Escuela Politécnica Superior de Jaén

Grado en Ingeniería Informática

UNIVERSIDAD DE JAÉN
Escuela politécnica superior de Jaén

Trabajo Fin de Grado

DESARROLLO DE UNA HERRAMIENTA HONEYPOT PARA UN USO EFICIENTE EN SEGURIDAD INFORMÁTICA

Alumno: Alejandro Cruz Fernández De Moya

Tutor: José María Serrano Chica
Dpto: Departamento de informática

Septiembre, 2017

Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Jose Maria Serrano Chica , tutor del Trabajo Fin de Grado titulado: Desarrollo de una herramienta Honeypot para un uso eficiente en seguridad informática, que presenta Alejandro Cruz Fernández De Moya, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Septiembre de 2017

El alumno:

Alejandro

Alejandro Cruz Fernández De Moya

El tutor:

SERRANO CHICA JOSE MARIA - 26490278M

Firmado digitalmente por SERRANO CHICA JOSE MARIA - 26490278M
Nombre de reconocimiento (DN): c=ES, serialNumber=IDCES-26490278M, givenName=JOSE MARIA, sn=SERRANO CHICA, cn=SERRANO CHICA JOSE MARIA - 26490278M
Fecha: 2017.09.05 12:30:54 +02'00'

Jose Maria Serrano Chica

Índice

1. Introducción

1.1. Motivación

En el mundo que nos rodea cada vez se ha hecho más necesario el uso de la tecnología, ya sea tanto como medio de ocio, o como herramienta para desempeñar un trabajo o facilitarlo. Muy a menudo ese uso implica de alguna manera el intercambio de información, ya sea proporcionando datos personales, como descargando archivos.

La información se compone de una cantidad enorme de elementos dispares y adquiere una infinidad de formas, algunos ejemplos de esta información son: nuestro datos personales, datos de cuentas de diversos servicios(bancos, redes sociales,..), posición geográfica, mediciones de dispositivos, e incluso archivos tales como fotos, videos, juegos,…

Esta información es muy codiciada tanto por las empresas a las que cedemos nuestros datos, previa aceptación de contrato de términos y condiciones, como por terceros a los que no hemos dado tal consentimiento. Y es codiciada, ya que en base a ella, se pueden montar negocios y servicios, legales o no.

Es delicada porque forma parte de nosotros, es una forma de identificación con respecto al resto de usuarios y cuidarla debe ser una prioridad por razones de seguridad, ya que de ella se puede deducir donde vivimos, donde estamos y corremos el riesgo de que alguien actue en nuestro nombre sin consentimiento.

Este valioso recurso que es nuestra información se guarda por lo general en medios físicos, bajo ciertas condiciones que la hacen accesible solo ante un limitadísimo grupo de usuarios (por lo general solo el propietario). Sin embargo, debido al carácter deseable de esta, estos soporte físicos donde es guardada son continuamente atacados por aquellos que la quieren de manera poco legitima. Ya que de obtenerla podrían tener acceso a recursos deseables que de otra manera no obtendrían, tales como cuentas bancarias, nuestra ubicación,etc

Estos recursos físicos donde es guardada la información pueden estar bajo el cuidado de la propia empresa que ofrece un determinado servicio y/o del propio usuario. Ambas variantes dan ciertos problemas a tener en cuenta, si es el usuario el custodio exclusivo de su información corre el riesgo de perderla para siempre, además de que no suele ser consciente ni de la importancia de su información ni de su cuidado(uso de dispositivos seguros, contraseñas complejas, etc), por no hablar de que los dispositivos de usuario(moviles, portátiles,etc), tienen una capacidad y unos recursos limitados para hacer frente a todos los ataques. Por otro lado, las maquinas de una empresa sufren un mayor número de ataques al tener esta una mayor cantidad de información.

Para proteger esta información se necesita que tanto los sistemas que la guardan como las aplicaciones sean seguras y robustas, asi como de personal preparado. La seguridad es una lucha constante y ,aunque todo sistema o aplicación puede poseer un mayor o menor nivel de seguridad, esta necesidad obliga a las empresas a dirigir parte de sus recursos a mejorar este aspecto. Normalmente las empresas son capaces de corregir por si mismas una gran cantidad de fallos en sus productos, sin embargo a veces ciertos agujeros son pasados por alto.

De esta necesidad de tapar lo que a las propias empresas se les pasa por alto nace la utilidad de los **honeypots**, sistemas trampa, tanto falsos como reales, que hacen de señuelo con la finalidad de ser atacados a propósito y desvelar así vulnerabilidades pasadas por alto, para poder corregirlas antes de que alguien con intenciones más que dudosas lo explote por su cuenta tal vulnerabilidad y deje comprometidos nuestros datos y archivos.

1.2. Objetivo

El objetivo del proyecto es montar una herramienta honeypot funcional que además de captar datos de atacantes, nos permita realizar un estudio sobre los métodos utilizados y el malware insertado en la máquina señuelo. Además, es también parte del objetivo el análisis y proposición de soluciones a los agujeros de

seguridad encontrados por la aplicación. Para conseguir este objetivo este TFG propone satisfacer los siguientes puntos:

- Proporcionar una herramienta para la detección, y análisis de accesos no autorizados que puedan afectar a la seguridad de equipos conectados en red.
- Diseñar dicha aplicación dentro del modelo “honeypot” para que resulte un medio aislado de detección de intrusiones, sin exponer a los recursos hardware de la red ni comprometer al resto de servicios software.
- Favorecer el trabajo del administrador del sistema con una aplicación que resulte ágil y sencilla de instalar, administrar y consultar.
- Extraer conclusiones y representarlas mediante la herramienta mas adecuada, de forma que ayuden a mejorar la seguridad en cualquier red informática.
- Realizar una migración del sistema a un entorno hardware como pueda ser la tarjeta programable Raspberry Pi.

1.3. Metodología

Para satisfacer los objetivos anteriormente necesarios, este proyecto se servira de la siguiente metodología:

- Instalación de una maquina virtual para la gestión de pruebas.
- Estudio, selección e implantación de un sistema honeypot ssh de media interacción.
- Creación de una interfaz de comunicación Web.
- Implementación del sistema de seguridad.
- Experimentación y análisis de los resultados.
- Exportación de la aplicación a un sistema hardware sobre RaspberryPi.

1.4. Resumen del trabajo

A lo largo del trabajo hemos tocado varios puntos, con el primero empezamos definiendo la importancia de la seguridad, describiendo el contexto en el que se engloba este trabajo y definiendo los objetivos a alcanzar, asi como algunos de sus medios.

En segundo lugar, nos hemos puesto en situación definiendo algunos tipos de amenzas, asi como herramientas a utilizar de manera general para hacerles frente o paliar el daño provocado. También se ha hablado de os honeypots en profundidad, definiendo con más detalle, que son, los tipos que existen, y en que lugares de la red podrían colocarse, junto con los riesgos y resultados que conlleva cada tipo y cada lugar.

A continuación, pasamos a describir el sistema y las herramientas utilizadas, asi como a proporcionar un breve tutorial sobre como instalarlos y sobre cómo preparar el terreno para hacerlas funcionar, con la intención de que este proyecto pueda replicarse.

Se ha comentado, además, el proceso de desarrollo del proyecto, incluyendo que decisiones se tomaron desde que se inició hasta que dificultades surguieron y como fueron superadas. También se incluyeron imagenes y comentarios acerca del funcionamiento de las herramientas y de los resultados que se obtuvieron de su uso. Además, se han comentado que medidas podrían tomarse para solventar algunos fallos a raíz de los resultados obtenidos.

Por último, tras las conclusiones descritas en esta sección, también se ha añadido una bibliografia de referencia, así como unos apéndices enfocados a dar más detalles acerca del proceso de instalación del sistema y de las herramientas.

2. Seguridad en internet

En esta sección explicaremos varios tipos de ataques y amenazas, definiremos mas en profundidad lo que es un honeypot, asi como cuales son las variantes y que diferencias hay entre ellas, por último, explicaremos en que lugar del router es posible colocar el honeypot.

2.1. Tipos de ataques

Haremos una breve reseña sobre algunos tipos de ataques comunes, relacionados con la intrusión, ya sea mediante la suplantación del usuario, como de la explotación de alguna vulnerabilidad o características del software:

2.1.1. Spoofing

Consiste en la suplantación de la identidad falseando la información de origen del atacante haciéndola pasar por información que la victima considere de confianza. Por ejemplo, cuando se comunica con un host, la dirección de este host ocupa un lugar determinado en la cadena de datos, al igual que nuestra dirección. Pues el spoofing consiste en manipular esa información para hacer creer al host de destino que la dirección de la que vienen nuestros mensajes es la de alguien conocido o de confianza.

Este tipo de ataque involucra tres máquinas:

- La máquina atacada A, a la que queremos engañar.
- La máquina suplantada B, por la que nos queremos hacer pasar que tiene la relación de confianza con A.
- La máquina atacante C, que suplantará B para poder comunicarse con A.

Para que C tenga exito en su ataque necesita establecer una comunicación falseada con A, intentando a su vez que B u otro equipo no interfiera en el ataque.

Un ejemplo es el ataque man-in-the-middle:

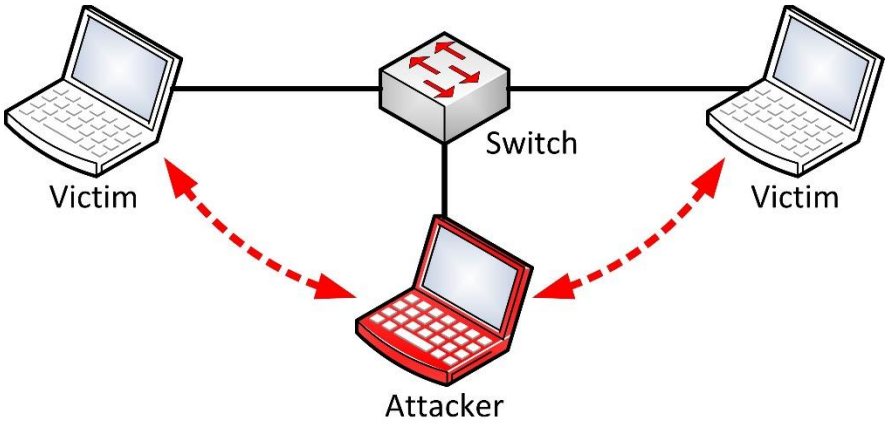


Ilustración 1.1

Los tipos de este ataque se clasifican en función de que han querido falsear:

- **IP Spoofing**: Consiste en sustituir la dirección IP de origen de un paquete TCP/IP por la dirección IP de la máquina que se desea suplantar. Puede usarse dentro de cualquier protocolo TCP/IP tales como ICMP,UDP o TCP, sin embargo, para TCP se han de tener en cuenta el comportamiento del protocolo con el envio de paquetes SYN y ACK con su ISN específico y teniendo en cuenta que el usuario suplantado puede cortar la conexión en cualquier momento al recibir paquetes no solicitados.
- **ARP Spoofing**(IP + MAC): Aqui lo que se falsea es la tabla ARP, que consiste en un protocolo que relaciona la dirección MAC con una determinada dirección IP. La tabla ARP es una tabla local guardada por switches y por host donde viene esta relación IP-MAC y acuden a ella para buscar o añadir direcciones válidas con la que establecer comunicaciones. Esta técnia solo puede utilizarse en redes LAN ya que trabaja a nivel de enlace de datos OSI y el nivel de red ,que esta una capa por encima, es el que identifica el enrutamiento con otras redes.
- **DNS Spoofing**: En este tipo de spoofing lo que se falsea es la relación “nombre de dominio - IP”. Esto consiste en falsear las entradas de esta relación gracias a alguna debilidad en el servidor de dominio o a la confianza en un dominio poco fiable.
- **Web Spoofing**: Consiste en enrutar la conexión de una víctima a través de una página falsa hacia otras páginas web, con el objetivo de obtener información de la víctima. La página web falsa actúa a modo de

intermediario solicitando información necesitada por la víctima para cada servidor original.

- **Mail Spoofing**: Suplantación del correo electrónico de una persona o entidad. Esta técnica es frecuentemente utilizada para el envio de correos basura, con la intención de realizar fishing.

Otros tipos de ataques son:

- Non-Blind Spoofing: El atacante esta en la misma red que la víctima. Al permancer en la misma red que la víctima la información, secuencias y números de reconocimiento, puede ser robada en lugar de tener que calcularla, lo que facilita el trabajo al atacante.
- Blind Spoofing: Aqui para intentar el robo de información se envian varios paquetes probando con varios números de secuencia. Aunque esta técnica ha dejado de ser efectiva desde que los sistemas operativos generan números de secuencia arbitrarios.
- Man in the middle: Este ataque consiste en interceptar la comunicación entre dos máquinas para controlar el flujo de esta. Al hacerlo se obtiene toda la información de ambas máquinas, permitiendo al atacante hacerse pasar por ambas.

Una buena forma de evitar este tipo de ataques es implementando filtros de entrada y salida en el router, mediante una lista de control de acceso(ACL).

En el 21 de julio de 2017 la universidad británica de New Castle advirtieron la aparición de un sitio web falso relacionado con la universidad. A través de esta página falsa se les pedia a los alumnos que ingresaran datos personales, asi como datos de sus tarjetas de crédito. Acceso a la noticia del diario BAE

Página noticia 1

2.1.2. Ataques de fuerza bruta

Método para sacar contraseña probando todas las combinaciones posibles hasta dar con la correcta. Es el método más simple e ineficiente. Peligroso si la contraseña es corta y simple.

Ilustración 2

Si queremos evitar que nos afecte se han de diseñar contraseñas largas y complejas, mezclando mayúsculas y minúsculas, números y signos de numeración, para hacer inviable el uso de esta técnica.

Ilustración 3

2.1.3. Ataques de diccionario

Método de cracking que consiste en intentar averiguar una contraseña a base de probar todas las palabras de un diccionario. Este algoritmo funciona de manera similar a un algoritmo de fuerza bruta.

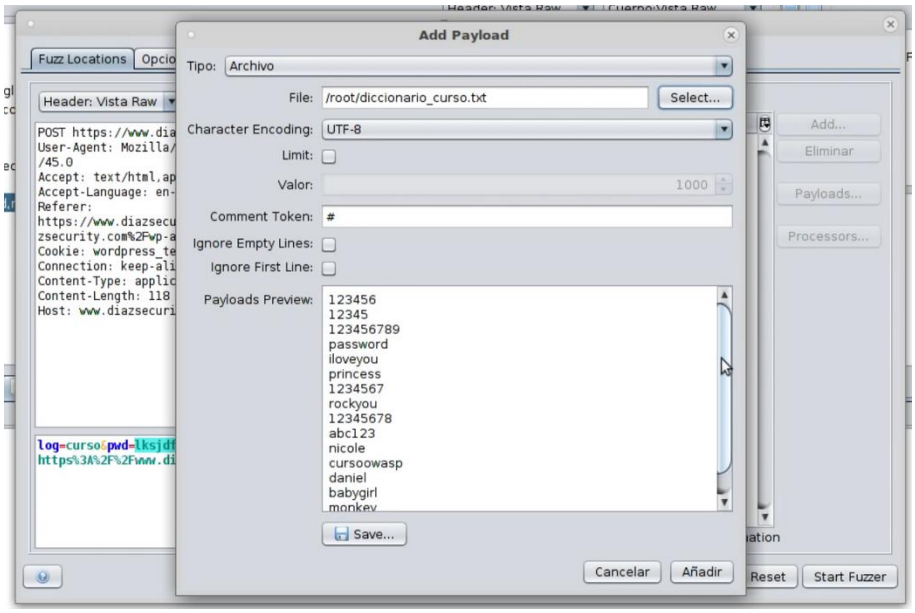


Ilustración 4

Es menos simple y más eficiente que un algoritmo de fuerza bruta y necesita de un diccionario de palabras en el idioma del atacado.

Al igual que en los ataques de fuerza bruta no hay una herramienta especifica, solo diseñar una contraseña que haga su uso inviable, mediante contraseñas largas, alternando mayúsculas y minúsculas, usando números, etc.

2.1.4. Exploits

Programa malicioso con el que se pretende aprovechar la vulnerabilidad de un sistema informático. No es considerado malware, aunque su propósito sea el de poder insertar malware en la máquina de la víctima.

Ilustración 5

Algunos ejemplos:

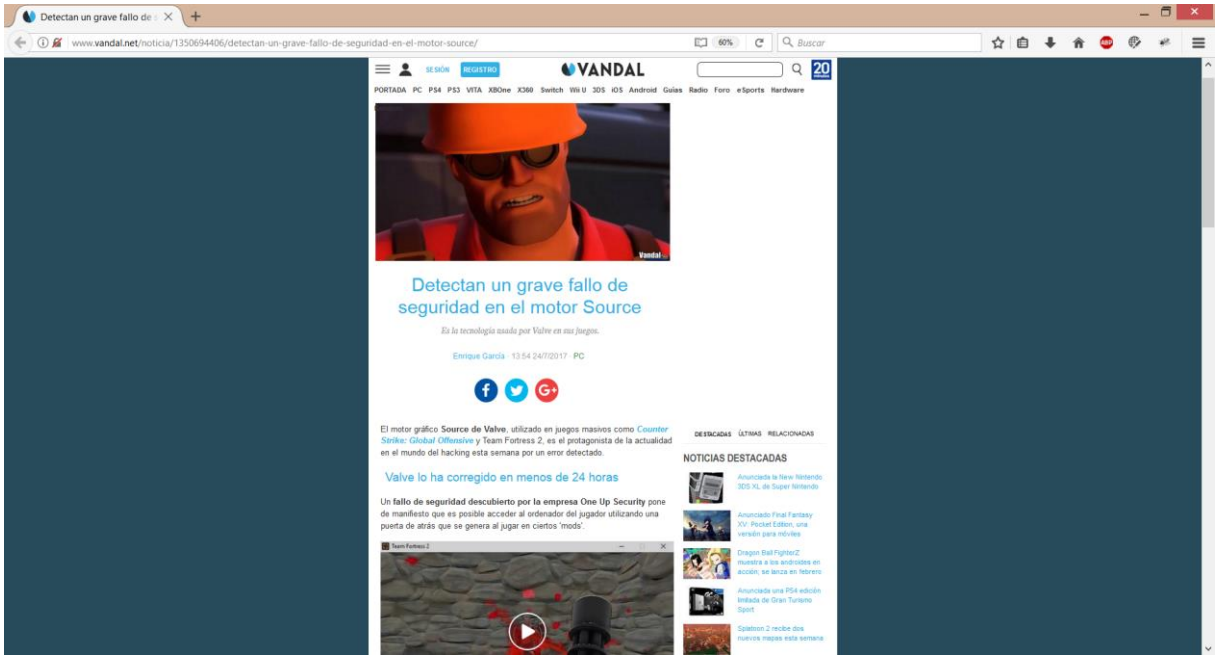
- Exploits remotos: Si se accede desde fuera del sistema, ya sea desde la red de trabajo o la propia internet.
- Exploits locales: Si la vulnerabilidad viene del propios sistema, como un usuario del sistema sin privilegios de administrador que consigue actuar como tal sin acceder a los privilegios de manera legítima.
- Exploit Zero Day: Exploit que no han sido publicados y permanecen en un ámbito privado dentro de la empresa.
- Exploit ClientSide: Son vulnerabilidades de aplicaciones instaladas en el sistema. El ataque se centra en el tipo de archivos que se leen por estas aplicaciones, como lectores de pdf, documentos, etc

Se recomienda el uso de firewall, y de un antivirus que analice el archivo antes de ser ejecutado o leído.

Si no es detectada y corregida, se puede acceder al sistema sin la necesidad de usuarios ni contraseñas.

Sus daños no tiene por qué limitarse solo a robar datos, también pueden dañar el sistema.

Recientemente se ha descubierto un exploit en el motor de físicas Source utilizado por la compañía Valve para realizar sus videojuegos. El fallo permita cargar código malicioso cuando se lograba matar a un jugador, dejando la puerta abierta para poder robar información o causar daños. Acceso a la noticia de vandal.



Página noticia 2

Lo mejor que se puede hacer contra este tipo de ataque es tener el software actualizado, ya que las compañías están continuamente revisando sus programas en busca de fallos.

2.1.5. Backdoors

Secuencia especial dentro del código de un determinado software, que permite al creador manipular libremente el programa. A diferencia de los exploits, los backdoors no son necesariamente un error, sino un atajo puesto a propósito.

Ilustración 6

Al tratarse de una secuencia que permite manipular libremente el programa, implica una falla de seguridad enorme, si es descubierta por un atacante, puede dejar en un estado más que vulnerable al sistema.

2.2. Honeypots

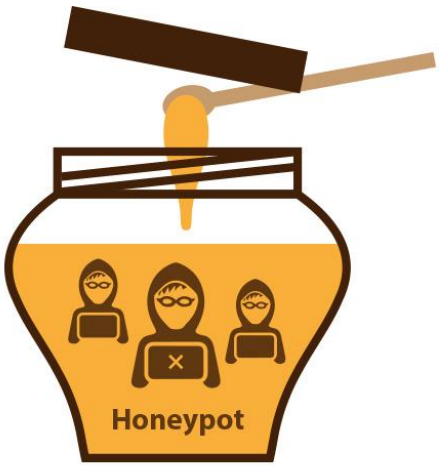


Ilustración 7

De la necesidad de corregir estos fallos, las empresas hacen uso de señuelos que les permiten averiguar antes de que sea tarde, las vulnerabilidades de sus sistemas u aplicaciones.

Un honeypot es una herramienta informática creada para hacer de señuelo, con la intención de obtener información de atacantes potenciales, y asi poder actuar antes de que cualquier ataque afecte al sistema legítimo. Otros usos suelen ser el de alertar de un posible ataque, o el de ralentizarlo (sticky honeypots).

Algunas de sus características :

- Generan un volumen pequeño de datos
- No existen falsos positivos
- Necesitan recursos mínimos.
- Son herramientas que implican riesgo potencial en la red.
- Tiene alto grado de detección por parte de intrusos.

Tambien se deben advertir algunos puntos:

- Un honeypot no sirve para eliminar o corregir fallos, mas bien para estudiar esos fallos.
- No es una medida de seguridad, si la red es vulnerable lo seguirá siendo.
- No va a evitar que el atacante fije su atención en nuestra red.

2.2.1. Tipos de honeypots

Los honeypots pueden dividirse en tres estamentos, segun el nivel de interaccion que permitan:

- **Baja interacción**: Emulan un servicio o aplicación vulnerable. Son fáciles de instalar y configurar, además de ser el tipo que menos en riesgo pone la red. Sin embargo, es el tipo mas limitado a la hora de obtener infomación ya que solo obtiene información sobre los intentos de conexión, la fecha y hora en que se intentaron, la ip y los puertos de origen y destino.
- Ventajas: Facil instalación y mantenimiento.
- Desventajas: Calidad de los datos limitada.

  Ejemplos: HoneyD, KFSensor, SPECTER, Dionaea

- **Media interacción**: Aqui los servicios que son emulados pueden responder al atacante, el cual puede acceder a recursos falsos. Son utilizados tambien para capturar malware dejado por el atacante en el sistema trampa. Al igual que los honeypots de interaccion baja, tambien puede emular servicios,pero ademas, pueden emular sistemas.

  Proporciona un mayor riesgo que los de interacción baja, pero sus resultados son muchos más completos al disponer de la capacidad de almacenar ese malware.

- Ventajas:Mejor calidad de los datos. En vez de almacenar datos en un log, en algunos casos se puede obtener el malware que se ha utilizado en el honeypot.
- Desventajas:Mas dificil de utilizar

  Ejemplos: kippo,cowrie

Especial mención al proyecto honeydrive realizado por Ioannis, el cual reune y configura varios de los honeypots nombrados, ademas de proveer de herramientas de analisis forense.

- **Alta interacción**: Son sistemas reales en equipos reales, son colocados en la red interna de la organización, son el tipo de honeypot que implica mayor riesgo, sin embargo, toda actividad detectada debe ser observada detenidamente. Estos honeypots dan acceso total al atacante, motivo por el cual se ponen en ambientes controlados, pero son tambien los que mayor cantidad y tipo de informacion recogen, estos son a su vez los mas dificiles de configurar. Son situados dentro de la red de la empresa protegidos por firewall, por lo que para atacarlos se han de superar todas las barreras previas.

  Las funciones del firewall serán la de permitir el tráfico hacia el honeypot, y evitar que se pueda atacar los equipos desde el honeypot.

- Ventajas: Es el tipo que mejor recopilacion de datos ofrece, entre tipos de datos y archivos recojidos, tanto en cantidad como en calidad.

- Desventajas: Es el tipo de honeypot mas arriesgado al poner menos restricciones al atacante.

  Ejemplos: Honeynet

Otra forma de clasificar los honeypots es en funcion de su ambiente de implementación entre los que podemos diferenciar

- **De produccion**: Su principal uso consiste en proteger una red, generalmente de una empresa, y suelen estar completos por honeypots de interacción completa.
- **De investigación**: Su finalidad consiste en utilizarlos como herramienta educativa para estudiar tipos y patrones de ataques. Suelen ser instalados en universidades y centros de estudios, este proyecto se basa en este tipo de herramientas.

2.2.2. Donde colocar un honeypot

Una cuestión a tener en cuenta antes de montar el honeypot, es en que parte de la red, en función del tipo de honeypot y la seguridad montada en nuestro sistema y entorno a el, podemos distinguir tres lugares en la red:

- **Antes del firewall(Front of firewall)**: En esta zona se evita en gran medida el riesgo de colocal el honeypot, al estar fuera de la zona segura, su desventaja es que estamos ciegos antes de ataques internos. Otra pega es que corremos el riesgo de generar mucho tráfico debido a que el sistema es facil de atacar.

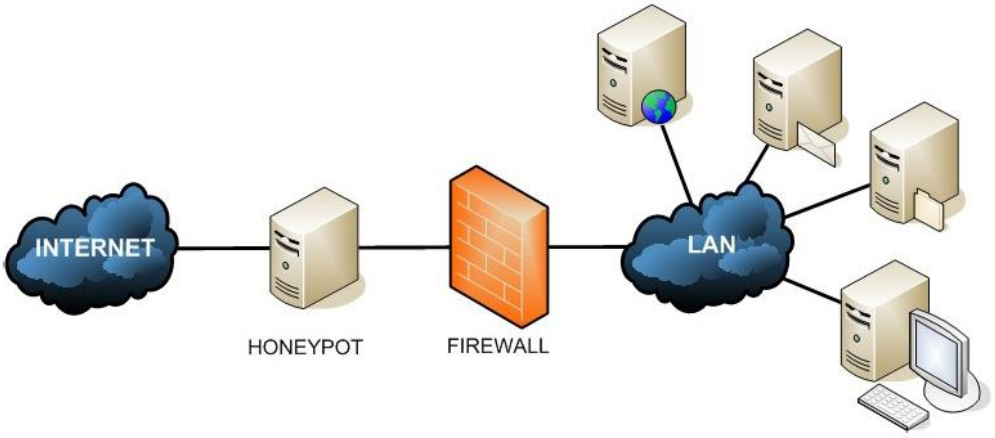


Ilustración 8

- **Detras del firewall(Behind the firewall)**: Si colocamos el honeypot tras el firewall, quedará afectado bajo las normas del firewall. Es el lugar más arriesgado donde colocar el honeypot, ya que esta dentro de nuestra red. Se han de configurar algunas reglas para permitir el tráfico hacia el honeypot, asi como firewalls extras o sistemas de control de acceso, con la intención de proteger el resto de equipos de la red.

  Su punto a favor es que permite la detección de atacantes internos.

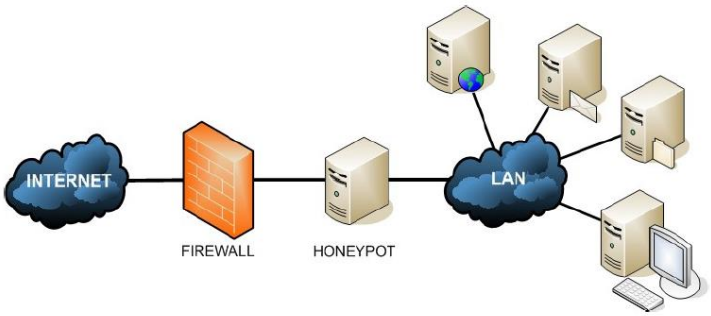
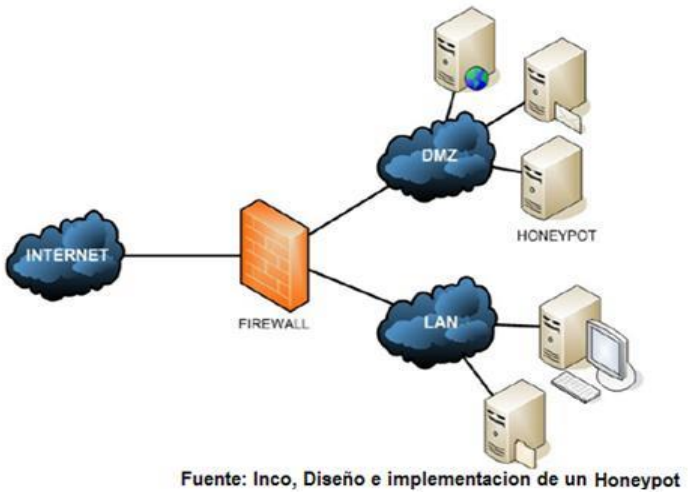


Ilustración 9

En la zona desmilitarizada (DMZ): Esta ubicación nos permite colocar el honeypot tras un firewall, y al mismo tiempo controlar el riesgo que supone tenerlo

en la red. Esto se debe a que la DMZ es una subred aparte con su propio firewall que aisla el sistema trampa del resto de la red.

Esta arquitectura nos permite detectar tanto ataques externos como internos, aunque la detección a los internos se ve debilitada. Con esta arquitectura conseguimos evitar dos cosas: por un lado, que el resto de sistemas de seguridad salten a la mínima cuando se accede al honeypot, y por otro, evitamos el riesgo de infectar la red al encontrarse el honeypot aislado de esta. Sin embargo, si que se podria acceder al honeypot desde la red local si esta se ve afectada.



Fuente: Inco, Diseño e implementacion de un Honeypot

Ilustración 10

3. Metodología

A continuación haremos una breve descripción de algunas herrramientas que vamos a utilizar, asi como de la maquina en la que vamos a montar el sistema. Para terminar esta sección se describirá como se instaló y montó el sistema, y como ha de ponerse en funcionamiento.

3.1. Descripción herramientas utilizadas

Este proyecto esta basado en la distribución honeydrive realizada por Ioannis(http://bruteforcelab.com/honeydrive), el cual es una distro de linux que contiene varias herramientas honeypots (de baja y media interacción), asi como diversas utilidades que van desde herramientas de análisis forense hasta frameworks para realizar auditorias de redes y tests de penetración.

A continuación describiremos algunas de sus herramientas:

Kippo: Honeypot SSH de interacción media, diseñado para registrar ataques de fuerza bruta, asi como la interacción shell realizada por el atacante. Esta inspirado, pero no basado en kojoney.

La información obtenida se guarda en logs que podemos exportar a una base de datos de mySQL. Tambien almacena dentro de la carpeta ‘dl’ los archivos que el atacante descarge mediante el comando wget.

Algunas de las carpetas a tener en cuenta a la hora de poner a funcionar esta herramienta son:

- dl/ : Carpeta donde se guardan los archivos dejados por el atacante a traves del comando wget.
- Log/ : carpeta donde se guardaran el log que utilizaremos para comprobar el funcionamiento del honeypot. Aqui se verán reflejados los intentos de

conexion, junto con los datos de logeo, ademas de las ordenes utilizadas si lo consiguen.

- Logt/tty: logs de sesion
- Utils/playlog.py : utilidad par reproducir logs de sesion.
- Utils/createfs.py : Usado para crear el sistema de archivos falso(guardado en fs.pickle).
- Honeyfs/ : Contenido para el sistema de ficheros falso.

Para iniciar el honeypot solo hay que ejecutar el archivo start.sh, si queremos ejecutarlo bajo alguna opcion determinada el comando que debemos mirar es el de twistd.

Actualmente hay una versión más actualizada de este honeypot, cowrie, el cual tambien esta siendo utilizado para este TFG.

En un principio, se optó por este en particular ya que ,aparte de cumplir con lo que el tfg, se encontraba mas información sobre su instalación y uso que de cualquier otro, tanto en español como en inglés. Además tiene interfaz gráfica propia, kippo-graph, elemento que también satisfacia otro de los objetivos del tfg.

Finalmente, y tras aprender a utilizar kippo, se optó por utilizar tambien Cowrie, el cual es una versión mas avanzada. Esta versión tambien puede utilizar kippo-graph con unos cambios mínimos que detallaremos mas adelante.

Cowrie: Versión mas actualizada del honeypot kippo desarrollada por Michel Oosterhof. Posee las características de kippo ya que se basa en el mismo proyecto, pero con algunos añadidos que lo hacen más completo, pero tambien algo mas complejo. Algunos de estos añadidos son:

- Soporte SFTP y SCP para subida de archivos.
- Soporte para la ejecución de comandos SSH.
- Logeado para intentos de conexión directa TCP.
- Capacidad para retransmitir conexiones SMTP a un honeypot SMTP.
- Registros en formato JSON.

Kippo2mysql: Script que nos permite pasar los datos obtenidos por los logs de kippo a la base de datos de mysql.

Mysql-server: Servidor de base de datos donde registraremos la actividad obtenida del honeypot.

Kippo-Graph: Interfaz web para kippo, con ella podremos visualizar estadísticas y obtener información visual acerca de los atacantes. La versión mas actualizada tiene ademas soporte para cowrie.

Myphpadmin: Interfaz web para administrar mysql-server.

Snort: Herramienta capaz de ser usada como sniffer y como sistema de deteccion de intrusiones de nuestra red. Podemos configurarlo para que logee paquetes en funcion del patrón que definamos.

ClamAV + ClamTK: Se han instalado para proporcionar un soporte antivirus al sistema raspbian. ClamAV es el antivirus y ClamTK su interfaz gráfica.

P0f: herramienta diseñada para realizar fingerprinting, que consiste en la suposición de un determinado SO a través del análisis del comportamiento observado cuando se realizan determinadas peticiones a una máquina, de forma pasiva. Es capaz de intentar averiguar el sistema de las maquinas que envian tráfico a nuestra máquina.

Podemos usarlo simultaneamente al honeypot de manera que a parte de la información que da el mismo honeypot , sobre de donde viene el ataque, podemos intentar además averiguar algunos datos mas del atacante como, por ejemplo , el sistema operativo que usa.

Wireshark: herramienta diseñada para analizar el tráfico de una red, posee además un analizador de protocolos. Posee una interfaz gráfica, con opciones de filtrado que nos permite ver de manera intuitiva el tráfico de nuestra red.

Puede usarse para leer los archivos pcap generados por p0f, en el caso de que queramos echar un vistazo e intentar averiguar algunos datos acerca de las máquinas que han intentado acceder a nuestro honeypot.

3.2. Descripción de la arquitectura del sistema

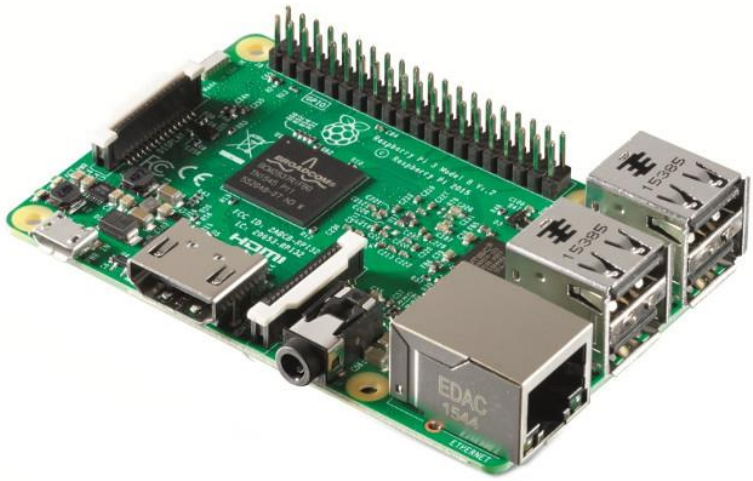


Ilustración 11

- Hardware
  - Raspberry Pi 3 Modelo B
  - CPU: 1.2 GHz, 64 bit quad-core ARMv8
  - GPU: Broadcom VideoCoreIV, OpenGL ES 2.0, MPEG-2 y VC-1(con licencia), 1080p30 H.264/MPEG-4 AVC
  - Memoria(SDRAM):1GB(Compartido con GPU)
  - Conectividad: 10/100 Ethernet(RJ-45) via hub USB, Wifi 802.11n, Bluetooth

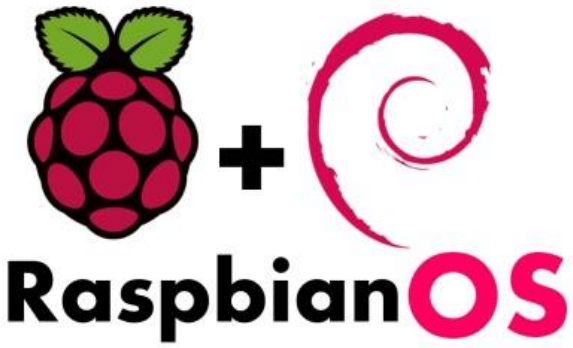


Ilustración 12

- SO:
  - Raspbian(Debian version Jessie)

Ilustración 13

3.3. Breve tutorial de instalación del sistema

1. Lo primero es instalarle una distribución a la tarjeta, para eso basta con ir a la página oficial de raspberry, y seleccionar un sistema. El resto consiste en seguir los pasos una vez encendemos la maquina con la tarjeta dentro.
2. Para instalar tanto kippo como cowrie es necesario copiarse el proyecto desde su directorio en git, e instalar las dependencias necesarias, las cuales estan anotadas en su pagina principal.
3. Estaría bien instalar ademas mysql, ya que el entorno web que tiene kippo para obtener los datos los coge de una base de mysql.
4. Descargamos kippo-graph.
5. Configuramos kippo y kippo-graph para que, el primero pueda escribir en la base de datos y el segundo pueda leer de ella, para realizar sus gráficas.

Para mas detalle mirar los apéndices A y C.

3.4. Tutorial para echar a andar el honeypot

1. Lo primero es colocar la máquina en un lugar de la red donde no afecte su actividad al resto de dispositivos, por lo que meteremos el honeypot en una zona desmilitarizada
    a. Para ello entramos en la configuración del router para activar esa opción e introducimos la dirección ip de nuestra máquina.
2. Para echar a andar kippo o cowrie lo primero es redirigir el tráfico del puerto 22 a otro que vaya a ser vigilado por el honeypot, en este caso el 2222.
3. Por último antes de echarlo a andar mirar los archivos de configuración de :
    a. Kippo: asegurarnos de que puerto va a estar mirando y de si tiene las credenciales correctas para poder meter datos en la base de mysql, hay que mirar tambien que algunos parámetros esten activados, como el que permite manipulación por parte del atacante para hacerle creer que tiene el control.
    b. Kippo-graph: comprobar que en el archivo de configuración esten bien introducidos los datos para acceder a la base de datos correcta. Si queremos cambiar entre kippo y cowrie solo deberemos cambiar los siguientes campos:
        i. Todos los relacionados con la base de datos, ya que con cowrie abremos creado un usuario y una base de datos exclusiva.
        ii. La variable back_end_engine la cual admite dos opciones ‘kippo’ o ‘cowrie’.
        iii. Back_end_path donde deberemos poner el directorio raiz de cowrie.(Solo en el caso de usar este honeypot, si no, no hace falta).

    Para mas detallle mirar apéndice B.

4. Descripción de la experiencia

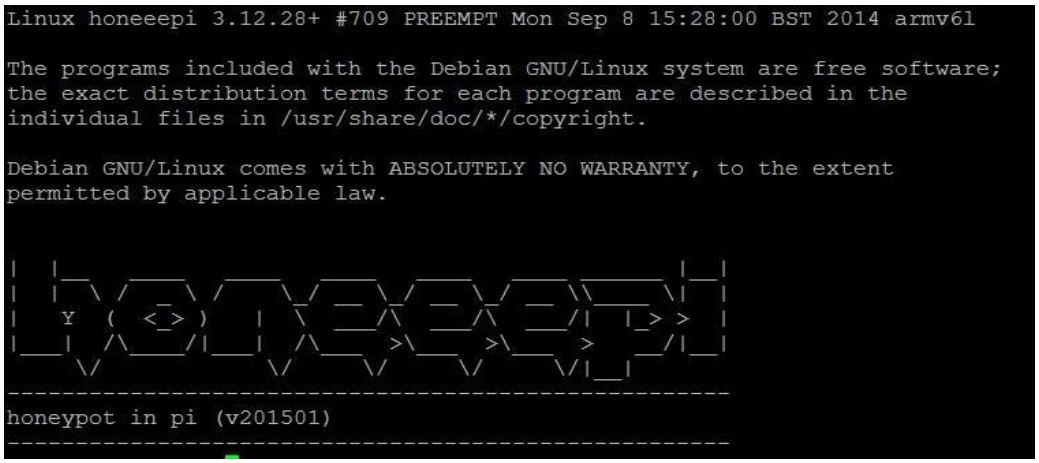
El objetivo de esta sección es exponer la evolución de este trabajo, asi como las conclusiones de sucesivas puestas en marchas de las raspberry.

4.1. Diario de desarrollo

El desarrollo del proyecto comenzó en febrero haciendo uso de la distro honeydrive ya mencionada en el apartado 3.1 en una máquina virtual. En este momento, se aprendió a manejar algunas de las herramientas, asi como el intento de captación de algunos datos aunque sin éxito.

Escritorio Honeydrive 1

Al llegar a marzo se empezó a montar el sistema dentro de una raspberry pi 3. Inicialmente se utilizó honeeepi, pero fue descartado al carecer tanto de interfaz gráfica para el sistema como de la interfaz kippo-web. Tras esto se tomó la decisión de instalar las herramientas necesarias para la ejecución del proyecto, es en este punto donde se tomó la decisión de tomar como referencia el proyecto honeydrive.



Apariencia honeeepi 1

Durante los meses de marzo y abril se estuvieron instalando varias de las herramientas listadas en la mencionada distribución, algunas quedaron descartadas, bien por ser obsoletas(o de usar dependencias abandonadas hace mucho) o bien por no ser totalmente necesarias.

Debido a la carga de trabajo durante los meses de abril, mayo y junio la ejecución fue pospuesta hasta julio. Es a mediados de este mes, cuando este proyecto fue retomado.

4.2. Informes de ejecución

4.2.1. Informe del 19 de julio de 2017(kippo)

- El 19 de julio se activó el honeypot kippo a las 11:01 y se mantuvo encendido durante 7 horas hasta las 18:13. En este tiempo se registraron varios intentos de loggeo. La mayoria fallidos:

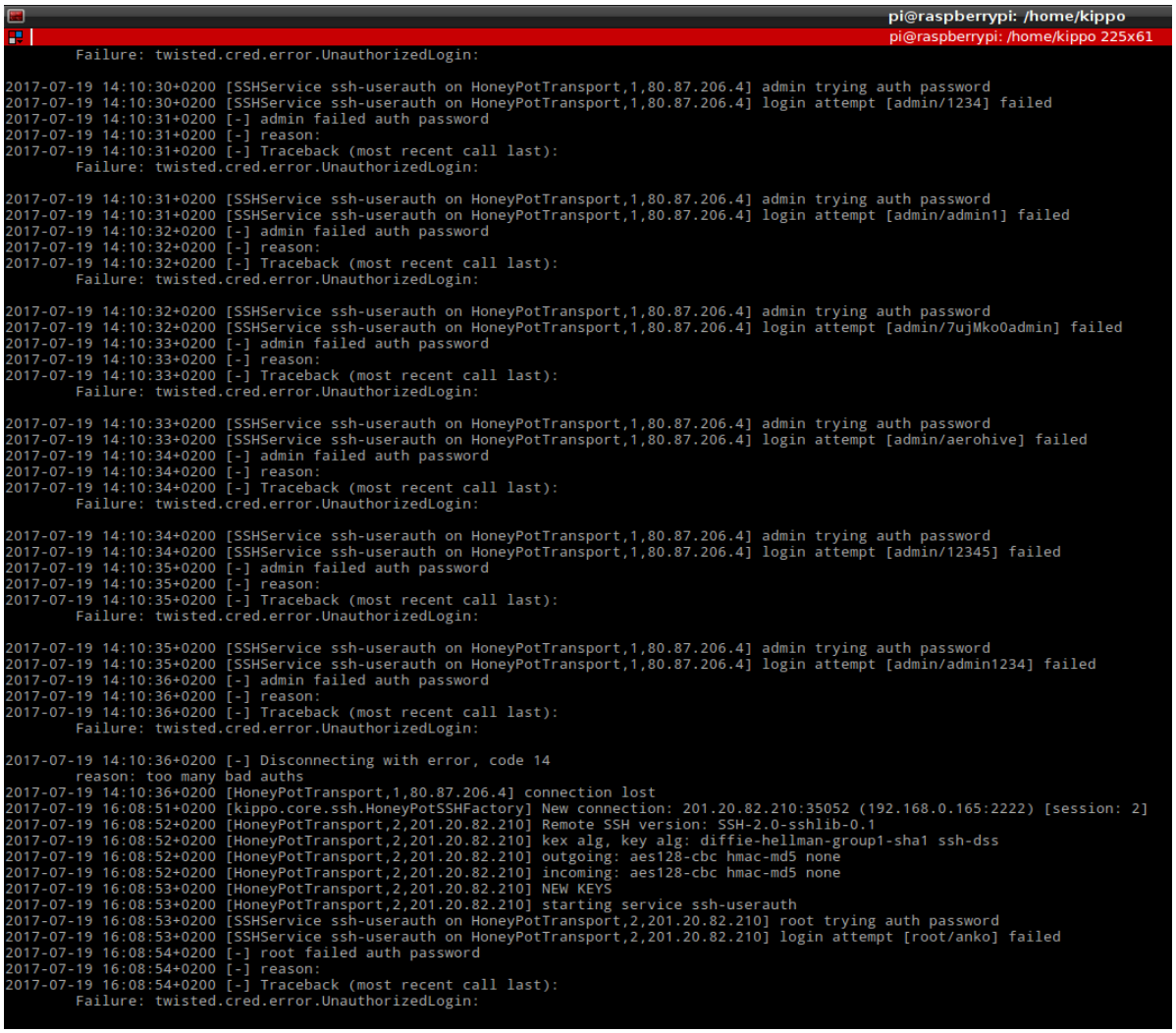


Foto de algunos de los intentos de logeo 1

En la imagen de arriba podemos observar los intentos según los va registrando kippo. Primero nos dice que es lo que esta intentando hacer el atacante, y despues con que usuario/password junto al resultado de su intento. Estos intentos se repiten hasta que el honeypot lo desconecta debido al número demasiado elevado de intentos fallidos. Despues llega otra conexión.

Esto nos sirve como pequeño avance al estudio que tenemos intencion de realizar. Gracias a la herramienta kippo-graph podemos recoger estos datos y sacar algunas conclusiones curiosas, tales como que contraseñas son las mas frecuentes cuando se intenta suplantar a alguien:

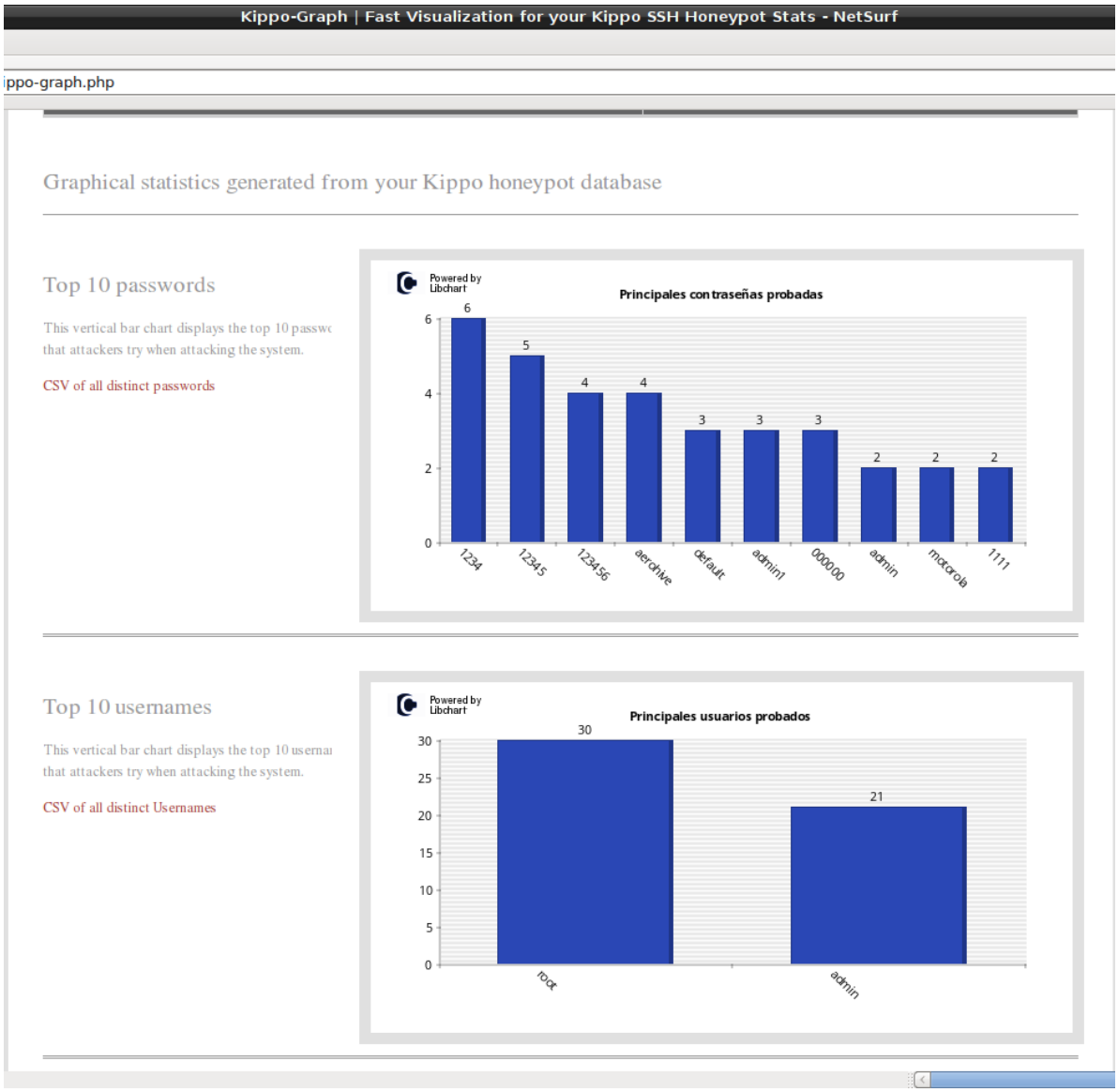


Foto de algunos de los intentos de logeo 2

En la imagen superior, podemos observar dos gráficas, una con los 10 nombres de usuarios mas usados y con las contraseñas, junto a cada barra tenemos el número de veces que se ha intentado.

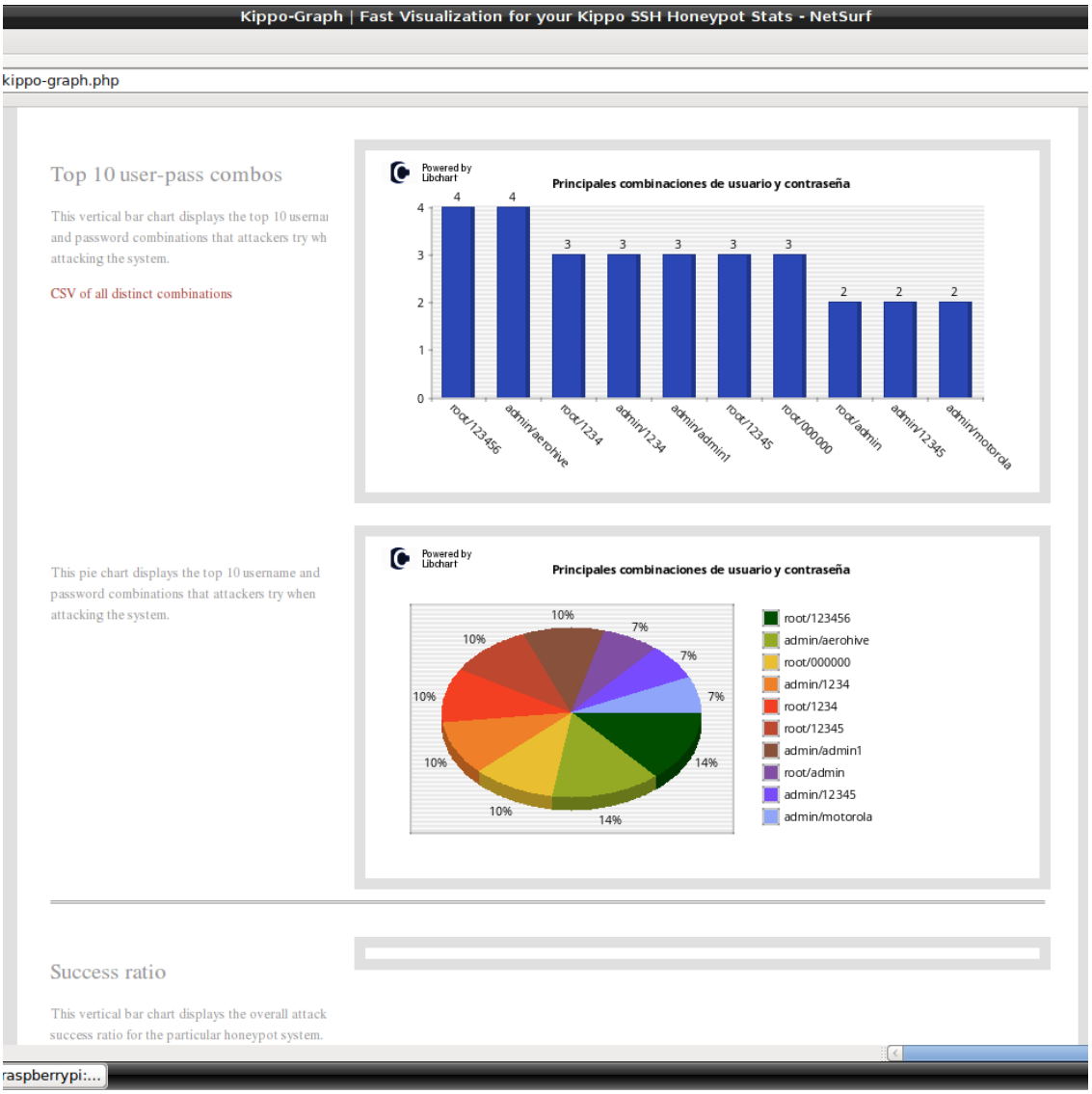


Foto de algunos de los intentos de logeo 3

Las gráficas de arriba muestran las combinaciones mas usadas de usuario/contraseña junto con su número de intentos(barras), y su porcentaje de uso (círculo).

Tambien se observa que al menos uno de los intentos de loggeo tuvo éxito, sin embargo, el programa falló lanzando la siguiente excepción:

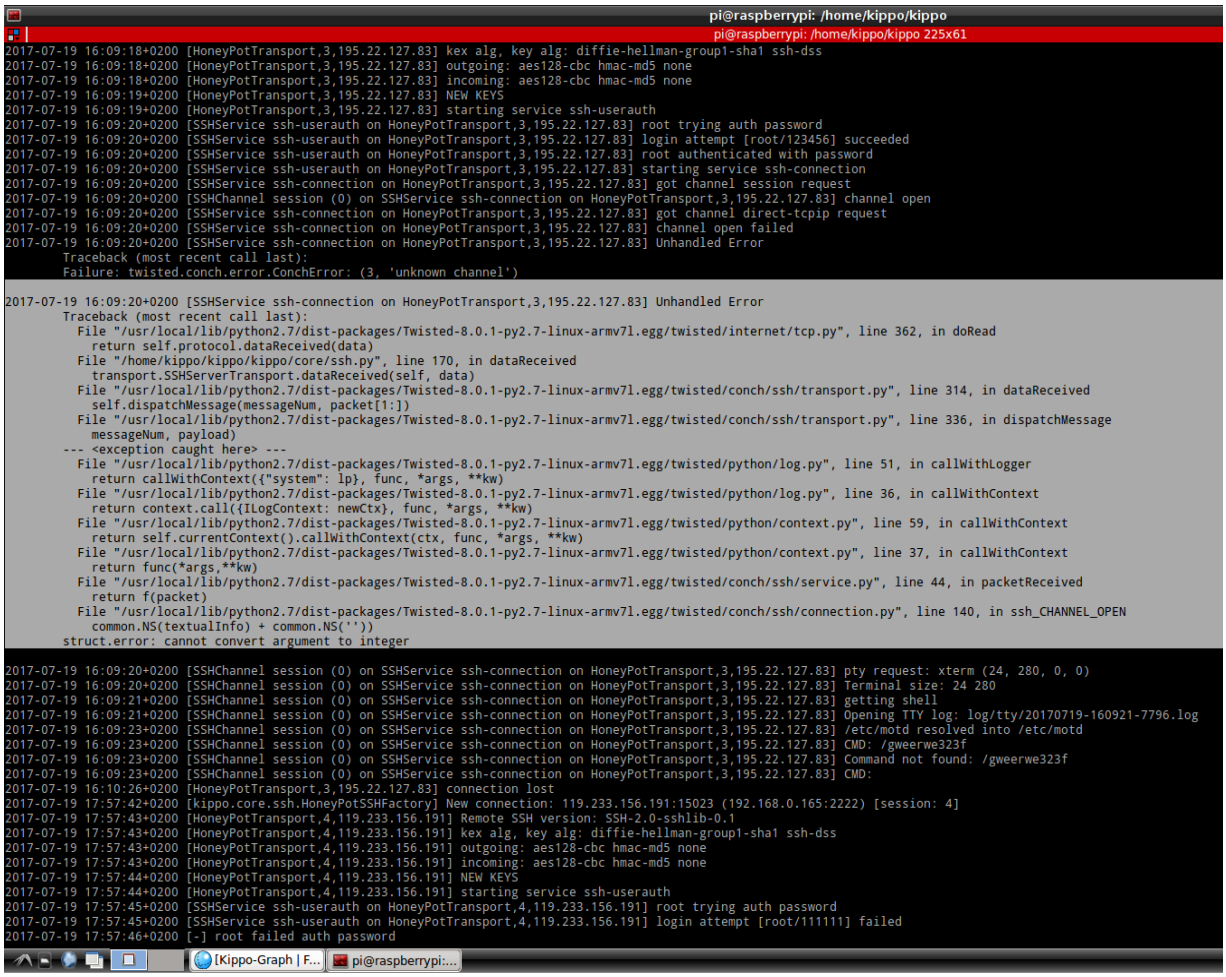


Foto de algunos de los intentos de logeo 4

Se investigará el mensaje de error para corregir el fallo.

4.2.2. Informe del 16 de agosto de 2017(kippo)

El 16 de agosto se volvió a encender kippo durante todo el día, esta vez, si ha habido intentos exitos de logeo que nos aportan algo mas de información acerca de los atacantes. Realizaremos un pequeño seguimiento de uno de esos accesos con éxito:

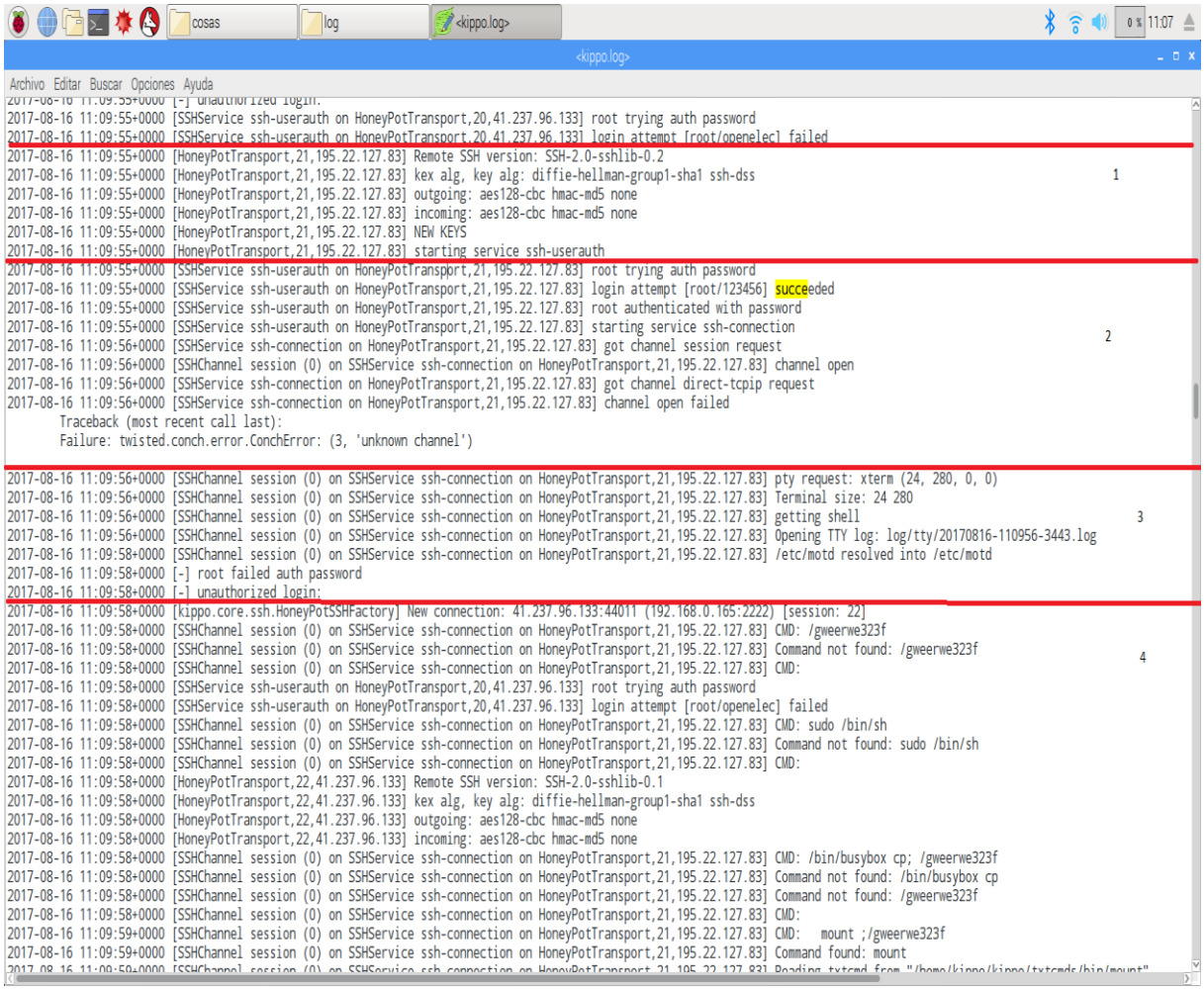


Ilustración de capturas del 16/8 1

Lo que vemos en la primera zona es el intento con éxito de logeo en nuestra máquina:

1.- En la primera zona, veremos algunas de las características que definimos en el fichero de configuración, con las que se iniciará el servicio.

2.- La segunda zona muestra un intento de logeo, en este caso con éxito. En la segunda linea podemos ver que usuario y password ha intentado el atacante.

3.- Tras la captura del loggeo con éxito se inicia el servicio que emula nuestro sistema falso.

4.- En la cuarta zona, vemos ya los comandos que va dejando el atacante. Hay que hacer especial mención al comando gweerwe323f. Se explicará esto mas adelante.

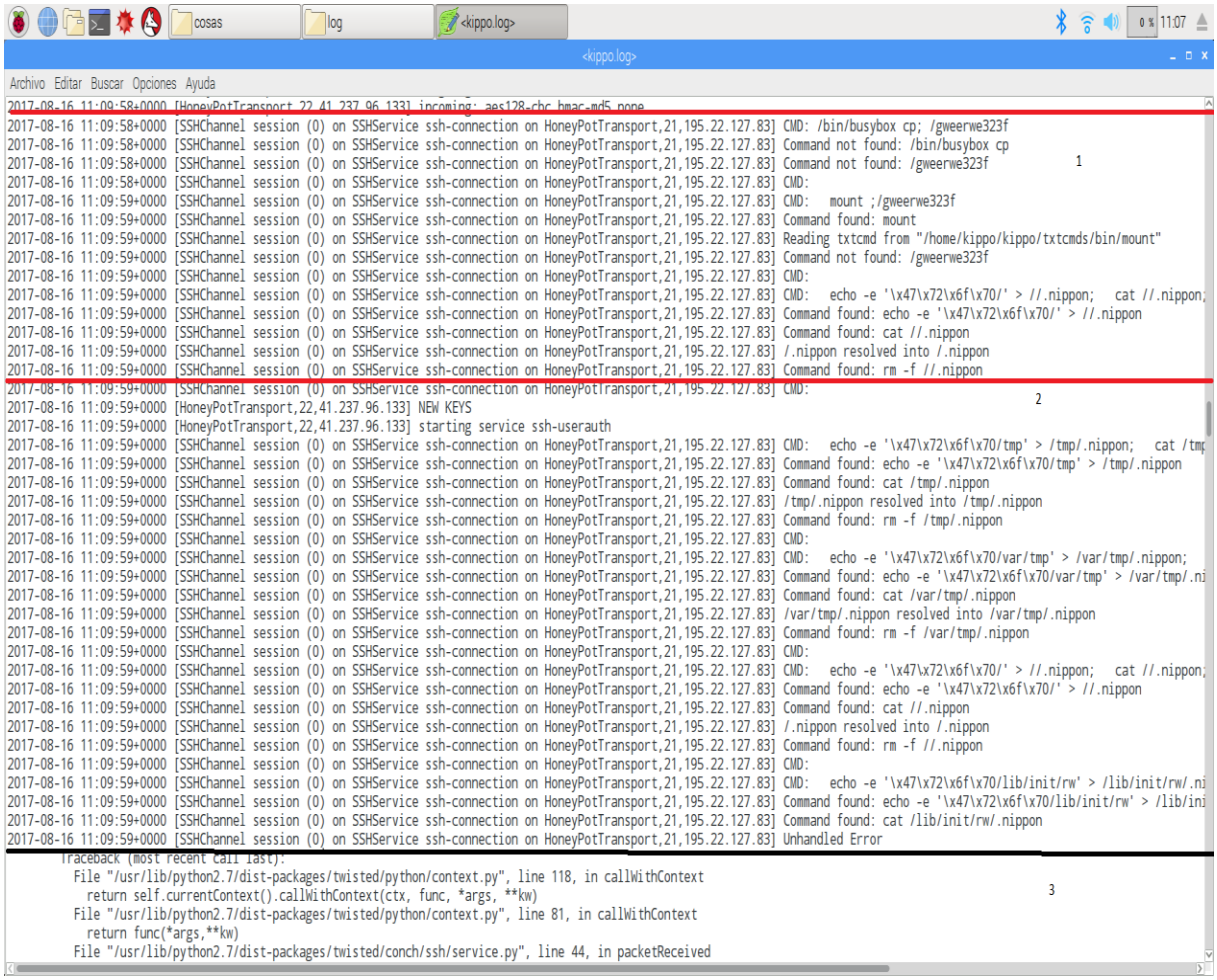


Ilustración de capturas del 16/8 2

En la primera zona de esta ilustración podemos observar que lo primero que hace es ver si el comando gweerwe323f existe, al no existir, intenta montarlo. En el momento en el que el honeypot ve un comando que existe dentro de los comandos que permitimos usar en el, lee el resto de comandos del directorio txtcmds.

El atacante no ha conseguido montar gweerwe323f por lo que intenta otra cosa. Crear archivos mediante la orden echo >, de esta manera si se crea el archivo en la máquina quiere decir que tiene permiso de escritura en ella, al hacer seguido la

orden cat al mismo archivo comprueba que tambien lo tiene de lectura.

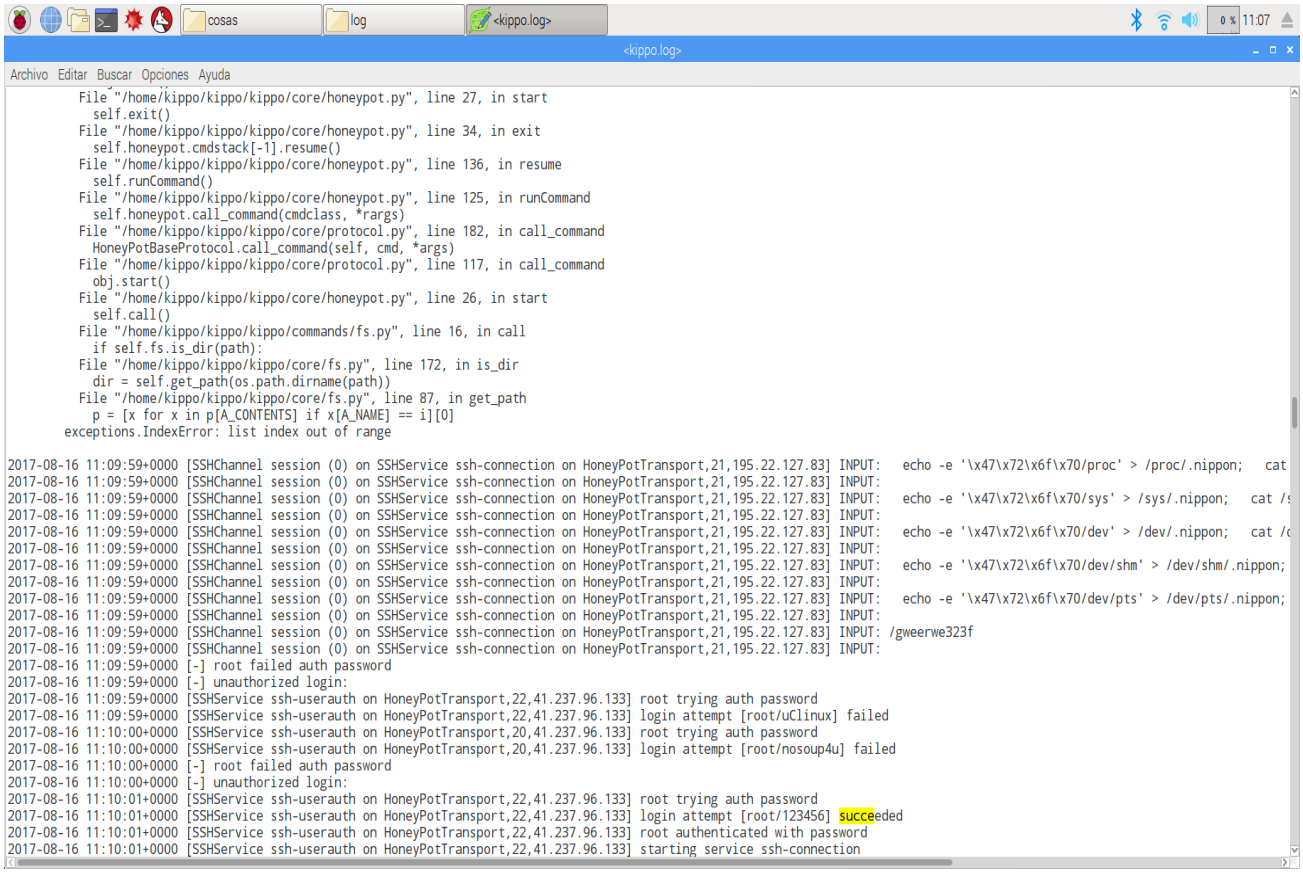


Ilustración de capturas del 16/8 3

La imagen superior nos enseña la continuación de la interacción del atacante con nuestra máquina. Volviendo a la aplicacion kippo-graph podremos echar un vistazo a estos datos de manera mas intuitiva:

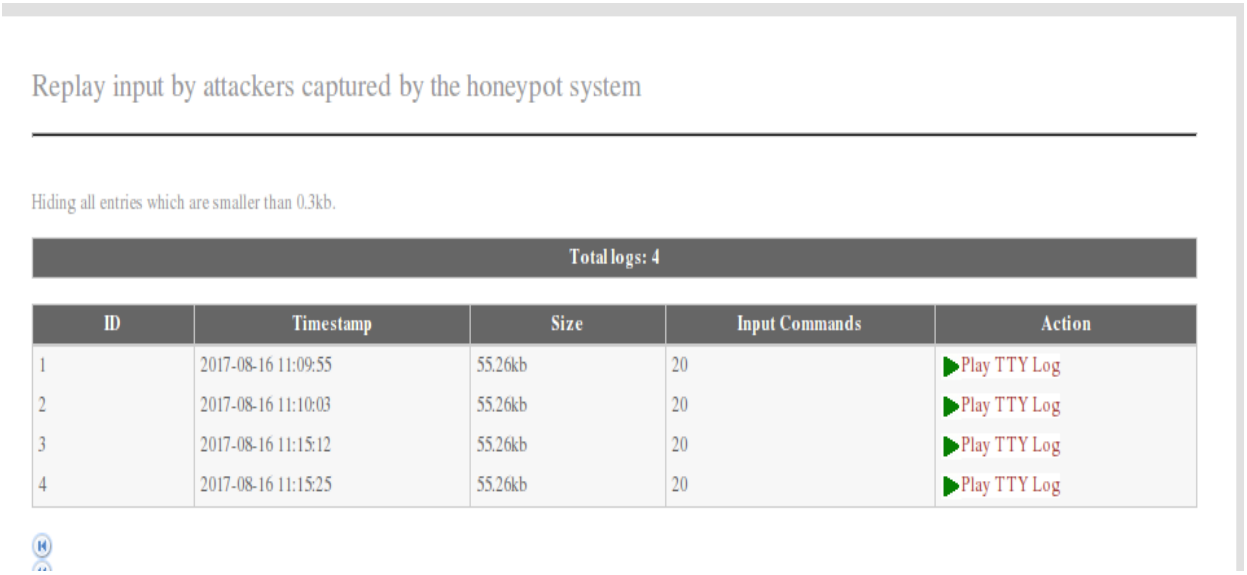


Ilustración de capturas del 16/8 4

La imagen superior nos muestra una tabla con los datos de las conexiones con éxito, su hora de inicio, el tamaño total, el número total de entradas y un enlace que nos muestra que escribió el atacante.

Si tenemos javascript activado podemos reproducir los comandos utilizados por el atacante. (imágenes inferiores)

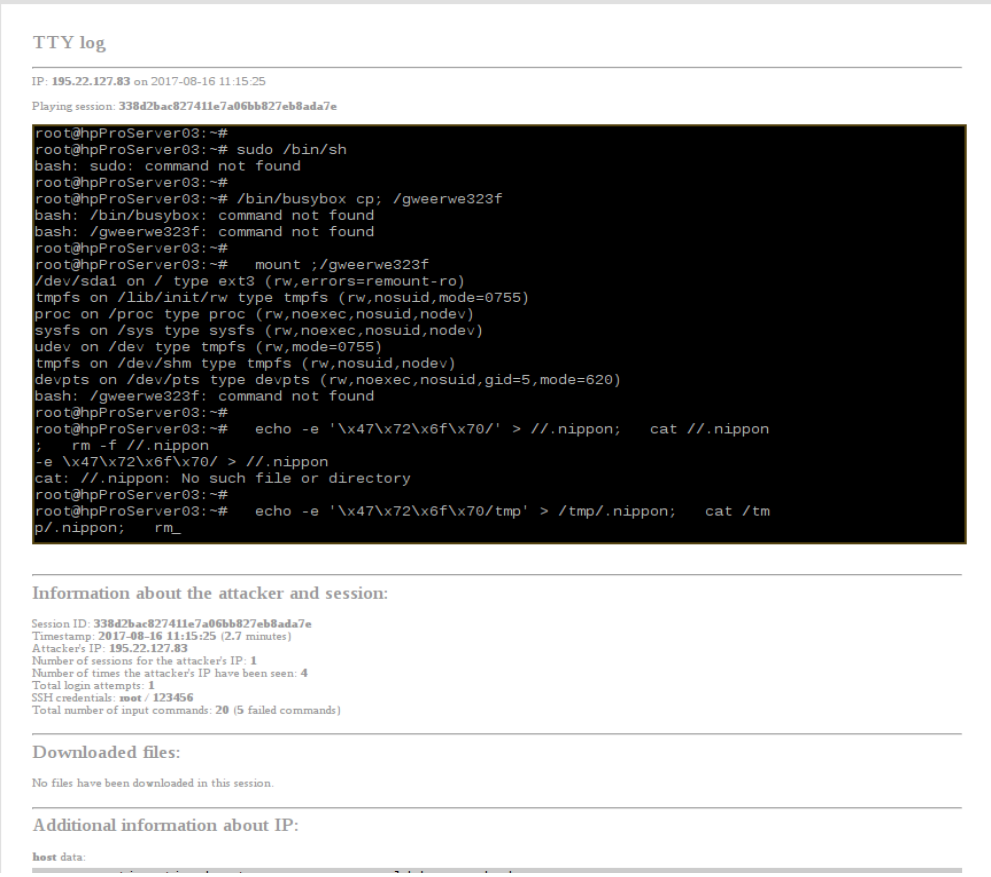


Ilustración de capturas del 16/8 5

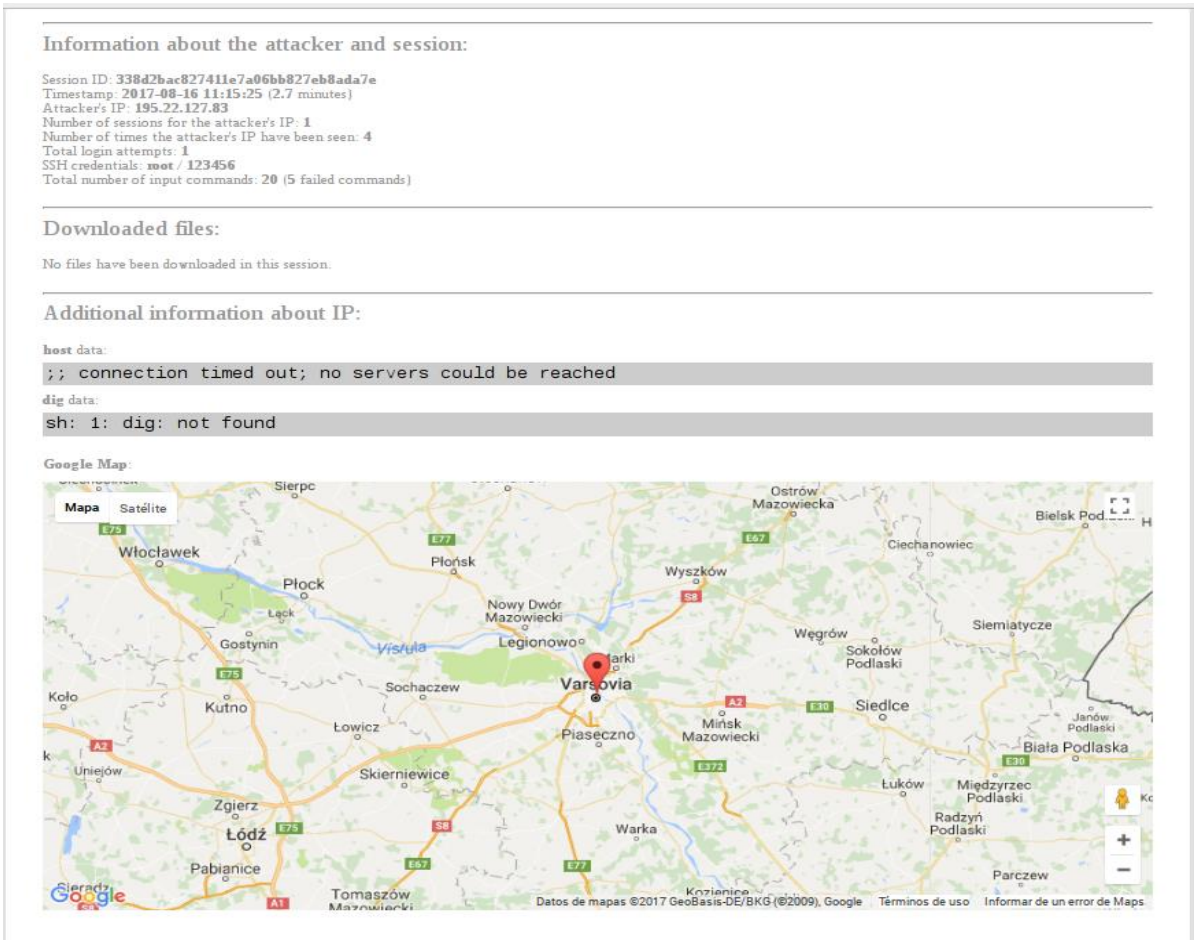
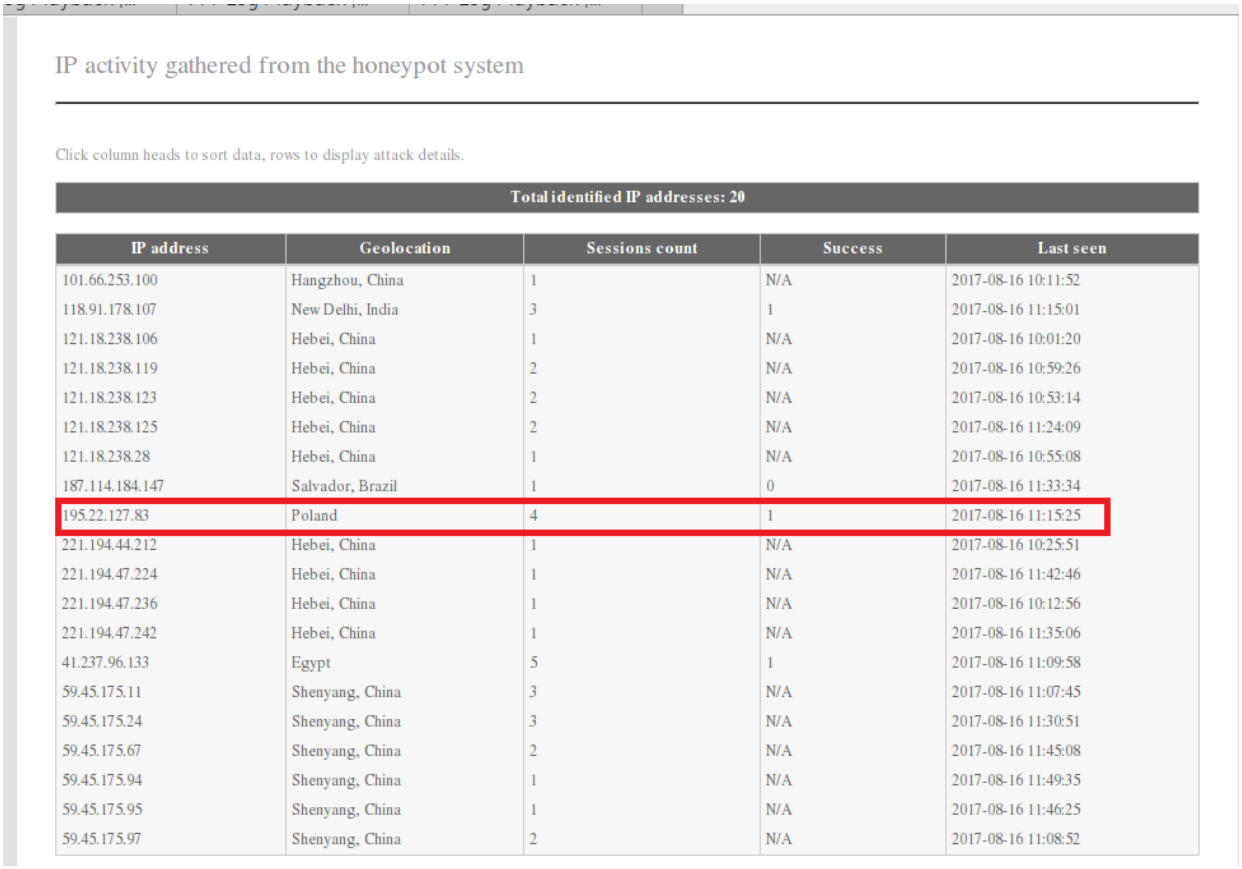


Ilustración de capturas del 16/8 6

En la pestaña ‘Network’, podemos observar algunos de los datos de las máquinas que han intentado acceder a nuestro honeypot, datos como: el país del que viene la dirección ip, la cantidad de éxitos que han tenido en nuestra máquina.(imagen inferior)



IP activity gathered from the honeypot system

Click column heads to sort data, rows to display attack details.

Total identified IP addresses: 20

IP address	Geolocation	Sessions count	Success	Last seen
101.66.253.100	Hangzhou, China	1	N/A	2017-08-16 10:11:52
118.91.178.107	New Delhi, India	3	1	2017-08-16 11:15:01
121.18.238.106	Hebei, China	1	N/A	2017-08-16 10:01:20
121.18.238.119	Hebei, China	2	N/A	2017-08-16 10:59:26
121.18.238.123	Hebei, China	2	N/A	2017-08-16 10:53:14
121.18.238.125	Hebei, China	2	N/A	2017-08-16 11:24:09
121.18.238.28	Hebei, China	1	N/A	2017-08-16 10:55:08
187.114.184.147	Salvador, Brazil	1	0	2017-08-16 11:33:34
195.22.127.83	Poland	4	1	2017-08-16 11:15:25
221.194.44.212	Hebei, China	1	N/A	2017-08-16 10:25:51
221.194.47.224	Hebei, China	1	N/A	2017-08-16 11:42:46
221.194.47.236	Hebei, China	1	N/A	2017-08-16 10:12:56
221.194.47.242	Hebei, China	1	N/A	2017-08-16 11:35:06
41.237.96.133	Egypt	5	1	2017-08-16 11:09:58
59.45.175.11	Shenyang, China	3	N/A	2017-08-16 11:07:45
59.45.175.24	Shenyang, China	3	N/A	2017-08-16 11:30:51
59.45.175.67	Shenyang, China	2	N/A	2017-08-16 11:45:08
59.45.175.94	Shenyang, China	1	N/A	2017-08-16 11:49:35
59.45.175.95	Shenyang, China	1	N/A	2017-08-16 11:46:25
59.45.175.97	Shenyang, China	2	N/A	2017-08-16 11:08:52

Ilustración de capturas del 16/8 7

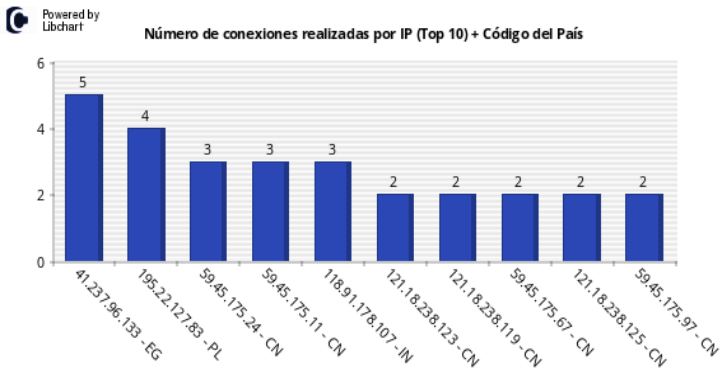
En la sección GeoIP, siguiente imagen, podemos tener algunos datos más de cada atacante y, en algunos casos, incluso podemos especificar de que ciudad vino el ataque:

Geolocation information gathered from the **top 10** IP addresses probing the system

The following table displays the top 10 IP addresses connected to the system (ordered by volume of connections).

ID	IP Address	Probes	City	Region	Country Name	Code	Latitude	Longitude	Hostname	IP Lookup
1	41.237.96.133	5			Egypt	EG	30.05	31.3666	host-41.237.96.133.tedata.net	
2	195.22.127.83	4			Poland	PL	52.2333	21.0167	195.22.127.83	
3	59.45.175.24	3	Shenyang	Liaoning	China	CN	41.7922	123.4328	59.45.175.24	
4	59.45.175.11	3	Shenyang	Liaoning	China	CN	41.7922	123.4328	59.45.175.11	
5	118.91.178.107	3	New Delhi	National Capital Territory of Delhi	India	IN	28.6	77.2	118.91.178.107	
6	121.18.238.123	2	Hebei	Hebei	China	CN	39.8897	115.275	121.18.238.123	
7	121.18.238.119	2	Hebei	Hebei	China	CN	39.8897	115.275	121.18.238.119	
8	59.45.175.67	2	Shenyang	Liaoning	China	CN	41.7922	123.4328	59.45.175.67	
9	121.18.238.125	2	Hebei	Hebei	China	CN	39.8897	115.275	121.18.238.125	
10	59.45.175.97	2	Shenyang	Liaoning	China	CN	41.7922	123.4328	59.45.175.97	

The following vertical bar chart visualizes the top 10 IPs ordered by the number of connections to the system.
Notice the two-letter country code to after each IP get a quick view of the locations where the attacks are coming from.



The following pie chart visualizes the top 10 IPs ordered by the number of connections to the system.
Notice the two-letter country code to after each IP get a quick view of the locations where the attacks are coming from.

Ilustración de capturas del 16/8 8

Otra cosa interesante de esta sección es que podemos ver la referencia que tiene este atacante, en páginas dedicadas a la seguridad como, por ejemplo virus total:

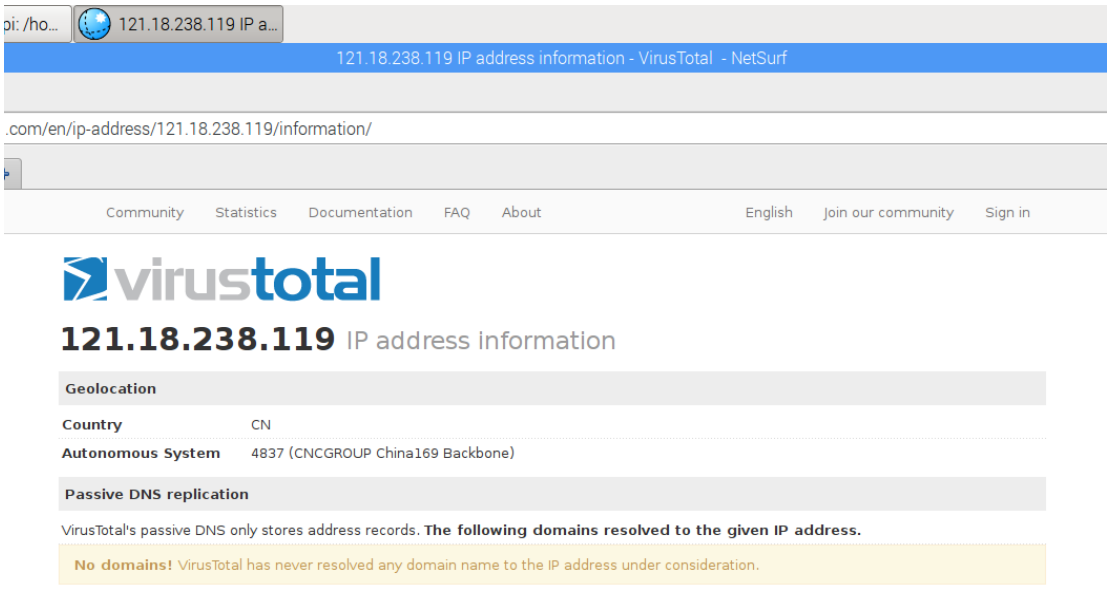
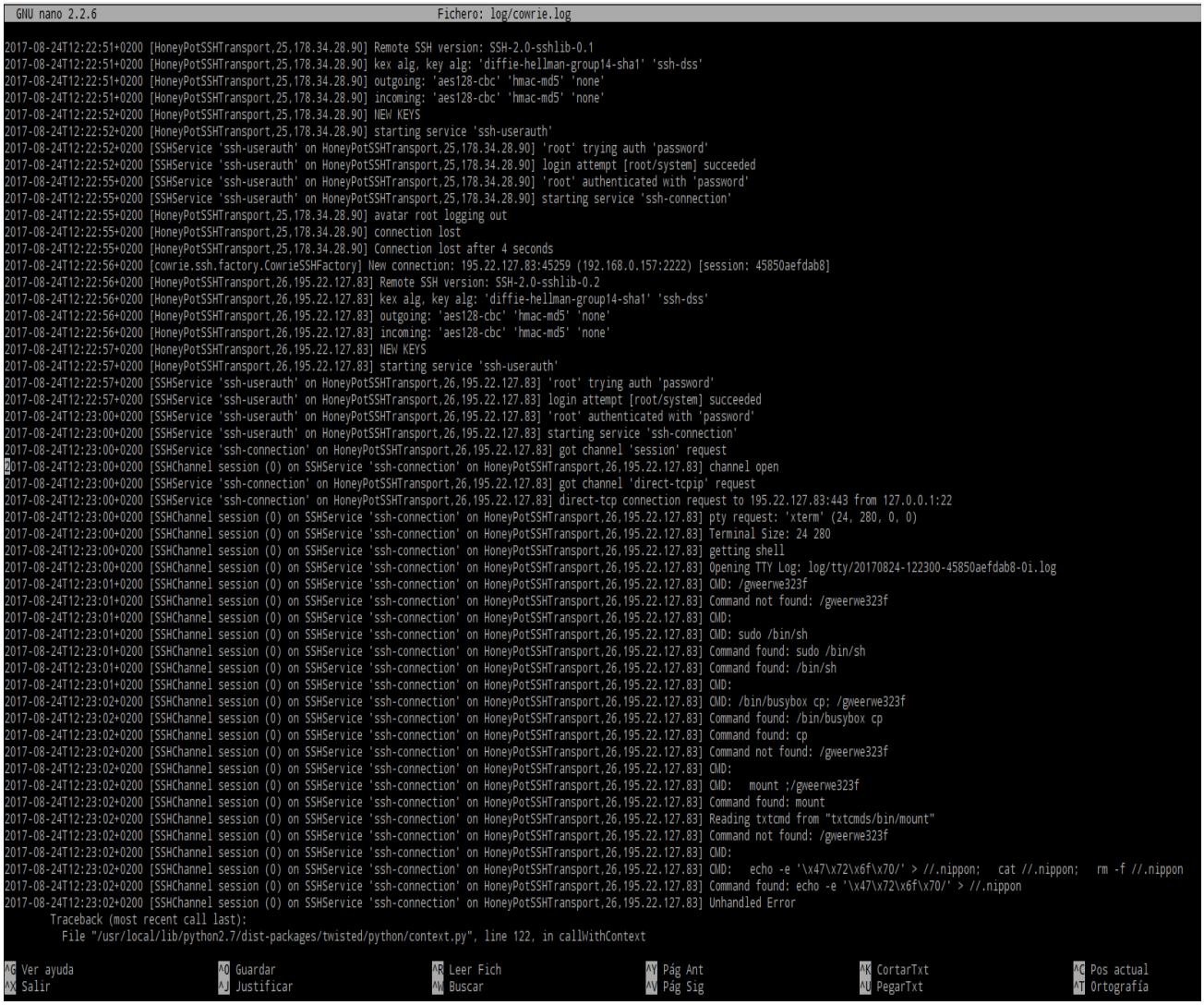


Ilustración de capturas del 16/8 9

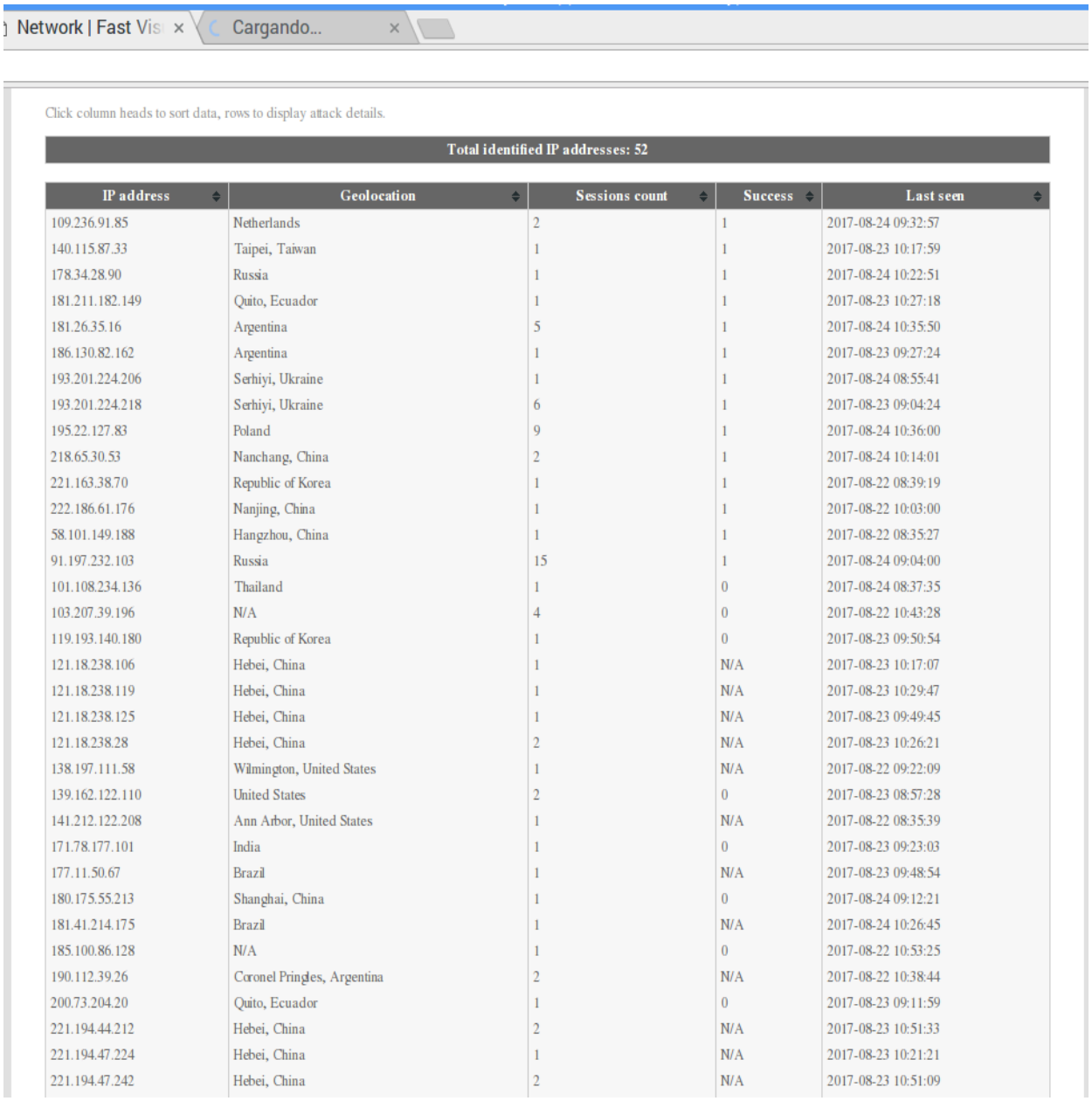
4.2.3. Informe del 24 de agosto (cowrie)

Como podemos ver, en un principio su forma de proceder es similar a kippo:



captura cowrie 1

Debido a que creamos una base de datos aparte para cowrie, podemos extraer correctamente los datos recogidos por este honeypot con tan solo cambiar las credenciales en el archivo kippo-graph.php. Como podemos ver durante los dias que hemos puesto cowrie, hemos tenido más éxito cazando a curiosos.



Network | Fast Vis × Cargando... ×

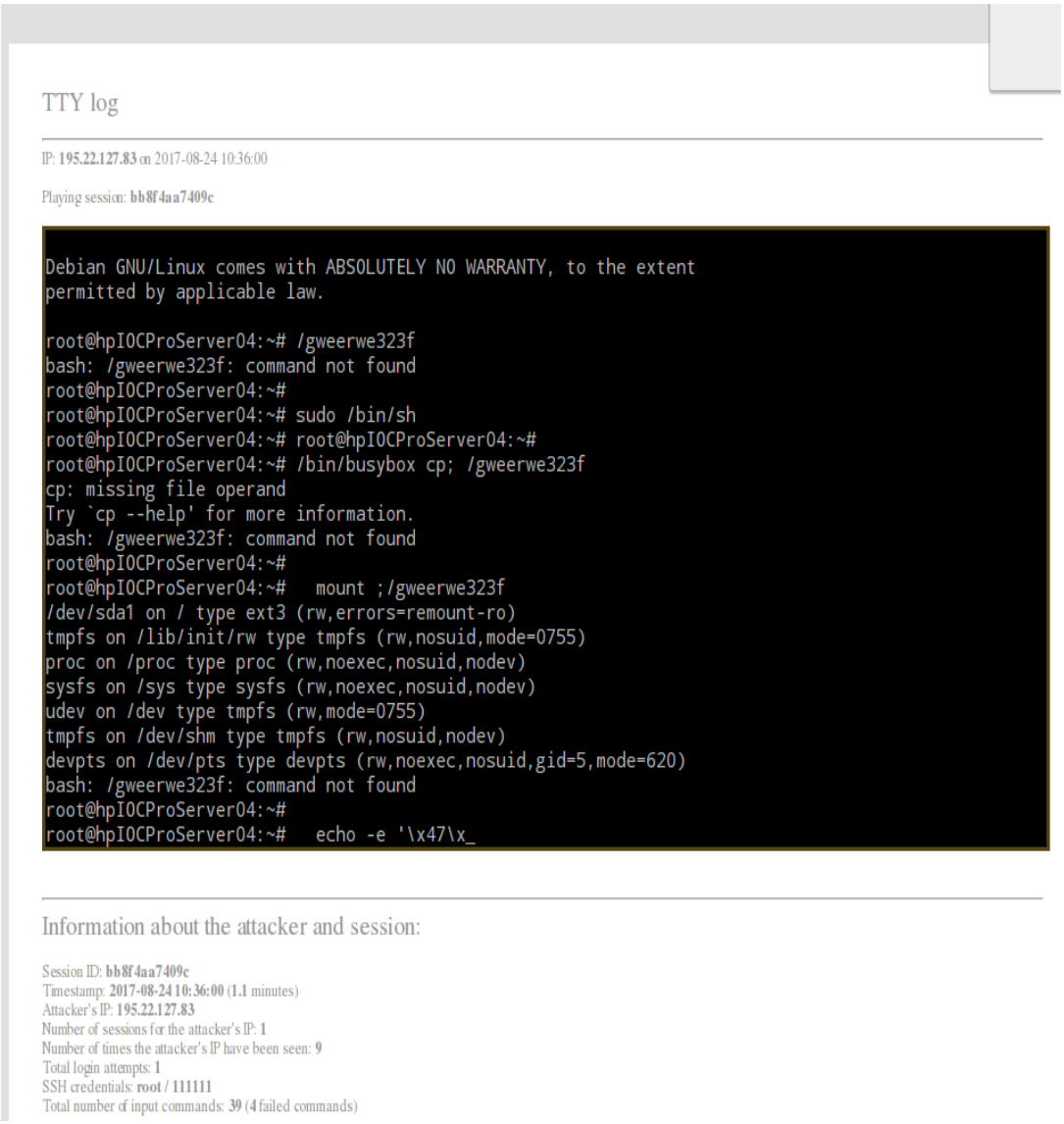
Click column heads to sort data, rows to display attack details.

Total identified IP addresses: 52

IP address	Geolocation	Sessions count	Success	Last seen
109.236.91.85	Netherlands	2	1	2017-08-24 09:32:57
140.115.87.33	Taipei, Taiwan	1	1	2017-08-23 10:17:59
178.34.28.90	Russia	1	1	2017-08-24 10:22:51
181.211.182.149	Quito, Ecuador	1	1	2017-08-23 10:27:18
181.26.35.16	Argentina	5	1	2017-08-24 10:35:50
186.130.82.162	Argentina	1	1	2017-08-23 09:27:24
193.201.224.206	Serhiyi, Ukraine	1	1	2017-08-24 08:55:41
193.201.224.218	Serhiyi, Ukraine	6	1	2017-08-23 09:04:24
195.22.127.83	Poland	9	1	2017-08-24 10:36:00
218.65.30.53	Nanchang, China	2	1	2017-08-24 10:14:01
221.163.38.70	Republic of Korea	1	1	2017-08-22 08:39:19
222.186.61.176	Nanjing, China	1	1	2017-08-22 10:03:00
58.101.149.188	Hangzhou, China	1	1	2017-08-22 08:35:27
91.197.232.103	Russia	15	1	2017-08-24 09:04:00
101.108.234.136	Thailand	1	0	2017-08-24 08:37:35
103.207.39.196	N/A	4	0	2017-08-22 10:43:28
119.193.140.180	Republic of Korea	1	0	2017-08-23 09:50:54
121.18.238.106	Hebei, China	1	N/A	2017-08-23 10:17:07
121.18.238.119	Hebei, China	1	N/A	2017-08-23 10:29:47
121.18.238.125	Hebei, China	1	N/A	2017-08-23 09:49:45
121.18.238.28	Hebei, China	2	N/A	2017-08-23 10:26:21
138.197.111.58	Wilmington, United States	1	N/A	2017-08-22 09:22:09
139.162.122.110	United States	2	0	2017-08-23 08:57:28
141.212.122.208	Ann Arbor, United States	1	N/A	2017-08-22 08:35:39
171.78.177.101	India	1	0	2017-08-23 09:23:03
177.11.50.67	Brazil	1	N/A	2017-08-23 09:48:54
180.175.55.213	Shanghai, China	1	0	2017-08-24 09:12:21
181.41.214.175	Brazil	1	N/A	2017-08-24 10:26:45
185.100.86.128	N/A	1	0	2017-08-22 10:53:25
190.112.39.26	Coronel Pringles, Argentina	2	N/A	2017-08-22 10:38:44
200.73.204.20	Quito, Ecuador	1	0	2017-08-23 09:11:59
221.194.44.212	Hebei, China	2	N/A	2017-08-23 10:51:33
221.194.47.224	Hebei, China	1	N/A	2017-08-23 10:21:21
221.194.47.242	Hebei, China	2	N/A	2017-08-23 10:51:09

captura kippo-grah cowrie 1

Tenemos activado javascript, por lo que podemos ver dos cosas, por un lado una representación en gif de los comandos según fueron insertados, y por otro, podemos incluso localizar hasta cierto punto en un mapa su localización.



TTY log

IP: **195.22.127.83** on 2017-08-24 10:36:00

Playing session: **bb8f4aa7409c**

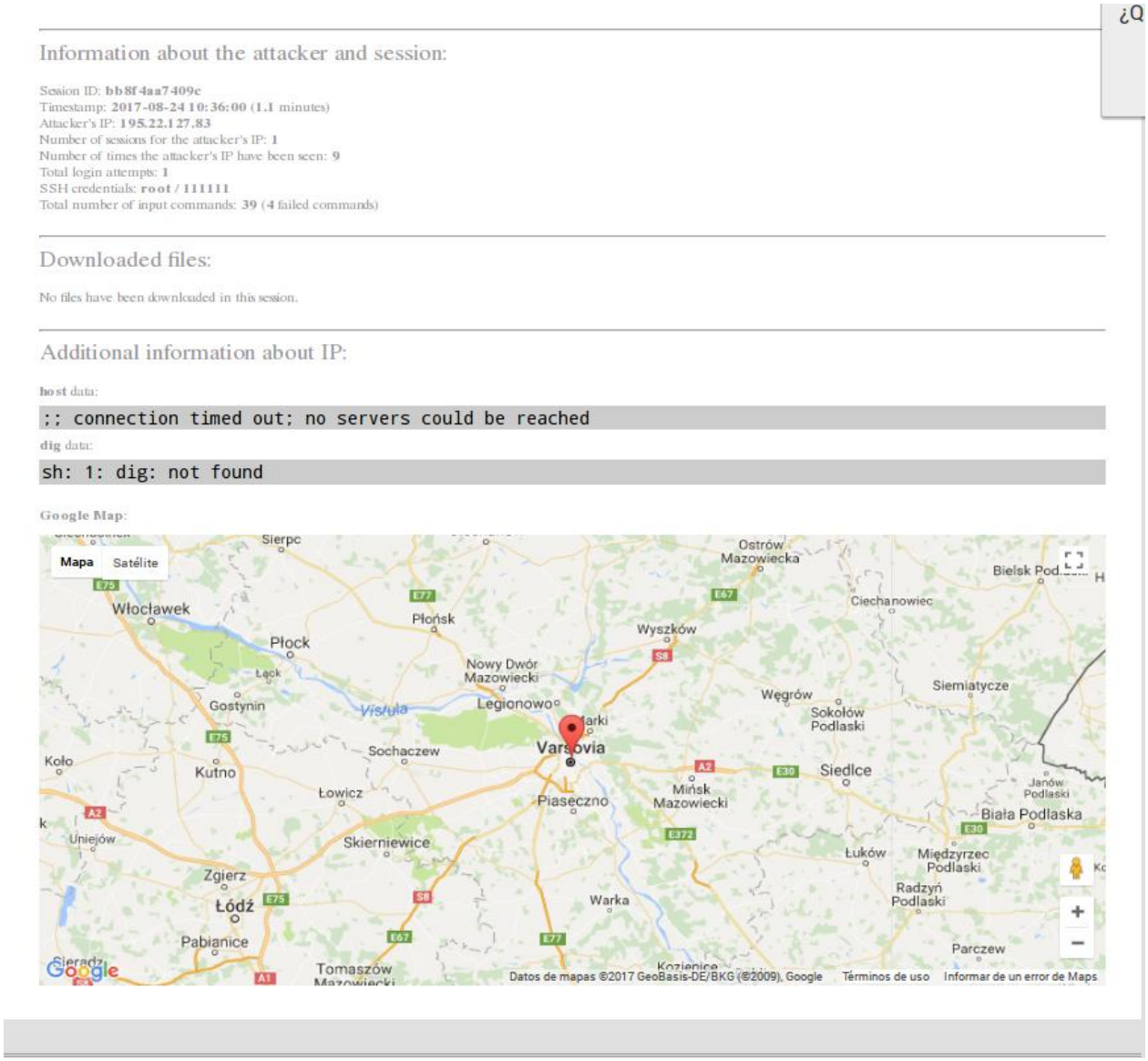
```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

root@hpIOCProServer04:~# /gweerwe323f
bash: /gweerwe323f: command not found
root@hpIOCProServer04:~#
root@hpIOCProServer04:~# sudo /bin/sh
root@hpIOCProServer04:~# root@hpIOCProServer04:~#
root@hpIOCProServer04:~# /bin/busybox cp; /gweerwe323f
cp: missing file operand
Try `cp --help' for more information.
bash: /gweerwe323f: command not found
root@hpIOCProServer04:~#
root@hpIOCProServer04:~#   mount ;/gweerwe323f
/dev/sda1 on / type ext3 (rw,errors=remount-ro)
tmpfs on /lib/init/rw type tmpfs (rw,nosuid,mode=0755)
proc on /proc type proc (rw,noexec,nosuid,nodev)
sysfs on /sys type sysfs (rw,noexec,nosuid,nodev)
udev on /dev type tmpfs (rw,mode=0755)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
devpts on /dev/pts type devpts (rw,noexec,nosuid,gid=5,mode=620)
bash: /gweerwe323f: command not found
root@hpIOCProServer04:~#
root@hpIOCProServer04:~#   echo -e '\x47\x_
```

Information about the attacker and session:

Session ID: **bb8f4aa7409c**
Timestamp: **2017-08-24 10:36:00** (**1.1** minutes)
Attacker's IP: **195.22.127.83**
Number of sessions for the attacker's IP: **1**
Number of times the attacker's IP have been seen: **9**
Total login attempts: **1**
SSH credentials: **root / 111111**
Total number of input commands: **39** (**4** failed commands)

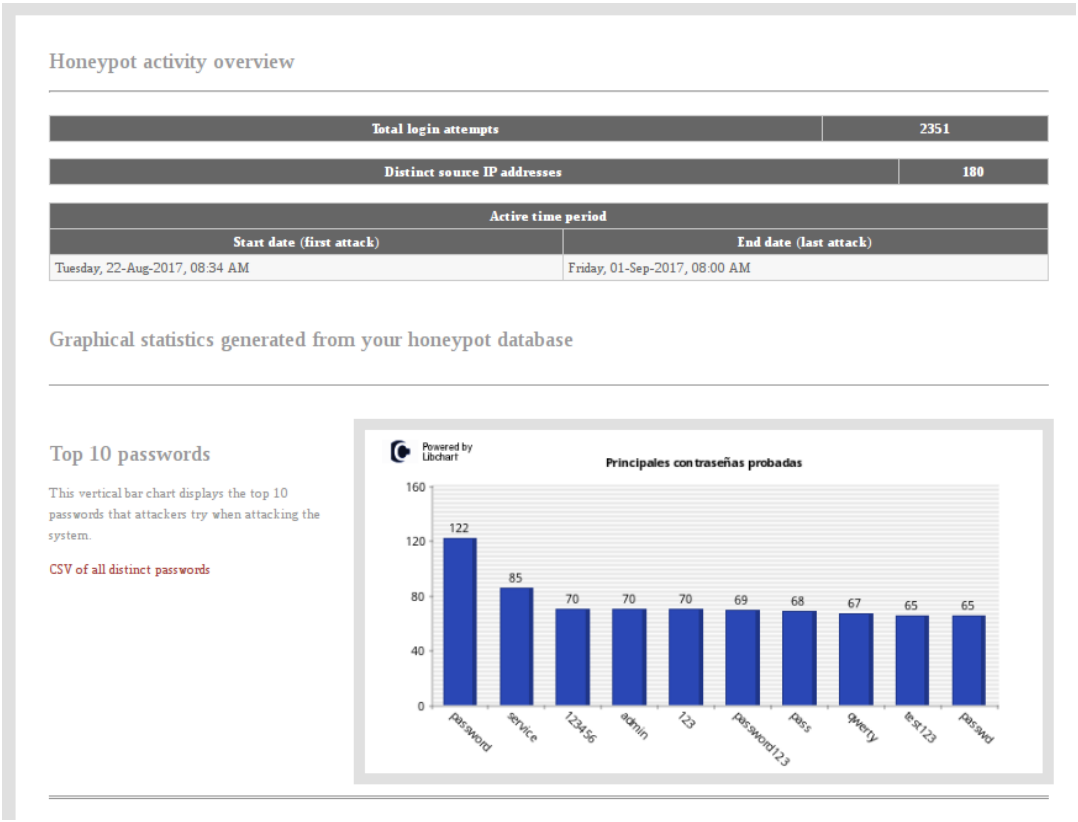
captura kippo-grah cowrie 2



captura kippo-grah cowrie 3

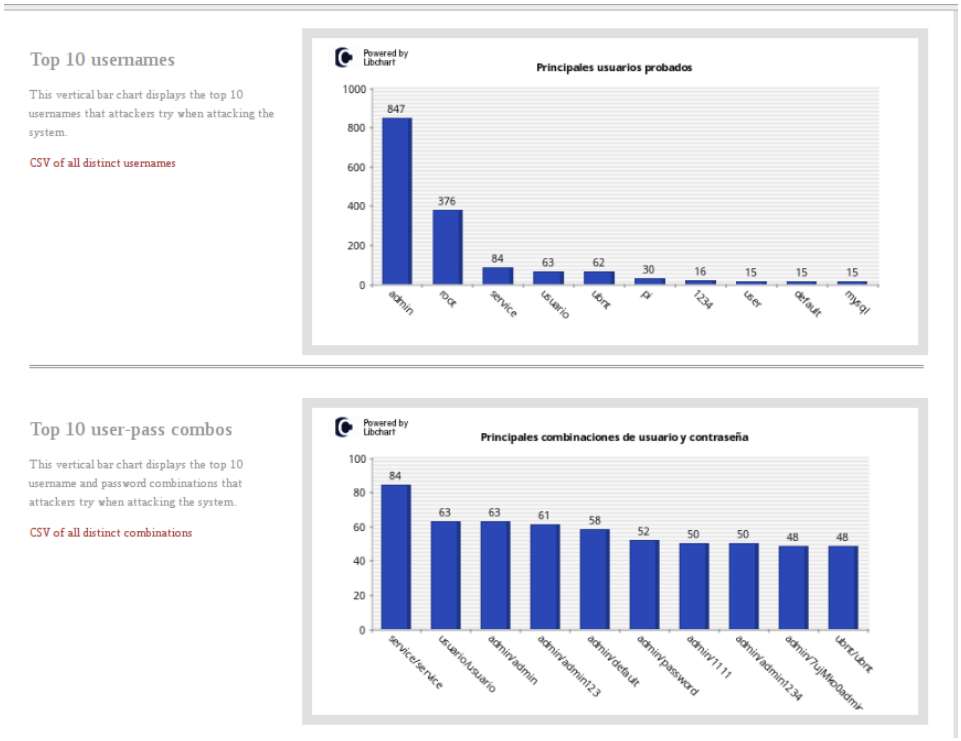
4.2.4. Informe cowrie del 22 de agosto al 1 de septiembre

Durante la franja de tiempo comprendida entre los días 22 de agosto y el 1 de septiembre se ha puesto en funcionamiento el honeypot cowrie.



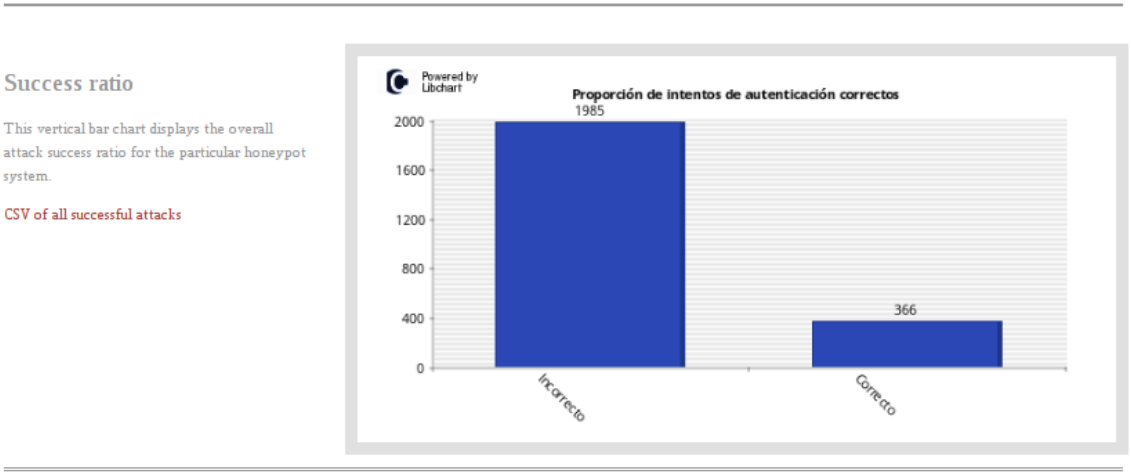
captura kippo-grah cowrie 4

Como podemos comprobar, no hay mucha novedad, las contraseñas más usadas por los atacantes son muy simples, lo frecuente es que el usuario medio no se complique a la hora de poner una contraseña.



captura kippo-grah cowrie 5

Podemos observar por la imagen de arriba los nombres de usuario más utilizados no son tampoco un gran misterio. Los más utilizados suelen coincidir con los que se tiene por defecto en un sistema, o ser lo más simple posible.

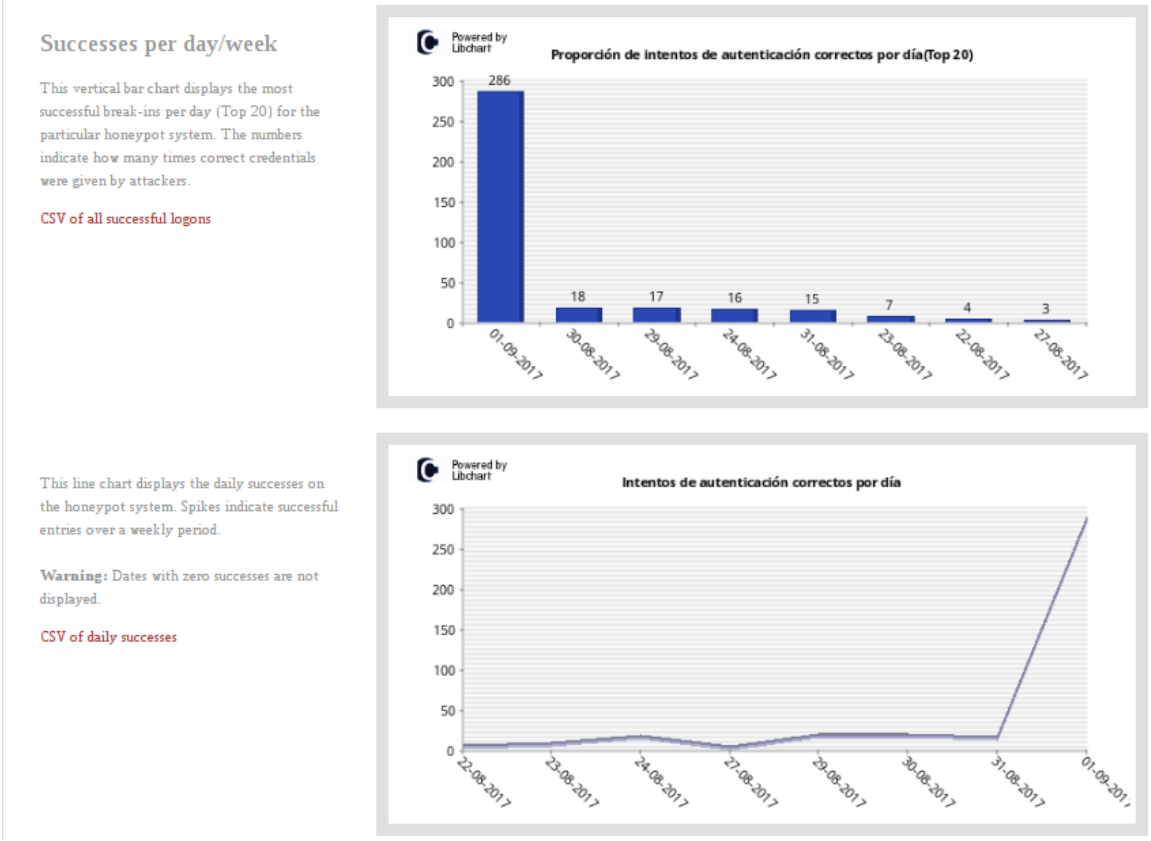


captura kippo-grah cowrie 6

Comprobamos por la imagen anterior y la siguiente, solo es cuestión de ir probando pues no pocas veces o no se cambia la contraseña o si se cambia suele ser a una no muy buena.

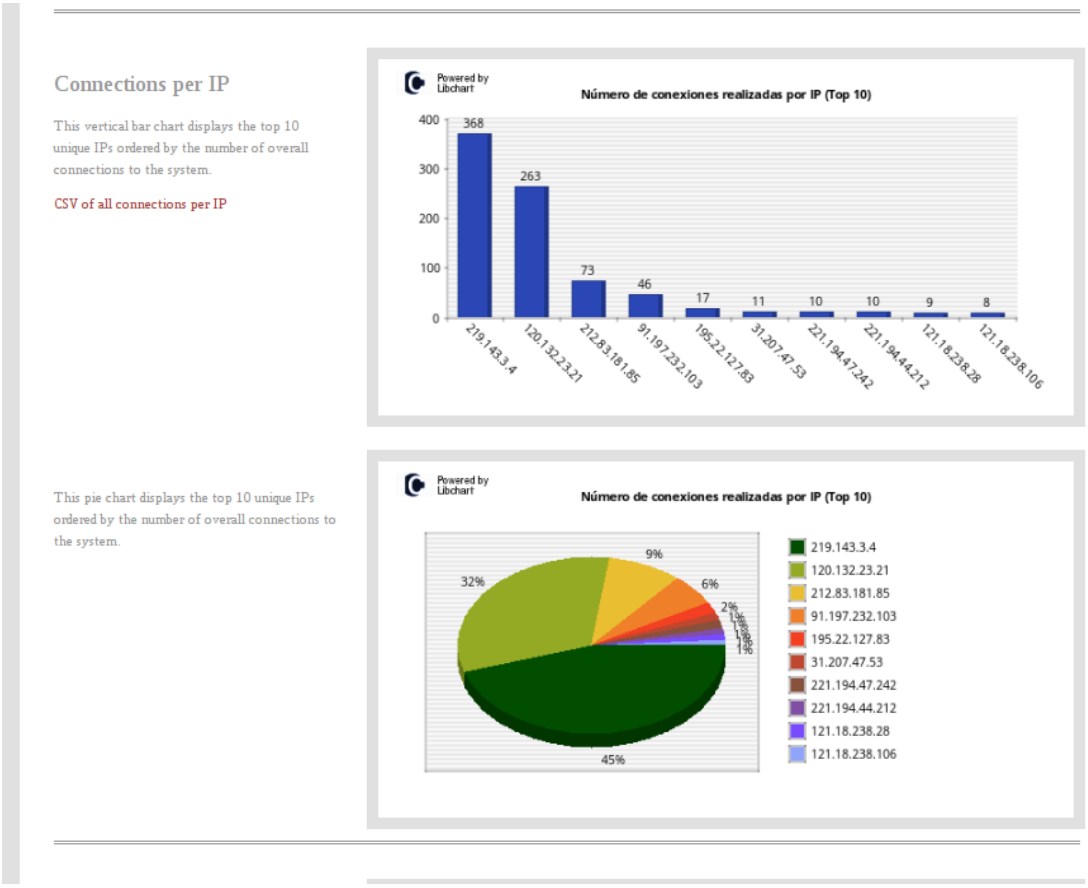
captura kippo-grah cowrie 7

Como no es habitual cambiar el nombre del usuario root, no es extraño que no sea solo el segundo más usado, sino el que más aciertos tiene.



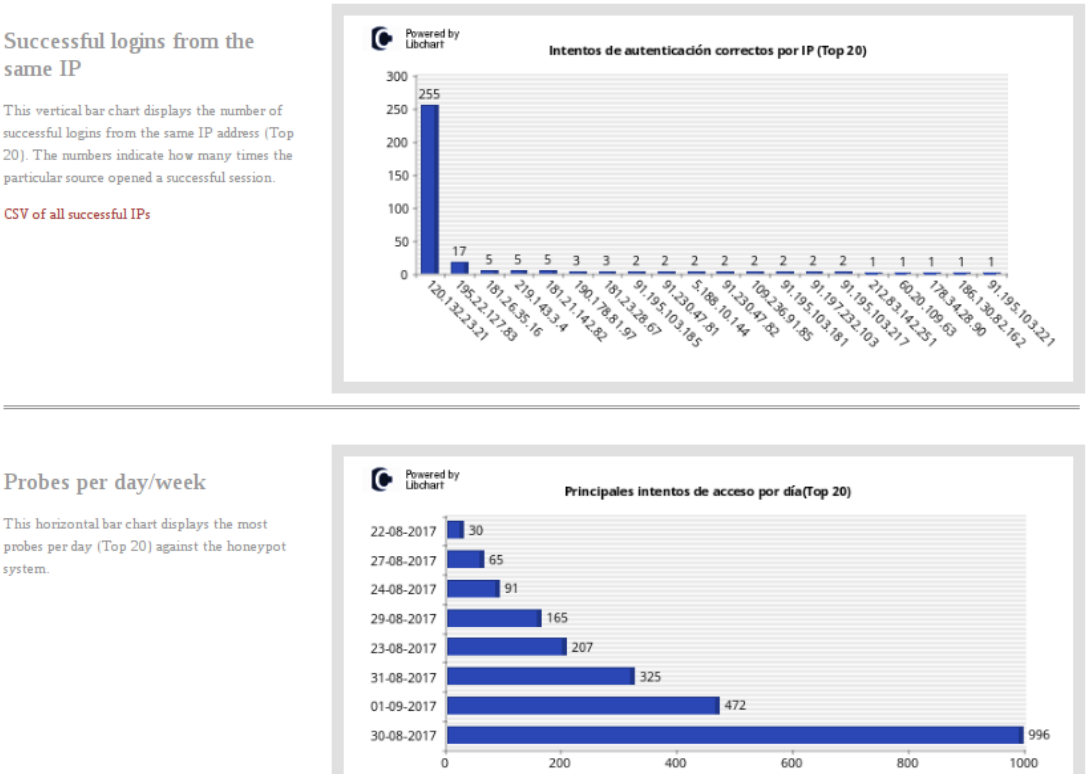
captura kippo-grah cowrie 8

Observamos además, por la imagen de arriba, cuanto más hemos puesto nuestro honeypot más fácil es que vuelva a ser visitado por algún visitante extraño.



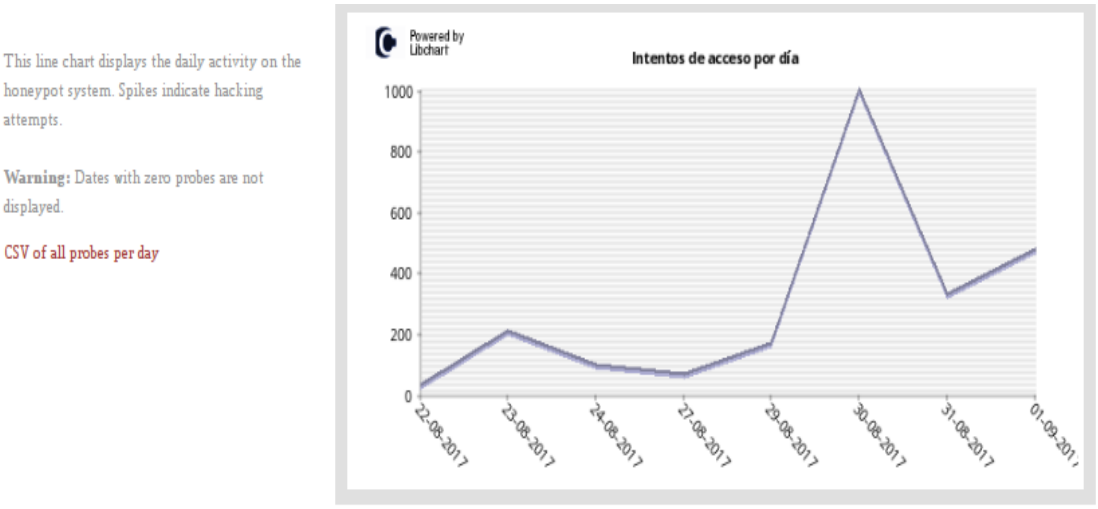
captura kippo-grah cowrie 9

En las gráficas de la imagen de arriba podemos ver cuáles son las ips que más han intentado acceder a nuestro honeypot, aunque no necesariamente con éxito como podemos ver. Podemos observar la insistencia de la dirección 120.132.23.21 que es la que mayor éxito tiene y es la segunda en mayor número de intentos.



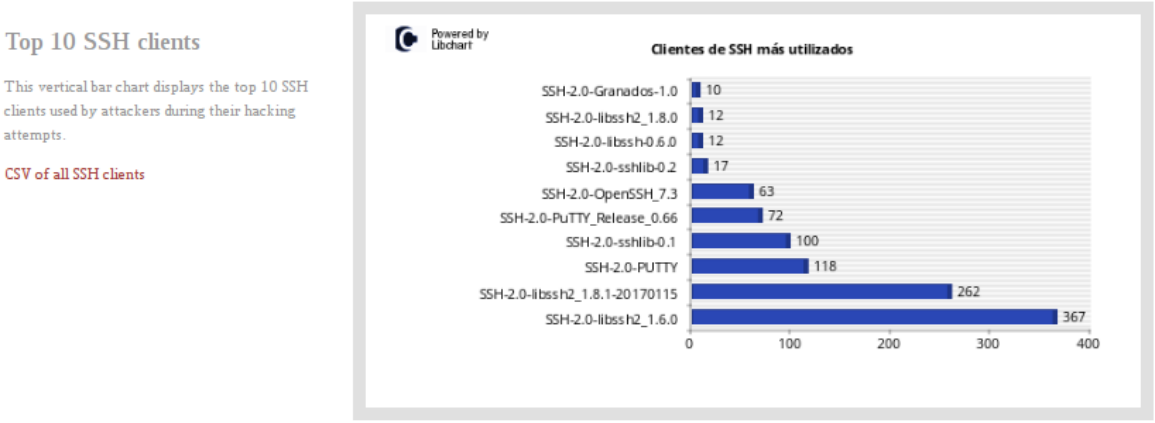
captura kippo-grah cowrie 10

Vemos que el dia 30 de agosto fue el dia con el mayor número de ataques.



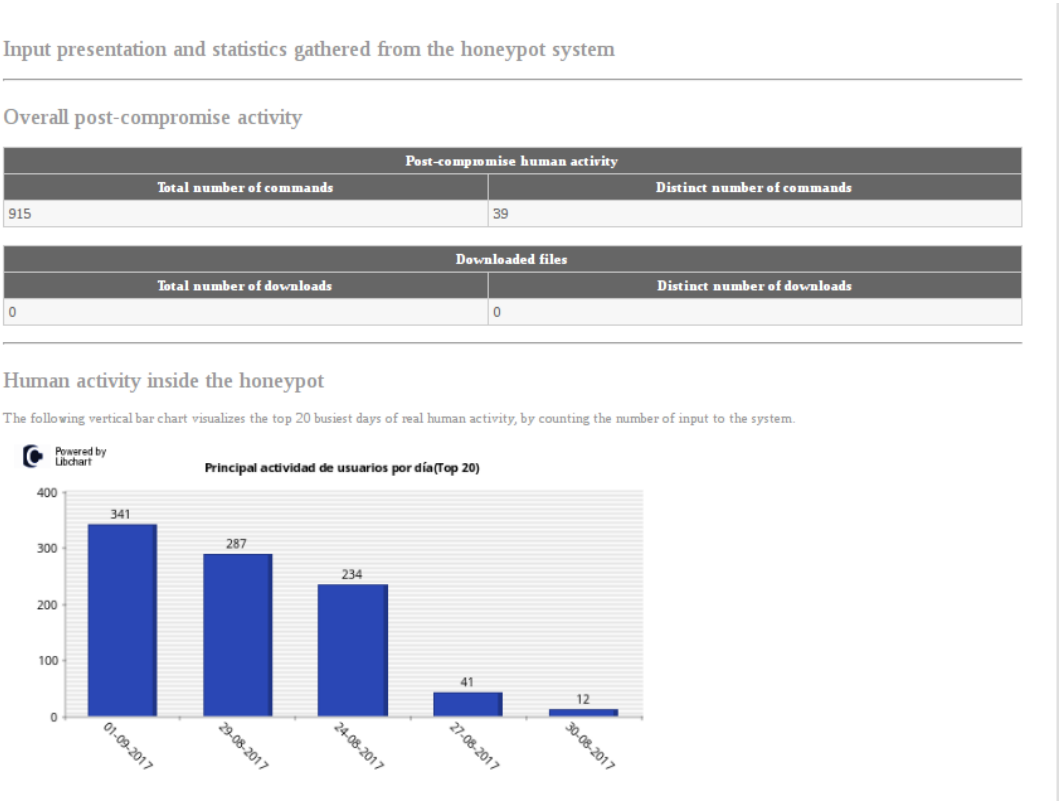
captura kippo-grah cowrie 11

La siguiente imagen nos muestra cuales han sido los clientes ssh más utilizados por los atacantes para acceder a nuestra máquina.



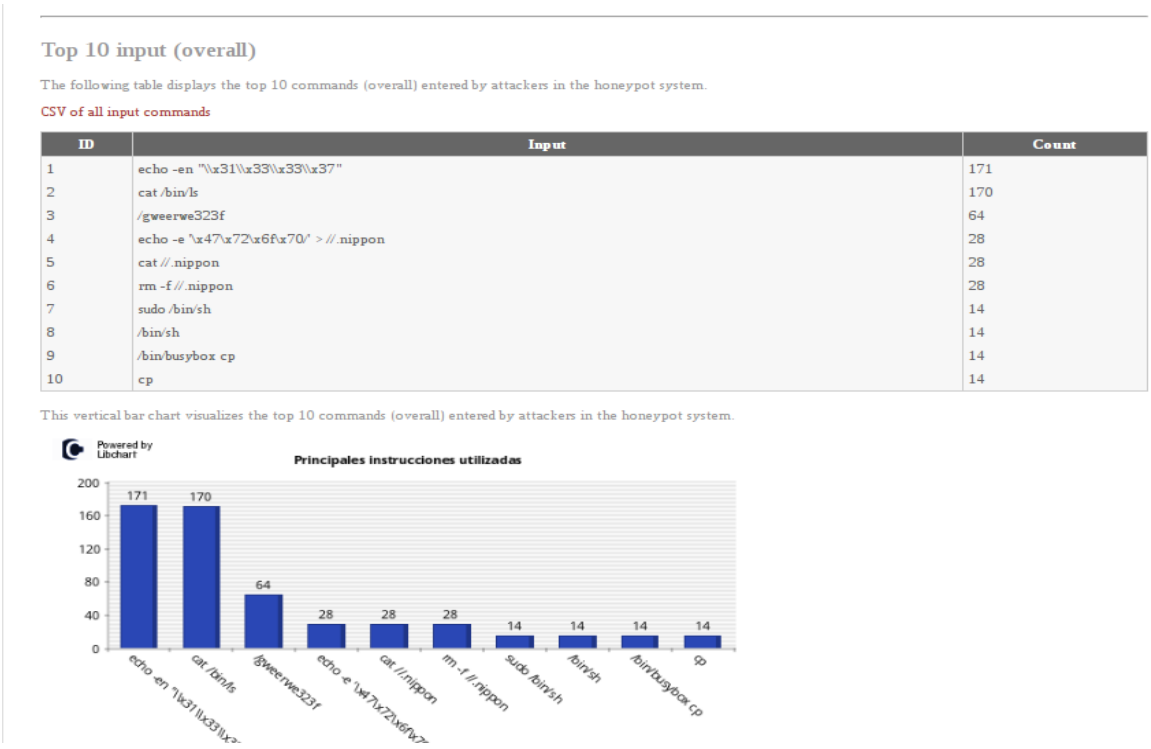
captura kippo-grah cowrie 12

Ahora pasaremos a ver algunos datos relacionados con la actividad del honeypot una vez que han conseguido entrar:



captura kippo-grah cowrie 13

En la imagen superior podemos comprobar que el día donde más actividad hubo fue el día 1 de septiembre (aunque previamente vimos que el día que más se intentó acceder fue el 30 de agosto).



Top 10 input (overall)

The following table displays the top 10 commands (overall) entered by attackers in the honeypot system.

CSV of all input commands

ID	Input	Count
1	echo -en "\\x31\\x33\\x33\\x37"	171
2	cat /bin/ls	170
3	/gweerwe323f	64
4	echo -e '\x47\x72\x6f\x70/' > //.nippon	28
5	cat //.nippon	28
6	rm -f //.nippon	28
7	sudo /bin/sh	14
8	/bin/sh	14
9	/bin/busybox cp	14
10	cp	14

This vertical bar chart visualizes the top 10 commands (overall) entered by attackers in the honeypot system.

captura kippo-grah cowrie 14

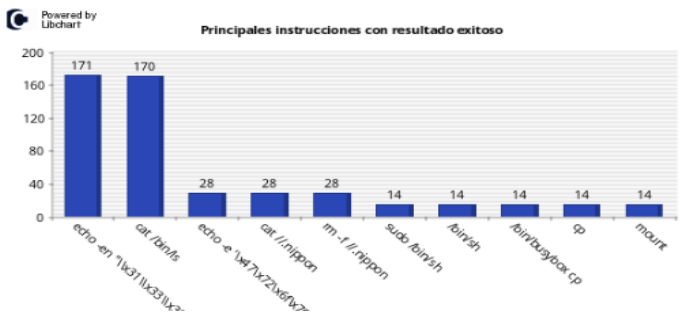
Top 10 successful input

The following table displays the top 10 successful commands entered by attackers in the honeypot system.

CSV of all successful commands

ID	Input (success)	Count
1	echo -en "\\x31\\x33\\x33\\x37"	171
2	cat /bin/ls	170
3	echo -e '\x47\x72\x6f\x70/' > //.nippon	28
4	cat //.nippon	28
5	rm -f //.nippon	28
6	sudo /bin/sh	14
7	/bin/sh	14
8	/bin/busybox cp	14
9	cp	14
10	mount	14

This vertical bar chart visualizes the top 10 successful commands entered by attackers in the honeypot system.



captura kippo-grah cowrie 15

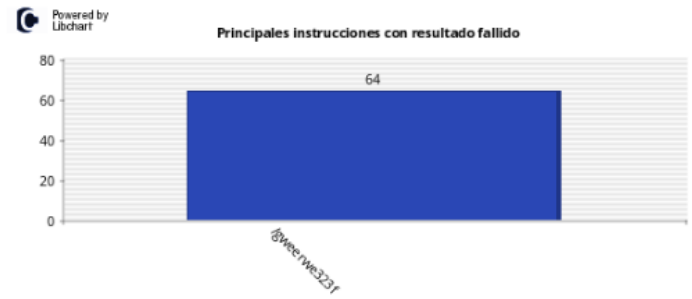
Top 10 failed input

The following table displays the top 10 failed commands entered by attackers in the honeypot system.

CSV of all failed commands

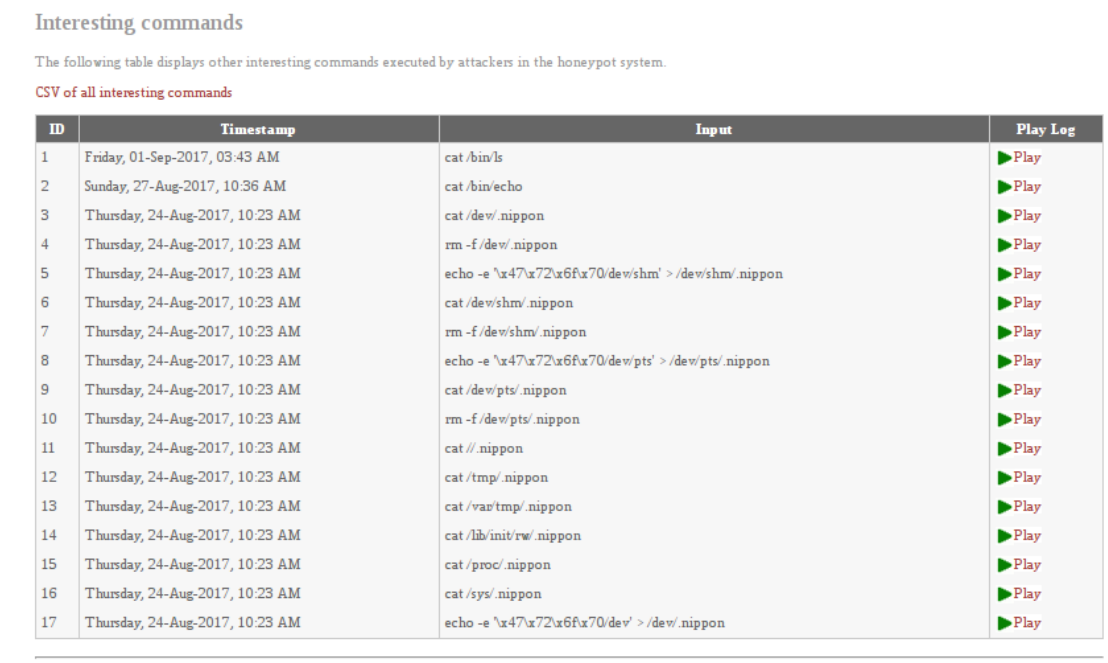
ID	Input (fail)	Count
1	/gweerwe323f	64

This vertical bar chart visualizes the top 10 failed commands entered by attackers in the honeypot system.



captura kippo-grah cowrie 16

En las tres imágenes anteriores podemos ver cuáles son los comandos más utilizados, cuales los que han tenido éxito y cuáles no. La mayoría de los comandos tienen como objetivo conocer que permisos tiene mediante la creación de un archivo o la lectura de este.



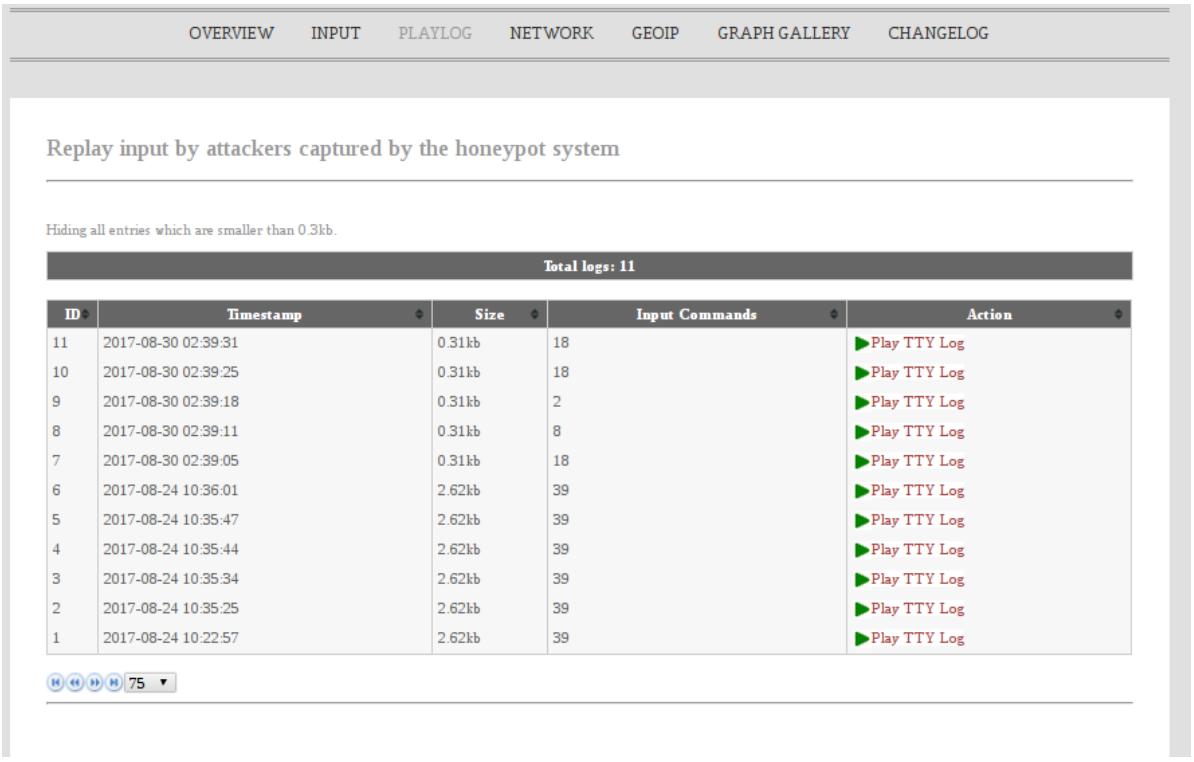
Interesting commands

The following table displays other interesting commands executed by attackers in the honeypot system.

CSV of all interesting commands

ID	Timestamp	Input	Play Log
1	Friday, 01-Sep-2017, 03:43 AM	cat /bin/ls	Play
2	Sunday, 27-Aug-2017, 10:36 AM	cat /bin/echo	Play
3	Thursday, 24-Aug-2017, 10:23 AM	cat /dev/.nippon	Play
4	Thursday, 24-Aug-2017, 10:23 AM	rm -f /dev/.nippon	Play
5	Thursday, 24-Aug-2017, 10:23 AM	echo -e '\x47\x72\x6f\x70/dev/shm' > /dev/shm/.nippon	Play
6	Thursday, 24-Aug-2017, 10:23 AM	cat /dev/shm/.nippon	Play
7	Thursday, 24-Aug-2017, 10:23 AM	rm -f /dev/shm/.nippon	Play
8	Thursday, 24-Aug-2017, 10:23 AM	echo -e '\x47\x72\x6f\x70/dev/pts' > /dev/pts/.nippon	Play
9	Thursday, 24-Aug-2017, 10:23 AM	cat /dev/pts/.nippon	Play
10	Thursday, 24-Aug-2017, 10:23 AM	rm -f /dev/pts/.nippon	Play
11	Thursday, 24-Aug-2017, 10:23 AM	cat //.nippon	Play
12	Thursday, 24-Aug-2017, 10:23 AM	cat /tmp/.nippon	Play
13	Thursday, 24-Aug-2017, 10:23 AM	cat /var/tmp/.nippon	Play
14	Thursday, 24-Aug-2017, 10:23 AM	cat /lib/init/rw/.nippon	Play
15	Thursday, 24-Aug-2017, 10:23 AM	cat /proc/.nippon	Play
16	Thursday, 24-Aug-2017, 10:23 AM	cat /sys/.nippon	Play
17	Thursday, 24-Aug-2017, 10:23 AM	echo -e '\x47\x72\x6f\x70/dev' > /dev/.nippon	Play

captura kippo-grah cowrie 17



OVERVIEW INPUT PLAYLOG NETWORK GEOIP GRAPH GALLERY CHANGELOG

Replay input by attackers captured by the honeypot system

Hiding all entries which are smaller than 0.3kb.

Total logs: 11

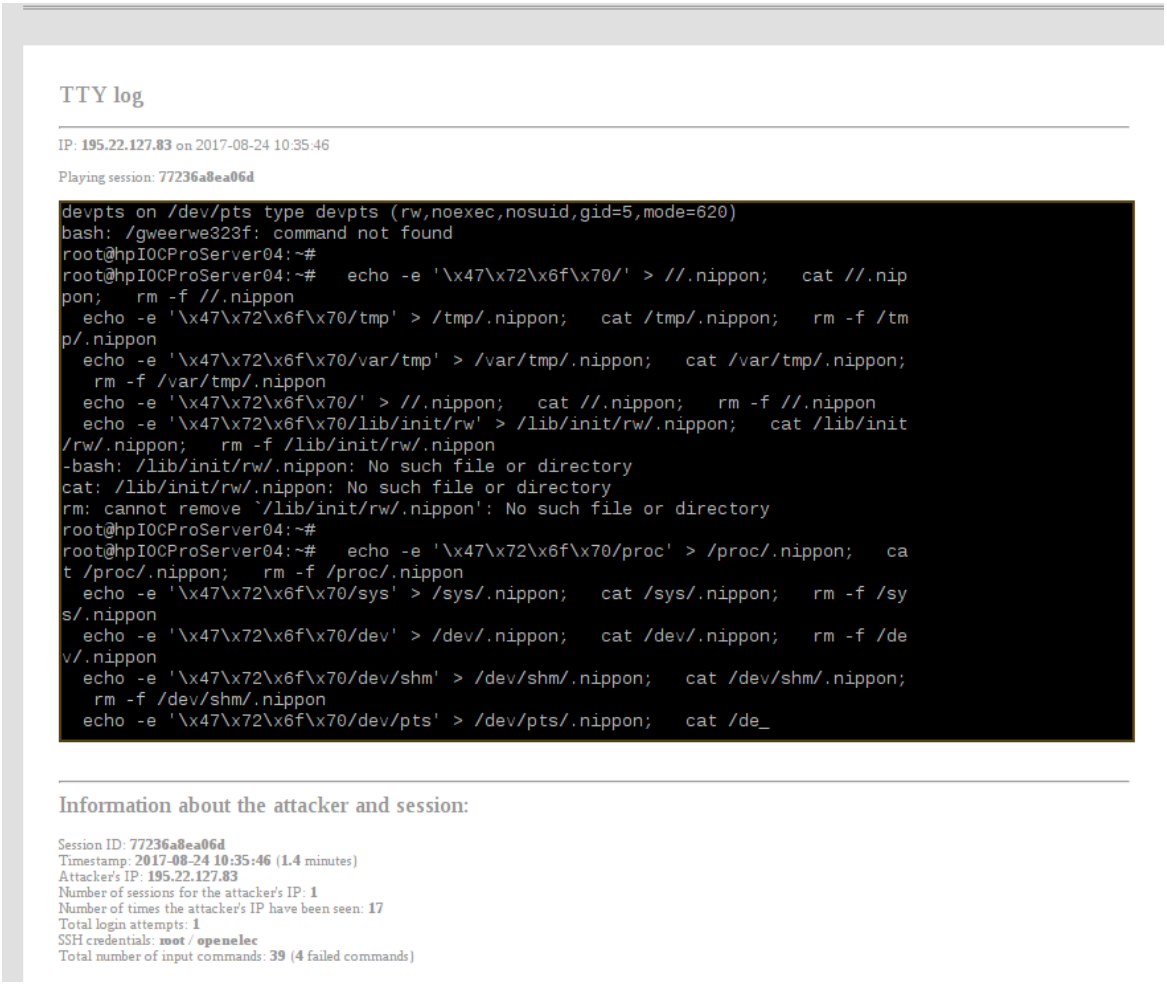
ID	Timestamp	Size	Input Commands	Action
11	2017-08-30 02:39:31	0.31kb	18	Play TTY Log
10	2017-08-30 02:39:25	0.31kb	18	Play TTY Log
9	2017-08-30 02:39:18	0.31kb	2	Play TTY Log
8	2017-08-30 02:39:11	0.31kb	8	Play TTY Log
7	2017-08-30 02:39:05	0.31kb	18	Play TTY Log
6	2017-08-24 10:36:01	2.62kb	39	Play TTY Log
5	2017-08-24 10:35:47	2.62kb	39	Play TTY Log
4	2017-08-24 10:35:44	2.62kb	39	Play TTY Log
3	2017-08-24 10:35:34	2.62kb	39	Play TTY Log
2	2017-08-24 10:35:25	2.62kb	39	Play TTY Log
1	2017-08-24 10:22:57	2.62kb	39	Play TTY Log

75

captura kippo-grah cowrie 18

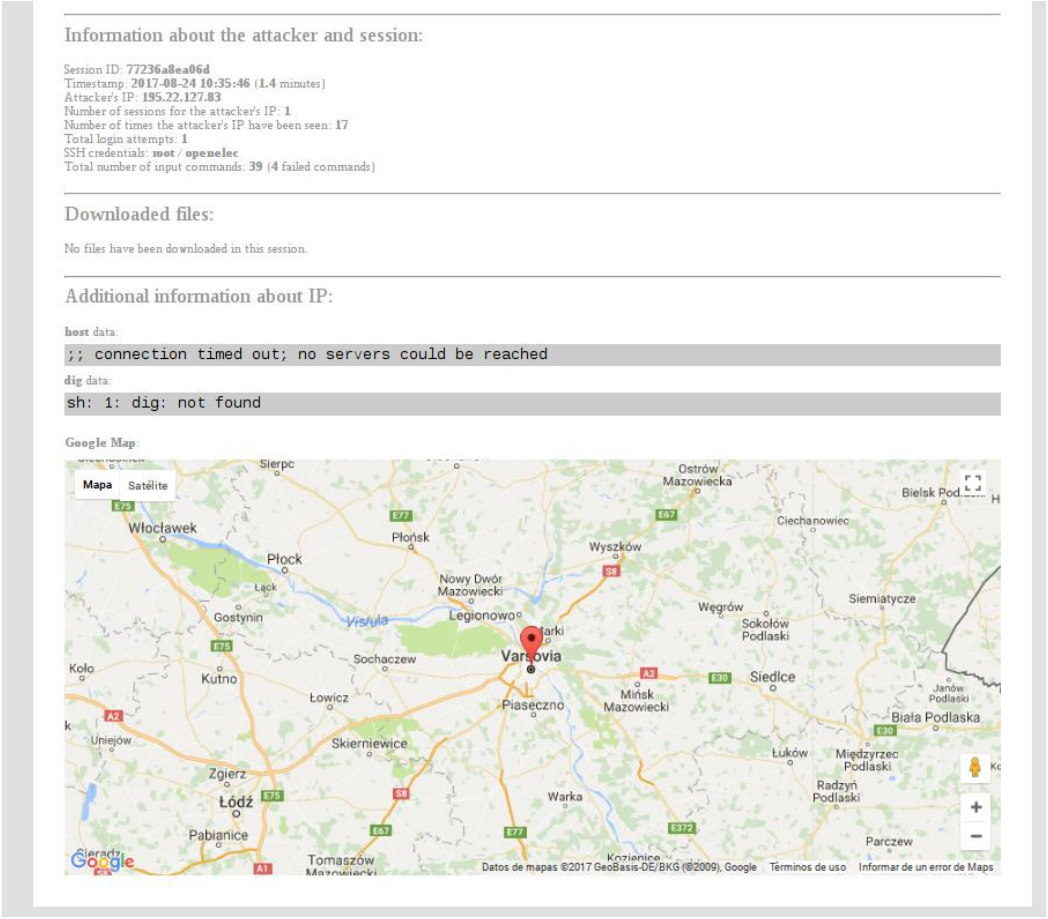
En la ventana playlog, podemos elegir algunas de las interacciones hechas en nuestra máquina, para reproducirlas y ver algunos datos de una interacción en particular mas detenidamente.

Un ejemplo:



captura kippo-grah cowrie 19

En la sección de kippo-graph relativa a la imagen superior podemos ir viendo cómo se van introduciendo los comandos. Como podemos comprobar se limita a volcar el contenido de la echo a un archivo mediante > para despues comprobar si puede leerlo.



captura kippo-grah cowrie 20

Debido a que kippo-graph hace uso de geoPlugin, a veces es capaz de geolocalizar la posición de una determinada IP.

Total identified IP addresses: 180

IP address	Geolocation	Sessions count	Success	Last seen
107.155.21.179	Herndon, United States	1	1	2017-09-01 01:34:59
109.236.91.85	Netherlands	2	1	2017-08-24 09:32:57
112.4.81.93	China	3	1	2017-08-31 21:23:37
112.81.58.45	Nanjing, China	1	1	2017-08-31 23:25:18
115.199.226.99	Hangzhou, China	1	1	2017-08-31 21:54:45
115.237.181.9	Shaoxing, China	1	1	2017-08-29 23:13:09
118.100.2.214	Malaysia	1	1	2017-08-31 22:52:59
120.132.23.21	Chaoyang, China	263	1	2017-09-01 05:34:27
122.162.70.77	New Delhi, India	1	1	2017-08-31 18:38:30
123.112.137.177	Beijing, China	1	1	2017-09-01 06:08:26
123.169.200.30	Jinan, China	1	1	2017-08-31 18:20:31
125.123.156.118	Jiaxing, China	1	1	2017-08-31 21:37:11
138.219.254.28	N/A	1	1	2017-09-01 07:00:52
14.47.125.208	Republic of Korea	1	1	2017-09-01 03:50:22
140.115.87.33	Taipei, Taiwan	1	1	2017-08-23 10:17:59
151.80.26.34	France	1	1	2017-08-30 01:41:17
175.6.27.205	Changsha, China	1	1	2017-09-01 04:57:25
177.221.107.193	Cuiabá, Brazil	1	1	2017-09-01 03:57:39
178.34.28.90	Russia	1	1	2017-08-24 10:22:51
178.44.149.37	Chkalov, Russia	1	1	2017-09-01 01:34:51
179.36.218.107	Argentina	1	1	2017-08-30 01:01:30
181.21.142.82	Ushuaia, Argentina	6	1	2017-08-30 04:37:49
181.211.182.149	Quito, Ecuador	1	1	2017-08-23 10:27:18
181.23.28.67	Argentina	5	1	2017-08-29 08:39:59
181.26.35.16	Argentina	5	1	2017-08-24 10:35:50
182.44.61.250	Jinan, China	1	1	2017-08-27 11:21:04
183.154.164.208	Jinhua, China	1	1	2017-08-29 23:21:16
186.119.173.57	Bogotá, Colombia	1	1	2017-08-29 23:22:52
186.129.160.244	Argentina	1	1	2017-08-31 20:17:46
186.130.82.162	Argentina	1	1	2017-08-23 09:27:24
190.152.193.54	Quito, Ecuador	1	1	2017-08-31 21:32:17
190.178.81.97	Río Gallegos, Argentina	5	1	2017-08-30 03:24:02
190.235.216.63	Peru	1	1	2017-08-29 09:29:13
190.51.101.127	Moron, Argentina	1	1	2017-08-31 21:14:40
193.201.224.206	Serhiyi, Ukraine	1	1	2017-08-24 08:55:41
193.201.224.218	Serhiyi, Ukraine	6	1	2017-08-23 09:04:24
195.22.127.83	Poland	17	1	2017-08-29 23:22:58
201.254.189.133	Argentina	2	1	2017-08-30 01:54:27
212.83.142.251	France	6	1	2017-08-30 01:50:28
218.109.238.45	Hangzhou, China	1	1	2017-09-01 02:26:54
218.65.30.53	Nanchang, China	2	1	2017-08-24 10:14:01
219.143.3.4	Beijing, China	368	1	2017-08-30 03:13:34
219.74.10.118	Singapore, Singapore	1	1	2017-08-31 19:11:55
221.163.38.70	Republic of Korea	1	1	2017-08-22 08:39:19
222.186.61.176	Nanjing, China	1	1	2017-08-22 10:03:00
24.43.146.106	Los Angeles, United States	3	1	2017-08-31 17:47:04
27.187.224.216	Hebei, China	1	1	2017-08-29 11:05:39
27.19.1.251	Wuhan, China	1	1	2017-08-31 19:44:31
27.54.162.253	India	1	1	2017-08-27 10:36:39
31.148.193.2	India	1	1	2017-08-31 19:29:13

captura kippo-grah cowrie 21

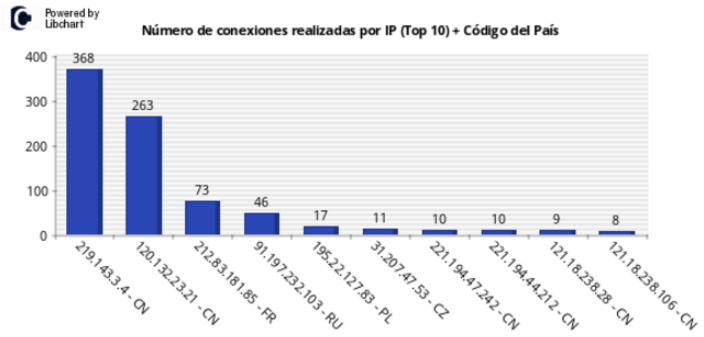
La imagen superior nos muestra algunos de los datos sobre los atacantes, tales como su dirección ip, el lugar desde el que nos atacan, la cantidad de intentos, cuántos de esos intentos han tenido éxito y la fecha del último. Como podemos comprobar la procedencia de los ataques está muy repartida.

Geolocation information gathered from the **top 10** IP addresses probing the system

The following table displays the top 10 IP addresses connected to the system (ordered by volume of connections).

ID	IP Address	Probes	City	Region	Country Name	Code	Latitude	Longitude	Hostname	IP Lookup
1	219.143.3.4	368	Beijing	Beijing Shi	China	CN	39.9289	116.3883	219.143.3.4	
2	120.132.23.21	263	Chaoyang	Liaoning	China	CN	41.5703	120.4586	120.132.23.21	
3	212.83.181.85	73			France	FR	48.86	2.35	212-83-181-85.rev.poneytelecom.eu	
4	91.197.232.103	46			Russia	RU	55.75	37.6166	91.197.232.103	
5	195.22.127.83	17			Poland	PL	52.2333	21.0167	195.22.127.83	
6	31.207.47.53	11			Czech Republic	CZ	50.0833	14.4167	31.207.47.53	
7	221.194.47.242	10	Hebei	Hebei	China	CN	39.8897	115.275	221.194.47.242	
8	221.194.44.212	10	Hebei	Hebei	China	CN	39.8897	115.275	221.194.44.212	
9	121.18.238.28	9	Hebei	Hebei	China	CN	39.8897	115.275	121.18.238.28	
10	121.18.238.106	8	Hebei	Hebei	China	CN	39.8897	115.275	121.18.238.106	

The following vertical bar chart visualizes the top 10 IPs ordered by the number of connections to the system.
Notice the two-letter country code to after each IP get a quick view of the locations where the attacks are coming from.

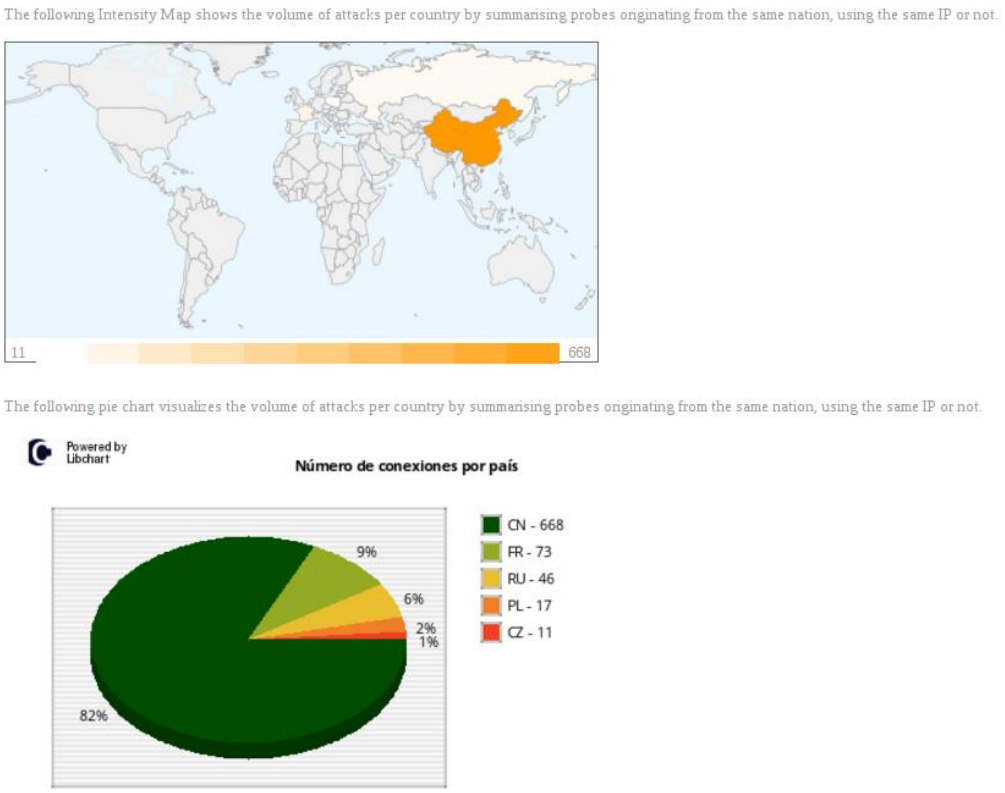


captura kippo-grah cowrie 22

En la sección “Geolocation”, podemos ver en mas detalle algunos de los datos de los 10 atacantes mas exitosos en nuestra máquina. Como podemos observar, la mayoría son de China.

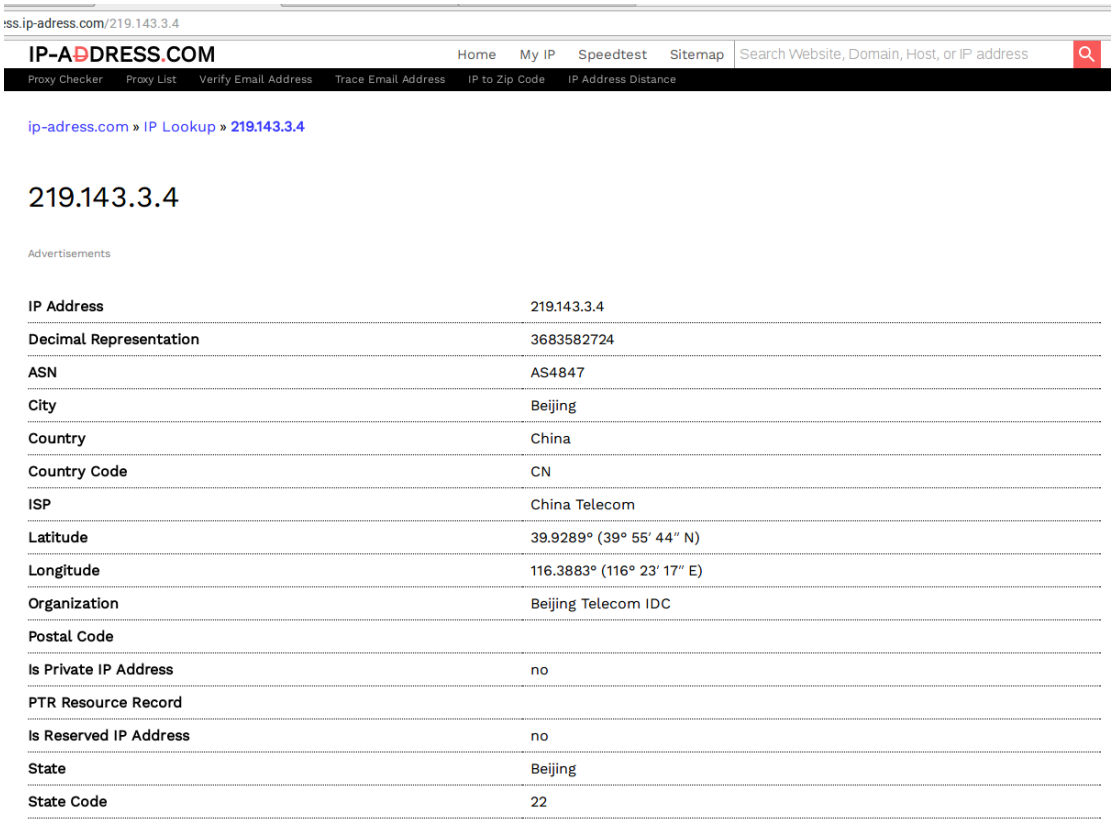
The following zoomable world map marks the geographic locations of the top 10 IPs according to their latitude and longitude values. Click on them to get the full information available from the database.

captura kippo-grah cowrie 23

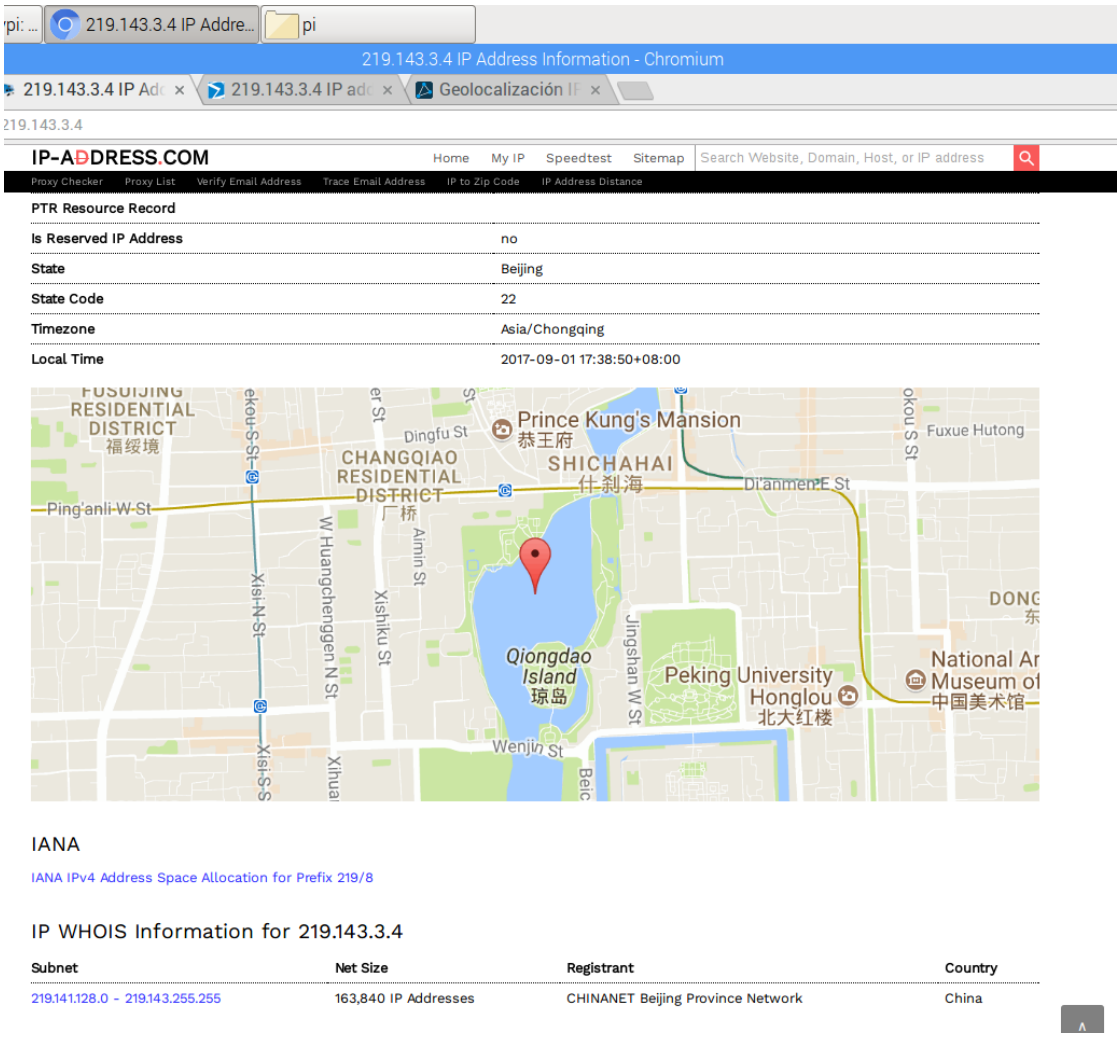


captura kippo-grah cowrie 24

Si pinchamos en cualquiera de los iconos de la sección “IP Lookup” nos redirigirá a páginas con más información sobre esa dirección en particular. Un ejemplo:



captura kippo-grah cowrie 25



captura kippo-grah cowrie 26

4.3. Informe de análisis forense, auditorias y eventos

4.3.1. Conclusiones del informe del honeypot kippo del 17 de julio de 2017

De momento, lo único que podemos sacar como conclusión para mejorar la seguridad es en utilizar contraseñas mas seguras.

4.3.2. Conclusiones del informe del honeypot kippo de 18 de agosto de 2017

Esta vez hemos recopilado mas información que la primera vez, ya que esta vez si se consiguió acceder al honeypot. Junto a la conclusión a la que llegamos en el anterior punto, podemos destacar:

- Busca una aplicacion llamada busyBox la cual posee varias de las ordenes mas comúnes de UNIX en un solo ejecutable.
- Gweerwe323f es un bot cuyo patrón de ataque es similar a uno ya descubierto por la comunidad llamado Mirai.

4.3.3. Conclusiones del informe de captura cowrie del 24 de agosto

Nuevamente tenemos la visita de gweerwe323f, luego las conclusiones son parecidas a las de kippo el 18 de agosto.

4.3.4. Conclusiones del informe de captura cowrie del 22 de agosto al 1 de septiembre.

Además de lo ya visto, podemos destacar varias cosas:

- La mayoria de los ataques registrados son procedentes de china.
- Casi todas las intrusiones se han limitado a conocer la organización del sistema del honeypot. No se ha descargado ningún archivo mediantes las ordenes get o curl, luego tampoco hay malware que se haya podido observar.
- Lo que si se ha guardado en la carpeta dl es el contenido de los archivos donde ha volcado el resultado de la orden echo, con la que ha podido tantear los derechos de escritura.

5. Análisis de la experiencia y conclusiones

En esta sección recapitularemos que hemos querido hacer y que hemos conseguido, así como que podríamos hacer de cara a un futuro, o como podríamos mejorar, tambien comentaremos algunos de los fallos que nos hemos encontrado y como los hemos solucionado.

5.1. ¿Se han cumplido los objetivos propuestos para el trabajo?

Los objetivos a cumplir han sido los siguientes:

- **Proporcionar una herramienta para la detección, y análisis de accesos no autorizados que puedan afectar a la seguridad de equipos conectados en red.**
- Junto a las herramientas honeypots, se ha añadido algunas mas que permiten el análisis y filtrado de paquetes. Esto nos permite tener una mayor información sobre el origen de un ataque.
- **Diseñar dicha aplicación dentro del modelo "honeypot" para que resulte un medio aislado de detección de intrusiones, sin exponer a los recursos hardware de la red ni comprometer al resto de servicios software.**
- Aparte de las herramientas citadas, se ha dispuesto de una descripción de bajo que condiciones ha de ponerse la herramienta para que su actividad no afecte, o afecte lo mínimo posible, a la seguridad de la red.
- **Favorecer el trabajo del administrador del sistema con una aplicación que resulte ágil y sencilla de instalar, administrar y consultar.**
- La mayoría de las herramientas citadas se utilizan mediante líneas de comandos, sin embargo, algunas poseen de una interfaz web desde la que podemos consultar los datos de una manera mas intuitiva.
- **Extraer conclusiones y representarlas mediante la herramienta mas adecuada, de forma que ayuden a mejorar la seguridad en cualquier red informática.**

- La mayoria de las herramientas utilizadas en este trabajo provee de una interfaz gráfica que permite visualizar y comprender la información recogida de manera intuitiva.
- **Realizar una migración del sistema a un entorno hardware como pueda ser la tarjeta programable Raspberry Pi.**
- Conseguido, de hecho la mayor parte de este trabajo se ha llevado a cabo en una tarjeta Raspberry Pi 3 modelo B.

5.2. ¿Hemos logrado algo más aparte de los objetivos del proyecto?

Se han instalado dos herramientas honeypots, con la intención de realizar una comparación, Kippo y Cowrie. Ambas son versiones distintas del mismo tipo de honeypot.

Entre las diferencias mas destacables estan:

- El logeado en formato JSON, ademas del que ya tenia kippo, para facilitiar el proceso de registro en la gestión de soluciones ya que JSON puede ser leido por cualquier lenguaje de programación.
- Capacidad para desviar las conexiones SMTP a un honeypot dedicado a este fin.
- Soporte para los protocolos SFTP y SCP para la subida de archivos.
- Ademas de guardar los archivos que dejase el atacante con wget, es capaz de guardar los que se descargan bajo el comando curl.
- Hace uso de un entorno virtual para que su actividad afecte lo menos posible a la máquina.

Debido a que Cowrie es una versión avanzada de kippo, tiene pulidos muchos de sus fallos y bugs.

5.3. ¿Qué problemas han surgido? ¿Cómo los hemos solucionado?

Algunos de los problemas que han surgido estan relacionados con las propias herramientas y sus dependencias.

Por un lado tenemos el echo de que algunas (herramientas y/o dependencias), estuvieran obsoletas, esto nos plantea dos problemas, ser poco recomendables de utilizar y ser dificiles de encontrar. En algunos casos se descartó el uso de determinadas herramientas por el excesivo tiempo que pasó desde la última vez que fueron actualizadas. En otros casos, su mantenimiento se continuó bajo el nombre de otro proyecto y su repositorio y/o fuente fue cambiado lo que complicó la búsqueda.

En este punto se hace especial referencia a la página de **arch linux** que recopila una buena cantidad de las herramientas y dependencias relacionadas con este tipo de proyectos.

5.4. ¿Cómo se podría continuar este trabajo?

El tema de la seguridad informática toca muchísimos aspectos con los que podria expanderse este trabajo. Ya que este trabajo ha ido sobre el despliege de una herramienta honeypot ssh de media interacción, podría expandirse realizando estudios de otras herramientas que emulen otros servicios, también podría expandirse haciendo un estudio, y alguna demostración controlada, de algunos de los ataques mas comunes, ya que, una manera de prepararnos para un ataque es estudiarlo desde el punto de vista del atacante, no solo del defensor.

Se ha intentado instalar el honeypot Dionaea, que soporta una cantidad considerable de protocolos (ftp, http,sip,tftp,…) y nos permite quedarnos con una copia del malware utilizado por el atacante. No se ha podido instalar debido a un error de compilación al hacer el make durante el proceso de instalación, debido a una posible incompatibilidad entre el kernel y el software.

6. Bibliografía

- Definición de Honeypot:
  - Wikipedia. Honeypot:https://es.wikipedia.org/wiki/Honeypot
  - Canditraps.¿Que es un honeypot?: https://candytraps.wordpress.com/que-es-un-honeypot/
  - Scribd. HoneyPot paso a paso. Rafael Rueda. 27 de julio de 2013: https://es.scribd.com/document/256999905/Honeypot-Paso-a-Paso
  - Donde colocar un honeypot:
    - Wordpress. Honeypots: Herramientas de seguridad: https://honeypots.wordpress.com/
  - Tipos de honeypots:
    - Wh0s. Honeypots ii. Clasificación. 1 de octubre de 2014.http://wh0s.org/2014/10/01/honeypots-ii-clasificacion/
- Tipos de ataques
  - Ecured. Ataque de autenticación: https://www.ecured.cu/Ataque_de_autenticaci%C3%B3n
- Spoofing
  - Hacking ético. Hablemos de spoofing.Carlos Garcia ciyi. 28 de junio de 2010: https://hacking-etico.com/2010/08/26/hablemos-de-spoofing/
  - Undercode. Ataques de autenticación. 18 de febrero de 2010: https://underc0de.org/foro/hacking/ataques-de-autenticacion/
  - Noticia:
    - Diario BAE. Un sitio web estafa a estudiantes de todo el mundo.21 de julio de 2017. http://www.diariobae.com/article/details/187236/un-falso-sitio-web-de-una-universidad-britanica-pide-dinero-para-cursos
- Ataques de fuerza bruta:
  - Faqoff.¿Que es un ataque de fuerza bruta?. Juan García. 7 de mayo de 2013: http://faqoff.es/que-es-un-ataque-por-fuerza-bruta/
- Uso de diccionarios

    - WebTriplex. Que es y como protegerse de un ataque de diccionario: http://www.webtriplex.com/vernoticias.php?Id=81&ti=Que-es-y-como-protegerse-de-un-Ataque-de-Diccionario&pageNum_listado=2&totalRows_listado=122

- Exploits
    - Wikipedia. Exploit: https://es.wikipedia.org/wiki/Exploit
    - Seguridadpc. Concepto de exploit: http://www.seguridadpc.net/exploit.htm
    - Seguinfo. Exploit: http://www.segu-info.com.ar/malware/exploit.htm
    - Caso real:
        - Vandal. Detectan un grave fallo de seguridad en el motor source. Enrique García. 24 de julio de 2017: www.vandal.net/noticia/1350694406/detectan-un-grave-fallo-de-seguridad-en-el-motor-source/
- Backdoors:
    - Wikipedia. Puerta trasera: https://es.wikipedia.org/wiki/Puerta_trasera
    - Hackersenlared. ¿Que son los backdoors?.15 agosto 2012 https://hackersenlared.wordpress.com/category/capacitacion/malware/que-son-backdoors/
- Tipos de honeypot
    - Whos.org. Honeypots II – Clasificación. 1 de octubre de 2014: http://wh0s.org/2014/10/01/honeypots-ii-clasificacion/
    - S3lab. Honeypots, atrayendo hackers con vulnerabilidades. Iskander Sanchez Rola. 10 de noviembre de 2015: http://s3lab.deusto.es/honeypots-hackers-vulnerabilidades/
    - Fwhibbit. Introducción a los sistemas señuelo – Honeypots. Diego Jurado. 2 de abril de 2016: https://www.fwhibbit.es/introduccion-a-los-sistemas-senuelo-honeypots
- Configuración previa antes de echar a andar el honeypot
    - Thehackerway. Honeypots Parte 1 – Kippo. Adastra. 24 de marzo de 2015: https://thehackerway.com/2015/03/24/honeypots-parte-1-kippo/

- Instalación de la herramienta:
    - Bruteforcelab. Installing kippo SSH honeypot on Ubuntu. Ion. 5 de diciembre de 2015: http://bruteforcelab.com/installing-kippo-ssh-honeypot-on-ubuntu.html
- Registrar eventos kippo con mysql
    - Bruteforcelab. Logging Kippo events using MySQL DB. Ion. 8 de diciembre de 2011: http://bruteforcelab.com/logging-kippo-events-using-mysql-db.html
- Manual snort(IDS):
    - Instalación: https://www.snort.org/
    - Descripción: https://es.wikipedia.org/wiki/Snort
    - Manual snort org. SNORT Users Manual 2.9.9.: http://manual-snort-org.s3-website-us-east-1.amazonaws.com/
    - Hacking-Etico. Instalación de un IDS con Snort parte 1.Juanjo Martinez. 20 de noviembre de 2015: https://hacking-etico.com/2015/11/20/instalacion-de-un-ids-con-snort-parte-i/
    - Mirar punto 1.4
- P0f:
    - Información sobre la herramienta:
        - Wikipedia. P0f. 27 de junio de 2017: https://en.wikipedia.org/wiki/P0f
        - bpsmind. La herramienta de fingerprinting p0f. Jose David Baena. 17 de julio de 207: https://bpsmind.wordpress.com/2008/07/17/la-herramienta-de-fingerprinting-p0f/
    - Información acerca de fingerprinting:
        - Bpsmind. Fingerprinting. Jose David Baena. 17 de julio 2008: https://bpsmind.wordpress.com/2008/07/17/fingerprinting/ ç
        - Bpsmind. Fingerprinting activo y pasivo. Jose David Baena. 17 de julio de 2008: https://bpsmind.wordpress.com/2008/07/17/fingerprinting-activo-y-pasivo/
    - Cowrie:
        - Instalación de la herramienta:

- GitHub. Cowrie. Michel Oosterhof: https://github.com/micheloosterhof/cowrie

  - ClamAV:
    - Instalación: https://www.clamav.net/
  - WireShark:
    - Instalación: https://www.wireshark.org/
    - Descripción:
      - Wikipedia.WireShark. 17 de agosto de 2017: https://es.wikipedia.org/wiki/Wireshark

- Página de Arch linux: https://www.archlinux.org/
- Dionaea:
  - Github. Dionaea. DinoTools: https://github.com/DinoTools/dionaea
  - Thehackerway.Honeypots Parte 2 – Introducción a Dionaea. Adastra. 6 de marzo de 2015: https://thehackerway.com/2015/03/26/honeypots-parte-2-introduccion-a-dionaea/
  - Readthedocs: https://dionaea.readthedocs.io/en/latest/

Apéndice A. Instalación del sistema y herramientas

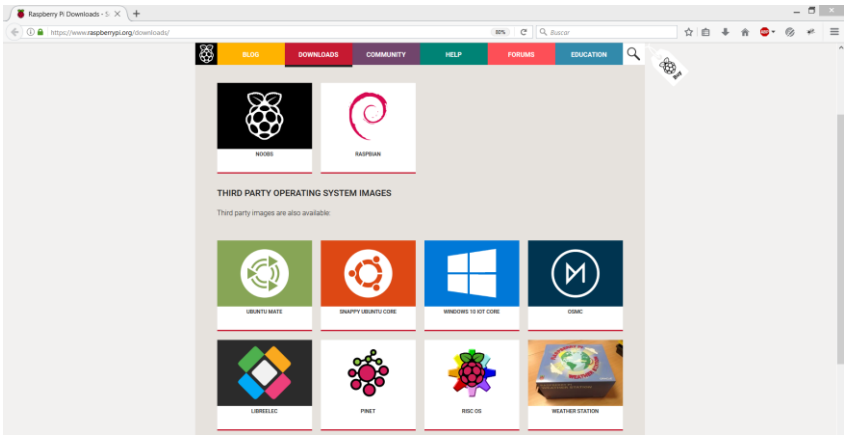
Este proyecto se realizará sobre una placa raspberry pi, los motivos por los que se ha decidido realizarlo en esta placa son sencillos:

- Por un precio asequible tenemos un ordenador de proposito general con una potencia (1 GB en el modelo utilizado) considerable para el tamaño que tiene y el consumo que gasta(1.8W a pleno rendimiento).
- Si se sabe como se puede utilizar para propósitos concretos, ya sea dentro del ámbito doméstico(miniPC, consola retro, o centro multimedia), como en el ámbito educativo(escaneo 3D, honeypot, archlinux).
- Al tener un tamaño reducido se puede utilizar en diseñor que implican movilidad(hay ejemplos qie lo usan como cámara digital, movil o consola portatil), lo cual la hace una placa muy versátil.
- Algunos ejemplos:
  - http://es.gizmodo.com/16-geniales-proyectos-para-tu-raspberry-pi-1657920779
  - https://www.xataka.com/accesorios/las-13-mejores-ideas-que-hemos-encontrado-hechas-con-raspberry-pi

A continuación describiremos como instalar el sistema en nuestro miniordenador.

Breve tutorial de instalación del sistema

A. Instalación del sistema raspbian
   a. Preparación de la tarjeta SD:
      i. Entramos en la sección de descargas de la página de raspberry (https://www.raspberrypi.org/downloads/)
      ii. Elegimos el sistema que queramos instalarle

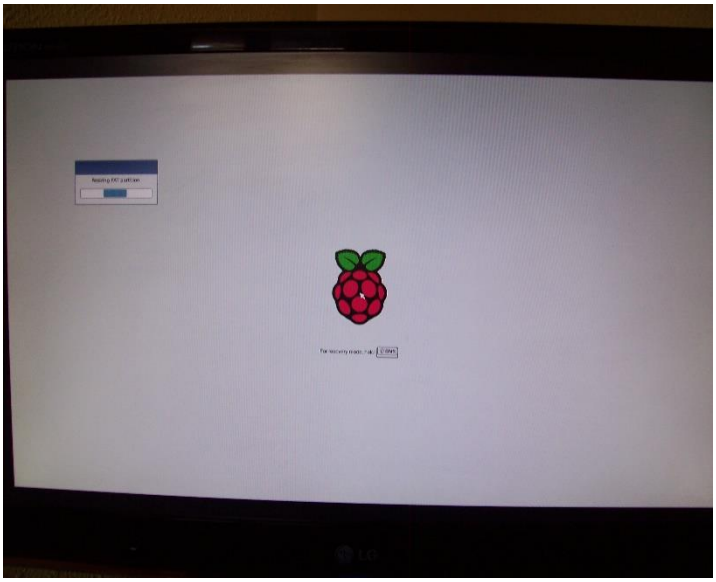


Instalación del sistema 1

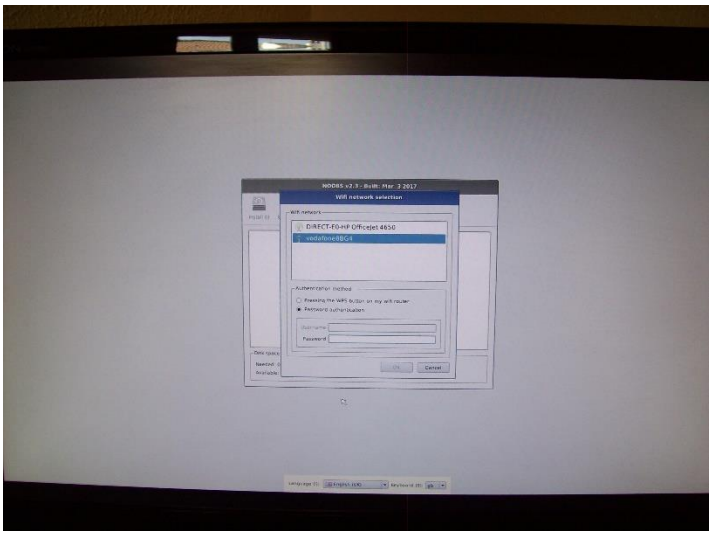
iii. Tras elegir la distribución descomprimimos el contenido en la tarjeta micro sd y la insertamos en la raspberry.

B. Instalación del sistema:

1. En nuestro ejemplo hemos elegido descarganos el instalador NOOBS, ya que es la que menos ocupa, pero necesitaremos internet para descargarnos la distribución que instalaremos.

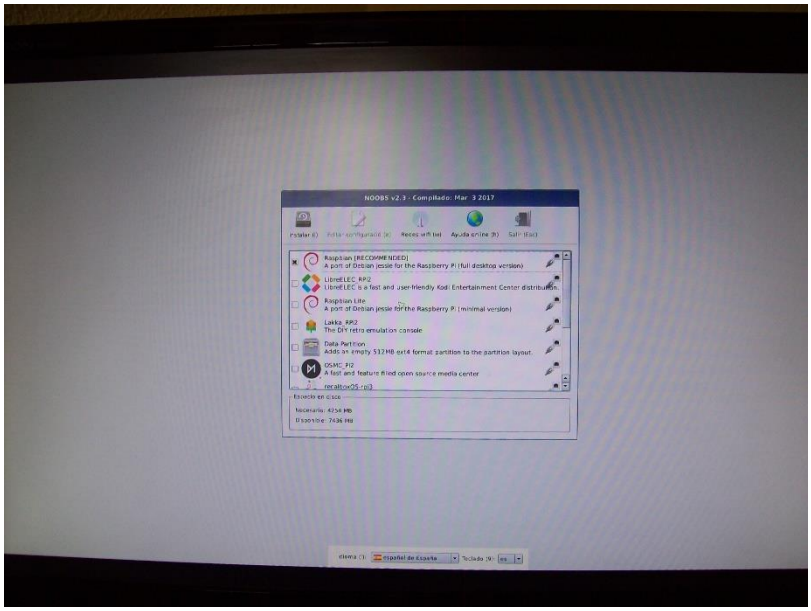


Instalador cargando



Selección de red 1

2. Una vez estemos conectados nos saldrá una lista de los sistemas oficiales soportados. Elegiremos raspbian



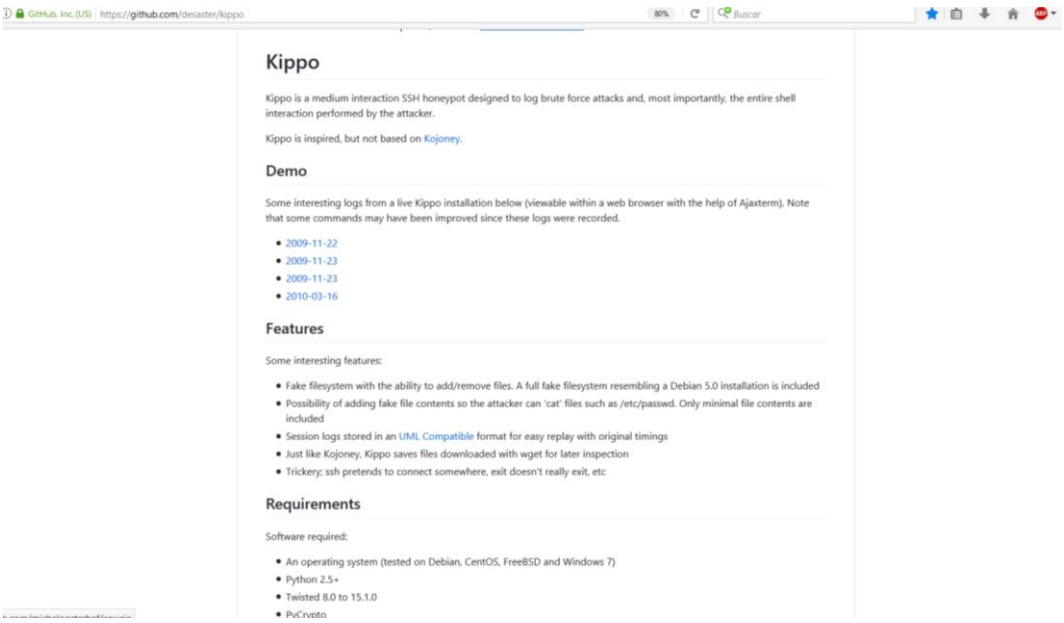
Selección de sistema operativo 1

C. Instalación del honeypot kippo

1. Primero abriremos el terminal y actualizamos el sistema con: **sudo apt-get update && apt-get upgrade.**
2. A continuacion mediante apt-get install, instalaremos los paquetes y librerias que necesita kippo:
    a. Python-dev
    b. Openssl

c. Python-openssl
d. Python-pyasn1
e. Python-twisted

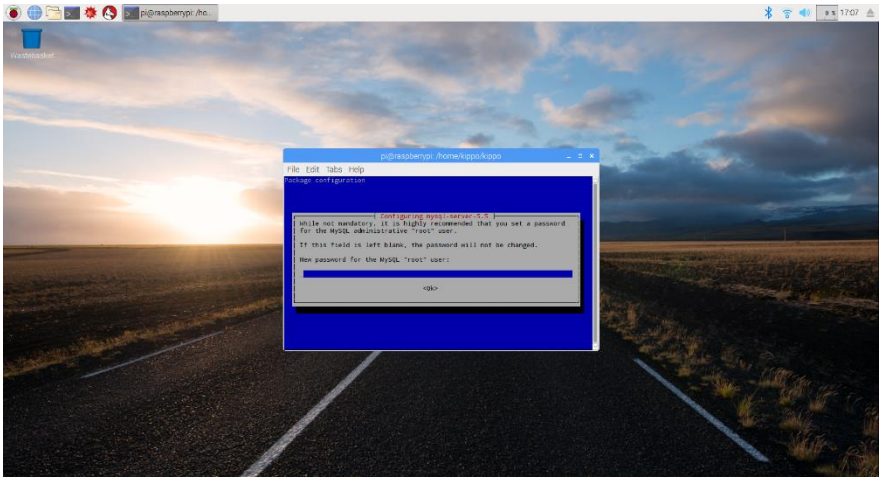
3. Para poder descargar kippo de su directorio descargaremos e instalaremos git.
4. Crearemos un usuario para manejar el honeypot y lo añadimos al grupo de usuarios que pueden usar el comando sudo.
5. Tras esto nos descargamos la versión mas actualizada de kippo de su repositorio con git clone:



Página kippo 1

D. Registrar eventos de kippo con mysql

1. Empezaremos por instalar python-mysqldb y mysql-server con apt-get install
2. Instalando mysql-server, llegará un momento de la instalación que nos pedirá configurarlo.
    a. Primero pidiendonos una contraseña para el root de mysqlserver

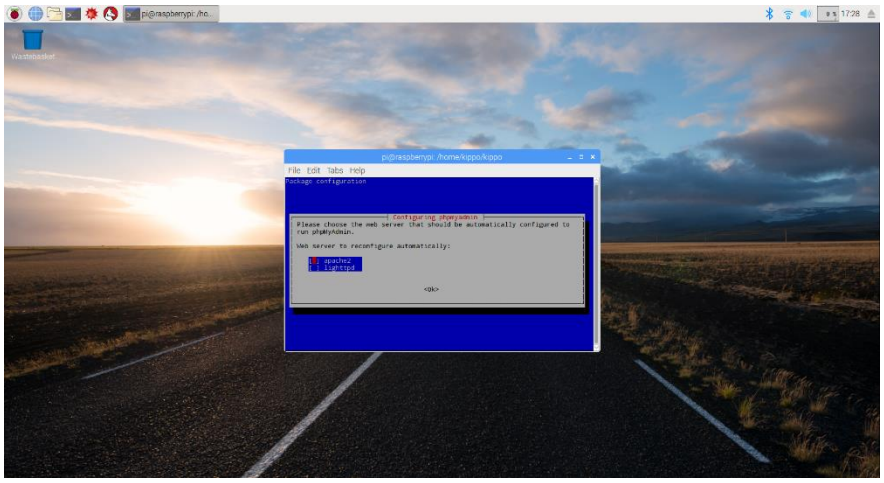


Configuración de mysql-server 1

b. Tras esto habrá que definir un usuario para kippo en mysql para poder registrar la actividad:
    i. Abrimos mysql como root(el usuario que se definió durante la instalación) con : mysql –u root –p
    ii. Una vez se inicie el programa realizaremos los siguientes pasos:
        1. Crearemos una tabla para kippo con: CREATE DATABASE kippo.
        2. Crearemos un usuario para kippo y le daremos derechos sobre la tabla con: GRANT ALL ON kippo.*(todo en la tabla kippo) TO ‘kippo’@’localhost’ IDENTIFIED BY ‘neverWas80’.
    iii. Nos moveremos hacia la carpeta donde tengamos instalado kippo y volvemos a ejecutar mysql, pero esta vez con nuestro usuario: mysql –u kippo –p
        1. Cambiamos a la bd que hemos creado para este usuario: USE kippo.
        2. Aqui cargaremos la tabla de estructuras en la bd con: source ./doc/sql/mysql.sql.
    iv. Tras esto deberemos modificar el archivo kippo.cfg en lo que se refiere al logeo en la bd tal como comentaremos en la sección 3.5.3.1.12.

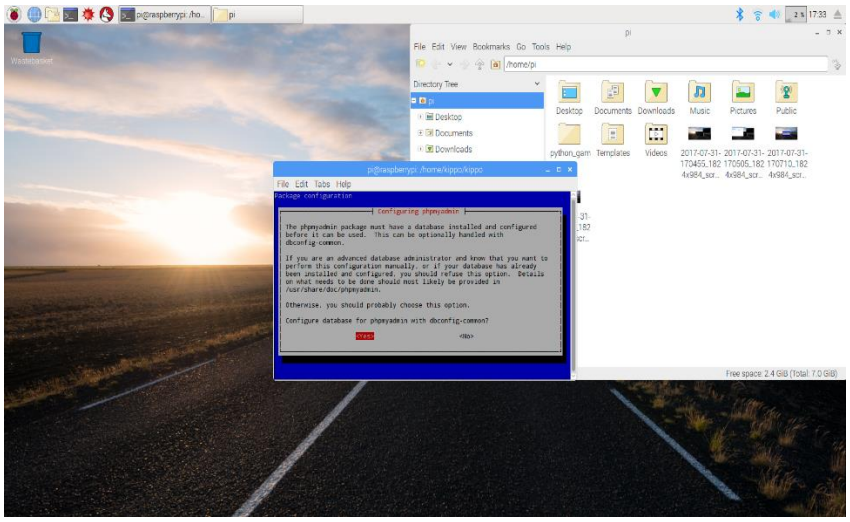
v. Para terminar con esta parte instalaremos phpmyadmin, que es un gui para mysql server.

1. Nos pedira que en que web server deberia ser configurado phpmyadmin, se eligió apache.



Configuración de phpmyadmin 1

2. Nos preguntará si tenemos una bd configurada y que si queremos configurar automáticamente a partir del archivo dbcommon, como ya la hemos configurado una podemos decir que si.



Configuración de phpmyadmin 2

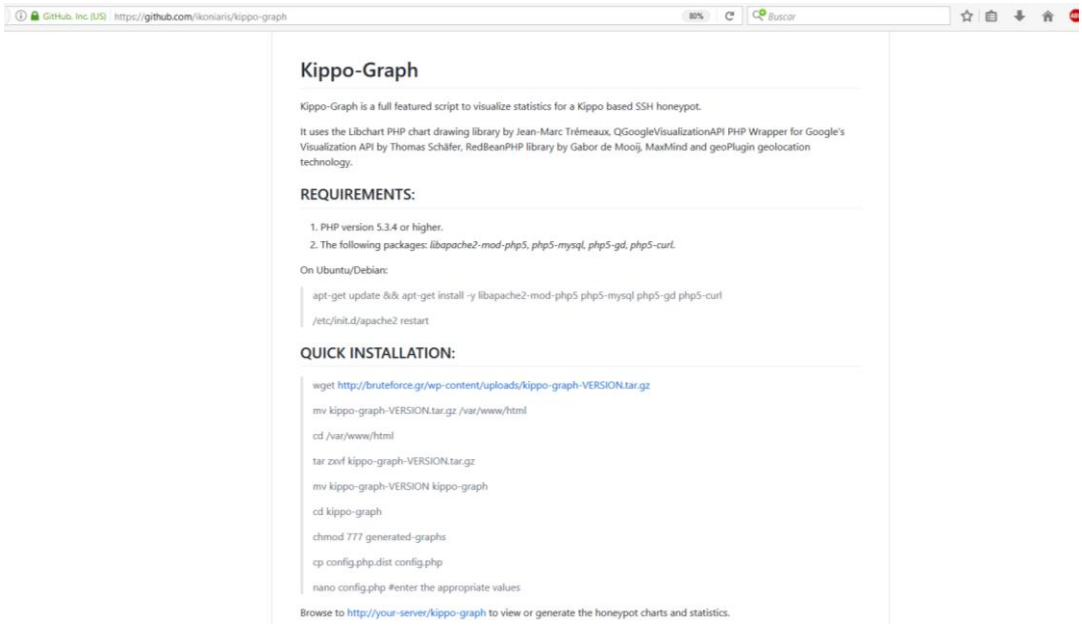
3. Nos pedirá la contraseña del root.
4. Nos pedriá a continuación una para el root de phpmyadmin.

vi. Por ultimo nos descargaremos kippo2mysql, que es un script que extrae algunos datos de los archivos log de kippo y los inserta en la base de datos de mysql.

1. Tras descargarlo deberemos modificar el script 'kippo2mysql.pl' para que tenga los datos de inicio de sesión de mysql. Para ello buscaremos las variables:
    a. Sql_user
    b. Sql_password
    c. Database
    d. Hostname
    e. port

E. Instalación de kippo-graph

a. Lo descargaremos de su repositorio de git hub:



Página kippo-graph 1

b. Comprobaremos y descargaremos las dependencias que necesite:
    i. Comprobaremos que dependencias tenemos instaladas e instalaremos las que no.
    ii. Tras esto reiniciamos apache con '/etc/init.d/apache2 restart'
c. Nos desplazaremos a la carpeta y daremos permisos de lectura, escritura y ejecución al archivo generated-graphs.

d. Tras esto copiamos el archivo config.php.dist bajo el nombre config.php y modificamos esta copia para que pueda iniciar sesión en mysql y pueda extraer los datos almacenados. Nuevamente deberemos de buscar los campos relacionados con el usuario de la base de datos, su contraseña, el nombre de la base y su puerto.
e. Se debe de mirar tambien las variables relacionadas con los directorios(principal, log y data de kippo) para asegurarse de que señalan los lugares correctos de los cuales extraer datos.

Apéndice B: Tutorial para echar a andar el honeypot

Antes de arrancar el honeypot, conviene tomar una serie de medidas:

1. Configuración de la maquina

   i) Definición de una DMZ.

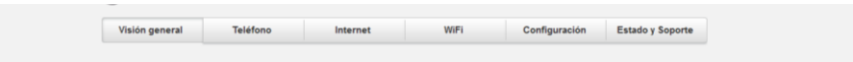
   (1) Donde colocar el honeypot: Teniendo en cuenta lo visto en el apartado 2.2.2 lo más ventajoso seria colocar el honeypot en una zona desmilitarizada ya que, por un lado, lo tenemos bajo la protección de un firewall, y por otro, lo tenemos aislado de la red.

   (2) Antes de colocar el honeypot es importante darle a este una dirección ip fija, el motivo, si es dinámica y va cambiando corremos el riesgo de que esa dirección le toque a otra máquina.

   (3) Para definir una dmz, lo primero es acceder a la configuración del router(normalmente poniendo la dirección ip en el navegador):

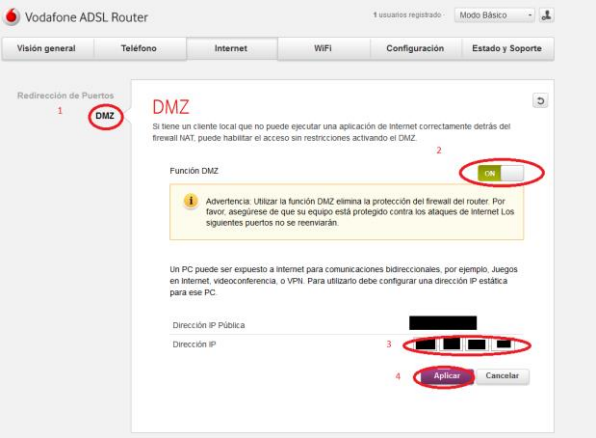
Inserción de la máquina en DMZ 1

   (4) Tras hacer login dentro del router, nos vamos hacia la pestaña 'Internet':



Inserción de la máquina en DMZ 2

ii) Vamos a la sección ‘DMZ’:



Inserción de la máquina en DMZ 3

(a) Activas la dmz

(b) Insertas la dirección ip de la raspberry

(c) Pinchas en activas.

2. Con kippo
    i. Una vez dentro del sistema raspbian se recomienda redirigir todas las entradas del puerto 22(ssh) a otro que no sea root y que este siendo vigilado por nuestro honeypot, en este caso lo redireccionaremos al puerto 2222 mediante la orden:
        a. iptables -t nat -A PREROUTING -p tcp –dport 22 -j REDIRECT –to-port 2222

    ii. Ahora pasaremos a configurar nuestro honeypot(kippo):
        1. En el archivo kippo.cfg veremos los campos que debemos cambiar para personalizar la configuración del honeypot:
            a. Ssh_addr = Interfaz de red en la que iniciará el honeypot, por defecto es 0.0.0.0(cualquiera).
            b. Ssh_port = puerto donde colocaremos el honeypot para escuchar, nosotros lo pondremos en el 2222.

c. Log_path=directorio donde se guardaran los registros de eventos.
d. Download_path= directorio donde se guardarán los archivos malware dejados por el atacante.
e. Download_limit_size=Tamaño máximo de archivo que se podrá guardar en download_path
f. Filesystem_file= Propiedad que se encarga de definir el sistema de ficheros del honeypot, incluyendo archivos y permisos
g. Data_path= En este directorio se incluye el fichero 'userdb.txt' que define los usuarios del sistema de ficheros.
h. Txtcmds_path= Aqui se podran definir los programas que se podrán usar una vez se entre en el honeypot.
i. exec_enabled= Propiedad que define si el atacante puede interactuar con el sistema. Por defecto esta desactivada, pero la activaremos para darle al atacante la oportunidad de generarnos eventos a su costa.
j. interact_enabled= Nos permitira interactuar con el atacante. Por defecto está desactivado
k. interact_port= Puerto de por el que interacturaremos en el caso de estar activada la opción anterior
l. database_mysql=Aqui definiremos las credenciales definidas para mysql
    i. host=localhost
    ii. database=kippo
    iii. username=kippo
    iv. password=neverWas80
    v. port=3360

iii. Una vez hayamos instalado el sistema, tengamos nuestro honeypot en una dmz, hallamos redirigido los puertos y configuremos nuestro honeypot, podremos echarlo a andar.

Apéndice C: Como instalar Cowrie

1. Para instalar cowrie, lo primero es instalar las dependencias que pueda necesitar: según la página del proyecto las dependencias a satisfacer son: python-virtualenv libssl-dev libffi-dev build-essential libpython-dev python2.7-minimal authbind y git.
2. Una vez satisfechas las dependencias crearemos un usuario que no posea derechos de root, tras esto descargamos cowrie desde su directorio de git.
3. Lo siguiente es instalar el entorno virtual, que es para lo que nos servía instalar python-virtualenv.
    a. Virtualenv cowrie-env(se recomienda este nombre para evitar problemas).
4. Tras instalarlo deberemos activarlo e instalar las dependencias.
    a. Source cowrie-env/bin/activate
    b. (una vez estemos en cowrie-env) pip install –r requirements.txt
5. También deberemos modificar el archivo de configuración, para ello al igual que con kippo copiaremos el archivo cfg.dist a otro cfg, que será el que modificaremos. El original se quedará sin modificar, para poder volver a empezar en el caso de algún fallo. Como cowrie es una versión avanzada de kippo la mayoría de las opciones a tocar nos serán familiares.
6. A continuación, es recomendable generar una clave DSA, ya que de esta manera nos podremos evitar problemas de compatibilidad con algunas versiones de twisted. Para crearla:
    a. Ssh-keygen –t dsa –b 1024 –f ssh_host_dsa_key.
        i. –t Para especificar el tipo
        ii. Para especificar el número de bits.(Para DSA debe ser 1024).
7. Antes de empezar a usar cowrie,tendremos que añadir el directorio de cowrie al path de python:
    a. Export PYTHONPATH=/home/usuario/cowrie
8. Si llamamos al entorno virtual cowrie-env, entonces podremos encenderlo directamente, de lo contraio habrá que modificar el archivo en bin/ cowrie.

Apéndice D: Directorios de herramientas y credenciales

Credenciales raspberry:

Usuario: pi Contraseña: raspberry - capaz de usar sudo

Usuario:kippo Contraseña: caraCarton47 – capaz de usar sudo

Usuario cowrie Contraseña: (no tiene) – usuario normal.

Usuario administrador: root contraseña:imNotAHero63

Contraseña administrador phpmyadmin: youShallNotPass

Kippo

Carpeta raíz: /home/kippo/kippo

Script inicio:/home/kippo/kippo/start.sh

Script parada:/home/kippo/kippo/stop.sh

Carpeta descargas: /home/kippo/kippo/dl

Carpetas de logs: /home/kippo/kippo/log

Base de datos de mysql: kippo

Credendiales base de datos: usuario: kippo contraseña:neverWas80

Kippo-graph

Carpeta raíz: /var/www/html/kippo-graph

Archivo de configuración: /var/www/html/kippo-graph/config.php

URL: localhost/kippo-graph

Usar como credenciales las de cowrie o kippo para acceder a sus respectivas bases de datos.

Cowrie

Carpeta raíz: /home/cowrie/cowrie

Script: /home/cowrie/cowrie/bin/cowrie (órdenes:start stop status)

Carpeta descargas: /home/cowrie/cowrie/dl

Carpeta de logs: /home/cowrie/cowrie/log

Base de datos de mysql: cowrie

Usuario base de datos de my sql: usuario: cowrie contraseña: tryItIfYouCan32

Dionaea

Carpeta raíz:/opt/dionaea

Apéndice E: Desglose de entrega

El presente proyecto se entregará de la siguiente manera:

- CD con la documentación y las herramientas en una carpeta.
  - La documentación consistirá en este documento en formato PDF.
  - Las herramientas se componen de los logs creados por la herramienta durante su actividad.
- Aplicación compuesta por una copia del sistema configurada en una tarjeta micro sd de 16 GB.