Explotación de Honeynets con Big Data.

Trabajo Final Integrador
de la
Especialización en Seguridad Informática

Autor: Coenda, Francisco Javier

Asesores
Tapia, Carlos

Noviembre 2017

Índice general

Índice de cuadros

Índice de figuras

1. Introducción

Ya no es noticia, ni tópico desconocido el hecho de que el volumen de datos que se genera año a año crece exponencialmente y las herramientas que se utilizan en el área de seguridad informática no son la excepción. Ya sea un firewall, IDS/IPS, honeypot o cualquier otro componente de seguridad de una organización, generan datos permanentemente. Estos se acumulan sin solución de continuidad y no son explotados de manera adecuada, inclusive se llegan a perder con el paso del tiempo [1], ya sea por falta de políticas de gestión de logs o porque son almacenados sin realizar el tratamiento adecuado.

En el campo de la seguridad informática, las organizaciones están desarrollando e invirtiendo en soluciones que les permitan detectar ciertos patrones en grandes volúmenes de datos [2, 3]. Incluso, buscan soluciones que van más allá del procesamiento masivo de datos y búsqueda de patrones, sino también se busca el estudio y desarrollo de modelos predictivos que permitan establecer medidas preventivas en las organizaciones [2].

Las honeynets tienen como propósito ser atacadas para obtener datos de las herramientas, las tácticas y las actividades que utilizan los cibercriminales, para posteriormente ser estudiados y obtener información de los ataques informáticos a fin de mejorar el nivel de seguridad de la red. Una honeynet está compuesta por uno o más sensores configurados específicamente para atraer y engañar a los ciberatacantes cuando intentan acceder a la red que está simulando. No es inusual que al cabo de algunos días de funcionamiento expuestos a internet, cualquiera de estos sensores haya almacenado cientos de miles de registros, los cuales eventualmente deberán ser filtrados y tratados. Esto permitirá obtener información valiosa que posibilitará priorizar las contramedidas y dirigir puntualmente cualquier inversión a realizarse en el área de seguridad informática.

El presente trabajo se enfocará en investigar y plantear un prototipo de infraestructura que permita almacenar y explotar los grandes volúmenes de datos que genera una honeynet, así como definir indicadores de interés para la toma de decisiones. Además se establecerán los mecanismos necesarios para visualizar dichos indicadores a través de un dashboard.

1.1. Situación Problemática

Después de varios meses de trabajo en el proyecto Red de Sistemas Señuelo para la Ciberdefensa, desde ahora llamado "proyecto red señuelo", se detectó que los honeypots generaban un gran volumen de datos que son depositados en distintos logs, entre los cuales no existe relación alguna. Esto trae como consecuencia la imposibilidad de realizar análisis de amplio espectro y además, dificulta la visualización y correlación de los indicadores en tiempo real imposibilitando de esta manera la realización de un seguimiento de los ataques y detenerlos a tiempo.

También se detectó que estos sensores no cuentan con visualizadores adecuados que presenten la información de manera amigable para el usuario final.

2. Problema

El proyecto red señuelo está compuesto de múltiples sensores que depositan los datos en distintos logs y no poseen relación alguna entre ellos. Cada log tiene su propia estructura de almacenamiento. Por otra parte existen sensores que almacenan los datos en múltiples logs, cada uno de ellos con una extensión y estructura de almacenamiento distinta, generando que la información no pueda ser visualizada adecuadamente. Además se detectó que los sensores no poseen un medio adecuado que permita presentar la información de manera amigable para el usuario final. Cualquier analista de seguridad informática debe contar con información clara y accesible rápidamente, dado que debe dedicar su valioso tiempo a estudiar la información y elaborar contramedidas, más que gastar esfuerzo en obtener la información y disponerla de manera que le sea útil.

3. Objeto de Estudio

Explotar mediante Big Data [4] los datos que genera los sensores dentro de la honeynet y exponerlos a través de algún medio de visualización adecuado que permita proveer información en tiempo y forma sobre lo que está sucediendo dentro de la infraestructura de simulación de la honeynet.

4. Campo de Acción

El campo de acción de este proyecto se limita a trabajar con la infraestructura montada en el proyecto red señuelo.

5. Objetivo General

Establecer los mecanismos necesarios para aprovechar los datos que producen los sensores de la red señuelo.

5.1. Objetivos Específicos

- Estudiar la salida del sensor Cowrie.
- Almacenar los datos en una base de datos que permita la explotación a través de herramientas Big Data.
- Definir los procesos de transformación que tendrán como origen los logs y como destino la base de datos [5].
- Consumir dichos datos mediante las CTools [6].
- Evaluar el funcionamiento de la solución propuesta.

6. Delimitación del Proyecto

Se analiza la salida del honeypot y se almacenan los datos que este genere en una base de datos NoSQL para posteriormente ser visualizados en un dashboard.

7. Desarrollo

7.1. Big Data

7.1.1. ¿Qué es el Big Data?

En los últimos años el Big Data ha irrumpido con una fuerza abrumadora en todos los ámbitos que nos podamos imaginar. La definición básica que se puede encontrar y que existe en el ideario colectivo, es que Big Data se refiere a grandes volúmenes de datos (estructurados, no estructurados y semi estructurados). Sin embargo, es difícil establecer a partir de cuándo se puede considerar el término "grandes volúmenes" (giga, tera, peta, etc) [7]. Big Data implica mucho más que grandes volúmenes de datos, hace referencia al conjunto de datos que no puede ser analizado y/o procesado a través de los métodos y herramientas tradicionales [7, 8].

7.1.2. Características del Big Data

El Big Data posee tres características que lo identifican [7–10].

- **Volumen:** Se refiere a la cantidad de datos. Resulta importante destacar, tal como se mencionó anteriormente, que no existe una convención respecto a lamagnitud a partir de la cual se puede considerar como grande el volumen de datos. Esto se debe principalmente a que el espacio de almacenamiento a seguido evolucionando con el paso del tiempo, permitiendo ampliar la capacidad de los discos para almacenar información.

- **Variedad:** Hace referencia a la variedad de datos que se puede obtener hoy en día. No solamente referido a datos estructurados y obtenidos de una base de datos relacional, si no, también apunta a datos de sensores (temperatura, presión, movimientos, etc), sistemas de posicionamiento satelital (GPS), audios, registro de eventos (logs), mails, etc. Hoy en día, la cantidad de fuentes de datos que existen son infinitas y a la vez son heterogéneas entre ellas.

- **Velocidad:** Este punto se refiere a la velocidad con la que se generan los datos en la actualidad. Es preciso tener en cuenta la existencia cada vez mayor de dispositivos que están generando datos en forma permanente. Por otra parte, este hecho implica que ante la gran cantidad de datos se necesite que las herramientas y procedimientos traten con mayor rapidez para facilitar el proceso de toma de decisión en tiempo y forma.

A las características clásicas que hemos mencionado anteriormente, se debe agregar una cuarta V que comenzó a ser propuesta por varias organizaciones [7–11].

- **Valor:** Los datos poseen un valor intrínseco el cual debe ser descubierto. Esto ha llevado a que las organizaciones se planteen formas de obtener los datos de la manera más rentable y eficiente posible, a la vez que estos datos deben aportar valor a la organización.

7.1.3. El Big Data en el mundo de la Seguridad Informática.

Todo ataque o intrusión informática queda registrado en un log. Sin embargo, estas huellas que quedan en los sistemas, suelen pasar desapercibidas y solamente son detectadas cuando se realiza una auditoría posterior a la sucesión de los eventos. Esto se suele denominar análisis post-mortem y se debe al simple hecho de que el volumen de datos es elevado para realizar análisis en tiempo real. Se le suma además el hecho de que los logs son almacenados por un tiempo o sobrescritos por las mismas aplicaciones para optimizar los espacios de almacenamiento [12].

Uno de los conceptos que se está comenzando a aplicar es el de "Visualización de la Información" como soporte en el área de seguridad informática. Se toma en cuenta el hecho de que la búsqueda de patrones y anomalías en el mar de datos plantea un reto muy grande para el ser humano. Es allí donde la visualización de información convierte los datos crudos en gráficos interactivos que son fáciles y rápidos de comprender por el ser humano [13, 14].

Este nuevo paradigma que ha irrumpido con fuerza ha llevado a que las organizaciones actualmente estén volcando sus recursos en desarrollar soluciones que permitan aprovechar el potencial del big data y enfoca en ofrecer productos específicos y puntuales que aprovechen la potencialidad que ofrecen los datos. A modo de ejemplo se puede mencionar el software Adaptive Defense de Panda Security, el cual distingue y clasifica el software como malware o goodware [15]. La estrategia de IBM de integrar Qradar con la plataforma de Big Data para ofrecer una solución que permite recoger, monitorear, analizar, explorar y reportar los datos de seguridad de la empresa de una forma que antes era imposible [16,17]. Por otra parte Telefónica a través de la aplicación tacyt puede detectar apps fraudulentas e inclusive correlacionar que otras apps han sido creadas por el mismo autor (a pesar de que este cambie su nombre/identificador en el appstore) [18].

Se puede seguir mencionando más empresas, cada una con una solución distinta y aplicada a diferentes problemáticas que impactan en cada una de estas organizaciones. Esto demuestra que el big data irrumpió con una gran fuerza en el mundo de la seguridad informática y que al día de hoy se desarrolla de una forma veloz con el fin de poder obtener información que permita mejorar los procesos de toma de decisión.

7.2. HoneyPots y HoneyNets

7.2.1. ¿Qué es un Honeypot?

Es una herramienta que se usa casi exclusivamente en el campo de la seguridad informática. Son sistemas o servidores que se utilizan como señuelo con la finalidad de atraer y analizar ataques realizados por bots o hackers. Básicamente se busca ver sus pautas de ataque, generar diccionarios para recopilar que palabras usan en ataques (para no usarlas en tu sistema), conocer al enemigo

y su perfil [19, 20]. También cabe recalcar que los honeypots no reemplazan los otros sistemas de seguridad, si no que son un nivel más de seguridad [20].

Los honeypots se pueden clasificar de acuerdo al nivel de interacción que ofrecen [21, 22]. A continuación se detallan los tres niveles de interacción en los que se puede clasificar un honeypot:

- **Honeypots de Baja Interacción:** Estos honeypots son los más fáciles de implementar y se caracterizan por emular los servicios e inclusive los sistemas operativos. Las actividades del atacante se limitan al nivel de emulación del honeypot, con lo cual no existe riesgo de comprometer los sistemas reales de la organización. Sin embargo el nivel de información que se puede extraer de este tipo de honeypots es muy limitada.

- **Honeypots de Media Interacción:** Este tipo de honeypots ofrece una mayor interacción que los honeypots de baja interacción. Aparte de emular servicios también se simula el software o soluciones en particular. También se obtiene una mayor cantidad de información, pero se debe tener en cuenta que los intrusos tienen un mayor nivel de interacción y a medida que se incremente las funcionalidades a la que tenga acceso el atacante, mayor será la posibilidad que este pueda llegar al los sistemas reales.

- **Honeypots de Alta Interacción:** Este tipo de honeypots se caracterizan por utilizar sistemas y aplicaciones reales. En este tipo de soluciones no se emula nada, lo que conlleva un riesgo bastante alto tomando en cuenta que el atacante tiene a su disposición los sistemas reales para interactuar. Debido a las características de estos honeypots, son bastante complejos de implementar y mantener, a la vez que son riesgosos, ya que desde estos honeypots el atacante puede llegar a los sistemas de producción. Sin embargo, la calidad de información que se obtiene de estos honeypots es excepcional.

Como se puede ver, los honeypots son una herramienta muy útil en el ámbito de la seguridad informática y aportan datos de gran valor para las organizaciones.

7.2.2. ¿Qué es una Honeynet?

Una honeynet es una red de sistemas compuesta por uno o más honeypots de alta interacción. Su objetivo es ser comprometida para obtener datos respecto a herramientas, tácticas, técnicas y motivos que existen detrás de los ciberataques. Estas redes señuelos, suelen ser configuradas con aplicaciones y servicios que simulan la red de la empresa [21–23]. La finalidad es crear una infraestructura en la que haya sistemas y servicios reales tales como DNS, HTTP, SMTP, etc. para permitir que el atacante se encuentre en un ambiente más realista. Se debe tomar en cuenta que una honeynet tiene dos requerimientos básicos [22]:

- **Control del flujo de datos:** Debido a que el objetivo de una honeynet es ser comprometida y atacada, se debe tener un control permanente y meticuloso del flujo de datos para evitar que el atacante los pueda llegar a utilizar en contra de nuestra infraestructura o infraestructuras de terceros.

- **Captura de Datos:** Si el nivel de vigilancia y captura de datos es elevado y permite que el atacante se percate de la honeynet, ésta dejara de ser efectiva. Esto lleva a que la captura de datos deba ser los más sigilosa posible evitando de este modo despertar sospechas en el atacante.

Por otra parte, se debe considerar que los datos capturados con la honeynet deben ser almacenados fuera de esta para evitar que sean comprometidas y generando la perdida de utilidad de la honeynet.

En función de lo mencionado con anterioridad se puede apreciar las honeynets son una solución mucho más compleja que permiten recopilar un mayor nivel de información de los ataques [24].

7.2.3. Tipos de Honeypots

Existen una amplia variedad de honeypots cada uno con un propósito específico. Por mencionar algunos ejemplos se tiene:

- **Elastichoney** [25, 26]**:** Un honeypot que emula una instancia vulnerable de Elasticsearch.
- **Glastopf** [27–29]**:** Un honeypot de baja interacción para aplicaciones web capaz de emular miles de vulnerabilidades web.
- **HoneyPy** [30, 31]**:** un honeypot que emula servicios UDP y TCP.
- **Conpot** [32, 33]**:** un honeypot que recaba información de los métodos y motivos que llevan a que se ataquen sistemas industriales.
- **Hale** [34]**:** Un honeypot que ayuda en la investigación y caza de botnets creando diversos sensores de red.
- **Ghost** [35–37]**:** Un honeypot para los malwares que se esparcen a través de dispositivos de almacenamiento USB.

7.2.4. Cowrie

Se trabaja con el honeypot Cowrie [38,39] desarrollado por Michel Oosterhof y basado en Kippo [40]. Es un sensor de media interacción que emula los servicios de SSH y Telnet. Diseñado con el propósito de registrar los ataques por fuerza bruta y la interacción que tiene el atacante con la Shell del sistema.

Características:

- Posee un fake filesystem con la capacidad de poder realizar distintas acciones incluidas la de agregar/eliminar archivos.
- Posibilidad de agregar archivos con contenido fake para que el atacante crea que está viendo archivos reales del sistema (ej /etc/passwd).
- Todas las sesiones son almacenadas para ser fácilmente reproducidas.
- Almacenamiento de los archivos que son descargados por el atacante para una posterior inspección

- Soporte para SSH exec commands.
- Soporte para la subida de archivos.
- Registro en formato JSON.

7.3. Propuesta de Solución

La solución que se busca plantear en este proyecto es el establecimiento de las bases necesarias para montar una infraestructura que permita a través del Big Data explotar los datos generados mediante los distintos honeypots que componen la honeynet. La finalidad es poder gestionarlos adecuadamente y visualizarlos en tiempo y forma.

7.3.1. Infraestructura

La infraestructura propuesta se compone:

- **ETLs** [41,42]**:** son script/procesos que permiten extraer, procesar y cargar los datos. Para este proyecto se diseñan un grupo de ETLs con el objetivo de extraer los datos almacenados en los distintos logs que se generan. Este grupo va a tomar los datos de los logs, los va a procesar/filtrar y posteriormente almacenar en un storage NoSQL.
- **Storage:** La propuesta de almacenamiento para los datos es utilizar un storage NoSQL [43–46]. Esta decisión se basa primera instancia en el hecho de que cada honeypot almacena y estructura los datos a su manera, lo que complejiza el trabajo de llevarlos a un storage relacional. Por otra parte, al tratarse de sensores que están logueando de manera permanente, el volumen de información que van a generar se torna elevado y afecta la performance de las consultas que se necesita realizar.
- **Dashboards:** Para la visualización de la información, una vez que los datos hayan sido procesados, se propone utilizar dahsboards. Se generan una serie de tableros de mandos con distintas gráficas que permitan las métricas e indicadores de interés. Para fines prácticos de este proyecto y con la finalidad de acotar el campo de trabajo, se va a trabajar con un grupo reducido de métricas e indicadores.

En la siguiente imagen podemos apreciar el esquema de la infraestructura propuesta en el proyecto.

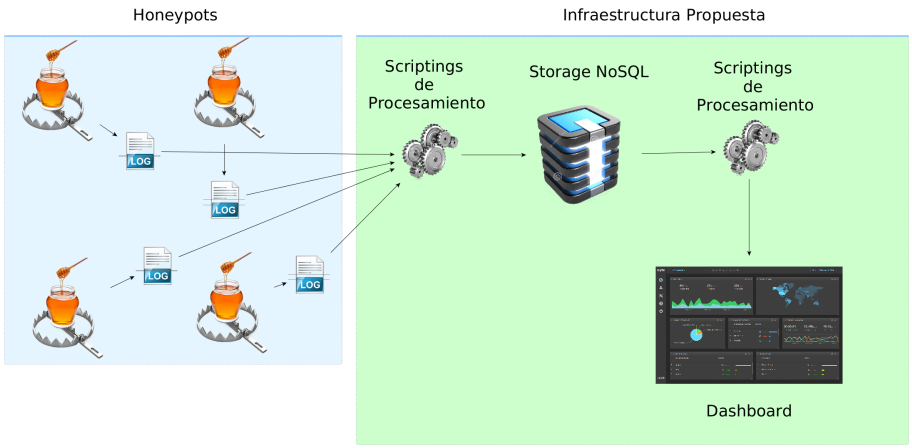


Figura 1: Infraestructura Propuesta

7.3.2. Desarrollo de la infraestructura propuesta

Para desarrollar la solución propuesta se trabaja con un entorno compuesto por dos máquinas virtuales. Una de las VM's (Virtual Machine) contiene el sensor que se utilizará, mientras que la segunda VM contiene el storage, dashboards y ETL's. Esta decisión se toma debido a la disponibilidad del hardware al momento de realizar el prototipo, ya que se trabaja con una maquina personal.

A continuación se presenta el detalle de cada una de las VM's:

Cuadro 1: Detalle de Virtual Machines

Nombre de VM	Recurso	Configuración
Dashboard, storage y ETL's	SO	ubuntu-16.10-desktop-amd64
	RAM	2.5 GB
	Procesador	2
	Disco	60 GB
Sensor	SO	ubuntu-15.04-desktop-amd64
	RAM	2.5 GB
	Procesador	1
	Disco	45 GB

Para tomar los datos que genera el sensor se trabaja con Apache Sqoop [47], el cual se ocupa de realizar el proceso de migración al storage NoSQL. Posteriormente se utiliza Pentaho Data Integration (PDI) [48,49] para desarrollar las transformaciones responsables de procesar los datos del sensor y subsiguientemente publicarlos para su consume mediante un dashboard.

7.3.2.1 Análisis de salida del sensor Cowrie

Cowrie posee una carpeta donde almacena los eventos que suceden en nuestra red.

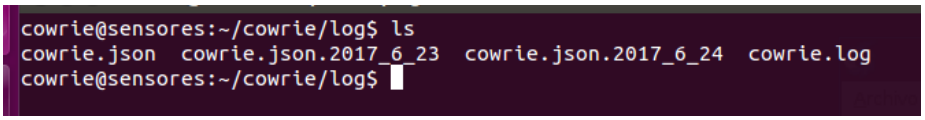


Figura 2: Directorio de logs de Cowrie

Como se puede apreciar en la imagen, Cowrie almacena los eventos en dos formatos, uno .log y otro .json. Incluso podemos comprobar que en el formato JSON [50, 51] se generan archivos secuenciales que almacenan los eventos del día en curso, permitiendo identificarlos mejor aún.

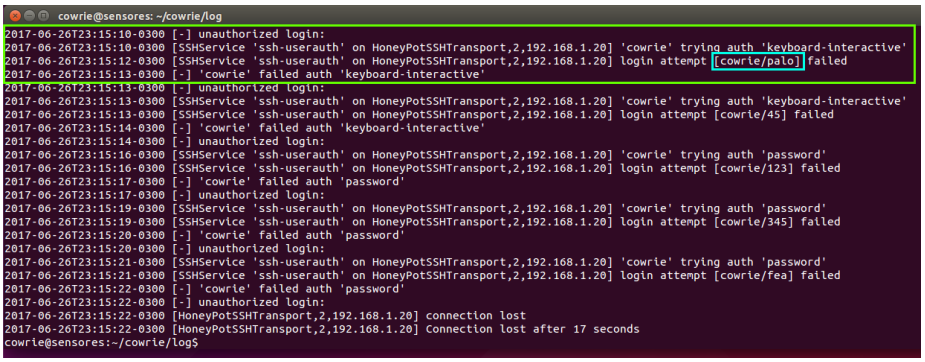


Figura 3: Formato de salida de un .log en Cowrie

En la figura 3 podemos apreciar el formato de salida que nos presenta Cowrie a través del archivo cowrie.log. Como se puede apreciar, el sensor detecta un login no autorizado y almacena la siguiente información:

- Timestamp del evento.
- Dirección ip de donde proviene el intento de conexión.
- User y password con la que se intento el login.

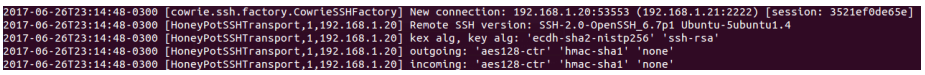


Figura 4: Información que muestra un log en Cowrie

Por otro lado, vemos que el archivo de login en formato json contiene la misma información organizada en el formato json y agrupada por cada intento de conexión.

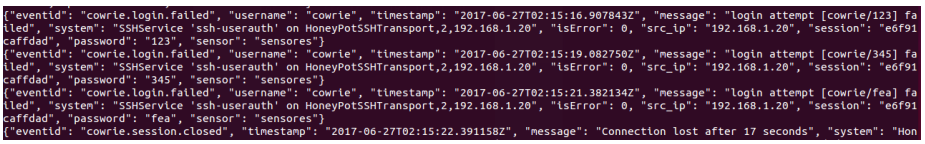


Figura 5: JSON de un .log en Cowrie

Aparte de contar con estos formatos de almacenamiento, Cowrie ofrece la posibilidad de realizar el login de los eventos en la base de datos MySQL [52,53]. En la siguiente imagen podemos ver las tablas que componen a la base de datos de cowrie

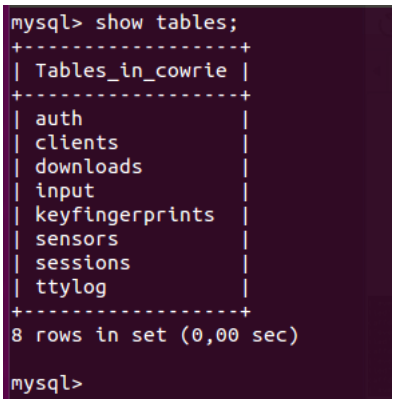


Figura 6: Tablas en las base de datos Cowrie

A continuación podemos visualizar las tablas que son pobladas por cowrie.

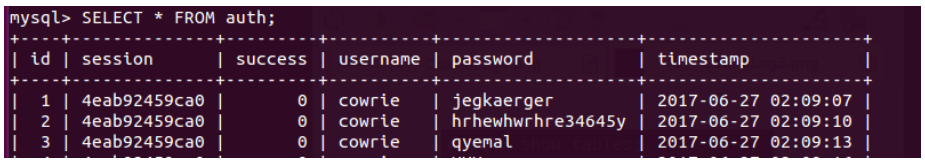


Figura 7: Tabla auth

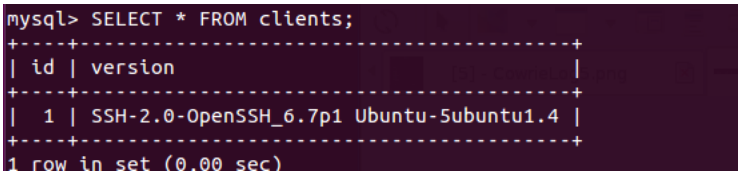


Figura 8: Tabla clients

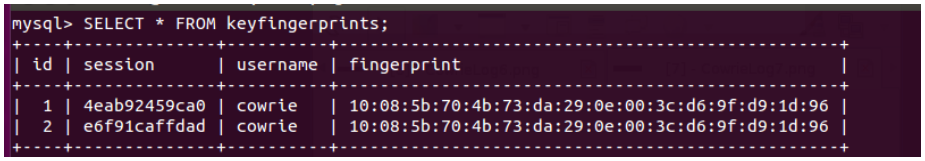


Figura 9: Tabla keyfingerprints

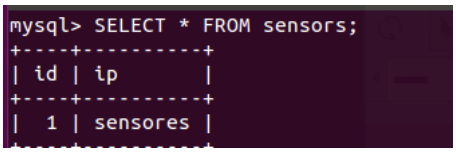


Figura 10: Tabla sensors

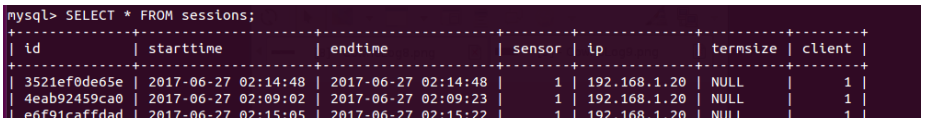


Figura 11: Tabla sessions

De estas tablas podemos obtener los siguientes datos:

- La sesión
- Conexión exitosa
- User y password probados
- Timestamp
- Versión del cliente que intento conectarse
- Sensor que detecto la conexión
- Timestamp de inicio y fin de la sesión.
- Ip de donde se produce la conexión.

Después de estudiar cada formato de salida se opta por trabajar con los datos almacenados en MySQL, debido a que estos se encuentran mejor estructurados y son más fáciles de extraer con consultas SQL.

7.3.2.2 Importar los datos al storage NoSQL

Para realizar la importación de los datos almacenados en Cowrie, se va a crear un directorio Cowrie en el cual se va a generar la estructura de archivos del ecosistema Apache HADOOP y se va a trabajar con HDFS.

En la figura 12 se puede apreciar la creación del directorio Cowrie y el comando que se ejecuta con Sqoop.

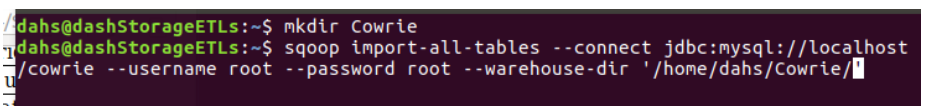


Figura 12: Importar datos con Sqoop

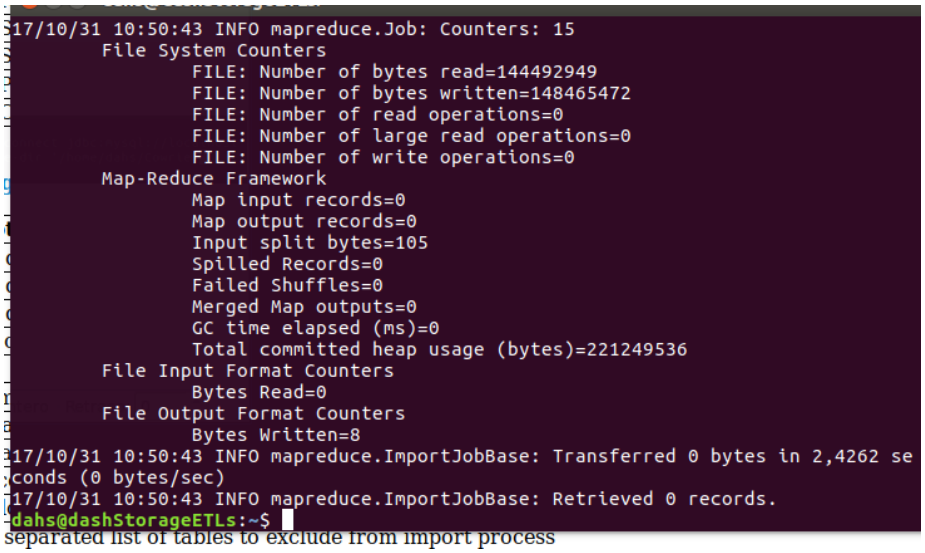


Figura 13: Proceso de importación finalizado

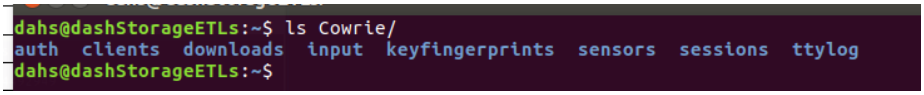


Figura 14: Directorio destino de la importación

Una vez que finaliza el proceso de importación (Figura 13), se puede corroborar que se a realizado la importación haciendo un `$ls` al directorio Cowrie (Figura 14).

7.3.3. Desarrollo de ETLs

7.3.3.1 ETL topUsername

Esta ETL es responsable de extraer los nombres de usuario y publicar cuales son los 5 nombres de usuario más usados.

Se configura la ruta de donde se va a tomar la información en el paso "**Hadoop Input File**" y como se muestra en la siguiente imagen, teniendo especial cuidado en el armado del comodín.

Figura 15: Hadoop Input File pestaña Fichero

En la solapa contenido se debe configurar el separador de campos por una "," y en el formato de archivo poner mixto.

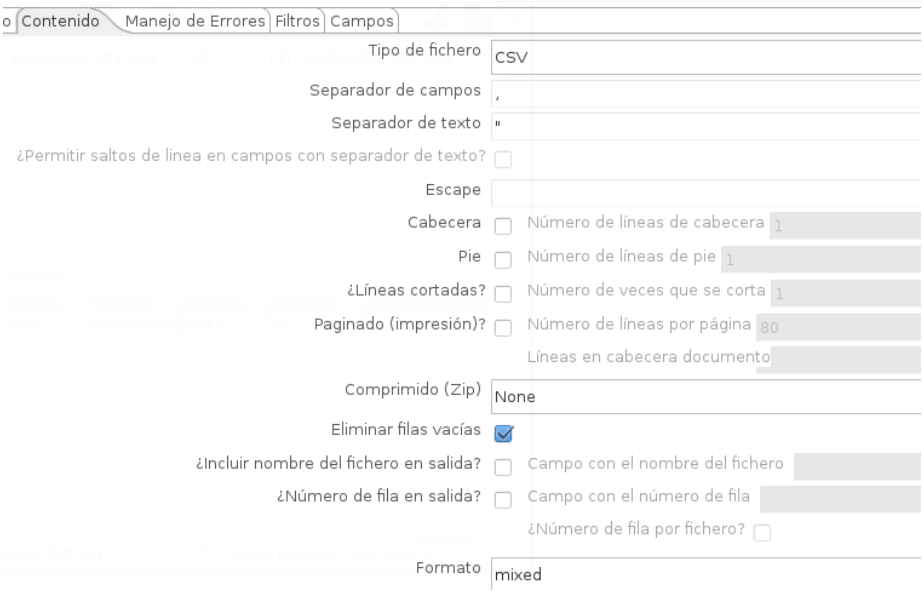


Figura 16: Hadoop Input File pestaña Contenido

Por último, se extraen los campos de los archivos.

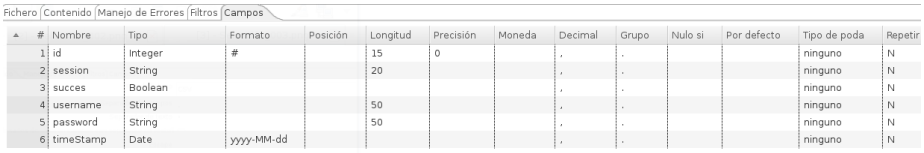


Figura 17: Hadoop Input File pestaña Campos

En el paso "**seleccionar username**", se selecciona los campos username y id.

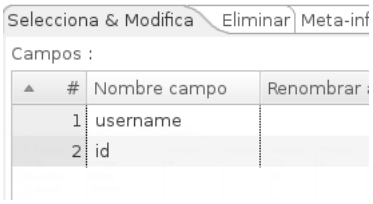
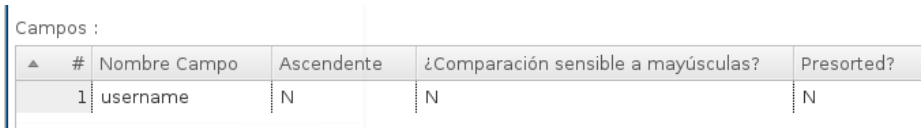


Figura 18: Selección de los campos username y id

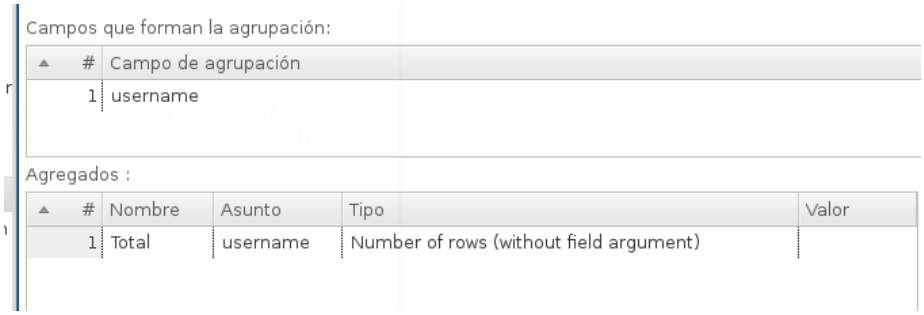
En el paso "**ordenar filas**" se ordenan los username a fin de filtrarlos correctamente.


Campos :

#	Nombre Campo	Ascendente	¿Comparación sensible a mayúsculas?	Presorted?
1	username	N	N	N

Figura 19: Ordenar filas por username

En el paso "**contar username**" se cuentan las veces que se repite un username y se nombra el campo resultante como "**Total**".


Campos que forman la agrupación:

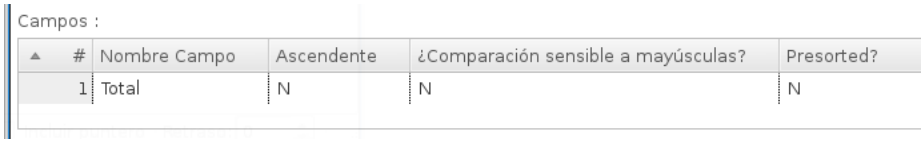
#	Campo de agrupación
1	username

Agregados :

#	Nombre	Asunto	Tipo	Valor
1	Total	username	Number of rows (without field argument)	

Figura 20: Contar usernames

En el siguiente paso "**ordenar filas**", se ordena las filas por el campo "**Total**".


Campos :

#	Nombre Campo	Ascendente	¿Comparación sensible a mayúsculas?	Presorted?
1	Total	N	N	N

Figura 21: Ordenar filas por el campo total

En el paso "**añadir secuencia**", se añade un nuevo campo con un número de secuencia. Posteriormente sevirá para filtrar los resultados.

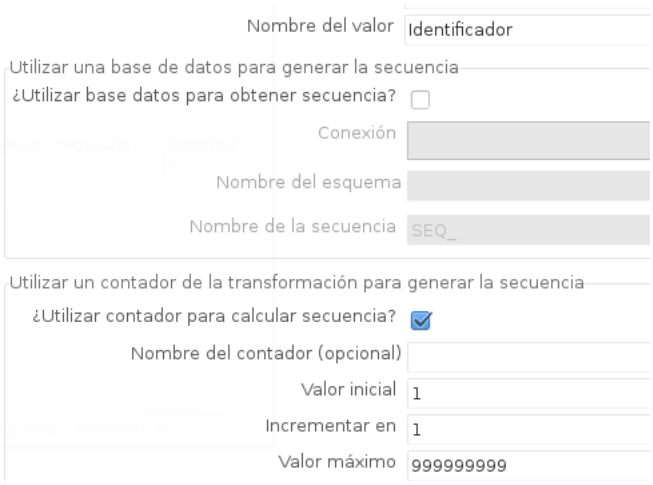


Figura 22: Añadir secuencia

En el paso "**limitar número de filas**", se configura para limitar el número de filas a la cantidad de cinco.

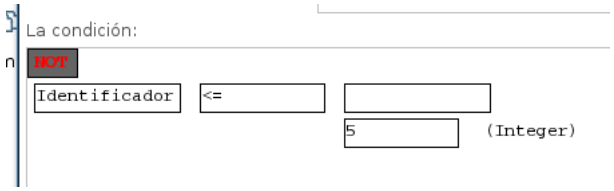


Figura 23: Filtrar filas

Por último el paso "**renombrar campo**" se seleccionan los campos que se quieren mostrar y el campo "**username**" se renombra a "**Nombre Usuario**".

Figura 24: Seleccionar y renombrar filas

A continuación, en la siguiente imagen, se visualiza el trabajo con todos los pasos concatenados.

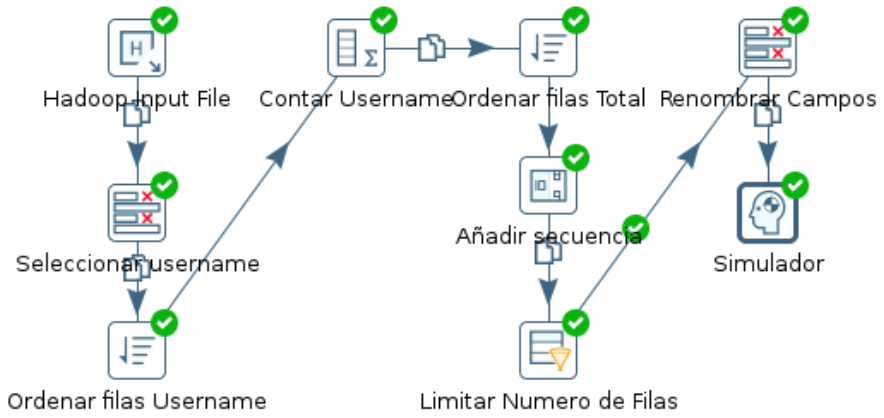


Figura 25: Transformación topUsername.ktr

7.3.3.2 ETL topPassword

La mencionada ETL es responsable de extraer los password que ingresan los usuarios y publicar cuales son los 5 password más usados.

Se configura la ruta de donde se va a tomar la información en el paso "**Hadoop Input File**" y como se muestra en la siguiente imagen, teniendo especial cuidado en el armado del comodín.

Figura 26: Hadoop Input File pestaña Fichero

En la solapa contenido se debe configurar el separador de campos por una "**,**" y en el formato de archivo poner mixto.

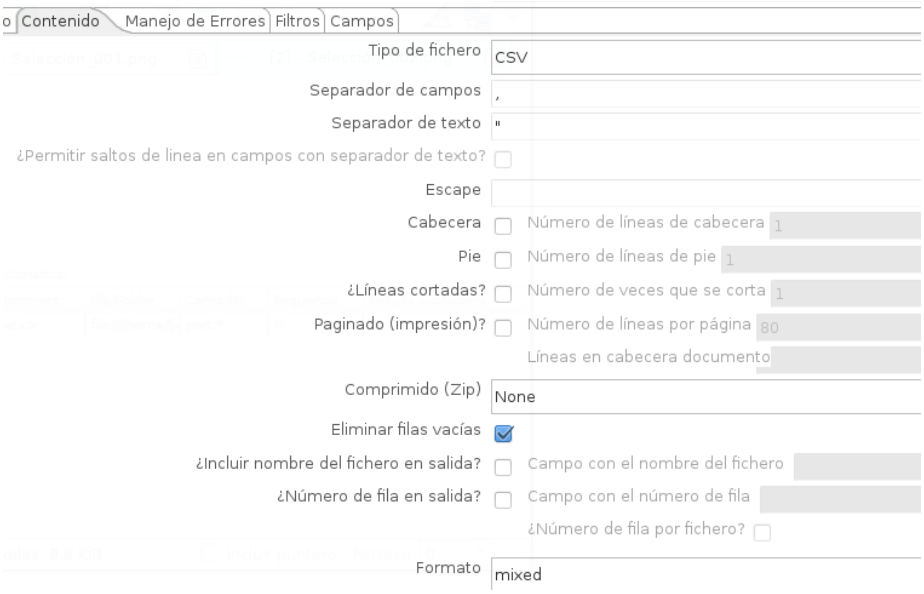


Figura 27: Hadoop Input File pestaña Contenido

Por último, se extraen los campos de los archivos.

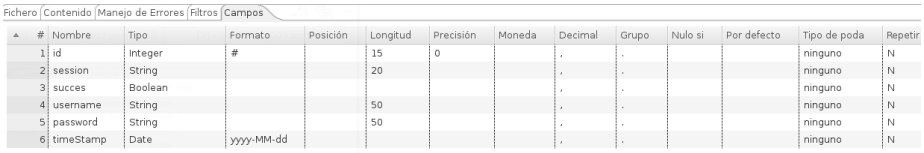


Figura 28: Hadoop Input File pestaña Campos

En el paso "**ordenar filas**" se procede a ordenar las filas por el campo password para facilitar las operaciones posteriores de las otras transformaciones.

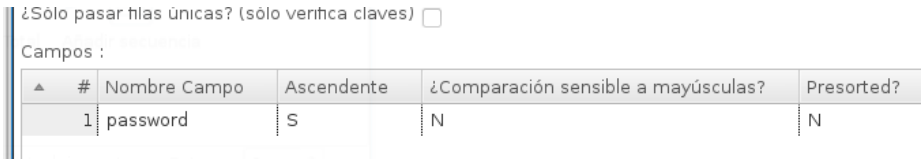


Figura 29: Ordenar filas por el campo password

El paso "**contar password**", cuenta las veces que se repite el mismo password.

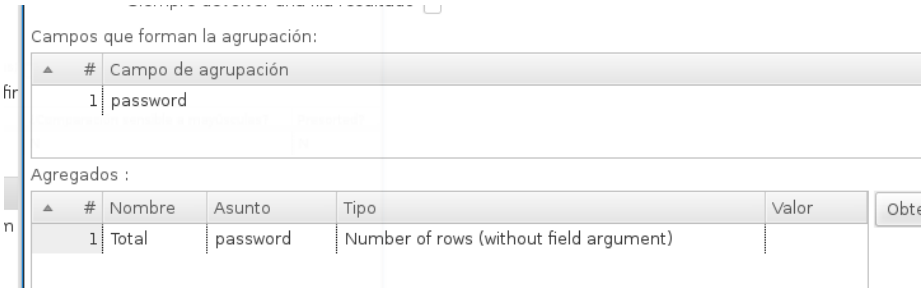


Figura 30: Paso contar password

El paso "**filtrar filas**" filtra las filas que contiene valor nulo, las cuales representan un password vacio.

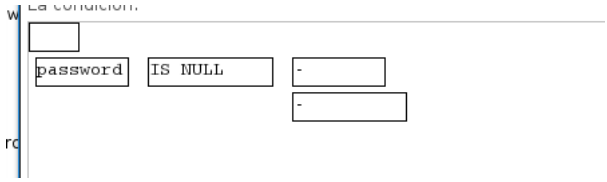


Figura 31: Filtrar los valores nulos de las filas

En el paso "**set empty password**" se remplaza el campo null por el string "**EMPTY PASSWORD**", con el fin de poder visualizar correctamente los dahsboards.

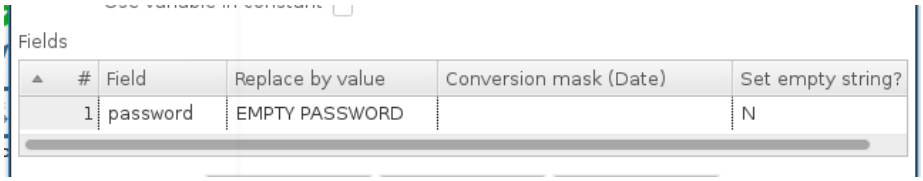


Figura 32: Paso set empty password

En el paso "**ordenar filas**" ordena las filas por el campo Total.

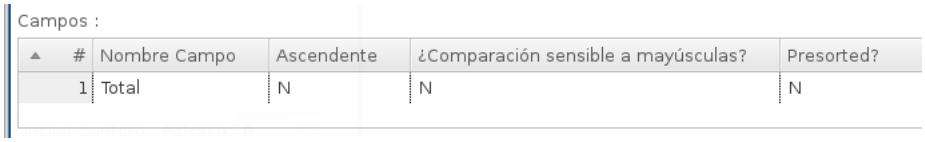


Figura 33: Ordenar Filas por el campo Total

En el paso "**añadir secuencia**" genera un nuevo campo con una secuencia numérica.

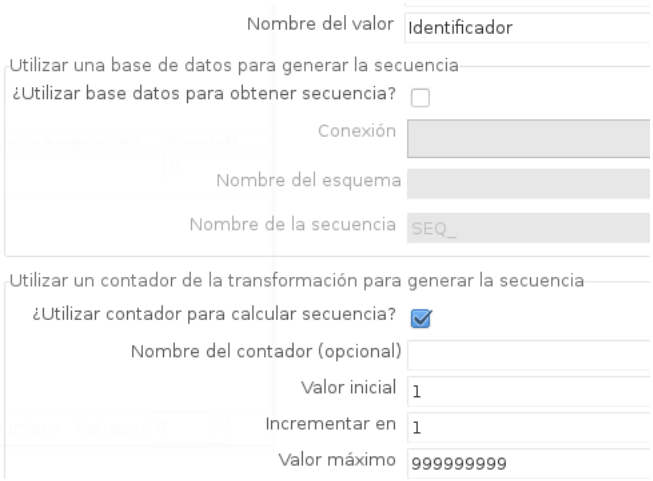


Figura 34: Generar secuencia

En el paso "**filtrar fila**" se limita el número de filas que se va a mostrar en el dashboard.

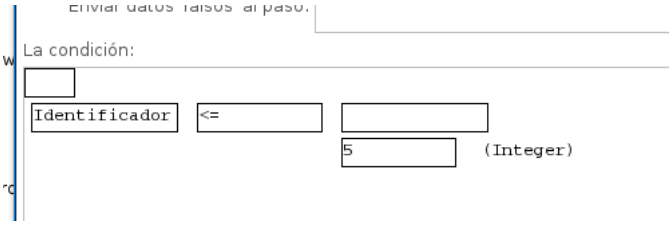


Figura 35: Paso filtrar filas

Por último, en el paso "**renombrar campo**" se cambia el nombre del campo password a Password y se selecciona los campos passwrod y Total.

Figura 36: Seleccionar y renombrar los campos password y total

A continuación, en la siguiente imagen, se visualiza el trabajo con todos los pasos concatenados.

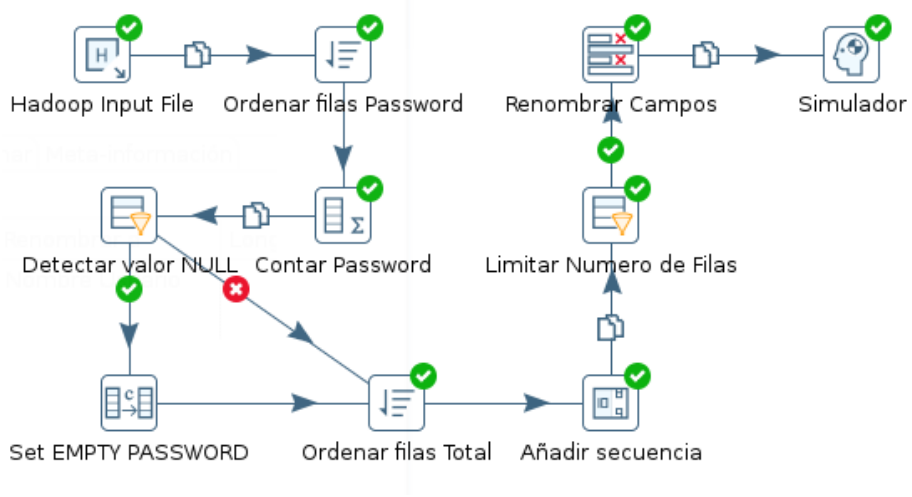


Figura 37: Transformación topPassword.ktr

7.3.3.3 ETL combinación

ETL responsable de procesar los datos para mostrar la combinación de usuario y password más utilizadas a la hora de los ataques que recibe la honeynet.

Se configura la ruta de donde se va a tomar la información en el paso "**Hadoop Input File**" y como se muestra en la siguiente imagen, teniendo especial cuidado en el armado del comodín.

Figura 38: Hadoop Input File pestaña Fichero

En la solapa contenido se debe configurar el separador de campos por una "**,**" y en el formato de archivo poner mixto.

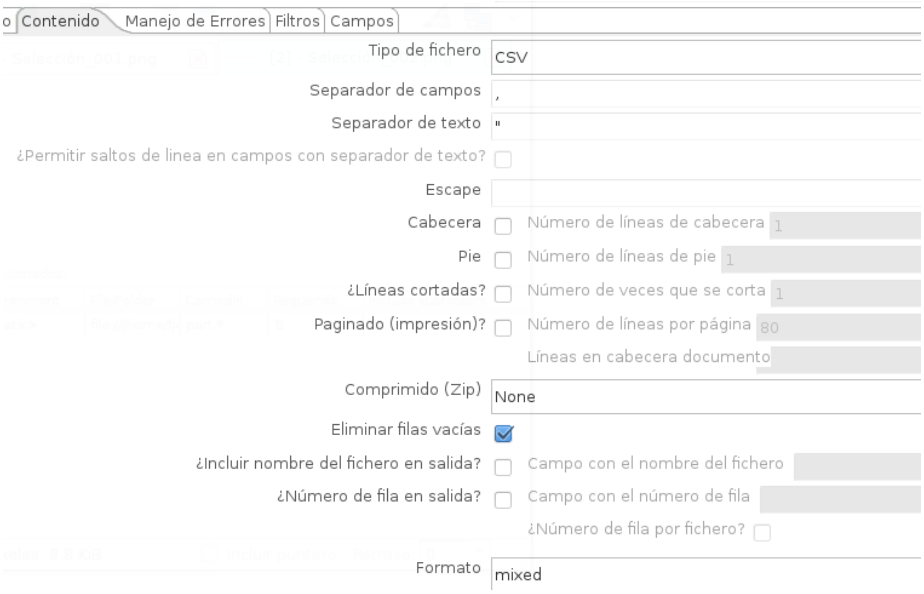


Figura 39: Hadoop Input File pestaña Contenido

Por último, se extraen los campos de los archivos.

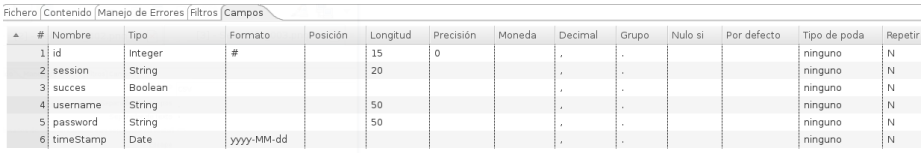


Figura 40: Hadoop Input File pestaña Campos

En el paso "**seleccionar campo**" se filtran los campos dejando solamente el campo username y password para ser procesados.

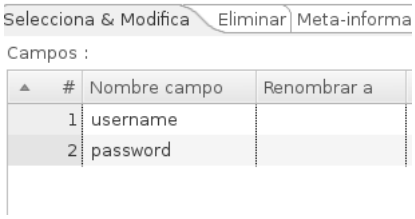


Figura 41: Seleccionar username y password

En el paso "**ordenar filas**" se ordenan las filas en primera medida por username y posteriormente por password.

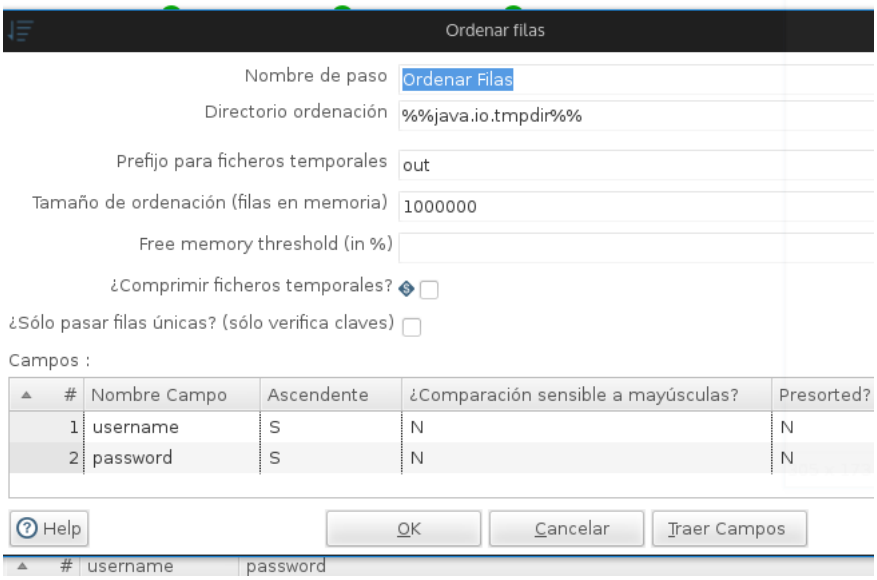


Figura 42: Ordenar Filas por username y password

En el paso "**contar ocurrencias**" se cuenta las veces en las cuales se repite la combinación de username y password y se nombra dicho campo resultante con la denominación "**Total**".

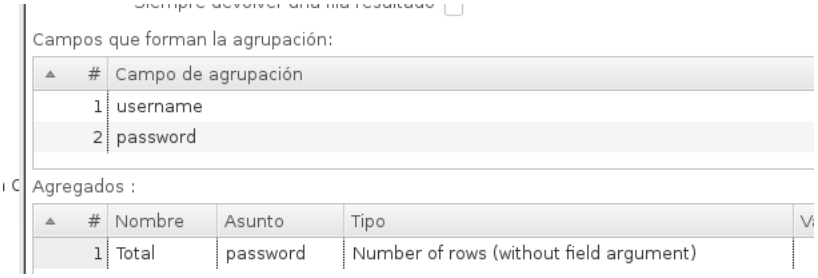


Figura 43: Paso contar ocurrencias

En el paso "**filtrar filas**" se filtra las filas que contienen valor nulo, las cuales representan un password vacío.

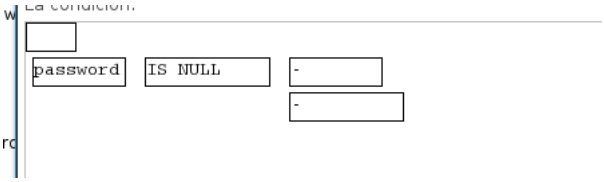


Figura 44: Filtrar valores nulos

En el paso "**set empty password**" se remplaza el campo "**null**" por el

string "**EMPTY PASSWORD**", con el fin de poder visualizar correctamente los dahsboards.

Figura 45: Paso set empty password

En el siguiente paso "**ordenar filas**", se ordenan las filas por el campo "**Total**".

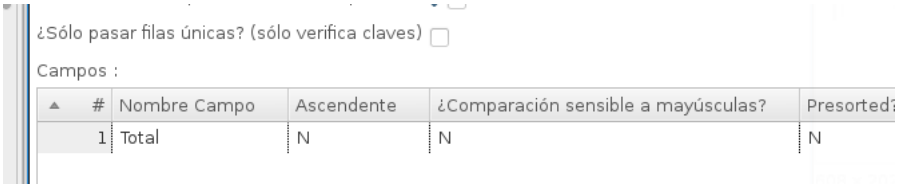


Figura 46: Paso ordenar filas

En el paso "**añadir secuencia**" se añade un nuevo campo con un número de secuencia. Posteriormente servirá para filtrar los resultados.

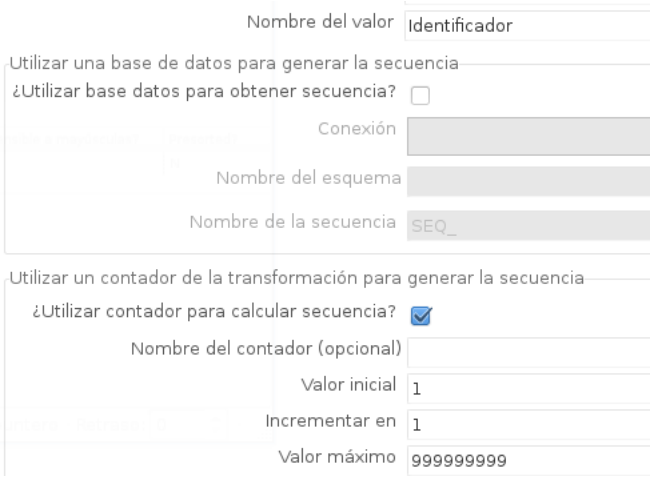


Figura 47: Generar secuencia

En el paso "**limitar número de filas**" se configura para limitar la cantidad de filas a cinco.

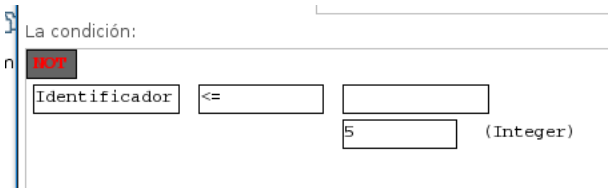


Figura 48: Limitar número de filas

Por último, en el paso "**renombrar campo**" se seleccionan los campos que se quieren mostrar. En este caso username, password y Total.

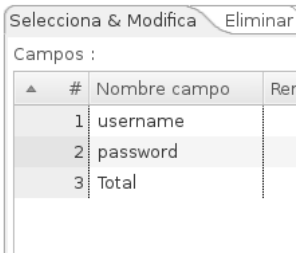


Figura 49: Seleccionar Campos

A continuación, en la siguiente imagen, se visualiza el trabajo con todos los pasos concatenados.

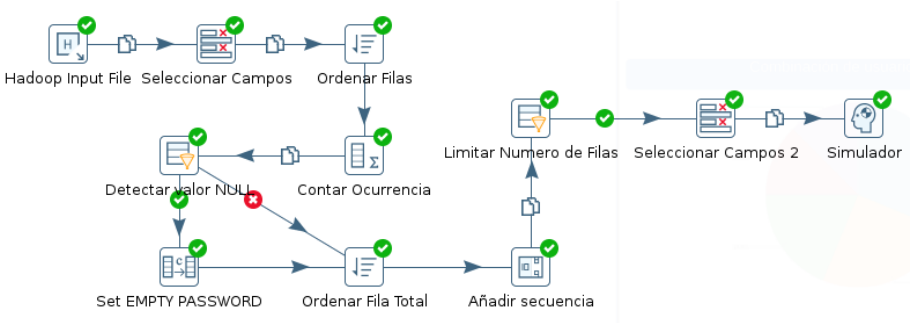


Figura 50: Transformación combinacion.ktr

7.3.4. Dashboards

7.3.4.1 Dashboards TopPasswords

El dahsboard TopPassword muestra los 5 passwords más utilizados en los intentos de acceso. Se visualiza un listado con el total de intentos de los passwords y se cuenta con un gráfico de torta para representar visualmente la proporción entre esas 5 passwords.

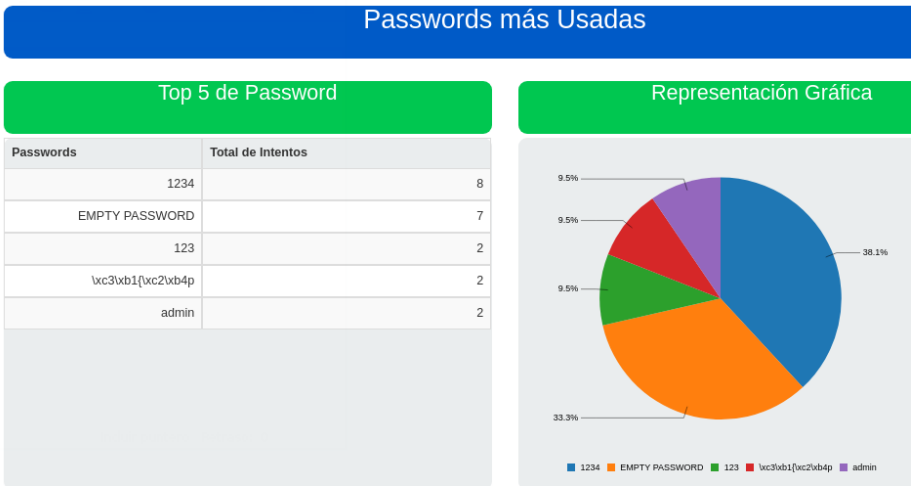


Figura 51: Dashboad passwords más usadas

Para confeccionar el dahsboards lo primero que se debe realizar es definir la fuente de dato con la que se va a trabajar.

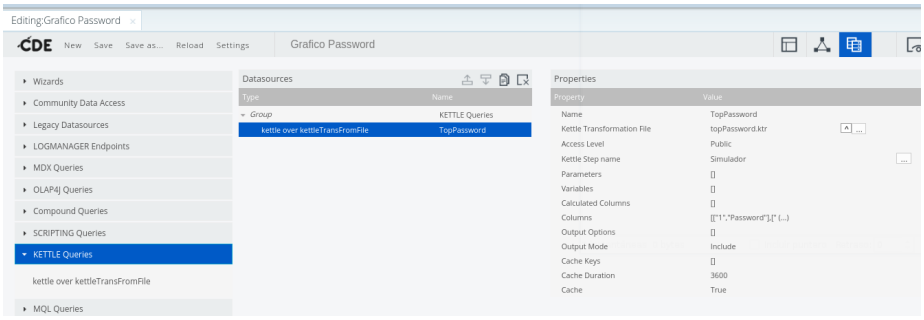


Figura 52: Definición de Fuente de datos para top password

En este caso se selecciona el recurso **Kettle over kettleTransFormFile** y se configura de la siguiente manera:

Cuadro 2: Kettle over kettleTransFormFile

Kettle Tranformation File	topPassword.ktr
Kettle Step name	Simulador
Columns	[[1, Password],[2, Total]]

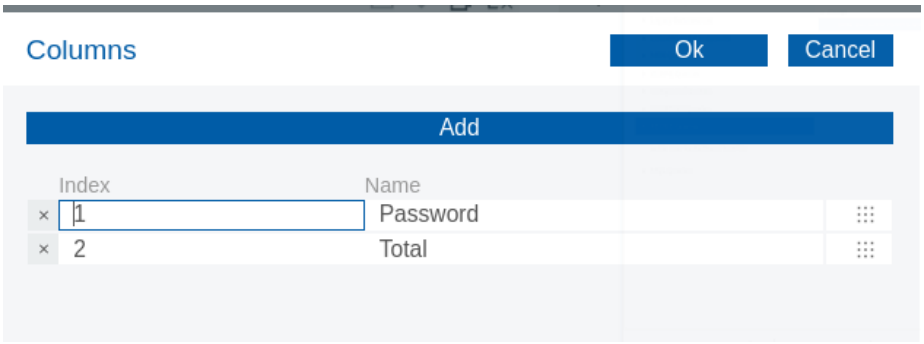


Figura 53: Definición de columnas para top password

Se debe seleccionar cual es el paso del archivo ktr que va a proveer los datos a mostrar en el dashboard. En el campo columnas indicamos cuales son las columnas que se van a utilizar.

El segundo paso a seguir es armar el layout del dashboard. Para ello, se diseñan 3 filas separadas por espacios en blanco. La primer fila contiene el encabezado/título del dashboard. La segunda fila contiene los dos encabezados por separado que representan subtitulos. Por último, la tercer fila contiene los elementos que se van a mostrar en el dashboard. En este caso una tabla y un gráfico de torta. Las cabeceras se editan en código HTML.

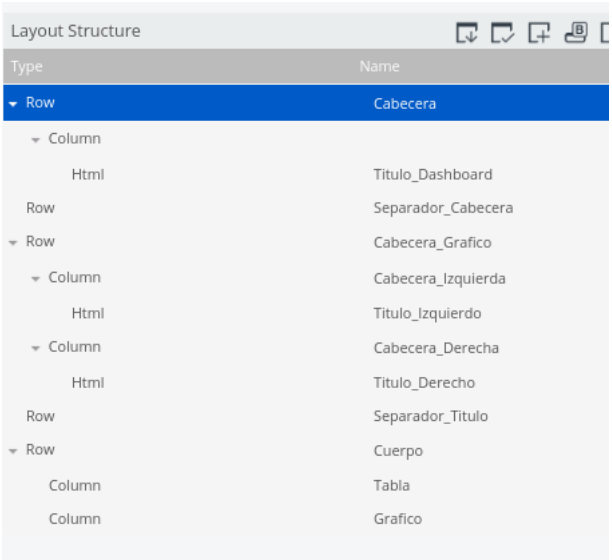


Figura 54: Configuración del layout para top password

Figura 55: Código HTML del encabezado Principal para top password

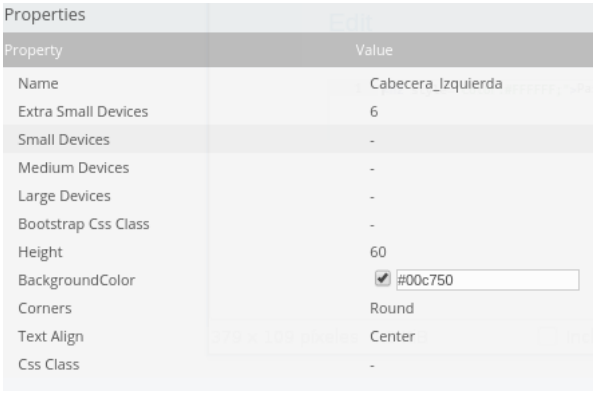


Figura 56: Configuración del fondo de una de las sub-cabeceras para top password

Figura 57: Código HTML de la tabla para top password

Figura 58: Código HTML del gráfico para top password

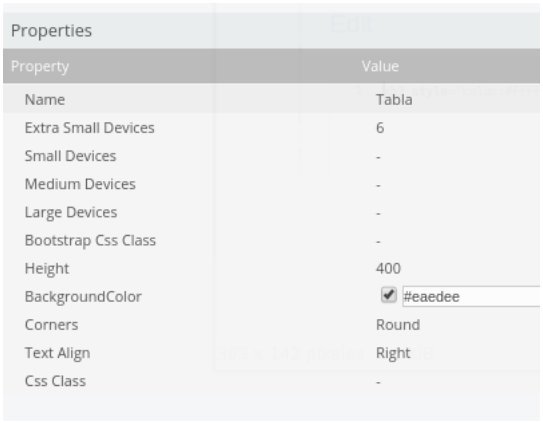


Figura 59: Configuración del contenedor para el gráfico top password

Por último, se configuran los elementos que se van a mostrar en el dahsboard. En este caso se utilizaron una tabla y un gráfico de torta. En la configuración del gráfico se procede a seleccionar el HTMLObject que va a contener el gráfico, el datasource y sus dimensiones.

Figura 60: Configuración del gráfico de torta top password

En las propiedades avanzadas del gráfico, se configura la manera en la que se muestran los valores. Seteamos el campo valuesMask de la siguiente manera:

```
valuesMask: value.percent
```

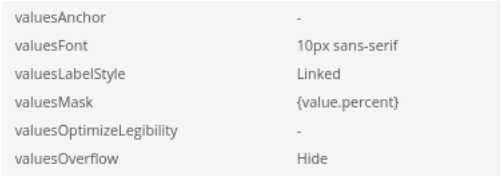


Figura 61: Configuración del value mask para el gráfico top password

Para la tabla se debe configurar la cabecera y parámetros a mostrar.

Cuadro 3: Cabecera y parámetros

Column Headers	[Password, Total]
Parameters	[[1, Password],[2, Total]]

Figura 62: Configuración cabecera y parametro para tabla top password

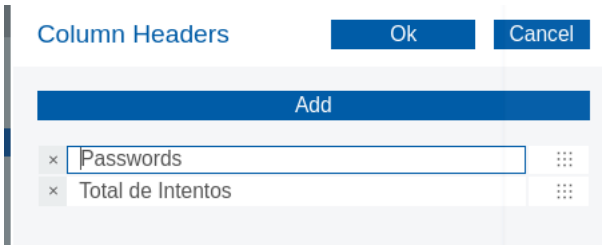


Figura 63: Configuración column header para tabla top password

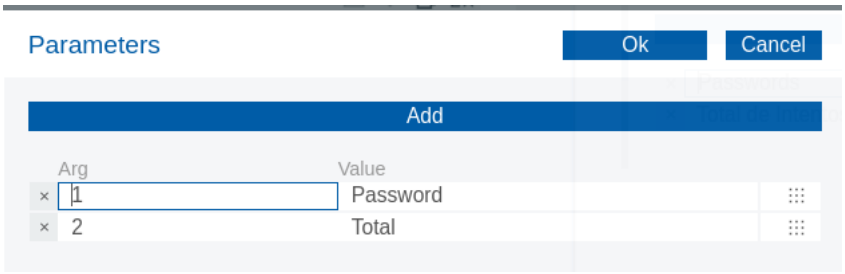


Figura 64: Configuración parameters para tabla top password

También se deben configurar las siguientes propiedades que se describen a continuación don el fin de mostrar una tabla básica:

Cuadro 4: Configuración de tabla para top password

Expand On Click	False
Show Filter	False
Info Filter	False
Length Change	True
Paginate	False
Sort Data	False

Por último se debe indicar el datasource a partir del cual se van a tomar los datos.

Figura 65: Definición del datasource para top password

7.3.4.2 Dashboards TopUsers

El dahsboard TopUsers muestra los 5 nombres de usuario más utilizadas en los intentos de acceso. Se visualiza un listado con el total de veces que se utilizo un nombre de usuario y se cuenta con un gráfico de torta para representar visualmente la proporción entre esos 5 nombres de usuario.

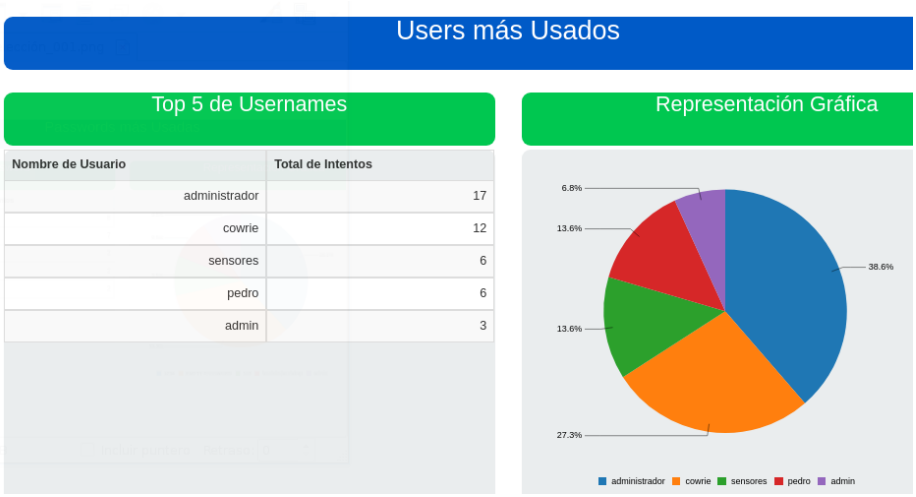


Figura 66: Dashboard top users

La configuración de este dashboard es igual a la de password, únicamente que varían algunos parámetros puntuales a la hora de seleccionar la fuente de datos y configurar los detalles de renderizado del dashboard.

Para este dashboard se debe seleccionar el origen del archivo ktr. En este caso va a ser "**topUser.ktr**" y las columnas de datos a utilizar (Nombre y Total).

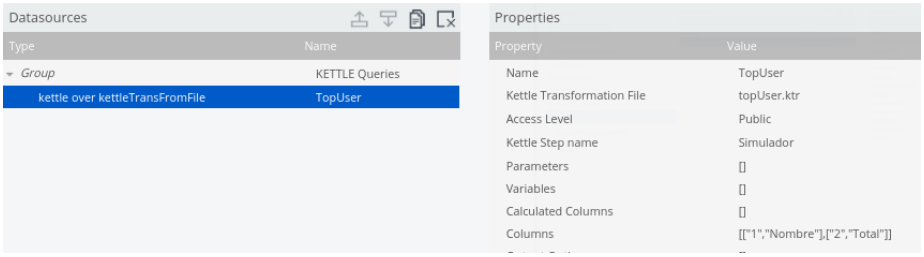


Figura 67: Definición del datasource topUser

En la distribución del layaout se deben configurar los encabezados que identifican al dashboard, a la tabla y al gráfico de torta.

Figura 68: Código HTML del encabezado del dashboard top user

Figura 69: Código HTML del encabezado de la tabla top user

Figura 70: Código HTML del encabezado del gráfico top user

Para la configuración de la tabla se debe configurar la cabecera de las columnas y los parámetros

Cuadro 5: Cabecera y parámetros de top user

Column Headers	[Nombre de Usuario, Total]
Parameters	[[1, Nombre], [2, Total]]

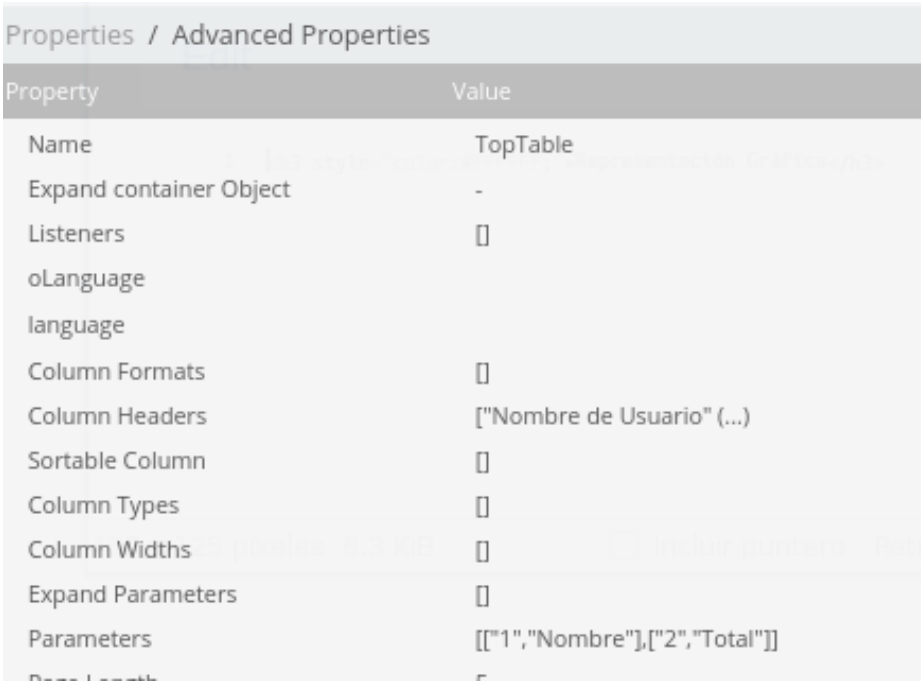


Figura 71: Configración de la cabecera y parametros de top user

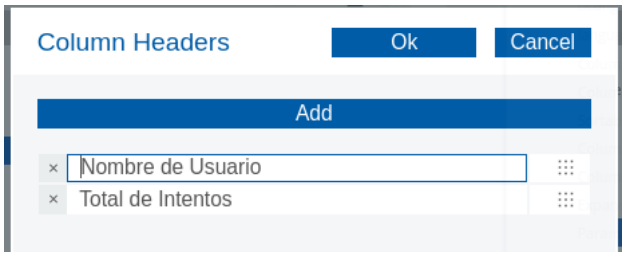


Figura 72: Configración del column headers

7.3.4.3 Dashboards combinación de user y password

El dahsboard muestra las 5 combinaciones de user y password más utilizadas en los intentos de acceso. Se visualiza un listado con el total de intentos de los passwords y se cuenta con un gráfico de torta para representar visualmente la proporción entre esos 5 passwords.

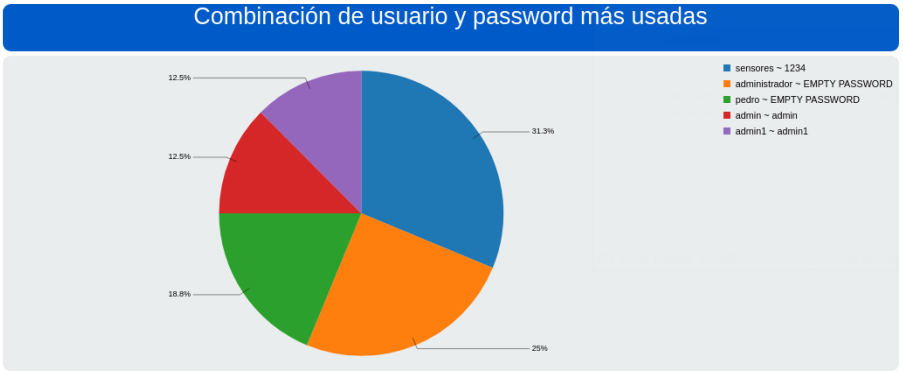


Figura 73: Dashboad combinación de usuario y password más usada

Para confeccionar el dahsboards, lo primero que se debe realizar es la definición de la fuente de dato con la que se va a trabajar. En este caso "**combinaciónUserPassword.ktr**". Se debe indicar cuál es el paso, Simulador, que va a proveer los datos.

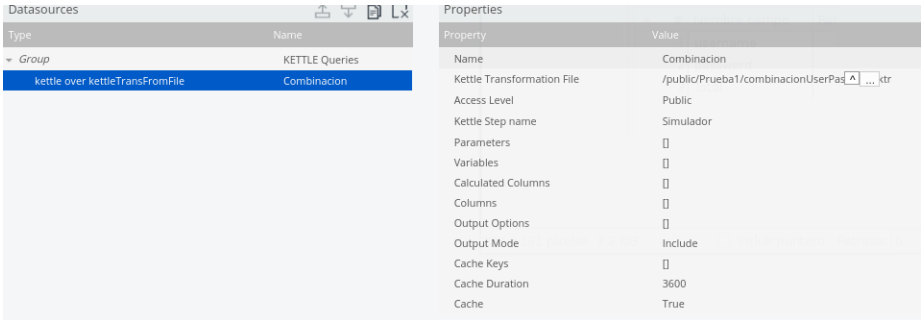


Figura 74: Definición del datasource para el dahsboard combinación

Posteriormente se define el layout. En este caso layout con dos filas y dos espacios. La primera fila contiene la cabecera del dashboard y la segunda fila contiene el gráfico que se va a visualizar.

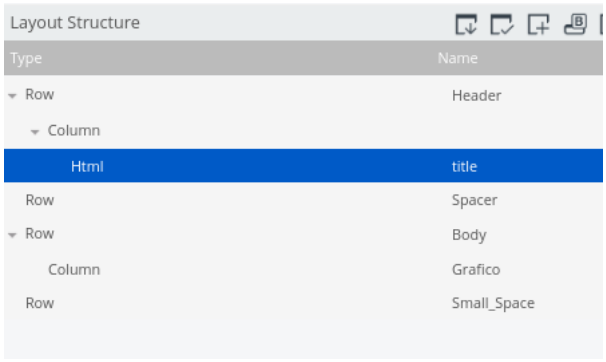


Figura 75: Layout del dahsboard combinación

Se cambia el título que muestra la cabecera.

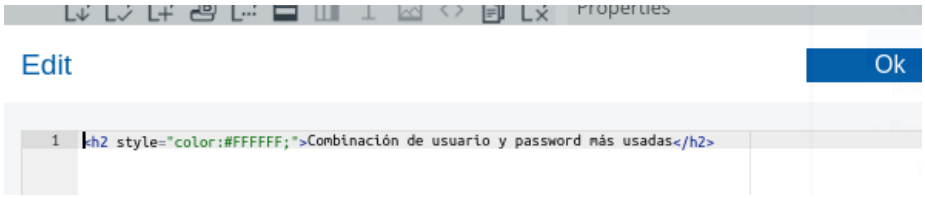


Figura 76: Código HTML de la cabecera del dashboard combinación

Al contenedor se le da un ancho de 12 puntos para que ocupe todo el espacio.

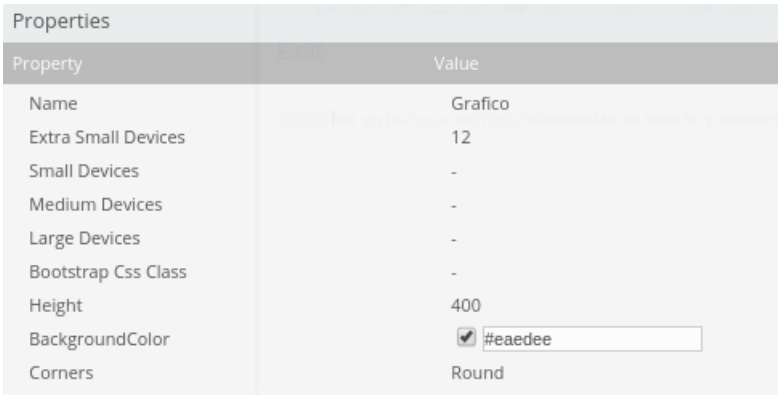


Figura 77: Configuración del contenedor del gráfico

Posteriormente se procede a configurar el gráfico que se va a mostrar. En este caso se utiliza un gráfico de torta.

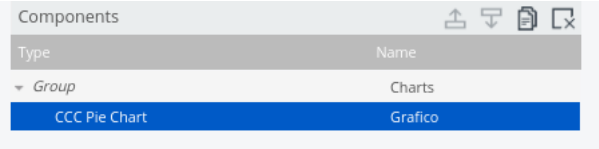


Figura 78: Selección del gráfico de torta

Se le da un nombre al componente. Del mismo modo se debe indicar cuál va a ser la fuente de datos a partir de la cual se va a alimentar el gráfico y se indica cual es el contenedor HTML en el cual se va a incrustar.

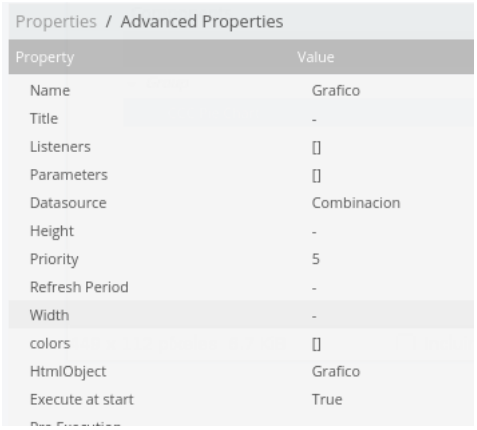


Figura 79: Configuración del gráfico de torta

Se configura la leyenda del gráfico cambiando el tamaño de la letra a 12 px y ubicandola a la derecha del gráfico.

Figura 80: Configuración del gráfico de torta 2

Por último, se configura la forma en la cual se muestran los valores en el gráfico con el fin de mostrar únicamente el porcentaje.

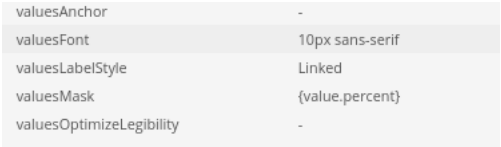


Figura 81: Configuración del gráfico de torta 3

8. Conclusión

Mediante el presente proyecto se han podido aprovechar los datos que genera el sensor Cowrie dentro de la honeynet brindándoles de ésta manera la posibilidad a los altos mandos de disponer de información relevante y de forma amigable.

A través de este trabajo se ha podido armar el prototipo de infraestructura que se había propuesto pudiendo almacenar en un storage NoSQL los datos que genera el honeypot Cowrie. Además se logro explotarlos exitosamente a través de las herramientas propuestas e inclusive pudiendo generar una serie de dashboards con información condensada para el usuario final.

Este desarrollo ha permitido establecer los mecanismos, tanto a nivel conceptual como práctico, necesarios que posibiliten el aprovechamiento de los datos de un sensor y por extensión se pueden llevar a aprovechar los datos que generen otros sensores.

Por otra parte, desde el rol de especialista de seguridad informática y más específicamente en relación a la explotación de datos que generan las herramientas de seguridad informática, se pueden aportar los conocimientos necesarios para el asesoramiento en el proceso de toma de decisiones a fin de mejorar la seguridad de la información en la organización.

Para finalizar, es posible dejar abiertas diversas líneas de trabajo a desarrollar. Entre ellas, se pueden mencionar la profundización de los procesos planteados en el presente trabajo y la explotación de los datos y eventos en tiempo real. Del mismo modo se le puede dar continuidad a través de la minería de datos para generar perfiles de ataque.

9. Anexo

9.1. Armado de Máquinas Virtuales

Para el armado de las maquinas virtuales se utiliza "**Virtual Box**" [54, 55] como software de virtualización. Lo primero es la creación de una maquina virtual nueva y seleccionar **Ubuntu de 64 bits** como sistema operativo. Posteriormente se aplica "**siguiente**" para avanzar en la pantalla.

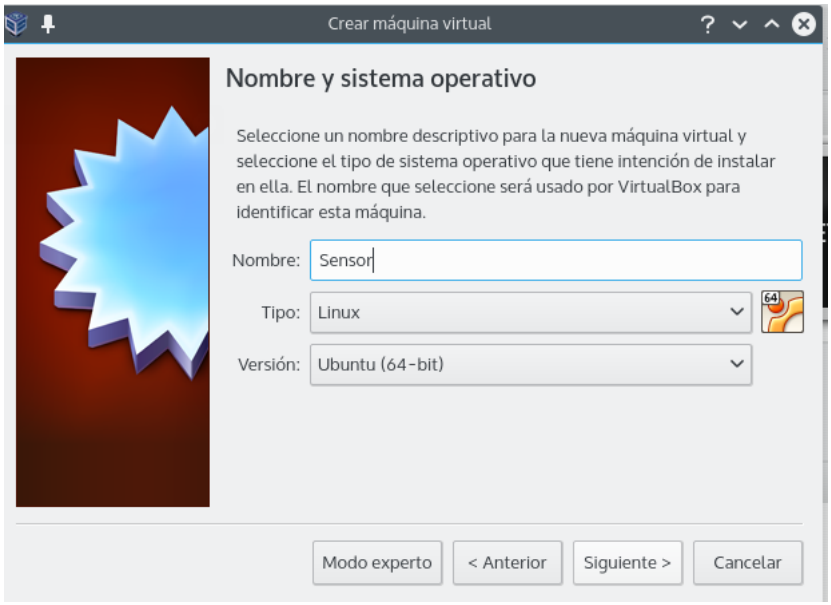


Figura 82: Creanción de la VM sensor

A continuación se procede a seleccionar la cantidad de memoria ram que va a utilizar la VM. En este caso 2.5 GB. Se debe hacer click en "**siguiente**" para pasar a la siguiente pantalla.

Figura 83: Configuración de memoria RAM

Posteriormente se debe seleccionar el tipo de disco y la capacidad del mismo. Para este caso se va a crear un disco virtual con capacidad de 60 GB, con la opción "**Reservado Dinámicamente**".

Figura 84: Creanción del disco duro de la VM sensor

Figura 85: Tipo de almacenamiento

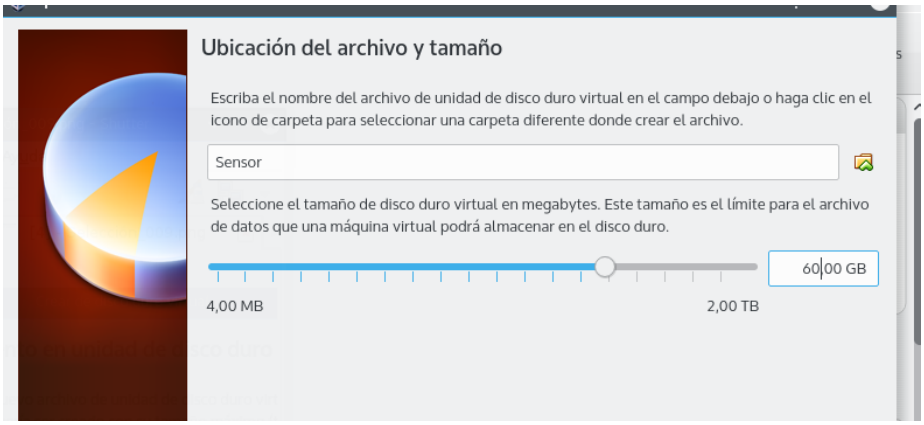


Figura 86: Ubicación y tamaño del disco

Una vez finalizado el proceso de crear el disco, también finaliza el proceso de creación de la maquina virtual.

Para crear la otra VM se procede de la misma manera. Una vez concluida la creación de dichas VMs se arma un grupo de trabajo para nuclear ambas Vms.

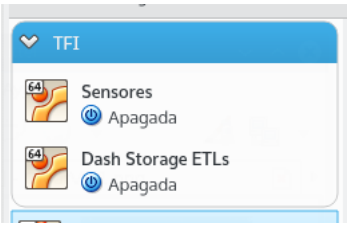


Figura 87: Agrupación de VMs

Una vez que se tienen las VM's creadas, se procede a instalar los sistemas

operativos en ellas. El procedimiento va a ser el mismo para ambas VM's. De esta manera se va a detallar la instalación del SO en una sola de las VM's.

En primera medida, se debe cargar la imagen de Ubuntu en la VM y esperar a que inicie el instalador. Una vez allí, seleccionar el idioma de instalación, en este caso "**Español**" y hacer click en instalar.

Figura 88: Pantalla de bienvenida de la instalación de Ubuntu

Cabe destacar que estos instaladores/wizar que poseen actualmente la mayoría de los sistemas GNU/Linux son sencillos y solamente basta con hacer click en "**siguiente**" para realizar una instalación por defecto.

Una vez que se pasa a la siguiente pantalla, se debe seleccionar la distribución del teclado. En este caso seleccionar "**Español (Latinoamericano)**" .

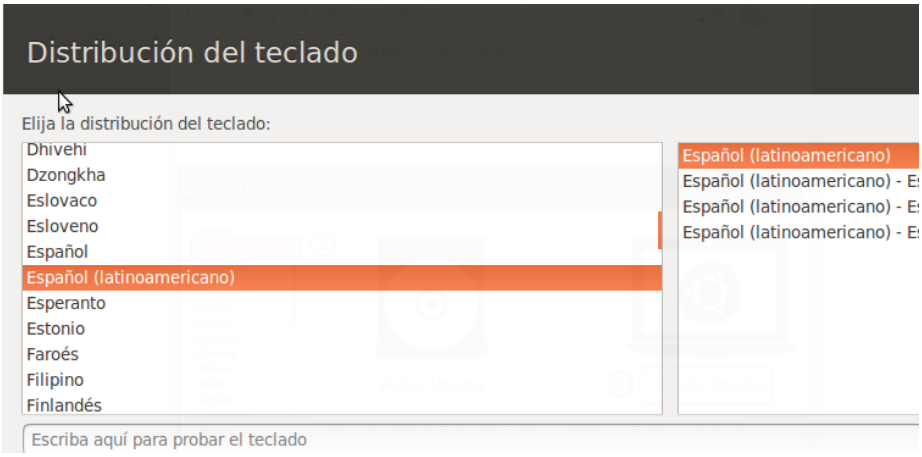


Figura 89: Distribución del teclado

Luego, se solicita si se va a utilizar todo el disco o particionar en distintos volúmenes. Usar la opción utilizar todo el disco.

Posteriormente se solicita crear el usuario, asignar un nombre a la maquina y establecer la contraseña.

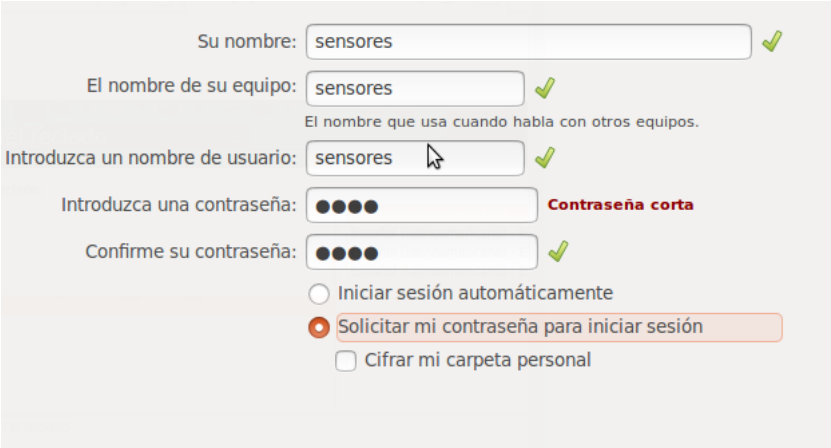


Figura 90: Definición del nombre de equipo, usuario y password

Una vez culminada la configuración, se comienza el proceso de instalación y descarga de paquetes necesarios. Finalizado este proceso, se puede utilizar la VM.

Como último paso, se guardan los datos de la VM en la parte de descripción con el fin de recordar si es necesario, alguna información importante.

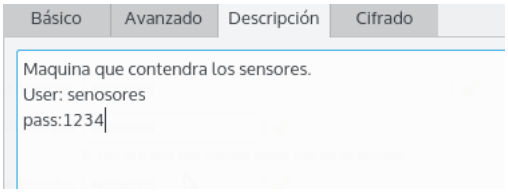


Figura 91: Solapa descripción de la VM

9.2. Instalación y configuración de Cowrie

Para instalar Cowrie se debe seguir la guía [56] de instalación que se encuentra en github. Dicha guía es bastante clara y bien detallada. Cabe aclara que el proceso de instalación que se ha implementado en este proyecto difiere en algunos puntos con la guía original debido a que existen opciones que no se han configurado por no ser vitales al momento de realizar la instalación.

Lo primero que se debe realizar es la instalación de las dependencias para Cowrie. Se instalan las dependencias para el entorno virtual de Python.

```
sensores@sensores:~$ sudo apt-get install git python-virtualenv libmpfr-dev libs
sl-dev libmpc-dev libffi-dev build-essential libpython-dev python2.7-minimal aut
hbind
```

Figura 92: Instalar dependencias

Posteriormente, se procede a clonar el repositorio

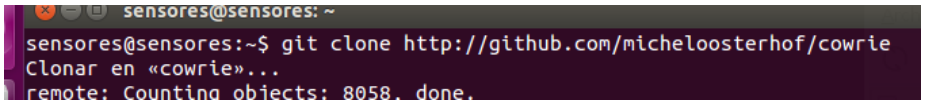


Figura 93: Clonar el repositorio

Se debe setear el entorno virtual, para ello es necesario ejecutar las siguientes instrucciones que se muestran en la siguiente imagen.

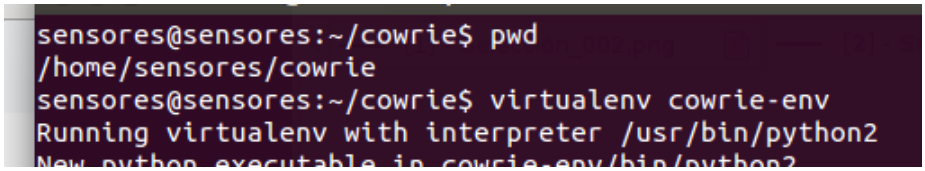


Figura 94: Setear entorno virtual

Una vez seteado el entorno virtual, se debe activarlo e instalar los paquetes necesarios.

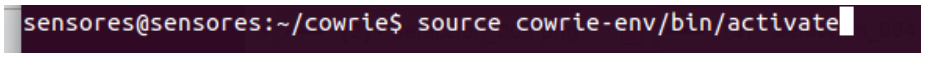


Figura 95: Activar entorno virtual

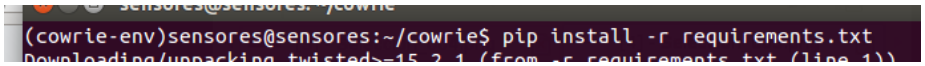


Figura 96: Instalar paquetes

Generar a DSA key

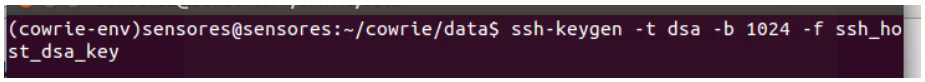


Figura 97: DSA key

Para que cowrie pueda correr sin problemas, es necesario que el directorio de nivel superior de los recursos se encuentre en el path del sistema de Python.

```
$export PYTHONPATH=/home/cowrie/cowrie
```

Por último queda correr cowrie

```
$bin/cowrie start
```

Bibliografía

[1] http://searchdatacenter.techtarget.com/es/opinion/el-impacto-de-big-data-a-la-seguridad-de-la-informacion. [Online]. Accedido: 02/08/2016.

[2] http://www.welivesecurity.com/la-es/2014/12/05/desafios-big-data-seguridad/. [Online]. Accedido: 02/08/2016.

[3] http://www.pandasecurity.com/spain/mediacenter/seguridad/big-data-eje-seguridad-empresas/. [Online]. Accedido: 02/08/2016.

[4] https://www.ibm.com/developerworks/vn/library/contest/dw-freebooks/tim_hieu_big_data/understanding_bigdata.pdf. [Online]. Accedido: 12/09/2016.

[5] Feris Thia Manoj R. Patil. *Pentaho for Big Data Analytics*. Packt Publishing, 2013.

[6] http://community.pentaho.com/projects/data-integration/. [Online]. Accedido: 12/09/2016.

[7] Luis Joayanes Aguilar. *Big Data Análisis de grandes volúmenes de datos en organizaciones*. Alfaomega, 2013.

[8] https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/. [Online]. Accedido: 28/03/2017.

[9] http://comein.uoc.edu/divulgacio/comein/es/numero37/articles/article-eva-ortoll.html. [Online]. Accedido: 06/04/2017.

[10] http://www.iic.uam.es/innovacion/big-data-caracteristicas-mas-importantes-7-v/. [Online]. Accedido: 06/04/2017.

[11] https://www.oracle.com/lad/big-data/index.html. [Online]. Accedido: 06/04/2017.

[12] http://searchdatacenter.techtarget.com/es/opinion/el-impacto-de-big-data-a-la-seguridad-de-la-informacion. [Online]. Accedido: 01/04/2017.

[13] http://web.ornl.gov/ jgoodall/goodall-vizsec07.pdf. [Online]. Accedido: 01/04/2017.

[14] https://www.welivesecurity.com/la-es/2014/12/05/desafios-big-data-seguridad/. [Online]. Accedido: 01/04/2017.

[15] http://www.pandasecurity.com/spain/mediacenter/seguridad/big-data-eje-seguridad-empresas/. [Online]. Accedido: 01/04/2017.

[16] https://www-03.ibm.com/security/solution/intelligence-big-data/. [Online]. Accedido: 02/04/2017.

[17] http://data.bsa.org/wp-content/uploads/2015/10/bsadatastudy_es. [Online]. Accedido: 10/04/2017.

[18] https://www.elevenpaths.com/es/tecnologia/tacyt/index.html. [Online]. Accedido: 13/04/2017.

[19] https://hacking-etico.com/2012/12/03/honeypot-un-tarro-de-miel-para-los-atacantes/. [Online]. Accedido: 15/04/2017.

[20] https://www.sans.org/security-resources/idfaq/what-is-a-honeypot/1/9. [Online]. Accedido: 15/04/2017.

[21] http://www.cybsec.com/upload/espe_honeypots.pdf. [Online]. Accedido: 15/04/2017.

[22] http://www.asc.unam.mx/descarga.dsc?arch=247. [Online]. Accedido: 17/04/2017.

[23] http://searchsecurity.techtarget.com/definition/honeynet. [Online]. Accedido: 20/04/2017.

[24] https://web.dit.upm.es/ jlopez/publicaciones/mundointernet04.pdf. [Online]. Accedido: 21/04/2017.

[25] https://github.com/jordan-wright/elastichoney. [Online]. Accedido: 20/11/2017.

[26] https://jordan-wright.com/blog/2015/03/23/introducing-elastichoney-an-elasticsearch-honeypot/. [Online]. Accedido: 20/11/2017.

[27] https://github.com/mushorg/glastopf. [Online]. Accedido: 20/11/2017.

[28] https://revista.seguridad.unam.mx/numero25/glastopf-honeypot-de-aplicaciones-web-i. [Online]. Accedido: 20/11/2017.

[29] https://revista.seguridad.unam.mx/numero26/glastopf-honeypot-de-aplicaciones-web-ii. [Online]. Accedido: 20/11/2017.

[30] https://github.com/foospidy/honeypy. [Online]. Accedido: 20/11/2017.

[31] https://labs.signalsciences.com/introduction-to-honeypy-honeydb. [Online]. Accedido: 20/11/2017.

[32] http://conpot.org/. [Online]. Accedido: 20/11/2017.

[33] https://github.com/mushorg/conpot. [Online]. Accedido: 20/11/2017.

[34] https://github.com/pjlantz/hale. [Online]. Accedido: 20/11/2017.

[35] https://www.honeynet.org/node/871. [Online]. Accedido: 20/11/2017.

[36] https://github.com/honeynet/ghost-usb-honeypot. [Online]. Accedido: 20/11/2017.

[37] http://www.hackplayers.com/2012/06/ghost-un-honeypot-para-malware-que-se.html. [Online]. Accedido: 20/11/2017.

[38] https://github.com/micheloosterhof/cowrie. [Online]. Accedido: 20/11/2017.

[39] http://www.micheloosterhof.com/cowrie/. [Online]. Accedido: 20/11/2017.

[40] https://github.com/desaster/kippo. [Online]. Accedido: 20/11/2017.

[41] http://www.dataprix.com/blogs/respinosamilla/herramientas-etl-que-son-para-que-valen-productos-mas-conocidos-etl-s-open-sour. [Online]. Accedido: 20/11/2017.

[42] https://www-01.ibm.com/software/data/etl/. [Online]. Accedido: 20/11/2017.

[43] https://www.mongodb.com/nosql-explained. [Online]. Accedido: 20/11/2017.

[44] https://www.youtube.com/watch?v=gqgengb1drq. [Online]. Accedido: 20/11/2017.

[45] https://www.youtube.com/watch?v=kklwu-fnkxw. [Online]. Accedido: 20/11/2017.

[46] https://www.couchbase.com/nosql-resources/why-nosql. [Online]. Accedido: 20/11/2017.

[47] https://aws.amazon.com/es/nosql/. [Online]. Accedido: 20/05/2017.

[48] http://sqoop.apache.org/. [Online]. Accedido: 20/05/2017.

[49] http://www.pentaho.com/product/data-integration. [Online]. Accedido: 20/05/2017.

[50] http://community.pentaho.com/projects/data-integration/. [Online]. Accedido: 30/05/2017.

[51] http://www.json.org/json-es.html. [Online]. Accedido: 30/05/2017.

[52] https://sehque.wordpress.com/2015/07/31/add-mysql-to-cowrie/. [Online]. Accedido: 05/06/2017.

[53] https://www.mysql.com/. [Online]. Accedido: 08/06/2017.

[54] https://www.virtualbox.org/. [Online]. Accedido: 05/05/2017.

[55] https://es.wikipedia.org/wiki/virtualbox. [Online]. Accedido: 05/05/2017.

[56] https://github.com/micheloosterhof/cowrie/blob/master/install.md. [Online]. Accedido: 05/05/2017.