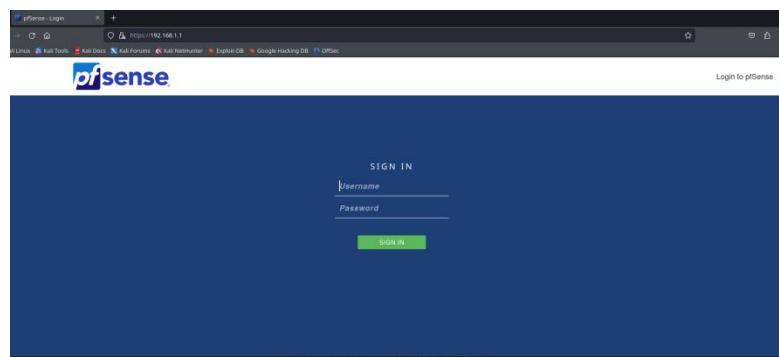
WAN (wan)      -> em0      -> v4/DHCP4: 192.168.1.55/24
LAN (lan)      -> em1      -> v4: 192.168.1.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults   13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: [
```

Ahora vamos a añadir nuestra maquina Kali a la LAN para configurar el PfSense.

Una vez ya tenemos las redes configuradas, ya podemos acceder a nuestro PfSense a través de la dirección ip: 192.168.1.1 desde el navegador de nuestro kali linux.





Nos logeamos con el **nombre**: admin. **contraseña**: pfsense, y ya estamos dentro para empezar con la configuración.

The screenshot shows the initial page of the pfSense Setup Wizard. At the top, there's a warning message: "WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager." Below this, the page title is "Wizard / pfSense Setup /". A sub-header says "pfSense Setup" and "Welcome to pfSense® software!". It provides a brief overview: "This wizard will provide guidance through the initial configuration of pfSense. The wizard may be stopped at any time by clicking the logo image at the top of the screen. pfSense® software is developed and maintained by Netgate®". There are "Learn more" and "Next" buttons at the bottom.

Vamos a System>setup wizzard rellenamos la configuración:

Nombre del host: UTM

Nombre de dominio: keepcoding.local.

Servidor dns primario: 127.0.0.1 (para que sea el propio pfsense el primer servidor dns).

Servidor dns secundario: 1.1.1.1. (este servidor dns son los servidores de cloudfair)

The screenshot shows the "General Information" step of the pfSense Setup Wizard, which is Step 2 of 9. The title is "Wizard / pfSense Setup / General Information".
The "General Information" section contains the following fields:

- Hostname:** UTM
Description: Name of the firewall host, without domain part.
Example: pfsense, firewall, edgefw
- Domain:** keepcoding.local
Description: Domain name for the firewall.
Example: home.arpa, example.com
Note: Do not end the domain name with '.local' as the final part (Top Level Domain, TLD). The 'local' TLD is widely used by mDNS (e.g. Avahi, Bonjour, Rendezvous, Airprint, Airplay) and some Windows systems and networked devices. These will not network correctly if the router uses 'local' as its TLD. Alternatives such as 'home.arpa', 'local.lan', or 'mylocal' are safe.
- Primary DNS Server:** 127.0.0.1
- Secondary DNS Server:** 1.1.1.1
- Override DNS:**
Allow DNS servers to be overridden by DHCP/PPP on WAN



Aquí, en Time server information lo configuramos de la siguiente manera:

Time server hostname: 2.pfsense.pool.ntp.org

Timezone: Europe/Madrid

https://192.168.1.1/wizard.php?xml=setup_wizard.xml

Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

pfSense COMMUNITY EDITION System Interfaces Firewall Services VPN Status Diagnostics Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Wizard / pfSense Setup / Time Server Information

Step 3 of 9

Time Server Information

Please enter the time, date and time zone.

Time server hostname: 2.pfsense.pool.ntp.org
Enter the hostname (FQDN) of the time server.

Timezone: Europe/Madrid

>> Next

Dejamos todo en modo DHCP ya que va a ser el PfSense el que actúa como servidor DHCP y permitimos el uso de IPs privadas (RFC1918 y Bogons), para que no nos bloquee las IPs internas que vamos a utilizar.

Wizard / pfSense Setup / Configure WAN Interface

Step 4 of 9

Configure WAN Interface

On this screen the Wide Area Network information will be configured.

SelectedType: DHCP

Static IP Configuration

IP Address: [empty]

Subnet Mask: 32

Upstream Gateway: [empty]

RFC1918 Networks

Block RFC1918 Private Networks: Block private networks from entering via WAN
When set, this option blocks traffic from IP addresses that are reserved for private networks as per RFC 1918 (10/8, 172.16/12, 192.168/16) as well as loopback addresses (127/8). This option should generally be left turned on, unless the WAN network lies in such a private address space, too.

Block bogon networks

Block bogon networks: Block non-Internet routed networks from entering via WAN
When set, this option blocks traffic from IP addresses that are reserved (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and obviously should not appear as the source address in any packets received.



Aquí, en la configuración de LAN, vamos a cambiar el rango de direcciones, ahora mismo las direcciones que tenemos son las de ifconfig de nuestro kali linux y lo que vamos a hacer es modificarla para que en lugar de ser esta la puerta de enlace 192.168.1.1 sea 192.168.100.1, para que no entre en conflicto:

LAN IP address: 192.168.100.1

Subnet: Mask: 24

The screenshot shows the pfSense setup wizard at Step 5 of 9, titled 'Configure LAN Interface'. It displays the configuration for the Local Area Network. The 'LAN IP Address' is set to 192.168.100.1 and the 'Subnet Mask' is set to 24. A note indicates that this is the fifth step of nine.

Ahora mismo si vamos al UTM deberíamos ver si le damos a intro que la dirección de nuestra puerta de enlace LAN ha cambiado a la nueva que hemos establecido

```
php-fpm[388]: /index.php: Successful login for user 'admin' from: 192.168.1.100
(Local Database)

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)

VirtualBox Virtual Machine - Netgate Device ID: 914eb5a50d5d03d866ca

*** Welcome to pfSense 2.7.2-RELEASE (amd64) on pfSense ***

WAN (wan)      -> em0      -> v4/DHCP4: 192.168.1.55/24
LAN (lan)      -> em1      -> v4: 192.168.100.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces           10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults   13) Update from console
5) Reboot system               14) Enable Secure Shell (sshd)
6) Halt system                 15) Restore recent configuration
7) Ping host                   16) Restart PHP-FPM
8) Shell

Enter an option: ■
```



Para poder acceder al pfSense ponemos la nueva dirección IP que hemos configurado.

The screenshot shows the pfSense Status / Dashboard interface. On the left, there's a 'System Information' table with details like Name (UTM.keepcoding.local), User (admin@192.168.100.10), System (VirtualBox Virtual Machine), BIOS (Vendor: innotech GmbH, Version: VirtualBox, Release Date: Fri Dec 1 2006), Version (2.7.2-RELEASE (amd64) built on Wed Dec 6 21:10:00 CET 2023 FreeBSD 14.0-CURRENT), CPU Type (Intel(R) Core(TM) i7-8565U CPU @ 1.80GHz), Hardware crypto (Inactive), Kernel PTI (Disabled), MDS Mitigation (Inactive), and Uptime (02 Hours 40 Minutes 49 Seconds). On the right, there's a 'Netgate Services And Support' section with a 'Contract type' set to 'Community Support' (Community Support Only). Below it is a 'NETGATE AND pfSense COMMUNITY SUPPORT RESOURCES' section with links to upgrade support, community resources, and professional services. A note at the bottom of this section states that if you purchase a Netgate Global TAC Support subscription, you must have your Netgate Device ID (NDI) from your firewall in order to validate support for this unit.

En primer lugar comprobamos si tenemos acceso a internet haciendo un ping 1.1.1.1, y vemos que si responde, pero en cambio, si hacemos un ping a mercadona.es no responde

```
> ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=55 time=24.6 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=55 time=9.25 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=55 time=8.84 ms
64 bytes from 1.1.1.1: icmp_seq=4 ttl=55 time=9.42 ms
^C
--- 1.1.1.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 8.842/13.021/24.569/6.670 ms
```

Vamos a nuestro cmd de windows y vamos a ver cual es la ip de mercadona.es hacemos ping mercadona.es y nos sale 35.201.121.112

```
C:\Users\Usuario>ping mercadona.es

Haciendo ping a mercadona.es [35.201.121.112] con 32 bytes de datos:
Respuesta desde 35.201.121.112: bytes=32 tiempo=7ms TTL=116
Respuesta desde 35.201.121.112: bytes=32 tiempo=6ms TTL=116
Respuesta desde 35.201.121.112: bytes=32 tiempo=31ms TTL=116

Estadísticas de ping para 35.201.121.112:
    Paquetes: enviados = 3, recibidos = 3, perdidos = 0
                (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 6ms, Máximo = 31ms, Media = 14ms
```



Por lo tanto, vemos que por tema de resolución de dominio no llegamos, pero poniendo la IP en nuestra kali si llegamos, por lo tanto vemos que tenemos un problema de resolución dns.

```
> ping 35.201.121.112
PING 35.201.121.112 (35.201.121.112) 56(84) bytes of data.
64 bytes from 35.201.121.112: icmp_seq=1 ttl=115 time=7.48 ms
64 bytes from 35.201.121.112: icmp_seq=2 ttl=115 time=8.37 ms
64 bytes from 35.201.121.112: icmp_seq=3 ttl=115 time=9.07 ms
64 bytes from 35.201.121.112: icmp_seq=4 ttl=115 time=8.77 ms
^C
--- 35.201.121.112 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 7.481/8.423/9.067/0.596 ms
```

Ahora vamos a **configurar la resolución DNS**.

Para ello vamos a Services-DNS resolver

- Comprobamos que está habilitado
- Quitamos el **DNSSec**. Es un protocolo que se utiliza para el firmado de las respuestas DNS, cuando haces una consulta DNS la consulta puede ir firmada para que se indique que nadie te la ha modificado por el camino (la consulta que te devuelva, te la envíe a ti y puedes confirmar esa firma para ver que no es un DNS fake)

General DNS Resolver Options

Enable Enable DNS resolver

● Habilitamos el **Forwarding mode** para que pueda realizar las consultas DNS en caso de que pfSense no sea capaz por sí mismo, en este caso lo que haría sería enviarlas al DNS secundario que sería el 1.1.1.1 de Cloudflare

DNS Query Forwarding Enable Forwarding Mode
If this option is set, DNS queries will be forwarded to the upstream DNS servers defined under System > General Setup or those obtained via dynamic interfaces such as DHCP, PPP, or OpenVPN (if DNS Server Override is enabled there).

Con esto ya tendríamos la configuración DNS. Lanzamos otra vez el ping a mercadona.es para comprobarlo.

```
> ping mercadona.es
PING mercadona.es (35.201.121.112) 56(84) bytes of data.
64 bytes from 112.121.201.35.bc.googleusercontent.com (35.201.121.112): icmp_seq=1 ttl=115 time=105 ms
64 bytes from 112.121.201.35.bc.googleusercontent.com (35.201.121.112): icmp_seq=2 ttl=115 time=9.02 ms
64 bytes from 112.121.201.35.bc.googleusercontent.com (35.201.121.112): icmp_seq=3 ttl=115 time=29.1 ms
^C
--- mercadona.es ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 9.018/47.715/105.066/41.370 ms
```



Ya tenemos nuestro DNS funcionando y configurado, ahora queremos poner un rango de direccionamiento a cada una de las redes, para ello nos vamos a configurar para que el servidor DHCP de nuestra red nos otorgue ese rango en las maquinas que tenemos en esa subred.

Para ello vamos a interfaces y le damos a LAN, aquí deberíamos tener la interfaz habilitada y en static ipv4.

Interfaces / LAN (em1)

General Configuration

Enable	<input checked="" type="checkbox"/> Enable interface
Description	LAN Enter a description (name) for the interface here.
IPv4 Configuration Type	Static IPv4

Ahora vamos en service a DHCP-SERVER-LAN.

- Modificamos el rango de direcciones de la LAN (192.168.100.100 to 192.168.100.200) con el fin de conseguir la subred que tenemos definido en nuestro esquema de red.

Primary Address Pool

Subnet	192.168.100.0/24
Subnet Range	192.168.100.1 - 192.168.100.254
Address Pool Range	From: 192.168.100.100 To: 192.168.100.200

- **Modificamos los servidores DNS**, ponemos el propio pfSense (hay que tener en cuenta que hay que poner la IP de puerta de enlace, por la que vamos a acceder a nuestra máquina Kali, que sería 192.168.100.1).
- En la red LAN y como **servidores DNS secundarios** ponemos el 1.1.1.1 (Cloudflare) y el 8.8.8.8 (Google), por si el DNS no fuese capaz de hacer la resolución de nombres.

Server Options

WINS Servers	WINS Server 1
	WINS Server 2
DNS Servers	192.168.100.1
	1.1.1.1
	8.8.8.8
	DNS Server 4



- Modificamos la puerta de enlace/Gateway, ponemos la misma dirección, 192.168.100.1

Other DHCP Options

Gateway	192.168.100.1
---------	---------------

The default is to use the IP address of this firewall interface as the gateway. Specify an alternate gateway here if this is not the correct gateway for the network. Enter "none" for no gateway assignment.

Ahora en nuestra kali vemos nuestra IP con ifconfig y es la 100.10
Vamos a desconectarnos y conectarnos de la red, para que nos asigne una nueva dirección IP del rango que hemos configurado en el servidor DHCP, que son la 100.10. hacemos ifconfig para comprobar.

```
> ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
        ether 02:42:52:bd:ae:82 txqueuelen 0 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.10 netmask 255.255.255.0 broadcast 192.168.100.255
        inet6 fe80::5d2e:960a:3176:4c0a prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:b7:26:75 txqueuelen 1000 (Ethernet)
            RX packets 6971 bytes 3591505 (3.4 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 6574 bytes 743307 (725.8 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 69076 bytes 5955197 (5.6 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 69076 bytes 5955197 (5.6 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Ya tenemos definida nuestra red, ya tenemos el direccionamiento que queremos, ya hemos cumplido siguiente objetivo que nuestra maquina kali este en la red LAN.

Ahora volvemos a PfSense para **configurar las DMZ y DMZ_2 de nuestra red.**

Interfaces > interfaces assignments (aquí vemos que podemos añadir mas interfaces de red)

Interfaces / Interface Assignments

Interface	Network port	
WAN	em0 (08:00:27:bc:3e:90)	<input type="button" value="Delete"/>
LAN	em1 (08:00:27:1d:b5:3d)	<input type="button" value="Delete"/>
Available network ports:	em2 (08:00:27:84:15:a0)	<input type="button" value="Add"/>



Estas interfaces de red son las que podemos ver en la propia maquina de UTM (si vamos a la configuración de la maquina, a red, donde definimos las 4 interfaces de red, ahora mismo solo estamos utilizando la 1 que seria la interfaz WAN y la 2 que seria la LAN) entonces vamos a añadir las que nos faltan, las DMZ, para ello le damos a añadir 2 veces y nos tienen que salir la em2 y la em3.

Interfaces / Interface Assignments

Interface has been added.

Interface	Network port	
WAN	em0 (08:00:27:bc:3e:90)	
LAN	em1 (08:00:27:1d:b5:3d)	
OPT1	em2 (08:00:27:84:15:a0)	
OPT2	em3 (08:00:27:7b:6b:a6)	

Save

Vemos que ahora ya nos aparece OPT1 y OPT2, vamos a **configurar estas interfaces de red**.

pfSense COMMUNITY EDITION

System ▾ Interfaces ▾ Firewall ▾

Interfaces / Interface Assignments

Interface has been added.

Assignments

- WAN
- LAN
- OPT1
- OPT2

Interface Assignments Interface Groups Wireless



Configuramos las interfaces → Interfaces-OPT1

- **Enable** → Habilitamos la interface
- **Description** → DMZ
- **IPv4 Configuration type** → Static IPv4 (esta dirección es la que vemos en la maquina de la UTM, es la de la puerta de enlace, para que no cambie tiene que ser estática para las maquinas que hay en la red sea su dirección de salida del router)

Interfaces / OPT1 (em2)

General Configuration

Enable	<input checked="" type="checkbox"/> Enable interface
Description	DMZ
Enter a description (name) for the interface here.	
IPv4 Configuration Type	Static IPv4

- **IPv4 address** → 192.168.200.1/24

Static IPv4 Configuration

IPv4 Address	192.168.200.1
/	24

Configuramos las interfaces → Interfaces-OPT1

- Ahora vamos a OPT2, hacemos lo mismo pero cambiando el nombre a DMZ_2 y el rango 192.168.250.1/24

Interfaces / OPT2 (em3)

General Configuration

Enable	<input checked="" type="checkbox"/> Enable interface
Description	DMZ_2
Enter a description (name) for the interface here.	
IPv4 Configuration Type	Static IPv4
IPv6 Configuration Type	None
MAC Address	XX:XX:XX:XX:XX:XX
This field can be used to modify ("spoof") the MAC address of this interface. Enter a MAC address in the following format: xx:xx:xx:xx:xx or leave blank.	
MTU	
If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.	
MSS	
If a value is entered in this field, then MSS clamping for TCP connections to the value entered above minus 40 for IPv4 (TCP/IPv4 header size) and minus 60 for IPv6 (TCP/IPv6 header size) will be in effect.	
Speed and Duplex	Default (no preference, typically autoselect)
Explicitly set speed and duplex mode for this interface. WARNING: MUST be set to autoselect (automatically negotiate speed) unless the port this interface connects to has its speed and duplex forced.	

Static IPv4 Configuration

IPv4 Address	192.168.250.1
/	24



Si nos vamos a la maquina de UTM y le damos a intro ya nos aparecer nuestras 4 interfaces bien definidas con los rangos que queremos.

```
FreeBSD/amd64 (UTM.keepcoding.local) (ttyv0)
VirtualBox Virtual Machine - Netgate Device ID: 914eb5a50d5d03d866ca
*** Welcome to pfSense 2.7.2-RELEASE (amd64) on UTM ***
WAN (wan)      -> em0      -> v4/DHCP4: 192.168.1.55/24
LAN (lan)      -> em1      -> v4: 192.168.100.1/24
DMZ (opt1)     -> em2      -> v4: 192.168.200.1/24
DMZ_2 (opt2)   -> em3      -> v4: 192.168.250.1/24

0) Logout (SSH only)          9) pfTop
1) Assign Interfaces          10) Filter Logs
2) Set interface(s) IP address 11) Restart webConfigurator
3) Reset webConfigurator password 12) PHP shell + pfSense tools
4) Reset to factory defaults 13) Update from console
5) Reboot system              14) Enable Secure Shell (sshd)
6) Halt system                15) Restore recent configuration
7) Ping host                  16) Restart PHP-FPM
8) Shell

Enter an option: ■
```

Vamos a la configuración del DHCP y habilitamos el DHCP server en esas interfaces

Vamos a configurar el direccionamiento que le vamos a dar a nuestros segmentos de red, que hemos definido en nuestro esquema, en nuestro DMZ y DMZ_2, junto con la habilitación del servidor DHCP.

- Enable **DHCP server** on DMZ interface
- **Range** 192.168.200.100 to 192.168.200.150

Primary Address Pool

Subnet	192.168.200.0/24
Subnet Range	192.168.200.1 - 192.168.200.254
Address Pool Range	<input type="text" value="192.168.200.100"/> <input type="text" value="192.168.200.150"/> From To

The specified range for this pool must not be within the range configured on any other address pool for this interface.

- **DNS servers:** 192.168.200.1 (la del pfsense), segundo servidor 1.1.1.1 y tercer servidor 8.8.8.8

Server Options

WINS Servers	<input type="text" value="WINS Server 1"/>
	<input type="text" value="WINS Server 2"/>
DNS Servers	<input type="text" value="192.168.200.1"/> <input type="text" value="1.1.1.1"/> <input type="text" value="8.8.8.8"/>

- **Gateway** 192.168.200.1 (la del pfsense)

Other DHCP Options

Gateway	<input type="text" value="192.168.200.1"/>
---------	--

The default is to use the IP address of this firewall interface as the gateway. Specify an alternate gateway here if this is not the correct gateway for the network. Enter "none" for no gateway assignment.



Ahora vamos a DMZ_2 y hacemos lo mismo

- El rango de direcciones IP 192.168.250.100 a 192.168.250.150

Primary Address Pool	
Subnet	192.168.250.0/24
Subnet Range	192.168.250.1 - 192.168.250.254
Address Pool Range	192.168.250.100
From	To
	192.168.250.150

- En los servidores DNS 192.168.250.1, luego el 1.1.1.1 y el 8.8.8.8

Server Options	
WINS Servers	WINS Server 1
	WINS Server 2
DNS Servers	192.168.250.1
	1.1.1.1
	8.8.8.8

- En la gateway 192.168.250.1

Other DHCP Options	
Gateway	192.168.250.1
<small>The default is to use the IP address of this firewall interface as the gateway. Specify an alternate gateway here if this is not the correct gateway for the network. Enter "none" for no gateway assignment.</small>	

Ahora mismo tenemos configurado el servidor DHCP y el DNS.

Ya tenemos en los adaptadores LAN, DMZ y DMZ_2 configurados el rango de IP y el servidor DHCP.

Ya hemos asignado a las interfaces de red diferentes segmentos de IP, ahora vamos a comprobar si lo que hemos hecho funciona.

Para comprobar que si metemos algo en kali-2 (ver esquema), nos da un rango dinámico de direccionamiento de DHCP correspondiente a 192.168.200.100 a 192.168.200.150, le tendríamos que cambiar la puerta de entrada al kali que tenemos.

Si ahora cogemos la kali, y en vez de conectarnos a la LAN, la conectamos a la DMZ, el servidor DHCP nos debería dar automáticamente una nueva IP.

Para comprobar que funciona, nos vamos a la kali, configuración y hacemos un cambio de red (en el adaptador 1, donde está la LAN, ponemos DMZ).

Adaptador 1	Adaptador 2	Adaptador 3	Adaptador 4
<input checked="" type="checkbox"/> Habilitar adaptador de red	Conectado a:	Red interna	
	Nombre:	DMZ	



Si ahora hacemos un ifconfig deberíamos ver que nos ha cambiado la IP, y que la que no da está dentro del rango de las direcciones de DMZ que hemos definido.

```
> ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
                ether 02:42:02:16:4a:76 txqueuelen 0 (Ethernet)
                RX packets 0 bytes 0 (0.0 B)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 0 bytes 0 (0.0 B)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.100.100 netmask 255.255.255.0 broadcast 192.168.100.255
                inet6 fe80::5d2e:960a:3176:4c0a prefixlen 64 scopeid 0x20<link>
                ether 08:00:27:b7:26:75 txqueuelen 1000 (Ethernet)
                RX packets 1246 bytes 774461 (756.3 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 1076 bytes 127894 (124.8 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
                inet6 ::1 prefixlen 128 scopeid 0x10<host>
                loop txqueuelen 1000 (Local Loopback)
                RX packets 9555 bytes 887799 (866.9 KiB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 9555 bytes 887799 (866.9 KiB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

> ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
        inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
                ether 02:42:02:16:4a:76 txqueuelen 0 (Ethernet)
                RX packets 0 bytes 0 (0.0 B)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 0 bytes 0 (0.0 B)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.200.100 netmask 255.255.255.0 broadcast 192.168.200.255
                inet6 fe80::5d2e:960a:3176:4c0a prefixlen 64 scopeid 0x20<link>
                ether 08:00:27:b7:26:75 txqueuelen 1000 (Ethernet)
                RX packets 1252 bytes 775669 (757.4 KiB)
```

Ya tenemos configuradas las redes y hemos comprobado que funciona.

Con estas configuraciones vemos que no tenemos acceso a ping ni a internet.

Nos cambiamos a la red LAN para poder acceder todos bien al Pfsense y poder ver un poco lo que nos esta pasando ahora mismo.

Red

Adaptador 1	Adaptador 2	Adaptador 3	Adaptador 4
<input checked="" type="checkbox"/> Habilitar adaptador de red			
Conectado a:	Red interna		
Nombre:	LAN		
Avanzado			



Creamos las **reglas de los FW** quedando la siguiente configuración:

Una vez en Pfsense, si nos vamos a firewall-rules, vemos en la LAN por defecto tenemos reglas configuradas.

Firewall / Rules / LAN

Floating WAN LAN DMZ DMZ_2

Rules (Drag to Change Order)

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓ 1/2.88 MiB	*	*	*	LAN Address	443 80	*	*	*	Anti-Lockout Rule	
✓ 63/1.60 MiB	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	
✓ 0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	

Pero en el resto de DMZ no tenemos nada.

Firewall / Rules / DMZ

Floating WAN LAN DMZ DMZ_2

Rules (Drag to Change Order)

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
No rules are currently defined for this interface All incoming connections on this interface will be blocked until pass rules are added. Click the button to add a new rule.										

Los firewalls por defecto rechazan automáticamente todo el tráfico, entonces al cambiarnos de red, si vamos a status-systems logs, y nos vamos a los logs de los firewalls vemos todas las peticiones DNS que hemos hecho están bloqueadas.

Status / System Logs / Firewall / Normal View

System Firewall DHCP Authentication IPsec PPP PPPoE/L2TP Server OpenVPN NTP Packages Settings

Normal View Dynamic View Summary View

Last 500 Firewall Log Entries. (Maximum 500)

Action	Time	Interface	Rule	Source	Destination	Protocol
✗	Mar 19 18:59:29	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:33332	1.1.1.1:53	UDP
✗	Mar 19 18:59:29	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:33332	1.1.1.1:53	UDP
✗	Mar 19 18:59:30	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:57481	1.1.1.1:53	UDP
✗	Mar 19 18:59:30	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:57481	1.1.1.1:53	UDP
✗	Mar 19 18:59:30	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:48778	1.1.1.1:53	UDP
✗	Mar 19 18:59:32	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:49003	8.8.8.8:53	UDP
✗	Mar 19 18:59:32	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:49003	8.8.8.8:53	UDP
✗	Mar 19 18:59:33	DMZ	Default deny rule IPv4 (1000000103)	192.168.200.100:48901	8.8.8.8:53	UDP



Configuración de los firewall:

Ahora vamos a hacer la configuración del firewall, para eso entramos en **firewall-rules**. y entramos a DMZ.

Para navegar a internet necesitamos habilitar el **protocolo http, puerto 4430,80,53** de DNS para que se lleve a cabo la resolución de nombre.

Vamos a crear las reglas de los firewall, para ello vamos a entrar dentro de firewall-aliases, y en **puertos**. Aquí vamos a poner el puerto 80 y el 443. Esto lo hacemos para cuando pongamos en una regla de firewall queramos habilitar el tráfico hacia la web, poniendo puertos web nos coge todo. No ponemos el puerto DNS porque utiliza lo que se denomina UDP(conexión no orientada) y HTTP y HTTPS utiliza TCP (en TCP son las tramas lo que se trata es que no se pierda ningún tipo de paquete cuando se establece la conexión)

Firewall / Aliases / Edit

Properties							
Name	<input type="text" value="webs"/> The name of the alias may only consist of the characters "a-z, A-Z, 0-9 and _".						
Description	<input type="text" value="PuertosWeb"/> A description may be entered here for administrative reference (not parsed).						
Type	<input type="text" value="Port(s)"/>						
Port(s)							
Hint	Enter ports as desired, with a single port or port range per entry. Port ranges can be expressed by separating with a colon.						
Port	<table border="1"><tr><td><input type="text" value="80"/></td><td><input type="text" value="HTTP"/></td><td><input type="button" value="Delete"/></td></tr><tr><td><input type="text" value="443"/></td><td><input type="text" value="HTTPS"/></td><td><input type="button" value="Delete"/></td></tr></table>	<input type="text" value="80"/>	<input type="text" value="HTTP"/>	<input type="button" value="Delete"/>	<input type="text" value="443"/>	<input type="text" value="HTTPS"/>	<input type="button" value="Delete"/>
<input type="text" value="80"/>	<input type="text" value="HTTP"/>	<input type="button" value="Delete"/>					
<input type="text" value="443"/>	<input type="text" value="HTTPS"/>	<input type="button" value="Delete"/>					

Ahora nos **vamos a crear las reglas en los firewalls**, nos metemos en **firewall-rules**. y nos metemos en DMZ, añadimos las reglas. En **action** ponemos en **pass**(porque estamos creando una regla para permitir el trafico)

Edit Firewall Rule

Action	<input type="text" value="Pass"/>
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.	



En **interface** ponemos DMZ, en **adress family** ponemos IPv4, en **protocol** ponemos TCP, para hacerlo mas seguro, en **source** marcamos **DMZ_subnets**, en **destination** ponemos from (others) en custom webs, y en to (other) y en custom webs.(hace referencia a los puertos que hemos definido antes en el alias puerto 443 y 80). en la **descripcion** podemos poner salida tráfico web.

Asi quedaría nuestra configuración:

Edit Firewall Rule

Action	Pass				
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.					
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.				
Interface	DMZ				
Choose the interface from which packets must come to match this rule.					
Address Family	IPv4				
Select the Internet Protocol version this rule applies to.					
Protocol	TCP				
Choose which IP protocol this rule should match.					
Source					
Source	<input type="checkbox"/> Invert match	DMZ subnets	Source Address	/	<input type="button" value="Display Advanced"/>
The Source Port Range for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.					
Destination					
Destination	<input type="checkbox"/> Invert match	Any	Destination Address	/	<input type="button" value="Display Advanced"/>
Destination Port Range	(other)	webs	(other)	webs	From Custom To Custom

Ahora añadimos la regla que nos falta, que sería la del protocolo DNS, la de UDP. nos metemos en firewall-rules. y nos metemos en DMZ, añadimos las reglas.

Cambiamos el **protocolo** ponemos UDP en **destinations** ponemos from DNS(53) a to DNS(53) y en la **description** ponemos permitir trafico DNS.

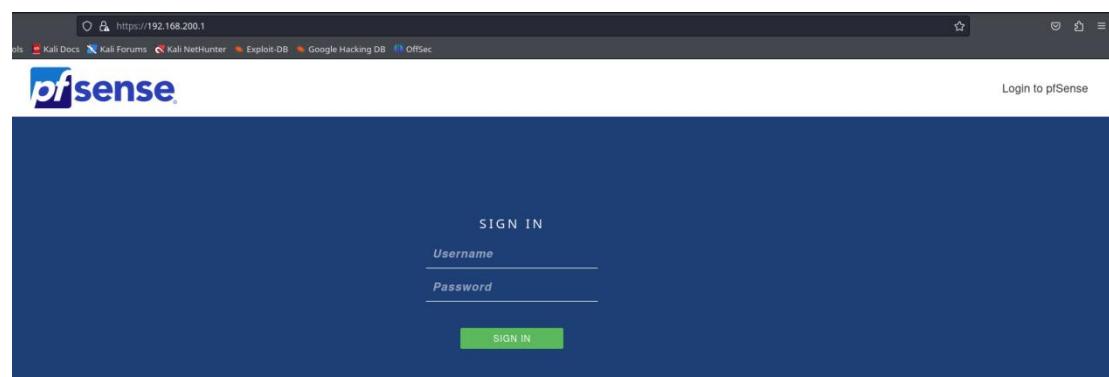
Edit Firewall Rule

Action	Pass				
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.					
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.				
Interface	DMZ				
Choose the interface from which packets must come to match this rule.					
Address Family	IPv4				
Select the Internet Protocol version this rule applies to.					
Protocol	UDP				
Choose which IP protocol this rule should match.					
Source					
Source	<input type="checkbox"/> Invert match	DMZ subnets	Source Address	/	<input type="button" value="Display Advanced"/>
The Source Port Range for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.					
Destination					
Destination	<input type="checkbox"/> Invert match	Any	Destination Address	/	<input type="button" value="Display Advanced"/>
Destination Port Range	DNS (53)		DNS (53)		From Custom To Custom



Ahora vamos a ver si los cambios han funcionado, cambiamos la maquina de red a la DMZ para que nos asigne la nueva IP. ahora vamos a la consola y ponemos ip a. nos metemos en el navegador y vemos que funciona, ya tenemos acceso. y si queremos acceder al PfSense (192.168.200.1) también tenemos acceso a la pagina del PfSense.

```
> ip a
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 noprefixroute
        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 08:00:27:b7:26:75 brd ff:ff:ff:ff:ff:ff
    inet 192.168.100.100/24 brd 192.168.100.255 scope global dynamic noprefixroute eth0
        valid_lft 5781sec preferred_lft 5781sec
    inet6 fe80::5d2e:960a:3176:4c0a/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:02:16:4a:76 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
> ip a
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 noprefixroute
        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 08:00:27:b7:26:75 brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.100/24 brd 192.168.200.255 scope global dynamic noprefixroute eth0
        valid_lft 7198sec preferred_lft 7198sec
    inet6 fe80::5d2e:960a:3176:4c0a/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:02:16:4a:76 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
```



Ahora que estamos dentro de la red, vamos a ver en la consola y nos gustaría saber si los servidores de google están bien. hacemos ping 8.8.8.8 y vemos que no funciona, porque el ping no va ni por TCP, ni por UDP, va por lo que se llama ICMP.

```
> ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
^C
--- 8.8.8.8 ping statistics ---
47 packets transmitted, 0 received, 100% packet loss, time 47096ms

[ping progress bar: 0% to 100%]
```



Para que nos funcione el ping deberíamos añadir otra **regla de permitido** en la DMZ desde el **protocolo ICMP**. En **source** ponemos desde la DMZ subnets y en **destino** lo dejamos en any.

Edit Firewall Rule

Action	Pass		
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.			
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.		
Interface	DMZ		
Choose the interface from which packets must come to match this rule.			
Address Family	IPv4		
Select the Internet Protocol version this rule applies to.			
Protocol	ICMP		
Choose which IP protocol this rule should match.			
ICMP Subtypes	any Alternate Host Datagram conversion error Echo reply		
For ICMP rules on IPv4, one or more of these ICMP subtypes may be specified.			
Source			
Source	<input type="checkbox"/> Invert match	DMZ subnets	Source Address /
Destination			
Destination	<input type="checkbox"/> Invert match	Any	Destination Address /

Si ahora hacemos ping ya nos responde.

```
› ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=11.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=10.3 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=8.16 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=8.37 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 8.159/9.703/11.941/1.546 ms
```



Nos faltaría configurar la DMZ_2.

Para ello volvemos a las reglas de firewall y añadimos una nueva regla con el **protocolo**: TCP, **origen** DMZ_2 subnets, **destino**: any (destination port range: other, custom: webs, to: other, custom: webs).

Edit Firewall Rule

Action	Pass				
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.					
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.				
Interface	DMZ_2				
Choose the interface from which packets must come to match this rule.					
Address Family	IPv4				
Select the Internet Protocol version this rule applies to.					
Protocol	TCP				
Choose which IP protocol this rule should match.					
Source					
Source	<input type="checkbox"/> Invert match	DMZ_2 subnets	Source Address	/	
Display Advanced The Source Port Range for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.					
Destination					
Destination	<input type="checkbox"/> Invert match	Any	Destination Address	/	
Destination Port Range	(other)	webs	(other)	webs	
From	Custom	To	Custom		

Ahora añadimos una nueva regla con el **protocolo**: udp, **origen**: any, **destino port range**: from dns(53) to dns (53).

Edit Firewall Rule

Action	Pass				
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.					
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.				
Interface	DMZ_2				
Choose the interface from which packets must come to match this rule.					
Address Family	IPv4				
Select the Internet Protocol version this rule applies to.					
Protocol	UDP				
Choose which IP protocol this rule should match.					
Source					
Source	<input type="checkbox"/> Invert match	Any	Source Address	/	
Display Advanced The Source Port Range for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.					
Destination					
Destination	<input type="checkbox"/> Invert match	Any	Destination Address	/	
Destination Port Range	DNS (53)	Custom	DNS (53)	Custom	
From		To			



Y por ultimo añadimos otro para el protocolo ICMP, **protocolo: ICMP**, **origen: DMZ_2 subnets** y **destino: any**

Firewall / Rules / Edit

Edit Firewall Rule

Action: Pass
Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.

Disabled: Disable this rule
Set this option to disable this rule without removing it from the list.

Interface: DMZ_2
Choose the interface from which packets must come to match this rule.

Address Family: IPv4
Select the Internet Protocol version this rule applies to.

Protocol: ICMP
Choose which IP protocol this rule should match.

ICMP Subtypes: any
Alternate Host
Datagram conversion error
Echo reply
For ICMP rules on IPv4, one or more of these ICMP subtypes may be specified.

Source: Source: Invert match, Destination: DMZ subnets, Source Address: /
Destination: Destination: Invert match, Any, Destination Address: /



El siguiente apartado que queremos conseguir es el kali

El apache es un servidor que vamos a tener dentro de nuestra red, así que nos interesa que siempre este dentro de la misma IP. si cambiamos constantemente el servdior de IP, al final no hay manera de encontrarlo, por lo que se suelen asignar IPs fijas a este tipo de maquinas.

Para ello nos vamos a conectar a la red DMZ_2, tenemos que ir al PfSense, a status-dhcp leases, le damos a leave, y aquí vemos las maquinas que tenemos conectadas y con IP dentro de nuestro PfSense, vemos la MAC de nuestra kali y la IP. esta IP esta asignada de manera dinámica por nuestro servidor de DHCP.

IP Address	MAC Address	Hostname	Description	Start	End	Actions
192.168.250.100	08:00:27:b7:26:75	kali		2024/03/19 20:37:14	2024/03/19 22:37:14	

Interface	Pool Start	Pool End	Used	Capacity	Utilization
DMZ_2	192.168.250.100	192.168.250.150	1	51	1% of 51

Puede ser útil para ademas de establecer direcciones IPs fijas, también establecer un match entre una dirección MAC y una IP para añadir una capa extra de seguridad.

Podemos, por ejemplo, configurar que para nuestra MAC, esta dirección IP es fija, es lo que se conoce como un filtro MAC.

Ahora vamos a añadir un mapeo estático de la dirección de nuestra kali, para que nuestro servidor apache no cambie.

Para ello le damos al + (el de fondo blanco) y aquí nos sale la MAC de nuestro servidor y la IP fija que queremos añadirle, en nuestro caso le añadimos la IP fija que pertenezca a nuestra subred.

Lo que vamos a hacer es darle una IP que no entre en ese rango dinámico que vemos en la imagen.

Para que no entre en conflicto con el servidor DHCP, por ejemplo 192.168.250.99 (que esta fuera del rango de DHCP pero esta dentro del rango de nuestra red DMZ)



Le damos a **arp table static entry** (lo activamos, ruta estática de nuestra tabla de arp, para que nos asocie para siempre esa MAC a esa IP), podemos poner una **descripción** si queremos para tener una referencia establecimiento estático de IP.

Static DHCP Mapping on LAN

DHCP Backend	ISC DHCP
MAC Address	08:00:27:b1:d0
	<input type="button" value="Copy My MAC"/>
MAC address of the client to match (6 hex octets separated by colons).	
Client Identifier	
An optional identifier to match based on the value sent by the client (RFC 2132).	
IP Address	192.168.100.99
IPv4 address to assign this client.	
Address must be outside of any defined pools. If no IPv4 address is given, one will be dynamically allocated from a pool. The same IP address may be assigned to multiple mappings.	
ARP Table Static Entry	<input checked="" type="checkbox"/> Create an ARP Table Static Entry for this MAC & IP Address pair.
Hostname	kali
Name of the client host without the domain part.	
Description	Establecimiento estatico de IP
A description for administrative reference (not parsed).	

(ip Address: 192.168.250.99)

Desconectamos y conectamos y hacemos un ip a y vemos la 100.99 como dirección asignada en eth0.

```
> ip a
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 noprefixroute
        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 08:00:27:b7:26:75 brd ff:ff:ff:ff:ff:ff
    inet 192.168.250.99/24 brd 192.168.250.255 scope global dynamic noprefixroute eth0
        valid_lft 7193sec preferred_lft 7193sec
    inet6 fe80::5d2e:960a:3176:4c0a/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:02:16:4a:76 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
```



APACHE:

Ahora por último vamos a levantar nuestro servidor de apache, para ello hacemos un **sudo service apache2 start**

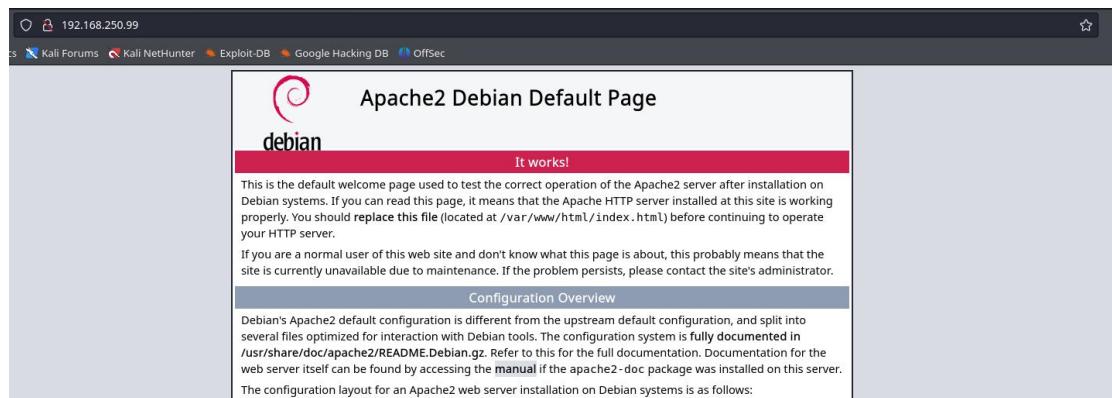
```
> sudo service apache2 start
[sudo] password for kali:
[  4s ]
```

Si ahora hacemos un **sudo service apache2 status** deberíamos ver que esta corriendo.

```
> sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; disabled; preset: disabled)
   Active: active (running) since Tue 2024-03-19 16:55:18 EDT; 1min 13s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 107388 (apache2)
   Tasks: 6 (limit: 9443)
    Memory: 18.4M (peak 18.7M)
      CPU: 97ms
     CGroup: /system.slice/apache2.service
             ├─107386 /usr/sbin/apache2 -k start
             ├─107391 /usr/sbin/apache2 -k start
             ├─107392 /usr/sbin/apache2 -k start
             ├─107393 /usr/sbin/apache2 -k start
             ├─107394 /usr/sbin/apache2 -k start
             └─107395 /usr/sbin/apache2 -k start

Mar 19 16:55:18 kali systemd[1]: Starting apache2.service - The Apache HTTP Server...
Mar 19 16:55:18 kali apachectl[107385]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally!
Mar 19 16:55:18 kali systemd[1]: Started apache2.service - The Apache HTTP Server.
[lines 1-20/20 (END)]
```

Si en el navegador ponemos la IP 192.168.250.99 deberíamos ver que esta corriendo el servidor de Apache en la IP que le hemos asignado.



Tenemos una IP fija en la kali y servidor apache (lo tratamos como una única máquina, ver esquema), tenemos la kali asignada con una IP fija que es la 192.168.250.99, y si nos vamos a nuestra kali, poniendo esa ip deberíamos acceder a nuestro portal del Apache.

Vamos a hacer una configuración para que los que estén dentro de la DMZ_2 puedan acceder al servidor, y gente que esté detrás de nuestro UTM también puedan acceder.

Ahora vamos a conectarnos desde nuestro ordenador.

Tenemos que hacerle la petición a la WAN, del Pfsense, que es la que tiene contacto con el router de nuestra casa, la petición va a ir al router de nuestra casa y de ahí va a ir al Pfsense (la va a derivar).



Si ponemos la dirección de la WAN en el navegador, 192.168.1.55, no nos llega a la kali.

Esto es porque no hemos creado un túnel o un paso entre la WAN del Pfsense y la DMZ_2 del Pfsense.

Ahora mismo solo tenemos reglas de firewall para entrada y salida de tráfico por el puerto 53, 80 y 443. como no estamos haciendo peticiones y estamos yendo solo a una única IP el Pfsense no es capaz de hacer nada con eso porque la IP a la que estamos dirigiéndonos es hacia la WAN del Pfsense.

Tenemos que crear una redirección para que el tráfico que estamos enviando a la WAN del Pfsense nos lo encapsule y nos lo mande hacia la DMZ_2, lo redirija (lo que sería un nateo).

Lo que vamos a hacer en pfsense es crear una regla para decirle que todo el tráfico que nos llegue al puerto 80 de nuestra interfaz WAN, nos lo reenvíe hacia nuestra DMZ_2. entonces vamos a ir a firewalls-Nat y añadimos una regla.

Aquí en **interface**: WAN, **protocolo**: TCP, **destino**: WAN address y **destination port** custom 80, custom 80, **redirección de IP**: address or alias 192.168.250.99 (nuestra kali), en **redirect target port**: other, custom: 80 (para crear la redirección), **description**: regla apache server

Edit Redirect Entry

Disabled	<input type="checkbox"/> Disable this rule			
No RDR (NOT)	<input type="checkbox"/> Disable redirection for traffic matching this rule This option is rarely needed. Don't use this without thorough knowledge of the implications.			
Interface	WAN			
Address Family	IPv4			
Protocol	TCP			
Source	Display Advanced			
Destination	<input type="checkbox"/> Invert match.	WAN address	/	
Destination port range	Other	80	Other	80
Redirect target IP	Address or Alias	192.168.250.99		
Redirect target port	Other	80		



Lo que estamos diciendo es que el trafico que entre hacia la WAN con dirección de la WAN nos lo redireccione (nateo) hacia la IP de la kali que tenemos en nuestra DMZ_2 y al puerto web de la kali (puerto 80, donde esta nuestro servidor de apache).

Si ahora nos vamos a nuestra maquina host y buscamos la IP de nuestra WAN (192.168.1.55), ya accede a nuestro Apache.



REGLAS DEL FIREWALL

WAN:

Firewall / Rules / WAN

Floating WAN LAN DMZ DMZ_2

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓ 0/2 KIB	IPv4 TCP	*	*	192.168.250.99	80 (HTTP)	*	none		NAT Regla Apache server	

Add Add Delete Toggle Copy Save Separator

LAN:

Firewall / Rules / LAN

Floating WAN LAN DMZ DMZ_2

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓ 0/0 B	*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule	
✓ 0/0 B	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	
✓ 0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	

Add Add Delete Toggle Copy Save Separator

DMZ:

Firewall / Rules / DMZ

Floating WAN LAN DMZ DMZ_2

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓ 0/0 B	IPv4 ICMP any	DMZ subnets	*	*	*	*	none			
✓ 0/0 B	IPv4 UDP	*	*	*	53 (DNS)	*	none		Permitir el tráfico DNS	
✓ 0/0 B	IPv4 TCP	DMZ subnets	*	*	webs	*	none		Salida tráfico web	

Add Add Delete Toggle Copy Save Separator

DMZ_2:

Firewall / Rules / DMZ_2

Floating WAN LAN DMZ DMZ_2

States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓ 0/0 B	IPv4 ICMP any	DMZ_2 subnets	*	*	*	*	none			
✓ 0/17 KIB	IPv4 UDP	*	*	*	53 (DNS)	*	none			
✓ 2/298 KIB	IPv4 TCP	DMZ_2 subnets	*	*	webs	*	none			

Add Add Delete Toggle Copy Save Separator



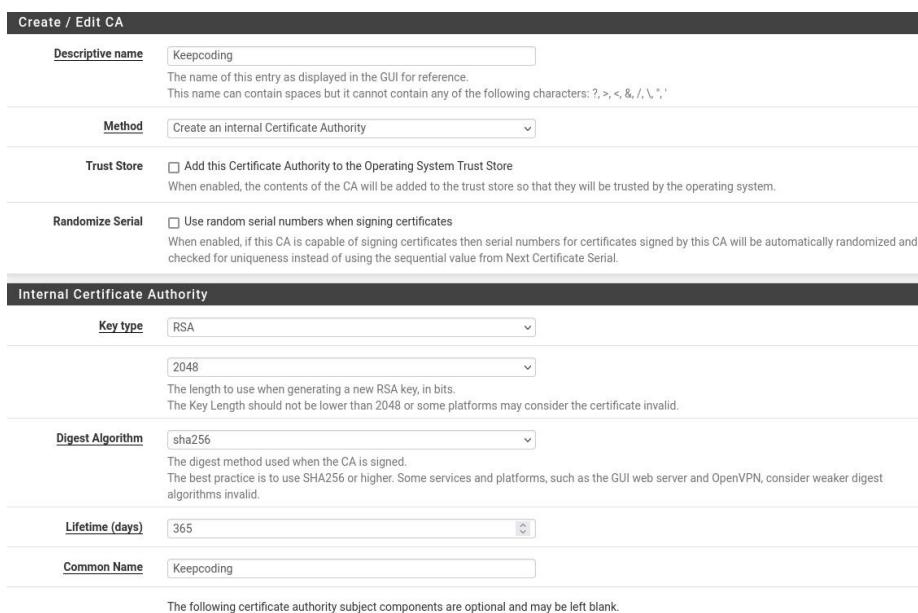
CREACION Y CONFIGURACION DE LA VPN:

Para ello tenemos que instalar un paquete que es el de openvpn, vamos en el pfSense a system-package manager-available packages y buscamos openvpn-client-export 1.9.2 (para poder exportar la configuración del cliente de vpn que vamos a configurar) y le damos a instalar.



Una vez instalado, vamos a ir a la instalación y creación de la vpn, para ello vamos a instalar los certificados con los que nos vamos a conectar a la vpn, cuando nos conectamos a la vpn necesitamos pasarle una serie de credenciales y de claves que confirmen que eres tú.

Vamos a generar los certificados para la conexión de la vpn, para ello vamos a system-certificates y tenemos que crear una entidad certificadora que es la que se va a encargar de generar y gestionar los certificados de nuestra vpn, le damos a add y en **descriptive name**: keepcoding, **method**: create an internal cert auth, **key type**: rsa 2048, **digest algorithm**: sha256, **lifetime**: 365, **common name**: keepcoding, **country code**: ES, **state or province**: Madrid, **city**: madrid, **organization**: keepcoding, **organization unit**: it



Create / Edit CA	
Descriptive name	Keepcoding The name of this entry as displayed in the GUI for reference. This name can contain spaces but it cannot contain any of the following characters: ?, >, <, &, /, \, ,'
Method	Create an internal Certificate Authority
Trust Store	<input type="checkbox"/> Add this Certificate Authority to the Operating System Trust Store When enabled, the contents of the CA will be added to the trust store so that they will be trusted by the operating system.
Randomize Serial	<input type="checkbox"/> Use random serial numbers when signing certificates When enabled, if this CA is capable of signing certificates then serial numbers for certificates signed by this CA will be automatically randomized and checked for uniqueness instead of using the sequential value from Next Certificate Serial.
Internal Certificate Authority	
Key type	RSA
2048	The length to use when generating a new RSA key, in bits. The Key Length should not be lower than 2048 or some platforms may consider the certificate invalid.
Digest Algorithm	sha256 The digest method used when the CA is signed. The best practice is to use SHA256 or higher. Some services and platforms, such as the GUI web server and OpenVPN, consider weaker digest algorithms invalid.
Lifetime (days)	365
Common Name	Keepcoding
The following certificate authority subject components are optional and may be left blank.	

Ahora tenemos que crear el certificado, vamos a la pestaña de al lado, **certificates** y le damos add, **descriptive name**: vpn, **lifetime**: 365,



certificate authority: keepcoding (la que hemos creado antes), **common name:** vpn.keepcoding.local, **certificate type:** server certificate
(IMPORTANTE porque este es el certificado que vamos a crear para nuestro propio servidor del pfsense y que luego vamos a exportar al exterior).

Add/Sign a New Certificate

Method: Create an internal Certificate

Descriptive name: VPN
The name of this entry as displayed in the GUI for reference.
This name can contain spaces but it cannot contain any of the following characters: ?, >, <, &, /, \, *, '

Internal Certificate

Certificate authority: Keepcoding

Key type: RSA

Key Length: 2048
The length to use when generating a new RSA key, in bits.
The Key Length should not be lower than 2048 or some platforms may consider the certificate invalid.

Digest Algorithm: sha256
The digest method used when the certificate is signed.
The best practice is to use SHA256 or higher. Some services and platforms, such as the GUI web server and OpenVPN, consider weaker digest algorithms invalid.

Lifetime (days): 365
The length of time the signed certificate will be valid, in days.
Server certificates should not have a lifetime over 398 days or some platforms may consider the certificate invalid.

Common Name: vpn.keepcoding.local
The following certificate subject components are optional and may be left blank.

Country Code: ES

State or Province: Madrid

City: Madrid

Organization: Keepcoding

Organizational Unit: IT

Certificate Attributes

Attribute Notes: The following attributes are added to certificates and requests when they are created or signed. These attributes behave differently depending on the selected mode.
For Internal Certificates, these attributes are added directly to the certificate as shown.

Certificate Type: Server Certificate
Add type-specific usage attributes to the signed certificate. Used for placing usage restrictions on, or granting abilities to, the signed certificate.

Alternative Names: FQDN or Hostname
Type: Value
Enter additional identifiers for the certificate in this list. The Common Name field is automatically added to the certificate as an Alternative Name. The signing CA may ignore or change these values.

Add SAN Row **+ Add SAN Row**

Ahora tenemos que crear el usuario y los rangos para la vpn que hemos definido, porque dentro de la vpn se va a crear un tunel, y en ese tunel tenemos que asignar una serie de direcciones IP, puerto que va a utilizar la vpn y toda la configuración.



Para ello vamos a ir **vpn-openvpn** y en servers le damos add, **description**: vpn-remota-lan, en **server mode**: remote access (ssl/tls + use auth), en **protocol**: tcp on ipv4 only, en **device mode**: tun - layer 3 tunnel mode (modo tunel, nivel 3 de la capa para conectar redes a redes y nivel 2 para tenerlo todo en el mismo fragmento de red), **protocol**: udp on ipv4 only, **interface**: WAN, en **local port**(esto es lo que vamos a cambiar, es seguridad por oscuridad, viene por defecto el que todo el mundo usa para vpn, le cambiamos el puerto para ocultarla): 4194, en **server certificate**: vpn (server: yes, ca: keepcoding) (el que hemos creado antes), en **hardware crypto**: si nos sale el hardware para poder acelerar el proceso criptografico de la vpn lo seleccionamos, en **ipv4 tunnel network** (ponemos la direccion ip que va a utilizar nuestro tunel de la vpn): 192.168.220.0/24, en **ipv4 local network** (la red para que sea accesible desde el exterior ponemos la de la LAN: 192.168.100.0/24 (esto seria el punto accesible que va a tener nuestra vpn, toda la red que cuelga de la parte de la lan), en **redirect ipv4 gateway** (no lo activamos): si lo activamos se redirige todo el trafico de la red de los usuarios, si queremos tener controlado absolutamente todo lo que hagan los usuarios, lo redireccionas, en **concurrent connections** (gente que se puede conectar a la vez): 10

General Information	
Description	VPN-Remota-LAN A description of this VPN for administrative reference.
Disabled	<input type="checkbox"/> Disable this server Set this option to disable this server without removing it from the list.
Mode Configuration	
Server mode	Remote Access (SSL/TLS + User Auth)
Backend for authentication	Local Database
Device mode	tun - Layer 3 Tunnel Mode "tun" mode carries IPv4 and IPv6 (OSI layer 3) and is the most common and compatible mode across all platforms. "tap" mode is capable of carrying 802.3 (OSI Layer 2.)
Endpoint Configuration	
Protocol	UDP on IPv4 only
Interface	WAN The interface or Virtual IP address where OpenVPN will receive client connections.
Local port	4194 The port used by OpenVPN to receive client connections.
Cryptographic Settings	
TLS Configuration	<input checked="" type="checkbox"/> Use a TLS Key



Automatically generate a TLS Key.

Peer Certificate Authority Keepcoding

Peer Certificate Revocation list No Certificate Revocation Lists defined. One may be created here: [System > Cert. Manager](#)

OCSP Check Check client certificates with OCSP

Server certificate VPN (Server: Yes, CA: Keepcoding)
Certificates known to be incompatible with use for OpenVPN are not included in this list, such as certificates using incompatible ECDSA curves or weak digest algorithms.

DH Parameter Length 2048 bit
Diffie-Hellman (DH) parameter set used for key exchange. [i](#)

ECDH Curve Use Default
The Elliptic Curve to use for key exchange.
The curve from the server certificate is used by default when the server uses an ECDSA certificate. Otherwise, secp384r1 is used as a fallback.

Data Encryption Algorithms
AES-128-CBC (128 bit key, 128 bit block)
AES-128-CFB (128 bit key, 128 bit block)
AES-128-CFB1 (128 bit key, 128 bit block)
AES-128-CFB8 (128 bit key, 128 bit block)
AES-128-GCM (128 bit key, 128 bit block)
AES-128-OFB (128 bit key, 128 bit block)
AES-192-CBC (192 bit key, 128 bit block)
AES-192-CFB (192 bit key, 128 bit block)
AES-192-CFB1 (192 bit key, 128 bit block)
AES-192-CFB8 (192 bit key, 128 bit block)

Available Data Encryption Algorithms
Click to add or remove an algorithm from the list

AES-256-GCM
AES-128-GCM
CHACHA20-POLY1305

Allowed Data Encryption Algorithms. Click an algorithm name to remove it from the list

The order of the selected Data Encryption Algorithms is respected by OpenVPN. This list is ignored in Shared Key mode. [i](#)

Fallback Data Encryption Algorithm AES-256-CBC (256 bit key, 128 bit block)
The Fallback Data Encryption Algorithm used for data channel packets when communicating with clients that do not support data encryption algorithm negotiation (e.g. Shared Key). This algorithm is automatically included in the Data Encryption Algorithms list.

Auth digest algorithm SHA256 (256-bit)
The algorithm used to authenticate data channel packets, and control channel packets if a TLS Key is present.
When an AEAD Encryption Algorithm mode is used, such as AES-GCM, this digest is used for the control channel only, not the data channel.
The server and all clients must have the same setting. While SHA1 is the default for OpenVPN, this algorithm is insecure.

Hardware Crypto Intel RDRAND engine - RAND

Certificate Depth One (Client+Server)
When a certificate-based client logs in, do not accept certificates below this depth. Useful for denying certificates made with intermediate CAs generated from the same CA as the server.

Strict User-CN Matching Enforce match
When authenticating users, enforce a match between the common name of the client certificate and the username given at login.

Client Certificate Key Usage Validation Enforce key usage
Verify that only hosts with a client certificate can connect (EKU: "TLS Web Client Authentication").

Tunnel Settings

IPv4 Tunnel Network 192.168.220.0/24
This is the IPv4 virtual network or network type alias with a single entry used for private communications between this server and client hosts expressed using CIDR notation (e.g. 10.0.8.0/24). The first usable address in the network will be assigned to the server virtual interface. The remaining usable addresses will be assigned to connecting clients.
A tunnel network of /30 or smaller puts OpenVPN into a special peer-to-peer mode which cannot push settings to clients. This mode is not compatible with several options, including Exit Notify, and Inactive.

IPv6 Tunnel Network
This is the IPv6 virtual network or network type alias with a single entry used for private communications between this server and client hosts



Redirect IPv4 Gateway	<input type="checkbox"/> Force all client-generated IPv4 traffic through the tunnel.
Redirect IPv6 Gateway	<input type="checkbox"/> Force all client-generated IPv6 traffic through the tunnel.
IPv4 Local network(s)	192.168.100.0/24 IPv4 networks that will be accessible from the remote endpoint. Expressed as a comma-separated list of one or more CIDR ranges or host/network type aliases. This may be left blank if not adding a route to the local network through this tunnel on the remote machine. This is generally set to the LAN network.
IPv6 Local network(s)	<input type="text"/> IPv6 networks that will be accessible from the remote endpoint. Expressed as a comma-separated list of one or more IP/PREFIX or host/network type aliases. This may be left blank if not adding a route to the local network through this tunnel on the remote machine. This is generally set to the LAN network.
Concurrent connections	10 Specify the maximum number of clients allowed to concurrently connect to this server.
Allow Compression	Refuse any non-stub compression (Most secure) <input type="checkbox"/> Allow compression to be used with this VPN instance. Compression can potentially increase throughput but may allow an attacker to extract secrets if they can control compressed plaintext traversing the VPN (e.g. HTTP). Before enabling compression, consult information about the VORACLE, CRIME, TIME, and BREACH attacks against TLS to decide if the use case for this specific VPN is vulnerable to attack. Asymmetric compression allows an easier transition when connecting with older peers.
Push Compression	<input type="checkbox"/> Push the selected Compression setting to connecting clients.
Type-of-Service	<input type="checkbox"/> Set the TOS IP header value of tunnel packets to match the encapsulated packet value.
Inter-client communication	<input type="checkbox"/> Allow communication between clients connected to this server
Duplicate Connection	<input type="checkbox"/> Allow multiple concurrent connections from the same user When set, the same user may connect multiple times. When unset, a new connection from a user will disconnect the previous session.

Ahora nos vamos a **system-user manager** y aqui añadimos un nuevo usuario que es nuestro usuario de la vpn.

le damos a add, **username**: kike, **password**: 12345 (nos tenemos que acordar), **full name**: kike, le damos a **certificate**: click to create a user certificate, se nos despliega un menu para crear el certificado del usuario para conectarnos con nuestra entidad de certificacion que hemos creado para generar estos certificados (maquinas a parte), **descriptive name**: kike, **certificate authority**: keepcoding, **key type**: rsa 2048, **digest algorithm** sha256, **lifetime**: 365, y le damos a guardar y ya tenemos nuestro usuario creado

User Properties	
Defined by	USER
Disabled	<input type="checkbox"/> This user cannot login
Username	kike
Password	*****
Full name	kike User's full name, for administrative information only
Expiration date	<input type="text"/> Leave blank if the account shouldn't expire, otherwise enter the expiration date as MM/DD/YYYY
Custom Settings	<input type="checkbox"/> Use individual customized GUI options and dashboard layout for this user.
Group membership	admins <input type="button" value="Move to 'Member of' list"/> <input type="button" value="Move to 'Not member of' list"/> Not member of Member of Hold down CTRL (PC)/COMMAND (Mac) key to select multiple items.
Certificate	<input checked="" type="checkbox"/> Click to create a user certificate
Create Certificate for User	
Descriptive name	kike
Certificate authority	Keepcoding



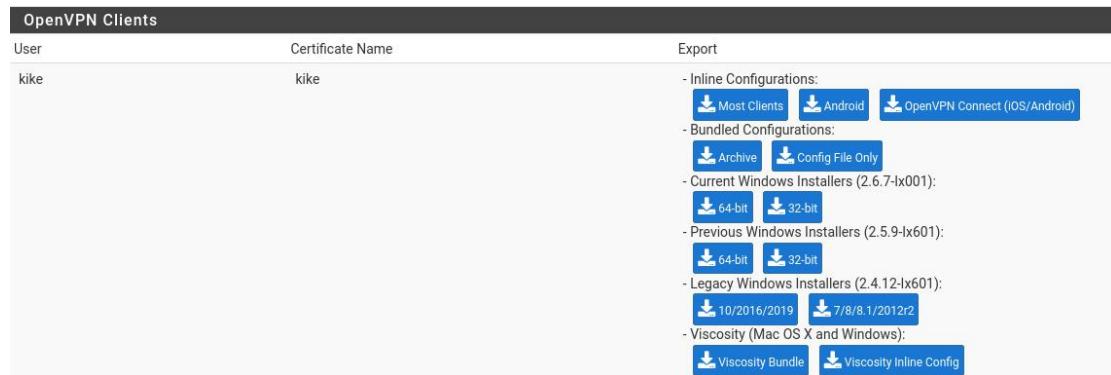
Ahora lo que nos queda seria exportar el certificado de nuestro usuario, para ello nos vamos a **vpn-openvpn** y en la barra roja de arriba a **clien export**.



Seleccionamos la vpn que hemos creado con el puerto que hemos configurado en **remote access serve**: vpn-remota-lan upd4:4194.



En openvpn clients nos tendría que salir el usuario que hemos creado y las configuraciones que hemos creado.

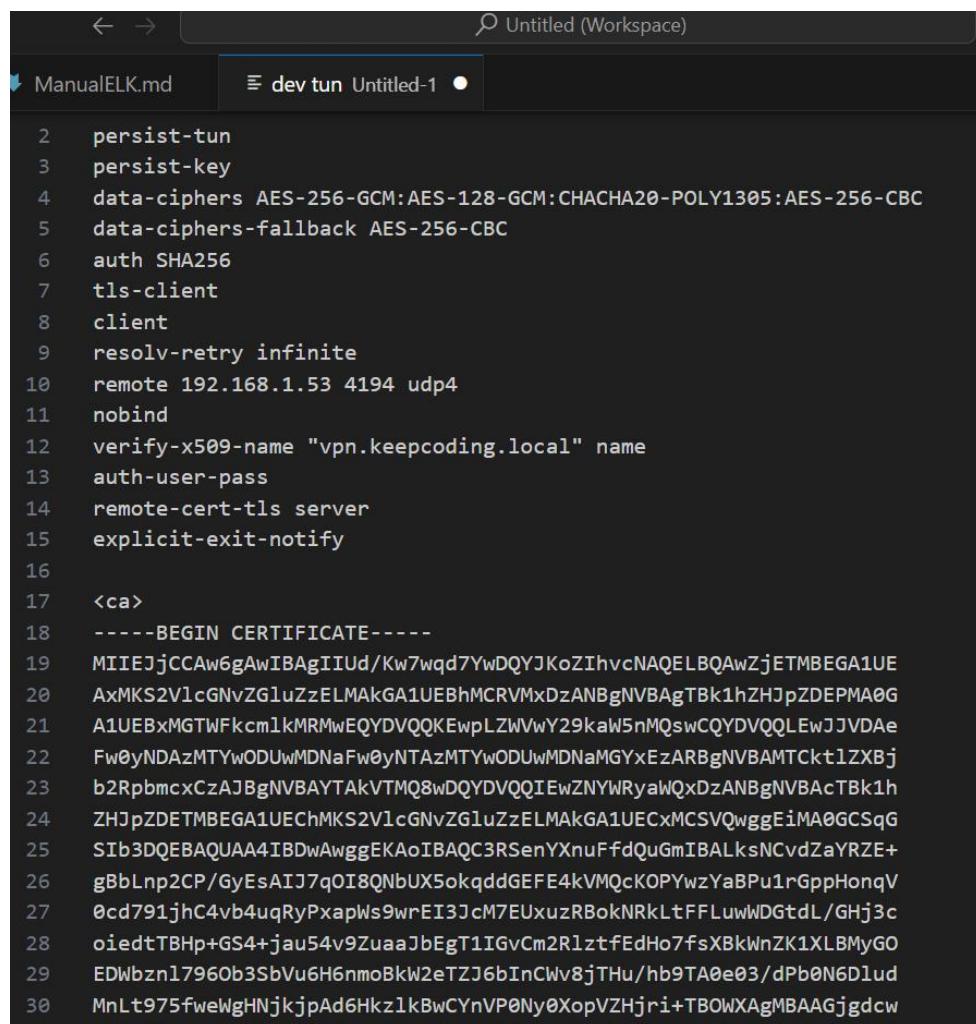


Descargamos **most clients**. se nos descarga un archivo **.ovpn**. ya tenemos el certificado descargado, lo abrimos con el editor de texto, y nos sale en la consola el certificado.

```
1 dev tun
2 persist-tun
3 persist-key
4 data-ciphers AES-256-GCM:AES-128-GCM:CHACHA20-POLY1305:AES-256-CBC
5 data-ciphers-fallback AES-256-CBC
6 auth SHA256
7 tls-client
8 client
9 resolv-retry infinite
10 remote 192.168.1.53 4194 udp
11 nobind
12 verify-x509-name "vpn.keepcoding.local" name
13 auth-user-pass
14 remote-cert-tls server
15 explicit-exit-notify
16
17 <ca>
18 -----BEGIN CERTIFICATE-----
19 MIIEjCCAwwGAwIBAgI1Ud/Kw7qd7YwDQYJKoZIhvNAQELBQAwZjETMBEGA1UE
20 AxMK52VlGNgVzGLuZzELMAkGA1UEBhMCRVMxszANBgNVBAgTBk1hZHjPZDEPMa0G
21 A1UEBxIGTWfkcm1kMRMwEQYDVQKKEwpLzWVwv29kaW5nQswCQYDVQQLEwJVDaE
22 Fw0yMDazNTywODUwMDNawFw0yNTAzMTYwODUwMDNawMGYxEzARBgNVBAMTCktLzVbj
23 b2RpbmCxZaJ8gNVBYATakVTM08wDQYDVQIIEwZNYRyaWQxDzANBgNVBAcTBk1h
24 ZHjPzDETMBEGA1UEChMKS2VlczGluZzELMAkGA1UECxMCsvQwgEiMA0GCSqG
25 S1b3DQEBAQAA4IBDwAwggEKAIBAQ3RSenXnuFdqUgn1BALksNcvdzAYZE+
26 gBbLnp2CP/GyEsA1J7qI8QNbUx5okqddGEFE4kVWQCKOPYwzyabPu1rGppHonqV
27 0cd791hC4vb4u4qrPxapws9wvEt3JcM7EUxzRb0kNRkLTFLuwWDGdL/Ghj3c
28 oiedTBH+GS+&jau54v9ZuaajbEgt1IGvcm2R1ztfedHo7fxBkWzK1XLByGO
29 EDWbznl7960b3SbvU6HnmobKw2tZJ6bInCwv8jTHu/hb9TA0e03/dpB0N6Lud
30 MnL9t9fweigHNjkjpaAd6hk1lkBwCynVP0Ny0xp0ZHjri+TBWxAgMBAAGjgdew
31 gdqWhQyDVROOBByEFGtdHpo2*x/N/0eSMBMLjsN6gkXWIMGXBgNVHSMEgYwgYyA
32 FGtdhp02*x/N/0eSMBMLjsN6gkXW0lwQkaDBmRMwEQYDVOQDewpLzWVwv29kaW5n
33 MQuwCQyDVQOGewJFUZEPMA0GAI1ECBMGTWFkcm1kMQBwDQYDVQOHwZNYWRyalQx
34 EzARBgNVBAoTckt1ZXbj2RpbmCxzaJ8gNVBaStAKIuggrh38rdvCp3tjaMBgNV
35 HRMEBTADAQH/MAsGA1UdDwQEcAwIBBJA0BgkqhkiG9wBAQgFAAACAOEALBjWnRvP
36 7Mjb1FOTTXIBCLix3f1MTafDpBuYiZta8kfQy9NVqr31Y9yjDVjQkEykdh
37 KE9y7JvYdu9PlvdoCxL2P5GFXhA1G6il19f8ipNNe20h31lwVGKghLYsDRV
38 AetY5kxPcZZOXLR120eCNPSzCmjiaS6L2kZXXL5TYJy1N+RHXAcxci+8WZ
39 PmxZXM1owLy77uJTeMi5iJ8giGmy0KBv0IJK3nPq6gdC63bH5iSygNzjH
40 h9EXG0x/g+s5pDwJbo59qBACjP88y1vWKv4eu97017/vNgQk1ogClg15VzR2e19c
41 ze3Kc+kx1ctyqg=
42 -----END CERTIFICATE-----
```



Copiamos el contenido y lo tenemos que tener en nuestra maquina host para conectarnos desde ella. Abrimos un editor, por ejemplo el Visual Studio Code y lo pegamos.



```
← → ⌂ Untitled (Workspace)
ManualELK.md dev tun Untitled-1 ●

2 persist-tun
3 persist-key
4 data-ciphers AES-256-GCM:AES-128-GCM:CHACHA20-POLY1305:AES-256-CBC
5 data-ciphers-fallback AES-256-CBC
6 auth SHA256
7 tls-client
8 client
9 resolv-retry infinite
10 remote 192.168.1.53 4194 udp4
11 nobind
12 verify-x509-name "vpn.keepcoding.local" name
13 auth-user-pass
14 remote-cert-tls server
15 explicit-exit-notify
16
17 <ca>
18 -----BEGIN CERTIFICATE-----
19 MIIEJjCCAw6gAwIBAgIIUd/Kw7wd7YwDQYJKoZIhvcNAQELBQAwZjETMBEGA1UE
20 AxMKS2VlcGNvZGluZzELMAkGA1UEBhMCRVMxDzANBgNVBAgTBk1hZHJpZDEPMA0G
21 A1UEBxMGTwFkcmIkMRMwEQYDVQQKEwpLZWVwY29kaW5nMQswCQYDVQQLEwJJVDAe
22 Fw0yNDAzMjMTYwODUwMDNaFw0yNTAzMTYwODUwMDNaMGYxEzARBgNVBAMTckt1ZXBj
23 b2RpbmcxCzAJBgNVBAYTAkVTMQ8wDQYDVQQIEwZNYWRyaWQxDzANBgNVBAcTBk1h
24 ZHJpZDETMBEGA1UEChMKS2VlcGNvZGluZzELMAkGA1UECxMCSVQwggEiMA0GCSqG
25 SIb3DQEBAQUAA4IBDwAwggEKAoIBAQC3RSenYXnuFFdQuGmIBALksNCvdZaYRZE+
26 gBbLnp2CP/GyEsAIJ7qO18QNbUX5okqddGEFE4kVMQcKOPYwzYaBPu1rGppHonqV
27 0cd791jhC4vb4uqRyPxapWs9wrEI3JcM7EUuzRBokNRkLtFFLuwlDGtdL/GHj3c
28 oiedtTBHp+GS4+jau54v9ZuaajbEgT1IGvCm2R1ztfEdHo7fsXBkWnZK1XLBMyGO
29 EDWbzn1796Ob3SbVu6H6nm0BkW2eTZJ6bInCwv8jTHu/hb9TA0e03/dPb0N6Dlud
30 MnLt975fweWgHNjkjpAd6HkzlkBwCYnVP0Ny0XopVZHjri+TBOWXAgMBAAGjgdew
```

ahora le damos a file-save as (lo guardamos en un sitio donde lo veamos, con formato no extension (*.) y con el nombre por ejemplo kike y la extension ovpn: kike.ovpn)



Ya con el certificado en el escritorio de windows, nos tenemos que descargar la aplicación de **vpnclient** y nos bajamos el cliente en windows (en el host en el que estemos) (este es el cliente de vpn al que le vamos a subir el certificado de nuestra vpn).



Abrimos la aplicación de el cliente y en upload file arrastramos el archivo con el certificado que hemos exportado, y nos debería salir la dirección de la WAN 192.168.1.53 y en el **nombre de usuario**(el que hemos configurado): kike, y le damos a connect y nos pide la **contraseña**: 12345, no se nos conecta porque nos falta la regla de firewall para habilitar el trafico de la vpn.

OpenVPN Connect

– ×

< Imported Profile

Profile Name
192.168.1.53 [kike]

Server Hostname (locked)
192.168.1.53

Username

Save password

PROFILES CONNECT

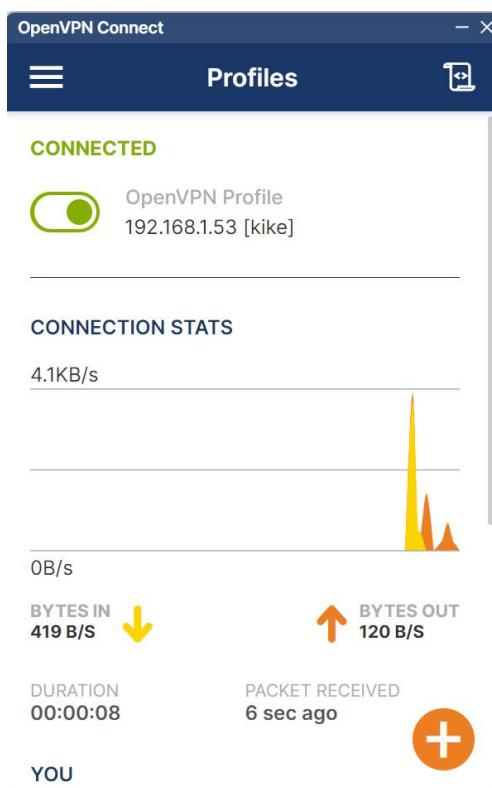


Vamos al pfsense firewall-rules y en la wan (barra de arriba) añdimos una nueva regla de firewall. **action**: pass, **interface**: wan, **address family**: ipv4, **protocol**: UDP, **source**(origen): any, **destination**: custom 4194 custom 4194(puerto que hemos configurado en nuestra vpn), **description**: regla WAN vpn.

Edit Firewall Rule

Action	Pass		
Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.			
Disabled	<input type="checkbox"/> Disable this rule Set this option to disable this rule without removing it from the list.		
Interface	WAN		
Choose the interface from which packets must come to match this rule.			
Address Family	IPv4		
Select the Internet Protocol version this rule applies to.			
Protocol	UDP		
Choose which IP protocol this rule should match.			
Source			
Source	<input type="checkbox"/> Invert match	Any	Source Address
<input type="button" value="Display Advanced"/>			
The Source Port Range for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, any.			
Destination			
Destination	<input type="checkbox"/> Invert match	Any	Destination Address
Destination Port Range	(other)	4194	(other)
From	Custom	To	Custom

Si volvemos a vpn ya se nos conecta.





HONEYBOT-COWRIE:

El honeypot que vamos a lanzar en dmz es el cowrie “ssh” a través el docker. Vamos a mandar los logs generados al archivo **cowrie.log**.

```
> sudo docker run -p 222:2222 cowrie/cowrie > cowrie.log
[sudo] password for kali:
/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:106: CryptographyDeprecationWarning: Blowfish has been deprecated
  b"blowfish-cbc": (algorithms.Blowfish, 16, modes.CBC),
/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:110: CryptographyDeprecationWarning: CAST5 has been deprecated
  b"cast128-cbc": (algorithms.CAST5, 16, modes.CBC),
/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:115: CryptographyDeprecationWarning: Blowfish has been deprecated
  b"blowfish-ctr": (algorithms.Blowfish, 16, modes.CTR),
/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:116: CryptographyDeprecationWarning: CAST5 has been deprecated
  b"cast128-ctr": (algorithms.CAST5, 16, modes.CTR),
|
```

El primer paso que hemos hecho es lanzar el contenedor docker para que arranque con este comando: **docker run -p 222:2222 cowrie/cowrie**. el puerto de la izquierda es del anfitrión y el de la derecha del invitado.

Con **docker ps** vemos que tenemos el contenedor corriendo.

```
> sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
3d3890f948bc        cowrie/cowrie      "/cowrie/cowrie-env/..."   3 minutes ago     Up 3 minutes          2223/tcp, 0.0.0.0:222->2222/tcp, :::222->2222/tcp   static_margulis
```

El que tenemos corriendo es el que hemos puesto que simula un honeypot de ssh y lo que le estamos diciendo es que nos lo abra en el puerto 222

Ahora lo que vamos a hacer es a través del cmd de nuestra maquina (windows), tenemos que hacer un ssh al puerto 222 con el usuario root y con la ip de nuestra maquina.

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.1.56  netmask 255.255.255.0  broadcast 192.168.1.255
      inet6 fe80::5d2e:960a:3176:4c0a  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:b7:26:75  txqueuelen 1000  (Ethernet)
          RX packets 20653  bytes 6035850 (5.7 MiB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 20643  bytes 2512761 (2.3 MiB)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

Abrimos cmd y ponemos **ssh -p 222 root@192.168.1.56** (para conectarnos desde la maquina windows) y nos debería salir la clave para poder establecer la sesión. Nos dice que el fingerprint no se conoce, es nuevo, y por lo tanto nos pregunta si estamos seguros de conectarnos, le damos a si y en la contraseña podemos poner lo que sea y ya estamos dentro.



```
Símbolo del sistema - ssh -p 222 root@192.168.1.56
Microsoft Windows [Versión 10.0.22631.3296]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Usuario> ssh -p 222 root@192.168.1.56
The authenticity of host '[192.168.1.56]:222 ([192.168.1.56]:222)' can't be established.
ED25519 key fingerprint is SHA256:9P3pDLLoEM6NJkVDupmFwb84WXDrMyMlw7UP82zspIg.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[192.168.1.56]:222' (ED25519) to the list of known hosts.
root@192.168.1.56's password:

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.
root@svr04:~#
```

Este honeypot consiste en capturar todos los comandos que se prueban dentro del servidor, si hacemos un **ls** en el cmd de windows nos sale en la consola del kali conectada.

```
root@svr04:~# ls
```

```
2024-03-23T10:58:30+0000 [twisted.conch.ssh.session#info] Getting shell
2024-03-23T11:01:08+0000 [HoneyPotSSHTransport,0,192.168.1.54] CMD: ls
2024-03-23T11:01:08+0000 [HoneyPotSSHTransport,0,192.168.1.54] Command found: ls
```

Es como simular un sistema de un servidor ssh y por lo tanto podríamos capturar los comandos, herramientas que intenten utilizar en nuestro sistema vulnerable y utilizar esa información para hacer threat hunting.



ELASTIC (SIEM):

La idea es que sepamos poder integrar fuentes, y mirar los logs que se recopilan.

Entramos en elastic.

The screenshot shows the Elastic Cloud interface. On the left, there's a sidebar with 'Cloud' selected. The main area has a 'Welcome to Elastic Cloud' banner with a message about serverless preview and a 'Join the waitlist' button. Below it is a table for 'Hosted deployments' with one entry: 'kike-keepcoding' (Status: Healthy, Version: 8.12.2, Cloud provider & region: AWS - Ireland (eu-west-1)). There are 'Create deployment' and 'Actions' buttons. To the right, there are sections for 'Role-based access control' (with a 'Skip tour' and 'Take me there' button), 'Modernizing financial services', 'Machine learning vs. AI', and a 'Community' section.

Vamos a añadir **Elastic Defend**.

Después deberemos copiar los comandos que nos dan según el sistema operativo en el que lo vayamos a instalar, en nuestro caso lo haremos en windows y linux.

The screenshot shows the 'Integrations' section of the Elastic Cloud interface, specifically the 'Elastic Defend' integration. It displays basic information like version (8.12.0), agent policies (1), and a 'Add Elastic Defend' button. Below this, there are tabs for 'Overview', 'Integration policies', 'Assets', 'Settings', 'Configs', and 'Advanced'. Under 'Elastic Defend Integration', there are sections for 'Requirements' and 'Permissions'.

Aquí vamos a **detect threats in my data with siem**.

The screenshot shows a landing page for 'What would you like to do first?'. It includes a search bar and filters for 'Search', 'Observability', and 'Security'. Three main options are listed: 'Detect threats in my data with SIEM' (status: 1 of 3 steps complete), 'Secure my hosts with endpoint security', and 'Secure my cloud assets with cloud security posture management (CSPM)'.



Ahora le damos a **add integration**, aquí ya lo que estamos haciendo es recopilar todos los eventos tanto de linux como de windows para recolectarlos en el siem.

Le estamos diciendo que coja todos los logs importantes o necesarios para el tema de seguridad.

Aquí podemos ver los eventos de kali:

Timestamp	event.dataset	Message
09:25:03.487	elastic_agent	[elastic_agent][info] Elastic Agent started
09:25:03.786	elastic_agent	[elastic_agent][info] Starting upgrade watcher
09:25:03.789	elastic_agent	[elastic_agent][info] Upgrade Watcher invoked
09:25:03.790	elastic_agent	[elastic_agent][info] releasing watcher 695
09:25:03.790	elastic_agent	[elastic_agent][info] APM instrumentation disabled
09:25:03.792	elastic_agent	[elastic_agent][info] Gathered system information
09:25:04.086	elastic_agent	[elastic_agent][info] Upgrade Watcher started
09:25:04.088	elastic_agent	[elastic_agent][info] update marker not present at '/opt/Elastic/Agent/data'

Aquí podemos ver los logs de windows:

Timestamp	event.dataset	Message
04:28:33.047	elastic_agent	[elastic_agent][warn] Possible transient error during checkin with fleet-server, retrying
04:28:33.072	elastic_agent	[elastic_agent][warn] Component state changed endpoint-default (HEALTHY->DEGRADED): Degraded : endpoint service missed 1 check-in
04:29:03.014	elastic_agent	[elastic_agent][info] Component state changed endpoint-default (DEGRADED->HEALTHY): Healthy: communicating with endpoint service
04:30:11.574	elastic_agent	[elastic_agent][info] signal "terminated" received
04:30:11.575	elastic_agent	[elastic_agent][info] Shutting down Elastic Agent and sending last events...
04:30:11.576	elastic_agent	[elastic_agent][warn] Possible transient error during checkin with fleet-server, retrying
04:30:11.578	elastic_agent	[elastic_agent][error] checkin retry loop was stopped
04:30:11.578	elastic_agent	[elastic_agent][error] failed accept conn info connection: accept tcp 127.0.0.1:6788: use of closed network connection
04:30:11.579	elastic_agent	[elastic_agent][info] stopping endpoint service runtime
04:30:11.801	elastic_agent	[elastic_agent][info] Shutting down completed.
04:30:11.804	elastic_agent	[elastic_agent][info] Stopping server
04:30:11.804	elastic_agent	[elastic_agent][info] Stats endpoint (127.0.0.1:6791) finished: accept tcp 127.0.0.1:6791: use of closed network connection

Aquí en el **fleet** vemos las políticas que tenemos:

Status	Host	Agent policy	CPU	Memory	Last activity	Version	Actions
Healthy	kali	My first agent policy rev. 2	0.20 %	27 MB	16 seconds ago	8.12.2	...
Healthy	desktop-gf86luv	My first agent policy rev. 2	0.03 %	27 MB	19 seconds ago	8.12.2	...
Healthy	35fff895cd0d	Elastic Cloud agent policy rev. 5	N/A	N/A	21 seconds ago	8.12.2	...



Esto son las partes de las políticas (conjunto de normas que le vamos a poner en nuestros sistemas)

Tanto la kali como el windows lo tenemos en **my first agent policy** (esta política solamente tienen la integración con el Elastic Defend).

The screenshot shows the Elastic Fleet interface with the 'Agent policies' tab selected. The page title is 'My first agent policy'. Key details shown are: Revision 2, Integrations 1, Agents 2 agents, Last updated on Mar 08, 2024. Below this, there's a table with columns: Name, Integration, Namespace, and Actions. One entry is visible: 'endpoint-1' with 'Elastic Defend v8.12.0' and 'default' namespace.

Queremos dividirlo en diferentes políticas para que los windows y los linux nos lo recoja con diferentes pluggins.

Para ello vamos a **agent policies** y creamos una nueva politica

The screenshot shows the 'Agent policies' section of the Fleet interface. It lists two policies: 'My first agent policy' (rev. 2) and 'Elastic Cloud agent policy' (rev. 5). Both policies have 1 integration and 2 agents. The 'Elastic Cloud agent policy' is described as 'Default agent policy for agents hosted on Elastic Cloud'. A 'Create agent policy' button is visible at the top right.

Aquí vamos a ponerle de nombre linux, y podemos dar a **collect system logs and metrics** para que nos recoja los logs de sistema y métricas de funcionamiento y creamos la política.

Data streams. Learn more

Agent monitoring

Collecting monitoring logs and metrics will also create an **Elastic Agent** integration. Monitoring data will be written to the default namespace specified above.

- Collect agent logs
- Collect agent metrics



Aquí tendríamos la política linux que hemos creado y la política **my first agent policy**.

Fleet

Centralized management for Elastic Agents.

Agents **Agent policies** Enrollment tokens Uninstall tokens Data streams Settings

Name	Description	Last update...	Agents	Integrations	Actions
Linux rev. 1		Mar 16, 2024	0	1	...
My first agent policy rev. 2		Mar 08, 2024	2	1	...
Elastic Cloud agent policy rev. 5	Default agent policy for agents hosted on Elastic Cloud	Mar 08, 2024	1	2	...

Vamos a añadirle una integracion a la politica de **my first agent policy**, le damos a **add integration**.

< View all agent policies

My first agent policy

Revision 2 | Integrations 1 | Agents 2 agents | Last updated on Mar 08, 2024 | Actions

Integrations Settings

Search... Namespace Add integration

Name ↑	Integration	Namespace	Actions
endpoint-1	Elastic Defend v8.12.0	default	...

Nosotros hemos añadido antes la de Elastic Defend, que es para proteger nuestros host y la que recoge los eventos de seguridad.

Vamos a añadirle la de **Suricata** para que recoja esos logs. le damos a **add suricata**.

elastic Find apps, content, and more. Live Chat Setup guide: step 1 Connection details

≡ D Integrations Suricata

Back to integrations

Suricata Version 2.21.0 Add Suricata

Elastic Agent Overview Settings Configs API reference

Suricata Integration Screenshots 1 of 2

Vamos a ponerla en la política de linux.

2

Where to add this integration?

New hosts Existing hosts

Agent policy

Agent policies are used to manage a group of integrations across a set of agents.

Agent policy

Linux

0 agents are enrolled with the selected agent policy.



Aquí podemos añadir un agente, pero en principio no vamos a añadir ninguno.

The dialog box is titled "Suricata integration added". It contains the message: "To complete this integration, add **Elastic Agent** to your hosts to collect data and send it to Elastic Stack." Below the message are two buttons: "Add Elastic Agent later" and "Add Elastic Agent to your hosts". At the bottom of the dialog are tabs for "New hosts" and "Existing hosts".

Aquí en el kali, vamos a cambiarle a la política de linux:

The dialog box is titled "Assign new agent policy". It shows the "Agent policy" dropdown set to "Linux". Below it, it says "The selected agent policy will collect data for 2 integrations: System, Suricata". At the bottom are "Cancel" and "Assign policy" buttons. To the right of the dialog is a sidebar with actions: "Assign to new policy", "Unenroll agent", "Upgrade agent", "View agent JSON", and "Request diagnostics.zip".

Aquí nos recoge los logs de sistema y los de suricata solamente para el equipo de la kali.

The page shows the "Agent details" tab for the "kali" host. The "Overview" section includes metrics like CPU (0.13 %), Memory (27 MB), Status (Healthy), Last activity (11 seconds ago), Last checkin message (Running), Agent ID (74b2a9a1-a08f-4f67-8269-ff5195346abe), Agent policy (Linux rev. 2), and Agent version (8.12.2). The "Integrations" section shows two entries: "system-1" and "suricata-1", each with an "Inputs" option.



El windows nos está recogiendo logs del end-point.

Agent details Logs Diagnostics

Overview

Integrations

Actions ▾

Una vez ya tenemos creadas las políticas, en la parte de **Discover** nos aparecen los logs que estamos recogiendo.

Discover

logs-* Filter your data using KQL syntax

1,021,507 hits

Break down by Select field

Documents Field statistics

Take the tour Demiss

Rows per page: 100 > 1 2 3 4 5 >

Aquí podemos crear los conjuntos de datos que queremos ver:

Discover

logs-* Filter your data using KQL syntax

Add a field to this data view

Manage this data view

Create a data view

Find a data view

.alerts-security.alerts-default,apm-*...*,winlogbeat-*,-elastic-cloud-logs-*

.kibana-event-log-*

logs-*

metrics-*

Try ES|QL Technical preview



Para ver los logs de Suricata, buscamos **logs-suricata***, ponemos los ultimos 30 dias y ya podemos ver los eventos de Suricata que están recopilados en el sistema.

Create data view

Name

Index pattern

logs-su*

Timestamp field

@timestamp

Select a timestamp field for use with the global time filter.

Show advanced settings

Your index pattern matches 1 source.

All sources Matching sources

Matching sources

logs-suricata.eve-default

Data stream

Rows per page: 10

Elasticsearch dashboard

Discover

Find apps, content, and more.

Live Chat Setup guide step 1

New Open Share Alerts Inspect

Last 30 days Refresh

Break down by Select field

Available fields

87,975 hits

Documents Field statistics

Get the best look at your search results

Add relevant fields, render and sort columns, resize rows, and more in the document table.

Take the tour

Document

event.start Mar 11, 2024 0 19:32:18.582 @timestamp Mar 11, 2024 0 19:32:18.582 agent.ephemeral_id a5c07971-5a73-4033-9e44-5479075721a agent.id 742c0a1-af0f-4f87-8269-f519534de agent.name kali1 agent.type filebeat agent.version 8.12.2 data_stream.dataset suricata.eve data_stream.namespace default data_stream.type logs destination.address 198.98.1.1 destination.bytes 169 destination.ip 198.98.1.1 destination.packets 53 eca.version 8.11.0 elastic.agent.id 742c0a1-af0f-4f87-8269-f519534de elastic.agent.snapshot false elastic.agent.version 8.12.2 event.agent_id.event_status verified.event.category network.event.created Mar 10, 2024 0 20:23:03.866 event.dataset suricata.eve event.duration 12,000,000 event.end Mar 11, 2024 0 19:31:48.781 event.ingested Mar 10, 2024 0 20:23:13.000 event.kind event.event.module suricata.event.start Mar 11, 2024 0 19:31:48.769 event.type connection input_type log log_file.path /var/log/suricata/eve.json log.offset 78,444,167 network.bytes 406 network.community_id 1-7c-57WWNFKYjyGUSEy45B= network.packets 2 network.protocol dns network.transport udp...

event.start Mar 11, 2024 0 19:32:18.582 @timestamp Mar 11, 2024 0 19:32:18.582 agent.ephemeral_id a5c07971-5a73-4033-9e44-5479075721a agent.id 742c0a1-af0f-4f87-8269-f519534de agent.name kali1 agent.type filebeat agent.version 8.12.2 data_stream.dataset suricata.eve data_stream.namespace default data_stream.type logs destination.address 198.98.1.1 destination.bytes 447 destination.ip 198.98.1.1 destination.port 53 eca.version 8.11.0 elastic.agent.id 742c0a1-af0f-4f87-8269-f519534de elastic.agent.snapshot false elastic.agent.version 8.12.2 2 event.agent_id.event_status verified.event.category network.event.created Mar 10, 2024 0 20:23:03.866 event.dataset suricata.eve event.duration 12,000,000 event.end Mar 11, 2024 0 19:31:48.781 event.ingested Mar 10, 2024 0 20:23:13.000 event.kind event.event.module suricata.event.start Mar 11, 2024 0 19:31:48.769 event.type connection input_type log log_file.path /var/log/suricata/eve.json log.offset 78,441,798 network.bytes 232 network.community_id 1-7c-57WWNFKYjyGUSEy45B= network.packets 2 network.protocol dns network.transport udp...

event.start Mar 11, 2024 0 19:32:18.582 @timestamp Mar 11, 2024 0 19:32:18.582 agent.ephemeral_id a5c07971-5a73-4033-9e44-5479075721a agent.id 742c0a1-af0f-4f87-8269-f519534de agent.name kali1 agent.type filebeat agent.version 8.12.2 data_stream.dataset suricata.eve data_stream.namespace default data_stream.type logs destination.address 198.98.1.1 destination.bytes 447 destination.ip 198.98.1.1 destination.port 53 eca.version 8.11.0 elastic.agent.id 742c0a1-af0f-4f87-8269-f519534de elastic.agent.snapshot false elastic.agent.version 8.12.2 destination.geo.country_name Spain destination.geo.location POINT (-3.684 40.472) destination.ip 198.98.1.1 destination.packets 53 eca.version 8.11.0 elastic.agent.id 742c0a1-af0f-4f87-8269-f519534de elastic.agent.snapshot false elastic.agent.version 8.12.2 event.agent_id.event_status verified.event.category network.event.created Mar 10, 2024 0 20:23:03.866 event.dataset suricata.eve event.duration 51,000,000 event.end Mar 11, 2024 0 19:31:48.781 event.ingested Mar 10, 2024 0 20:23:13.000 event.kind event.event.module suricata.event.start Mar 11, 2024 0 19:32:10.562 agent.ephemeral_id a5c07971-5a73-4033-9e44-5479075721a agent.id 742c0a1-af0f-4f87-8269-f519534de agent.name kali1 agent.type filebeat agent.version 8.12.2 data_stream.dataset suricata.eve data_stream.namespace default data_stream.type logs destination.address 198.98.1.1 destination.bytes 319 destination.ip 198.98.1.1 destination.packets 1 destination.port 53 eca.version 8.11.0 elastic.agent.id 742c0a1-af0f-4f87-8269-f519534de elastic.agent.snapshot false elastic.agent.version 8.12.2 2 event.agent_id.event_status verified.event.category network.event.created Mar 10, 2024 0 20:23:03.866 event.dataset suricata.eve event.duration 7,000,000 event.end Mar 11, 2024 0 19:30:41.865 event.ingested Mar 10, 2024 0 20:23:13.000 event.kind event.event.module suricata.event.start Mar 11, 2024 0 19:30:41.838 event.type connection input_type log log_file.path /var/log/suricata/eve.json log.offset 78,442,798 network.bytes 445 network.community_id 1-f1cc818-62085a79fc3XAHk= network.packets 2 network.protocol dns network.transport udp...



INTEGRACIÓN DE LOS LOGS DEL HONEYBOT EN ELASTIC

Vamos a incluir los logs que hemos recopilado antes en el archivo cowrie.log a Elastic con la ayuda de la integración **Custom Logs**:

The screenshot shows the 'Custom Logs' package details in the Elastic Stack interface. The package is version 2.3.1 and is categorized under 'Custom, Custom Logs'. It includes sections for Overview, Settings, Config, API reference, and Details. The Details section shows the package version 2.3.1, category 'Custom, Custom Logs', and a note about ECS Field Mapping.

The screenshot shows the 'Add Custom Logs integration' configuration screen. Under 'Configure integration', the 'Integration settings' section shows an integration name 'log-honeybot-cowrie' and a description 'logs del honeypot'. The 'Custom log file' section specifies a log file path '/home/kali/cowrie.log'. Under 'Dataset name', the dataset is named 'cowrie'. A note states that changing the dataset name will send data to a different index.

2 Where to add this integration?

The screenshot shows the 'Where to add this integration?' configuration screen. It allows selecting 'New hosts' or 'Existing hosts'. Under 'Agent policy', it shows a 'Linux' policy selected, with a note indicating 1 agent is enrolled. The 'Existing hosts' tab is currently selected.



Aquí ya podemos ver como nos llegan los logs de nuestro Honeypot al Elastic:

Discover

logs-cowrie-default

12 hits

Available fields: @timestamp, agent.ephemeral_id, agent.id, agent.name, agent.type, agent.version, cloud.account_id, cloud.availability_zone, cloud.image_id, cloud.instance_id, cloud.instance.name, cloud.machine_type, cloud.provider, cloud.region, container.id, container.image.name, container.name, data_stream.dataset, data_stream.namespace

Documents Field statistics

Get the best look at your search results

Take the tour Dismiss

Mar 23, 2024 @ 16:58:36.993 #timestamp Mar 23, 2024 @ 16:58:36.993 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 16:58:36.993 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 16:58:36.993 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 16:58:36.993 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 16:58:36.993 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 16:58:36.993 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 16:58:36.993 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 16:58:36.993 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 16:58:36.993 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 16:58:36.993 host.architecture x86_64...

Discover

logs-cowrie-default

32 hits

Available fields: @timestamp, agent.ephemeral_id, agent.id, agent.name, agent.type, agent.version, cloud.account_id, cloud.availability_zone, cloud.image_id, cloud.instance_id, cloud.instance.name, cloud.machine_type, cloud.provider, cloud.region, container.id, container.image.name, container.name, data_stream.dataset, data_stream.namespace

Documents Field statistics

Get the best look at your search results

Take the tour Dismiss

Mar 23, 2024 @ 17:55:46.737 - Mar 23, 2024 @ 17:59:07.454

11592
1
"message": [
"2024-03-23T16:59:10+0000
[honeypotSSHtransport,3.192.168.1.
\$4] CND: cd .."
],
"data_stream.type": [
"logs"
]
Copy value

#timestamp Mar 23, 2024 @ 17:59:10.455 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 17:59:10.455 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 17:59:07.454 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 17:59:07.454 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 17:59:07.454 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 17:59:07.454 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 17:59:07.454 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 17:59:07.454 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 17:59:07.454 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 17:59:07.454 host.architecture x86_64...
#timestamp Mar 23, 2024 @ 17:59:07.454 agent.ephemeral_id S3886449-9cdd-498d-9cde-3d8d22b88999 agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe agent.name kali agent.type filebeat agent.version 8.12.2 data_stream.dataset cowrie data_stream.namespace default data_stream.type logs ecs.version 8.0.0 elastic_agent.id 7402a9e1-a08f-4f67-8269-ff5193346abe elastic_agent.snapshot false elastic_agent.version 8.12.2 event.agent.id status verified.event.dataset cowrie.event.ingested Mar 23, 2024 @ 17:59:07.454 host.architecture x86_64...