**MHN**

**2020-2021**

1. **¿Qué es MHN?**

Es un sistema creado por la organización MHN Anomali. A través de él podemos gestionar una red de honeypots que contará con un servidor y con varios honeypots conectados a él. Cada uno de esos honeypots recogerá información sobre un campo concreto. Con toda la información recibida, el servidor gestionará la información de los posibles atacantes.

Podemos controlarlo desde el navegador, y tiene una interfaz muy visible y útil.

1. **Configuración**
   1. **Estructura de la red**

Tenemos una red local: 192.168.0.0/24. En ella tenemos varias máquinas con el sistema operativo Ubuntu Server 18.04:

* + - Servidor *mhn* en la dirección 192.168.0.100.
    - Un honeypot *cowrie* en la dirección 192.168.0.101.
    - Un honeypot *dionaea* en la dirección 192.168.0.102.

Para conectar esta red con el mundo disponemos de una máquina GNU/Linux. Esta máquina dispone de dos tarjetas de red:

* + - *eth0*: tiene la IP 10.88.99.33: nuestra IP pública.
    - *eth2*: tiene la IP 192.168.0.1/24 y es el gateway de la red local.
  1. **Puertos a abrir en el router**

Estos son los puertos que tendremos que abrir y redirigir en nuestra máquina *router*.

| **Puerto** | **IP de destino** | **Puerto de destino** |
| --- | --- | --- |
| 81 | 192.168.0.100 (*mhn*) | 80 |
| 22 | 192.168.0.101 (*cowrie*) | 22 |
| 445 | 192.168.0.102 (*dionaea*) | 445 |

Además de redigir el tráfico, tendrás que permitirlo expresamente en el script */etc/iptables.sh*. Si quieres, olvida lo hecho hasta ahora, y ejecuta [este sencillo script](https://docs.google.com/document/d/1WZH6D1T9wL8mEpRY5aAhm8FrYjn6H95VO8zCiRSKUQU/edit) (hace las redirecciones de la tabla de arriba).

* 1. **Instalar el servidor**

En la máquina *mhn* seguiremos los siguientes pasos:

* + - Instalar Git:

apt update

apt install git –y

* + - Instalar MHN:

cd /opt/

git clone https://github.com/pwnlandia/mhn.git

cd mhn/

./install.sh

Pasado un tiempo (10-15 minutos), nos aparecerán estas preguntas y campos (recuerda que X.X.X.X es nuestra IP pública):

Do you wish to run in Debug mode?: y/n n

Superuser email: zure\_emaila@domeinua.com

Superuser password:

Server base url ["http://X.X.X.X"]: 10.88.99.33

Honeymap url ["http://10.88.99.33:3000"]:

Mail server address ["localhost"]:

Mail server port [25]:

Use TLS for email?: y/n n

Use SSL for email?: y/n n

Mail server username [""]:

Mail server password [""]:

Mail default sender [""]:

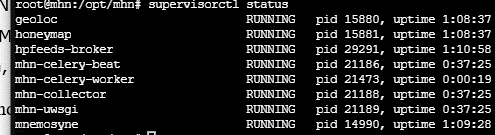
Path for log file ["mhn.log"]:

Por último, nos preguntarán si queremos integrar SPLUNK y ELK. Responderemos que no. Además, nos preguntarán si queremos activar las reglas correspondientes al MHN en la UFW. Responderemos a ello afirmativamente.

Para ver si todo ha ido bien, ejecutaremos el siguiente comando:

supervisorctl status

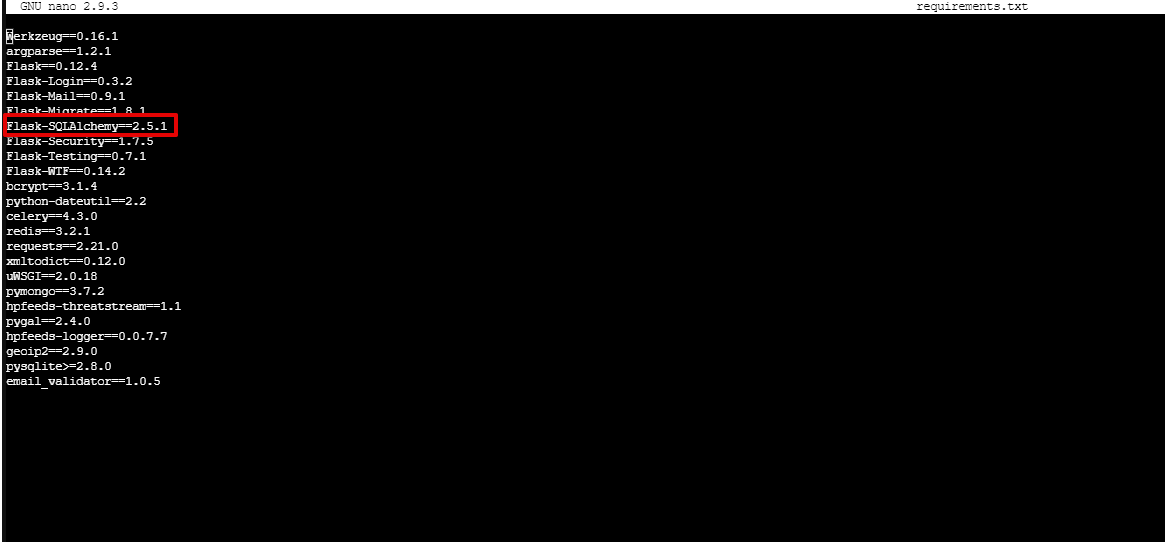
Nos tienen que aparecer varios servicios en estado RUNNING:



Comprobamos la versión de la extensión Flask-SQLAcademy.

nano /opt/mhn/server/requirements.txt

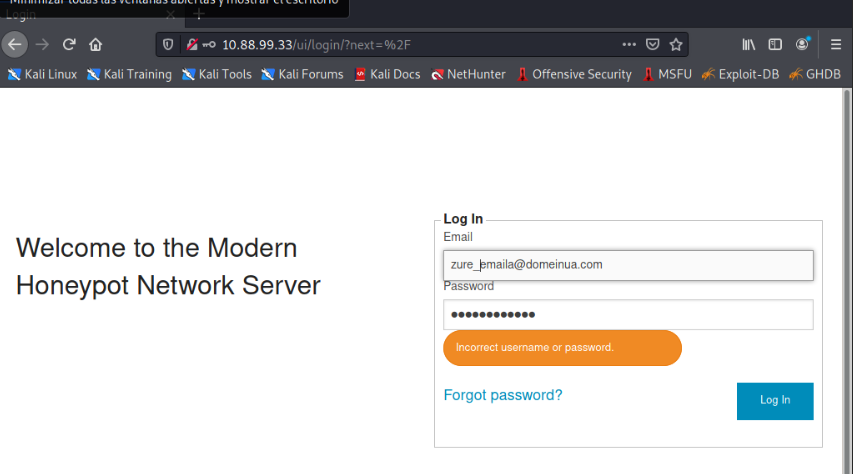
En ese fichero, el parámetro Flask-SQLAlchemy debe estar establecido en el valor 2.5.1 (si no lo tiene, cámbialo).



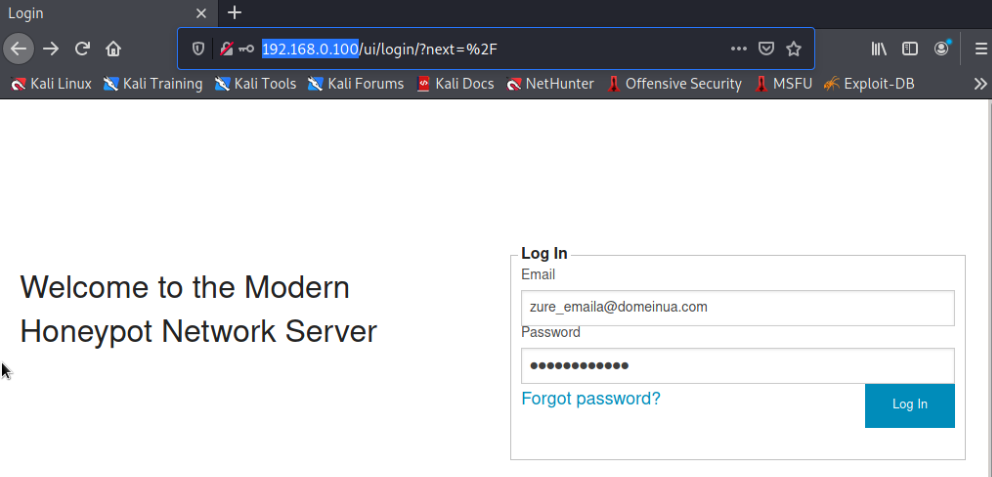
Para cambiar algún parámetro de la configuración del servidor (dirección de correo, contraseña…), tendremos que reescribir el fichero */opt/mhn/server/config.py*, y, después, ejecutar este comando:

service supervisor restart

Ya tendremos accesible el servidor MHN a través de la web, escribiendo http://10.88.99.33:81 en el navegador de una máquina externa; para realizar la prueba utilizaremos la máquina *kali* (recuerda que debe estar ubicada en nuestro Internet simulado). Para autenticarnos, deberemos utilizar las credenciales indicadas en la instalación.

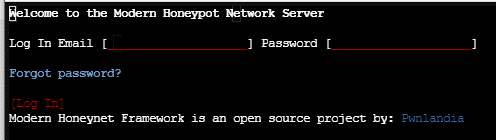


Por supuesto, el servidor también estará accesible desde el navegador de cualquier otro dispositivo de la misma red de la máquina *mhn*; si colocamos la máquina kali en esa red, podremos acceder utilizando la dirección 192.168.0.100:



Si no tuviéramos ninguna máquina con entorno gráfico en la red, podríamos hacer la prueba utilizando la aplicación *w3m*, por ejemplo desde la máquina router:

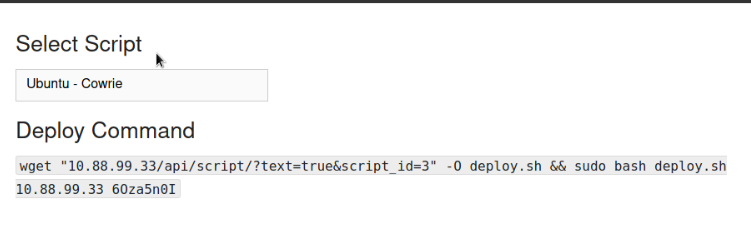




* 1. **Configurar los honeypots**
     1. Cowrie

Cowrie es un honeypot SSH. Para autenticarse en su servicio SSH, hay que utilizar el usuario *root* y cualquier contraseña. Una vez autenticado, el atacante ejecutará los comandos y recibirá respuestas, pero éstas serán ficticias, ya que lo visto en pantalla será un falso GNU/Linux. Los intentos de autenticación con cualquier otro usuario quedarán registrados, por lo que este honeypot puede servir para analizar los ataques de fuerza bruta o por diccionario que se utilizan habitualmente mediante SSH, ya que registra todos los nombres y contraseñas de los usuarios probados.

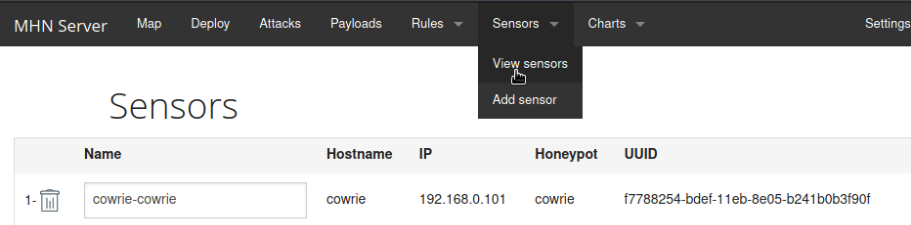
Para instalar Cowrie, seleccionamos "*Ubuntu Cowrie*" dentro de "*Select Script*" del apartado "Deploy" del MHN y copiamos el comando del mismo.



A continuación, vamos a pegar el texto copiado en la máquina *cowrie*, pero, atención: sustituiremos la IP pública por la IP privada de la máquina *mhn*. ¡Atención! El comando de debajo será variable. No lo copies tal cual en tu máquina:

wget "[**192.168.0.100**/api/script/?text=true&script\_id=3](http://10.88.99.33/api/script/?text=true&script_id=3)" -O deploy.sh && sudo bash deploy.sh **192.168.0.100** 6Oza5n0I

Pasado un tiempo -y si todo ha ido bien-, tendremos el honeypot preparado, y aparecerá una nueva entrada en el apartado “*Sensors -> View Sensors*”:



Como se ha comentado, en la máquina *cowrie* estará abierto el puerto 22, quedando registrados todos los intentos de acceso ssh a la misma. No será posible autenticarse sin utilizar el usuario *root*, ni siquiera con las credenciales de la máquina *cowrie*. Entrando con el usuario *root*, la consola ssh no ejecutará ningún comando, por lo que no se podrá dañar la máquina. Si nosotros mismos queremos simular ataques, estableceremos una conexión ssh con la IP pública desde una máquina externa a nuestra red (*kanpokoa*, por ejemplo). Realizaremos varios intentos con diferentes nombres de usuario:





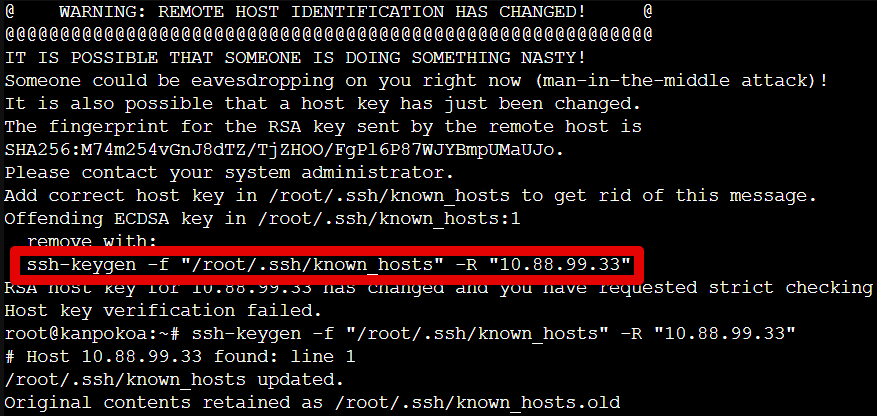




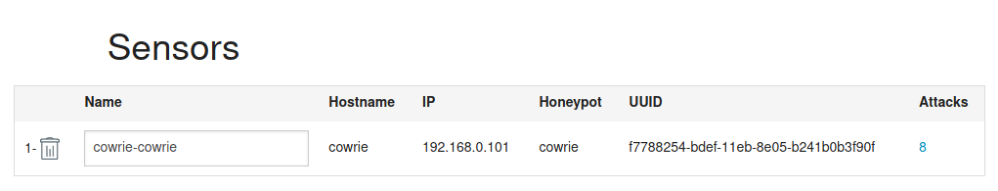


¡¡Atención!! Si al intentar conectarte te aparece el mensaje de debajo, ejecuta el comando que aparece ahí:

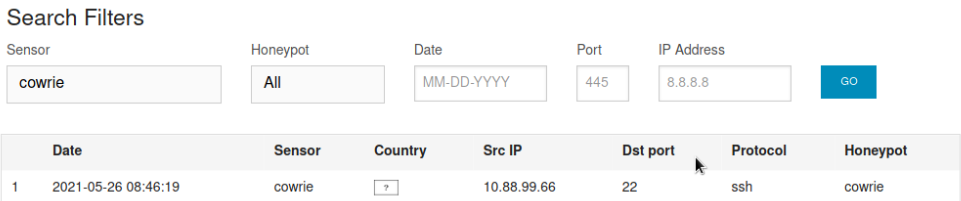
sh-keygen -f "/root/.ssh/known\_hosts" -R "10.88.99.33"



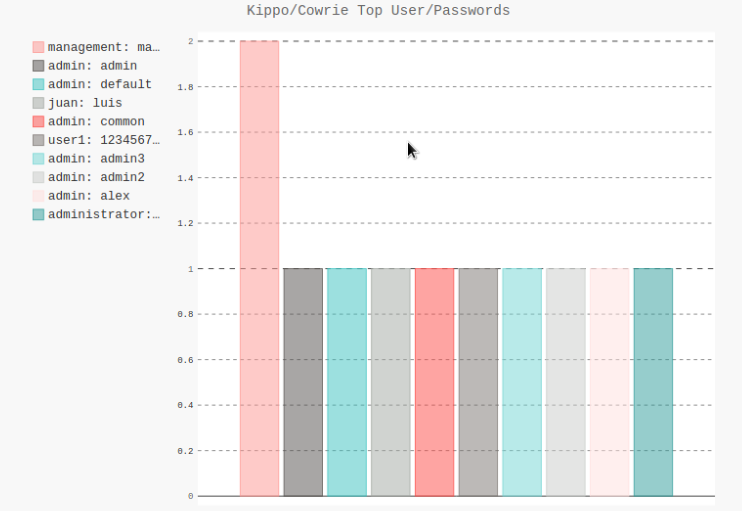
Para consultar la hora, país… de los ataques en el servidor MHN, haremos clic en la entrada *cowrie* del apartado “*Sensors -> View Sensors*”.

**

Si hacemos clic en el número que está debajo del encabezado *Attacks*, veremos todos los ataques recibidos; entre otros, el que acabamos de realizar desde la máquina *kanpokoa* (“Src IP”: 10.88.99.66).



Asimismo, será posible ver otra información interesante sobre esos ataques mediante gráficos. Por ejemplo, si seleccionamos “*Charts→ Kippo/Cowrie Top users/passwords*”, veremos los usuarios y contraseñas de los intentos realizados.

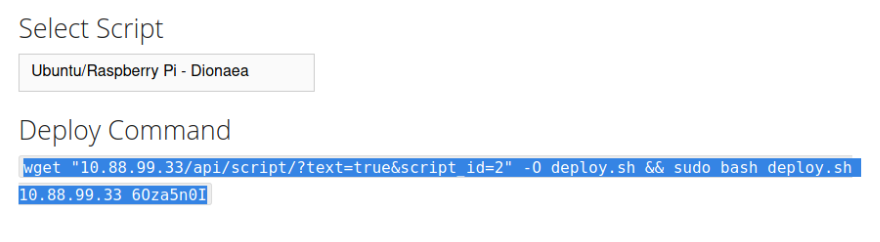


Así, con el tiempo, podremos tener una base de datos con los usuarios y contraseñas más utilizados en los ataques de fuerza bruta.

* + 1. Dionaea

El honeypot *Dionaea* deja abiertos varios servicios, y está preparado para recibir ataques de malware.

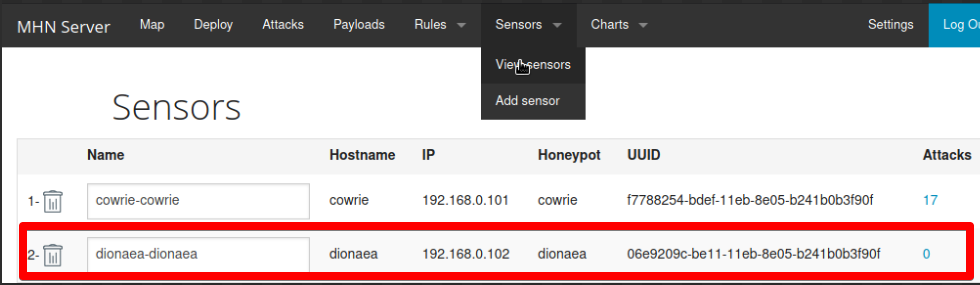
Para instalar el honeypot, en el apartado “*Deploy*” de *MHN*, seleccionaremos “*Ubuntu/Raspberry Pi - Dionaea*” de “Select Script”, y copiaremos el comando que aparece ahí.



﻿A continuación, pegamos en la máquina *dionaea* el texto copiado, pero, atención: sustituimos la IP pública por la IP privada de la máquina *mhn*. ¡Atención! El comando de debajo será variable. No lo copies tal cual en tu máquina:

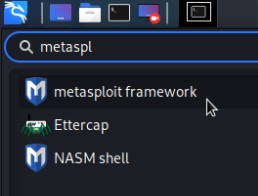
wget "**192.168.0.100**/api/script/?text=true&script\_id=2" -O deploy.sh && sudo bash deploy.sh **192.168.0.100** 6Oza5n0I

Pasado un tiempo, y si todo ha ido bien, tendremos preparado el honeypot, y nos aparecerá una nueva entrada en el apartado “*Sensors -> View Sensors*”:



﻿En la máquina *dionaea* estará abierto el puerto 445 (en nuestra máquina *router* está redirigido el tráfico hacia el puerto 445) y quedarán registrados todos los intentos de ataques smb recibidos. Si queremos simular un ataque, colocaremos la máquina *kali* fuera de nuestra red (IP 10.88.99.222 en *vmbr3*) y seguiremos los siguientes pasos:

* Ejecutar “*Metasploit Framework*”.



* Configurar un exploit.

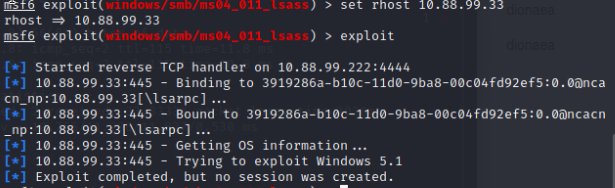
Escribir este texto

use exploit/windows/smb/ms04\_011\_lsass

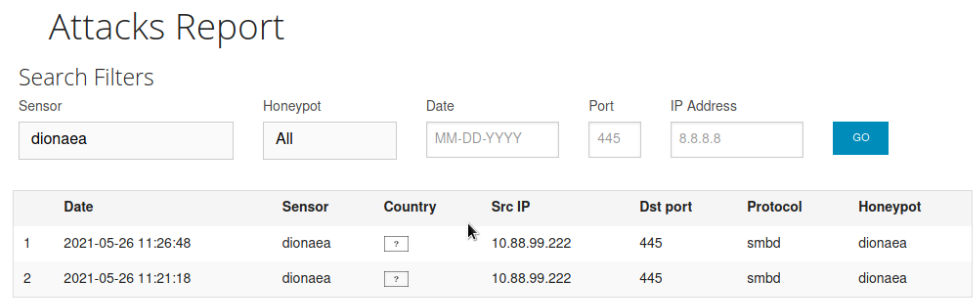


* Indicar la IP de la víctima y poner en marcha el ataque.

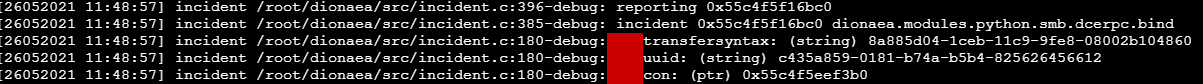
set rhosts 10.88.99.33  
exploit



﻿Los ataques recibidos en nuestra IP pública quedarán registrados. Para consultar la hora, el país... de los ataques en el servidor, haremos clic en la entrada *dionaea* del apartado "*Sensors -> view Sensors*".

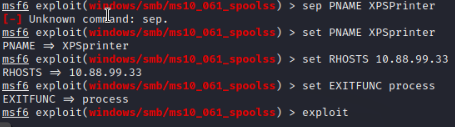


Si queremos tener más información acerca de los ataques, iremos a la máquina *dionaea*. En el final del fichero */opt/dionaea/var/log/dionaea/dionaea.log* (“Esc”+’/’ para ir al final), veremos registrado el ataque.

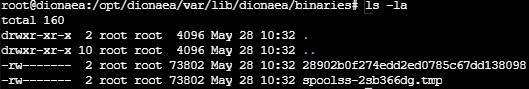


Sin embargo, encontrar esta información puede ser difícil, y no resulta fácil de utilizar. Así que haremos una prueba con un ataque más visual; usaremos un exploit que enviará un fichero a la víctima.





El fichero recibido en el honeypot *dionaea* está en el directorio */opt/dionaea/var/lib/dionaea/binaries*.



Veamos las características del fichero con mayor profundidad:



Como podemos ver, es un fichero ejecutable de Windows.

1. **Bibliografia**

[Ataques contra honeypot Dionaea](https://thehackerway.com/2015/04/14/honeypots-parte-3-configuracion-y-analisis-de-malware-con-dionaea/)

[Cómo instalar los honeypots dionaea y cowrie](https://neil-fox.github.io/Setting-up-Dionaea-&-Cowrie-with-MHN/)

[Esquema de la arquitectura de MHN. Vídeo](https://www.youtube.com/watch?v=Zd1Br8TW1mk)

[Cómo corregir errores comunes. Guía](https://github.com/pwnlandia/mhn/wiki/MHN-Troubleshooting-Guide)



