Reducción del riesgo de exposición e infección mediante la implementación de un Firewall DNS

Juan Zambrano Burgos

Máster Universitario en Ciberseguridad
M1.887 - TFM - Seguridad empresarial

Nombre Tutor/a de TF
Borja Guaita Perez
Profesor/a responsable de la asignatura
Victor Garcia Font

Fecha Entrega
28/07/2024

Universitat Oberta de Catalunya

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Reducción del riesgo de exposición e infección mediante la implementación de un Firewall DNS*
Nombre del autor:	*Juan Zambrano Burgos*
Nombre del consultor/a:	*Borja Guaita*
Nombre del PRA:	*Victor Garcia Font*
Fecha de entrega (mm/aaaa):	*07/2024*
Titulación o programa:	*Máster Universitario en Ciberseguridad y Privacidad*
Área del Trabajo Final:	*M1.887 - TFM - Seguridad empresarial*
Idioma del trabajo:	*Castellano*
Palabras clave	*DNS, C2C, IOC, RPZ*

Resumen del Trabajo

La seguridad cibernética es un aspecto crucial en la gestión de redes informáticas, especialmente para las pequeñas y medianas empresas (pymes). La prevención de amenazas y la detección temprana de máquinas infectadas son fundamentales para salvaguardar la integridad de la red y proteger los activos digitales de la organización. En este sentido, el protocolo DNS emerge como una herramienta clave debido a su omnipresencia en cualquier actividad de navegación web. Al aprovechar las capacidades del protocolo DNS, las empresas pueden fortalecer sus defensas cibernéticas al prevenir el acceso a sitios maliciosos y detectar rápidamente posibles infecciones en la red.

El objetivo principal de este proyecto de tesis es diseñar y desarrollar una herramienta efectiva destinada a reducir el riesgo de exposición e infección de las máquinas en el entorno de las pequeñas y medianas empresas (pymes). Esta herramienta estará enfocada en dos aspectos fundamentales de la seguridad cibernética: la prevención del acceso a sitios maliciosos y la detección temprana de infecciones en la red. Su implementación permitirá fortalecer las defensas de las pymes contra las amenazas cibernéticas, garantizando así la integridad y la continuidad de sus operaciones comerciales.

Para ello se ha realizado una investigación y análisis sobre herramientas empresariales que poseen características similares para posteriormente realizar un diseño y desarrollo de una herramienta que funcionalmente pueda ser utilizada libremente en Pymes. Posteriormente se ha desarrollado una herramienta que se basa en el protocolo DNS sobre un servidor Ubuntu que permite alojar un servicio DNS con una interfaz de gestión web. Esta

herramienta es alimentada de feeds de Indicadores de Compromiso (IoCs) de IPs maliciosas para detectar y bloquear amenazas de manera proactiva.

Finalmente, la herramienta propuesta proporciona una estrategia accesible y efectiva para fortalecer la seguridad de las redes de pymes, protegiendo sus activos digitales y garantizando la continuidad de sus operaciones en un entorno cada vez más hostil.

Abstract

Cybersecurity is a crucial aspect in managing computer networks, especially for small and medium-sized enterprises (SMEs). Preventing threats and early detection of infected machines are essential to safeguard the network integrity and protect the organization's digital assets. In this regard, the DNS protocol emerges as a key tool due to its omnipresence in any web browsing activity. By leveraging the capabilities of the DNS protocol, companies can strengthen their cyber defenses by preventing access to malicious sites and quickly detecting potential infections on the network.

The main objective of this thesis project is to design and develop an effective tool aimed at reducing the risk of exposure and infection of machines in the environment of small and medium-sized enterprises (SMEs). This tool will focus on two fundamental aspects of cybersecurity: preventing access to malicious sites and early detection of infections on the network. Its implementation will enable SMEs to strengthen their defenses against cyber threats, thus ensuring the integrity and continuity of their business operations.

To achieve this, research and analysis have been conducted on enterprise tools with similar characteristics, followed by the design and development of a tool that can be freely used in SMEs. Subsequently, a tool based on the DNS protocol on an Ubuntu server has been developed, allowing hosting a DNS service with a web management interface. This tool is fed with feeds of Indicators of Compromise (IoCs) of malicious IP addresses to detect and block threats proactively.

Ultimately, the proposed tool provides an accessible and effective strategy to strengthen the security of SME networks, protecting their digital assets and ensuring the continuity of their operations in an increasingly hostile environment.

Índice

Lista de figuras

1. Introducción

La seguridad cibernética es un aspecto crucial en la gestión de redes informáticas, especialmente para las pequeñas y medianas empresas (pymes). La prevención de amenazas y la detección temprana de máquinas infectadas son fundamentales para salvaguardar la integridad de la red y proteger los activos digitales de la organización. En este sentido, el protocolo DNS emerge como una herramienta clave debido a su omnipresencia en cualquier actividad de navegación web. Al aprovechar las capacidades del protocolo DNS, las empresas pueden fortalecer sus defensas cibernéticas al prevenir el acceso a sitios maliciosos y detectar rápidamente posibles infecciones en la red.

El objetivo principal de este proyecto de tesis es diseñar y desarrollar una herramienta efectiva destinada a reducir el riesgo de exposición e infección de las máquinas en el entorno de las pequeñas y medianas empresas (pymes). Esta herramienta estará enfocada en dos aspectos fundamentales de la seguridad cibernética: la prevención del acceso a sitios maliciosos y la detección temprana de infecciones en la red. Su implementación permitirá fortalecer las defensas de las pymes contra las amenazas cibernéticas, garantizando así la integridad y la continuidad de sus operaciones comerciales.

Para ello se ha realizado una investigación y análisis sobre herramientas empresariales que poseen características similares para posteriormente realizar un diseño y desarrollo de una herramienta que funcionalmente pueda ser utilizada libremente en Pymes. Posteriormente se ha desarrollado una herramienta que se basa en el protocolo DNS sobre un servidor Ubuntu que permite alojar un servicio DNS con una interfaz de gestión web. Esta herramienta es alimentada de feeds de Indicadores de Compromiso (IoCs) de IP maliciosas para detectar y bloquear amenazas de manera proactiva.

Finalmente, la herramienta propuesta proporciona una estrategia accesible y efectiva para fortalecer la seguridad de las redes de pymes, protegiendo sus activos digitales y garantizando la continuidad de sus operaciones en un entorno cada vez más hostil.

1.1. Contexto y justificación del Trabajo

La seguridad cibernética se ha vuelto cada vez más crucial en el panorama empresarial actual, especialmente a raíz de las crecientes amenazas y ataques cibernéticos que enfrentan las organizaciones en todo el mundo. Según el informe de Check Point 2023 [1], los CISOs tuvieron que lidiar con una cantidad considerable de desafíos en 2022. Los ataques globales aumentaron un 28% en el tercer trimestre de 2022 en comparación con el mismo período en 2021, y el promedio semanal de ataques por organización en todo el mundo superó los 1.130. Esta tendencia no muestra signos de desaceleración para 2023, con un aumento en los exploits de ransomware y el hacktivismo movilizado por el estado impulsado por conflictos internacionales.

Las pequeñas y medianas empresas (pymes), en particular, enfrentan desafíos únicos en cuanto a la seguridad cibernética, ya que muchas veces no tienen los

recursos financieros para invertir en soluciones de seguridad costosas. Sin embargo, existen opciones disponibles para las pymes, como el uso de software de código abierto (opensource), que ofrece alternativas de bajo costo para proteger sus redes y activos digitales. Implementar una herramienta opensource que permita crear salvaguardas para prevenir amenazas y detectar infecciones puede reducir significativamente el riesgo de exposición a amenazas cibernéticas para las pymes.

La reducción del riesgo de exposición a amenazas se asocia directamente con la habilitación de salvaguardas o controles de seguridad. Las pymes, al tener menos recursos disponibles para invertir en ciberseguridad, pueden beneficiarse enormemente de la implementación de herramientas opensource que les permitan crear estas salvaguardas y proteger sus redes de manera efectiva.

Después de implementar esta herramienta, el objetivo es reducir significativamente las infecciones de los usuarios y detectar rápidamente cuando están infectados, garantizando así la integridad y la continuidad de las operaciones comerciales de las pymes en un entorno cibernético cada vez más hostil.

1.2. Objetivos del Trabajo

Los principales objetivos de este trabajo de fin de máster son los siguientes:

Objetivos de investigación:

- Investigación sobre cómo utilizar un Firewall DNS para funciones de prevención de amenazas durante la navegación web.
- Investigación sobre como detectar dispositivos infectados dentro la red.
- Investigación sobre las posibilidades que ofrecen las herramientas de código abierto para pymes y su integración con terceros.
- Estudio del funcionamiento y configuración de BIND 9 en el contexto de un Firewall DNS.

Objetivos de implantación:

- Instalación y configuración de un Firewall DNS basado en BIND 9 montado en un Ubuntu Server.
- Aprender a configurar reglas y políticas de seguridad en el Firewall DNS para prevenir el acceso a sitios maliciosos y filtrar contenido inapropiado.
- Integrar un SIEM ELK con el Firewall DNS para mejorar la detección de amenazas y realizar pruebas de detección.
- Configurar un sistema de monitorización para el Firewall DNS y facilitando su gestión y visualización.

Objetivos de entrega:

- Desarrollar y entregar las entregas parciales del proyecto en tiempo y forma.

- Elaborar la memoria final del trabajo, documentando el proceso de investigación, implantación y resultados obtenidos.

1.3. Impacto en sostenibilidad, ético-social y de diversidad

Para nuestro proyecto de Firewall DNS de código abierto, podemos identificar los siguientes impactos positivos en la dimensión de reconocimiento y respeto a la diversidad funcional, social, cultural, económica, política, lingüística y de género:

Implementar un Firewall DNS de código abierto promueve un entorno inclusivo y accesible para todos los usuarios, independientemente de su diversidad funcional, social, cultural, económica, política, lingüística y de género. Al ofrecer una solución de seguridad cibernética de código abierto, se elimina la barrera económica para acceder a tecnologías de protección, lo que permite que personas de diferentes contextos y recursos puedan beneficiarse de la seguridad en línea. Además, el enfoque de código abierto fomenta la diversidad cultural y lingüística al permitir la participación de comunidades globales en el desarrollo y mejora del proyecto, asegurando que las soluciones propuestas sean culturalmente relevantes y accesibles para una amplia gama de usuarios. En última instancia, esto contribuye a crear un entorno inclusivo y equitativo donde todos tienen la oportunidad de contribuir y beneficiarse de la seguridad cibernética, independientemente de sus diferencias individuales.

Al diseñar y poner en práctica un Firewall DNS, se promueve una cultura ética y responsable en el ámbito de la seguridad cibernética. Esto implica comprender y aplicar los principios éticos relacionados con la privacidad, la seguridad y el respeto a los derechos digitales de los usuarios. Se busca diseñar políticas de filtrado y acceso que respeten la privacidad de los usuarios y protejan sus datos sensibles de manera ética y responsable.

Además, al desarrollar el proyecto de Firewall DNS, se fomenta una actitud ética, honesta y cívica en el trabajo académico y profesional. Se evita el plagio y cualquier forma de uso indebido de información o trabajo de terceros, lo que refleja un compromiso con la integridad intelectual y el reconocimiento adecuado de las contribuciones de otros en el ámbito académico y profesional. Se incentiva la colaboración responsable y se promueve un entorno de respeto hacia el trabajo y las ideas de los demás, lo que contribuye a una comunidad académica y profesional más ética y transparente.

1.4. Enfoque y método seguido

El enfoque de este proyecto, adaptado al contexto de las pequeñas y medianas empresas (Pymes), se centrará en evaluar la viabilidad y utilidad de la tecnología DNS para la prevención de amenazas y la detección de tráfico malicioso en redes empresariales.

El enfoque de este proyecto, adaptado al contexto de las pequeñas y medianas empresas (Pymes), se centrará en dos partes principales:

Parte teórica de investigación:

En esta etapa, se llevará a cabo una investigación exhaustiva sobre las herramientas pertinentes para el proyecto, que incluyen:

- Exploración de herramientas para el control de consultas DNS de los usuarios de la red, centrándose en DNS BIND 9.
- Investigación sobre la capacidad de correlación de información utilizando un SIEM (Security Information and Event Management) basado en la herramienta ELK Stack.
- Evaluación de las capacidades de integración de estas herramientas y análisis de los casos de uso disponibles.
- Identificación y análisis de dashboards o tableros de control disponibles para la supervisión y gestión de amenazas.

Implementación:

En esta fase, se procederá a la implementación práctica de las herramientas identificadas. Esto incluirá:

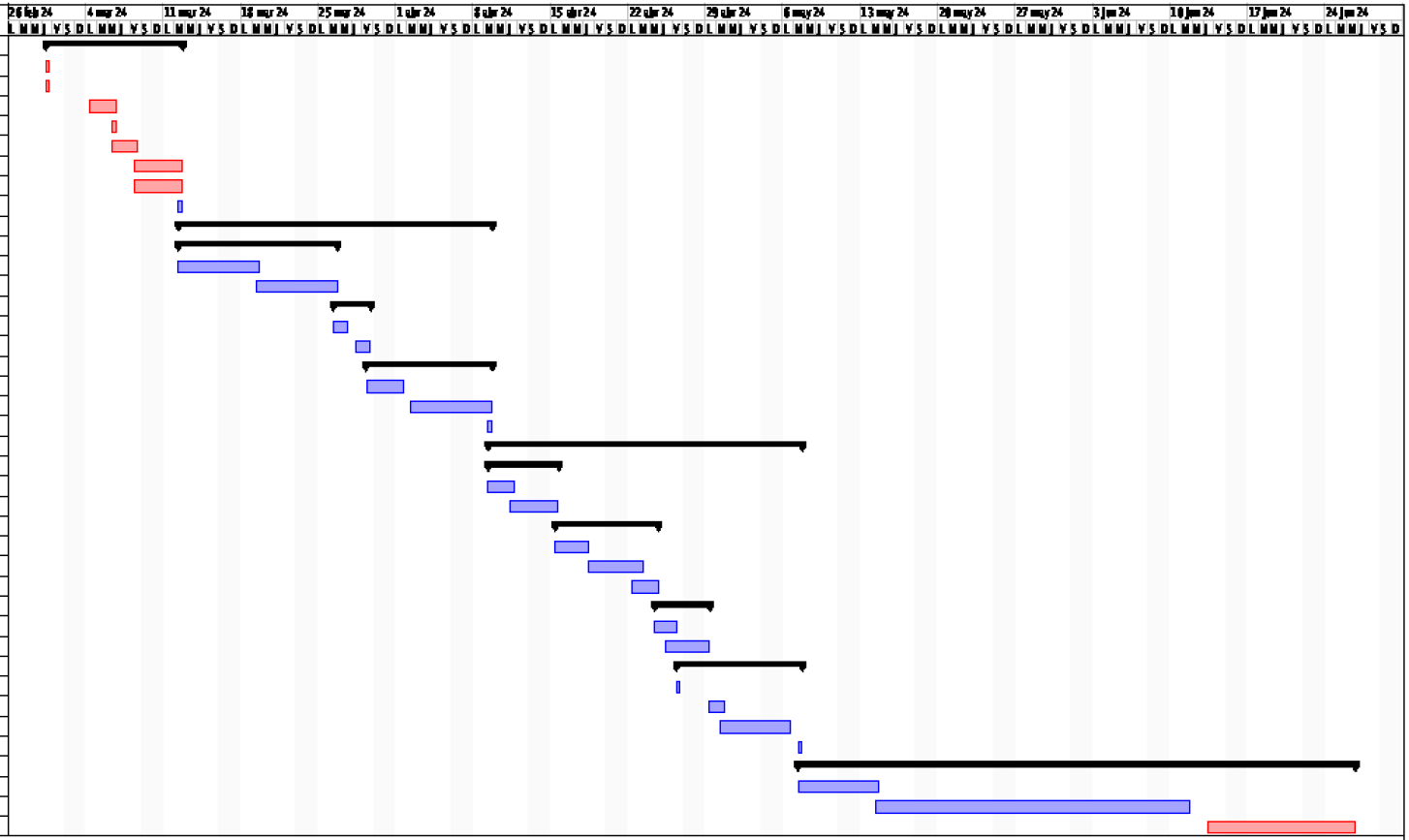
- Desarrollo de la instalación de las herramientas ELK Stack y DNS BIND 9 en el entorno de la red de la Pyme.
- Diseño e implementación de la integración de estas herramientas, asegurando su funcionamiento conjunto y su capacidad para compartir información relevante.
- Desarrollo de la integración con herramientas de terceros, lo que podría incluir sistemas de alerta temprana o feeds de información sobre amenazas.

1.5. Planificación del Trabajo

Se presenta la planificación temporal del trabajo mediante diagrama de Gantt:

	Nombre	Duracion	Inicio	Terminado
1	**1. Planificación**	9 days?	**29/02/24 8:00**	**12/03/24 17:00**
2	Contexto y justificación del Trabajo	1 day?	29/02/24 8:00	29/02/24 17:00
3	Objetivos del Trabajo	1 day?	29/02/24 8:00	29/02/24 17:00
4	Impacto en sostenibilidad, ético-social y de diversi	3 days?	4/03/24 8:00	6/03/24 17:00
5	Enfoque y método seguido	1 day?	6/03/24 8:00	6/03/24 17:00
6	Planificación del Trabajo	3 days?	6/03/24 8:00	8/03/24 17:00
7	Breve sumario de productos obtenidos	3 days?	8/03/24 8:00	12/03/24 17:00
8	Breve descripción de los otro capitulos de la memo	3 days?	8/03/24 8:00	12/03/24 17:00
9	Entrega del Plan de trabajo PEC1	1 day?	12/03/24 8:00	12/03/24 17:00
10	**2. Investigación**	21 days?	**12/03/24 8:00**	**9/04/24 17:00**
11	**Investigación sobre DNS**	11 days?	**12/03/24 8:00**	**26/03/24 17:00**
12	Estudio de BIND DNS	6 days?	12/03/24 8:00	19/03/24 17:00
13	Evalución de casos de uso con DNS BIND	6 days?	19/03/24 8:00	26/03/24 17:00
14	**Investigación sobre operación de SIEM**	4 days?	**26/03/24 8:00**	**29/03/24 17:00**
15	Estudio de ELK Stack	2 days?	26/03/24 8:00	27/03/24 17:00
16	Evaluación de casos de uso asociado a DNS	2 days?	28/03/24 7:00	29/03/24 17:00
17	**Investigación de Integración**	8 days?	**29/03/24 7:00**	**9/04/24 17:00**
18	Investigación sobre opciones de Integración	2 days?	29/03/24 7:00	1/04/24 17:00
19	Investigación sobre fuentes de terceros	6 days?	2/04/24 7:00	9/04/24 17:00
20	Entrega del PEC2	1 day?	9/04/24 8:00	9/04/24 17:00
21	**3. Implantación**	21 days?	**9/04/24 8:00**	**7/05/24 17:00**
22	**Disponibilización de recursos**	5 days?	**9/04/24 8:00**	**15/04/24 17:00**
23	Instalación de VMware Player	3 days?	9/04/24 8:00	11/04/24 17:00
24	Implementación de Servidor basado en Ubuntu	3 days?	11/04/24 8:00	15/04/24 17:00
25	**Implementación de DNS BIND**	7,875 days	**15/04/24 9:00**	**24/04/24 17:00**
26	Instalación del Servicio DNS	3 days	15/04/24 9:00	18/04/24 9:00
27	Configuración del servicio DNS para navegación	3 days	18/04/24 9:00	23/04/24 9:00
28	Configuración de una RPZ	3 days	21/04/24 9:00	24/04/24 17:00
29	**Implementación de ELK Stack**	3 days	**24/04/24 9:00**	**29/04/24 9:00**
30	Instalación de ELK	2 days	24/04/24 9:00	26/04/24 9:00
31	Configuración basica de Dashboard	2 days	25/04/24 9:00	29/04/24 9:00
32	**Integración de entre sistemas**	8 days?	**26/04/24 8:00**	**7/05/24 17:00**
33	Integración servicio DNS BIND y ELK	1 day	26/04/24 8:00	26/04/24 17:00
34	Implementación de dashboard con data integrad	2 days	29/04/24 8:00	30/04/24 17:00
35	Integración con fuentes abiertas	5 days	30/04/24 8:00	6/05/24 17:00
36	Entrega PEC 3	1 day?	7/05/24 8:00	7/05/24 17:00
37	**4. Presentación**	117 days?	**7/05/24 8:00**	**16/10/24 17:00**
38	Conclusiones sobre el trabajo	6 days?	7/05/24 8:00	14/05/24 17:00
39	Elaboración de la memoria final	21 days?	14/05/24 8:00	11/06/24 17:00
40	Presentación y defensa del TFM	79 days?	28/06/24 8:00	16/10/24 17:00

	Nombre	Duracion	Inicio	Terminado
1	**F1. Planificación**	**9 days?**	**29/02/24 8:00**	**12/03/24 17:00**
2	Contexto y justificación del Trabajo	1 day?	29/02/24 8:00	29/02/24 17:00
3	Objetivos del Trabajo	1 day?	29/02/24 8:00	29/02/24 17:00
4	Impacto en sostenibilidad, ético–social y de diver...	3 days?	4/03/24 8:00	6/03/24 17:00
5	Enfoque y método seguido	1 day?	6/03/24 8:00	6/03/24 17:00
6	Planificación del Trabajo	3 days?	6/03/24 8:00	8/03/24 17:00
7	Breve sumario de productos obtenidos	3 days?	8/03/24 8:00	12/03/24 17:00
8	Breve descripción de los otro capítulos de la me...	3 days?	8/03/24 8:00	12/03/24 17:00
9	Entrega del Plan de trabajo PEC1	1 day?	12/03/24 8:00	12/03/24 17:00
10	**E2. Investigación**	**21 days?**	**12/03/24 8:00**	**9/04/24 17:00**
11	**Investigación sobre DNS**	**11 days?**	**12/03/24 8:00**	**26/03/24 17:00**
12	Estudio de BIND DNS	6 days?	12/03/24 8:00	19/03/24 17:00
13	Evalución de casos de uso con DNS BIND	6 days?	19/03/24 8:00	26/03/24 17:00
14	**Investigación sobre operación de SIEM**	**4 days?**	**26/03/24 8:00**	**29/03/24 17:00**
15	Estudio de ELK Stack	2 days?	26/03/24 8:00	27/03/24 17:00
16	Evaluación de casos de uso asociado a DNS	2 days?	28/03/24 7:00	29/03/24 17:00
17	**Investigación de Integración**	**8 days?**	**29/03/24 7:00**	**9/04/24 17:00**
18	Investigación sobre opciones de Integración	2 days?	29/03/24 7:00	1/04/24 17:00
19	Investigación sobre fuentes de terceros	6 days?	2/04/24 7:00	9/04/24 17:00
20	Entrega del PEC2	1 day?	9/04/24 8:00	9/04/24 17:00
21	**E3. Implantación**	**21 days?**	**9/04/24 8:00**	**7/05/24 17:00**
22	**Disponibilización de recursos**	**5 days?**	**9/04/24 8:00**	**15/04/24 17:00**
23	Instalación de VMware Player	3 days?	9/04/24 8:00	11/04/24 17:00
24	Implementación de Servidor basado en Ubuntu	3 days?	11/04/24 8:00	15/04/24 17:00
25	**Implementación de DNS BIND**	**7,875 days**	**15/04/24 9:00**	**24/04/24 17:00**
26	Instalación del Servicio DNS	3 days	15/04/24 9:00	18/04/24 9:00
27	Configuración del servicio DNS para navegación	3 days	18/04/24 9:00	23/04/24 9:00
28	Configuración de una RPZ	3 days	21/04/24 9:00	24/04/24 17:00
29	**Implementación de ELK Stack**	**3 days**	**24/04/24 9:00**	**29/04/24 9:00**
30	Instalación de ELK	2 days	24/04/24 9:00	26/04/24 9:00
31	Configuración basica de Dashboard	2 days	25/04/24 9:00	29/04/24 9:00
32	**Integración de entre sistemas**	**8 days?**	**26/04/24 8:00**	**7/05/24 17:00**
33	Integración servicio DNS BIND y ELK	1 day	26/04/24 8:00	26/04/24 17:00
34	Implementación de dashboard con data integra...	2 days	29/04/24 8:00	30/04/24 17:00
35	Integración con fuentes abiertas	5 days	30/04/24 8:00	6/05/24 17:00
36	Entrega PEC 3	1 day?	7/05/24 8:00	7/05/24 17:00
37	**E4. Presentación**	**37 days?**	**7/05/24 8:00**	**26/06/24 17:00**
38	Conclusiones sobre el trabajo	6 days?	7/05/24 8:00	14/05/24 17:00
39	Elaboración de la memoria final	21 days?	14/05/24 8:00	11/06/24 17:00
40	Presentación y defensa del TFM	10 days	13/06/24 8:00	26/06/24 17:00



1.6. Análisis de riesgos

Para asegurar la viabilidad y el éxito de un proyecto, es esencial anticipar los diversos riesgos que pueden surgir a lo largo de sus diferentes fases de desarrollo. Comprender estos riesgos y preparar estrategias de contingencia adecuadas es fundamental para mitigar su impacto potencial. Este enfoque proactivo no solo mejora la resiliencia del proyecto frente a imprevistos, sino que también contribuye a una gestión más eficiente y a la optimización de recursos, asegurando así una mayor probabilidad de alcanzar los objetivos establecidos.

1.6.1. Retraso en actividades del cronograma del proyecto

El retraso en las actividades previstas en el cronograma del proyecto puede surgir por múltiples factores, como la subestimación de la complejidad técnica, demoras en la adquisición de hardware o software necesario, o retrasos en la entrega de componentes por parte de los proveedores. Este riesgo puede llevar a una prolongación del tiempo de desarrollo e implementación, afectando la planificación general y posiblemente incrementando los costos asociados.

Este es un riesgo aceptado, ya que se han estimado un tiempo holgado para la entrega final del proyecto.

1.6.2. Falla en el despliegue de la herramienta

La posibilidad de que surjan problemas técnicos durante el proceso de despliegue de los distintos Componentes de la herramienta de Firewall DNS es un riesgo inherente a proyectos de esta naturaleza. Esto puede deberse a incompatibilidades de software, configuraciones incorrectas o errores en el código. Tales fallos pueden resultar en un mal funcionamiento de la herramienta, comprometiendo la seguridad de la red y la eficacia del firewall.

Existe la amenaza de que parte de las funcionalidades afecte el sistema. Para la mitigación consideraremos la exclusión de la funcionalidad.

1.6.3. Falta de recursos en el servidor

Un cálculo inadecuado de los recursos necesarios en el servidor, como capacidad de procesamiento, memoria y almacenamiento, puede llevar a un rendimiento subóptimo de la herramienta de Firewall DNS. Esto puede afectar la capacidad del sistema para procesar eficientemente el tráfico de red, aumentando la latencia y disminuyendo la seguridad.

Para esta amenaza como mitigación se considera la adquisición de recursos para ampliar las capacidades del servidor, afectando los costos del proyecto.

1.6.4. Cortes energéticos

Los cortes de energía pueden causar interrupciones inesperadas en el servicio de firewall, dejando la red expuesta a ataques externos durante el periodo de inactividad. Además, pueden provocar pérdidas de datos o daños en el hardware, lo cual afectaría negativamente la continuidad del proyecto.

La PYME ha definido mitigar los riesgos utilizando una PSU dando continuidad a sistemas críticos como el del proyecto durante la etapa de implementación y puesta en marcha.

1.6.5. Disponibilidad del profesional

La dependencia de un profesional específico para la implementación puede convertirse en un riesgo si surge alguna situación que limite su disponibilidad, ya sea por enfermedad, cambio de empleo o cualquier otro compromiso personal. Esto puede resultar en retrasos significativos, ya que encontrar un reemplazo o transferir conocimientos especializados puede tomar tiempo.

Esta es una amenaza aceptada, dado que el proyecto solo puede depender del profesional designado al proyecto.

1.6.6. Costos del proyecto

La gestión inadecuada del presupuesto o el surgimiento de gastos no previstos pueden llevar a un exceso en los costos del proyecto. Esto puede suceder por variaciones en los precios de mercado de los componentes necesarios, la necesidad de adquirir hardware o software adicional no contemplado inicialmente, o la contratación de servicios externos para resolver problemas técnicos complejos. Un control presupuestario deficiente podría limitar la capacidad para completar el proyecto con éxito dentro de los limites financieros establecidos.

Como punto clave de mitigación de esta amenaza está la de mantener un presupuesto de emergencia del 20% del proyecto para imprevistos.

1.7. Evaluación económica del proyecto

La evaluación económica de un proyecto es un componente esencial en la fase de planificación y desarrollo, ya que proporciona una visión integral del impacto financiero y la viabilidad económica del proyecto. Este análisis implica una estimación detallada de todos los costos asociados a la implementación y operación del proyecto, incluyendo, pero no limitándose a, costos de adquisición de hardware, software y costos laborales.

Costo del Hardware
Para el proceso de evaluación de costos de mercado, se han realizado cotizaciones sobre servidores que cumplan con los requerimientos técnicos. En

la evaluación se obtiene un valor promedio, el cual es utilizado como costo del hardware.

Requerimientos técnicos:

- Servidor con 4 Core de CPU, 16GB RAM, 1TB de Disco.

Servidor evaluado	**USD**	**EUR**
HPE ProLiant Microserver Gen10	2,339	2,156
HPE Servidor torre Proliant ML30 Gen10	2,796	2,577
HPE ProLiant ML30 Gen10 Plus Intel Xeon E-2314/16GB	1,105	1,019
PE ProLiant MicroServer Gen10+ v2 Intel Xeon E-2314 4-Core	1,179	1,087
Servidor Dell EMC PowerEdge T150	1,532	1,412
Servidor DELL PowerEdge T150 E-2336G	1,526	1,407
Promedio	**1,746 USD**	**1,609 EUR**

Tipo de cambio 1 USD – 0,92 EURO.

Costo asociado al Software

Para el proceso de evaluación de costos asociados al software, se han considerado las versiones más básicas que cumplan con las funcionalidades mínimas requeridas. Durante el proceso se evaluación solo la herramienta Elastic ELK requirió un costo de mantención mensual por el soporte del fabricante.

Tipo de Software	**USD**	**EUR**
Elastic ELK	1,140	1,059
Ubuntu S.O.	0	0
BIND 9	0	0
Costo Total	**1,140 USD**	**1,059 EUR**

Costo asociado al profesional

Para el desarrollo e implementación del proyecto se ha considerado un Profesional con título de Ingeniero en Informática, Telecomunicaciones o carrera a fin. Este profesional obtendrá una remuneración por día trabajado y desplegará la herramienta en 21 días, según lo que define la etapa “implantación” en la carta Gantt.

Días trabajados	**USD**	**EUR**
21	2,916	2,688

Costo total de la implementación del proyecto

El costo total anual para el proyecto se detalla en la siguiente tabla.

Costo	**USD**	**EUR**
Hardware - Servidor	1,746	1,609
Software - Herramienta	1,14	1,059
Profesional - 21 días	2,916	2,688
Total	**5,802 USD**	**5,356 EUR**

2. Investigación

El presente capítulo se enfoca en el análisis detallado de dos tecnologías fundamentales para el desarrollo de este proyecto: DNS y SIEM.

El proceso de investigación se inicia con el “capitulo 2.1 Sobre DNS” donde se presenta el análisis de operación del protocolo DNS y de cómo este es utilizado en un ciberataque. En este proceso se analiza la operación de una zona RPZ y también algunas de las herramientas de protección existentes en el mercado empresarial. Finalmente, se presenta a BIND9 debido a sus características de ser opensource y soportada por el Internet Systems Consortium, Inc.

Durante el “capitulo 2.2 Investigación sobre operación del SIEM” se analizan las distintas opciones de SIEM y se presenta como tecnología seleccionada para el proyecto a “ELK Stack”, dado sus beneficios de ser Open Source y a los casos de uso disponibles para graficar la información del protocolo DNS.

Finalmente, en el “capitulo 2.3 Investigación de opciones de integración” se analizan las capacidades de integración de BIND9 utilizando una zona RPZ. En este punto se analizan las fuentes de terceros disponibles para alimentar una zona RPZ y se presentan los beneficios de uso de ioc2rpz para una infraestructura opensource.

2.1. Sobre DNS

El Protocolo de Sistema de Nombres de Dominio (DNS) es como el directorio de teléfonos de Internet. Cuando escribes una dirección web en tu navegador, como "www.ejemplo.com", tu dispositivo necesita saber la dirección IP asociada a ese nombre de dominio para conectarse al servidor correcto. Entonces, tu dispositivo envía una solicitud al servidor DNS local, preguntando por la dirección IP de "www.ejemplo.com".

El servidor DNS local primero revisa su caché para ver si ha resuelto esa dirección recientemente. Si la respuesta está en la caché y no ha expirado, la devuelve de inmediato. Si no está en la caché, el servidor DNS local comienza una búsqueda preguntando a los servidores DNS de nivel superior, como los servidores raíz, dónde puede encontrar información sobre el dominio ".com".

Luego, sigue preguntando a los servidores DNS responsables de manejar los dominios ".com" y luego los servidores de nombres autoritativos de "ejemplo.com", hasta que finalmente recibe la dirección IP de "www.ejemplo.com" de los servidores de nombres autoritativos.

Una vez que tiene la dirección IP, el servidor DNS local devuelve esta información a tu dispositivo. Ahora, tu dispositivo sabe a qué dirección IP debe conectarse para cargar la página web de "www.ejemplo.com".

“Los primeros sistemas de ARPANET (a partir de los cuales evolucionó Internet) asignaban nombres a direcciones utilizando un archivo de hosts que se distribuía a todas las entidades cada vez que se producían cambios. Operativamente, tal

sistema se volvió rápidamente insostenible una vez que había más de 100 entidades en red, lo que llevó a la especificación e implementación del Sistema de Nombres de Dominio que usamos hoy en día." [2]

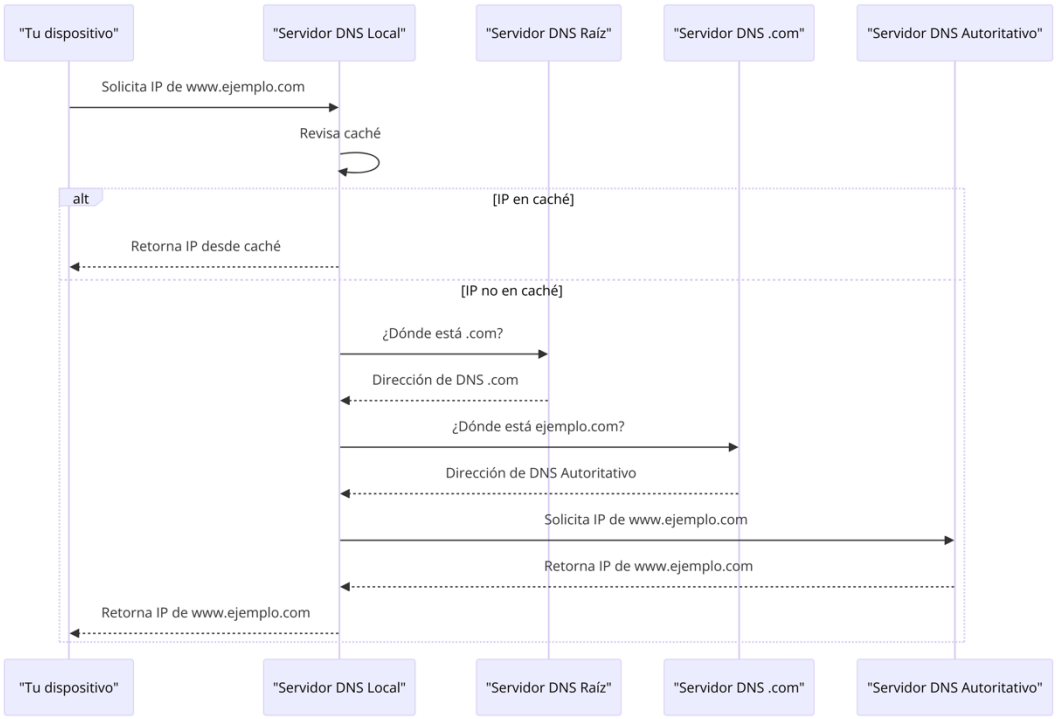


Ilustración 1: Operación DNS

2.1.1. Zonas en DNS

En el contexto del protocolo DNS (Domain Name System), una "zona" se refiere a una parte administrativa de la estructura de nombres de dominio en Internet. Específicamente, una zona es un segmento del espacio de nombres global de DNS que es administrado por una entidad específica o un administrador de sistemas. Cada zona representa una sección de la jerarquía de dominios y es responsable de responder a las consultas relacionadas con los nombres de dominio que se encuentran dentro de su área específica.

El DNS está organizado en una estructura jerárquica de dominios, que se asemeja a un árbol invertido con la "raíz" en la parte superior. Cada nodo o hoja en este árbol representa un dominio, que puede tener múltiples subdominios. Una zona comienza en un nodo específico y puede incluir todos los registros de ese nodo y de sus subdominios, hasta donde la autoridad de la zona se extiende. No todas las partes de un dominio necesariamente caen dentro de una sola zona; un dominio puede ser dividido en múltiples zonas, cada una gestionada de manera independiente.

Por ejemplo, en el dominio **ejemplo.com**, toda la información relacionada con este dominio, como los registros para **www.ejemplo.com**, **mail.ejemplo.com**,

etc., estarían dentro de la "zona" de **ejemplo.com**, siempre que no se deleguen subdominios a otras zonas.

La información de una zona se almacena en un archivo de zona, que contiene registros DNS. Estos registros incluyen información como:

- **A (Address)**: Asocia un nombre de dominio con una dirección IP.
- **MX (Mail Exchange)**: Especifica los servidores de correo para el dominio.
- **NS (Name Server)**: Indica qué servidores son autoritativos para la zona.
- **CNAME (Canonical Name)**: Permite asociar un nombre de dominio con otro nombre de dominio.

Los administradores de una zona DNS tienen la capacidad de controlar y actualizar esta información, permitiéndoles gestionar cómo se resuelven los nombres de dominio dentro de su zona. Esto es esencial para el funcionamiento de las redes y los servicios en Internet, ya que asegura que los nombres de dominio se resuelvan en las direcciones IP correctas.

2.1.2. Servidor de nombres autoritativo

Un servidor de nombres autoritativo (Authoritative Name Server en inglés) es un tipo de servidor DNS responsable de proporcionar respuestas definitivas y autorizadas sobre consultas de nombres de dominio para una zona de dominio específica. En otras palabras, un servidor de nombres autoritativo tiene la autoridad para proporcionar información oficial sobre los nombres de dominio dentro de una zona determinada.

Cuando un servidor de nombres autoritativo recibe una consulta DNS para un nombre de dominio dentro de la zona que gestiona, puede proporcionar una respuesta directa y precisa, sin necesidad de consultar a otros servidores. Si recibe una consulta para un nombre de dominio fuera de su zona, puede redirigir la consulta a otro servidor DNS que tenga la autoridad para responder.

Supongamos que tenemos un sitio web llamado "ejemplo.com" y un servidor de nombres autoritativo designado para manejar las consultas DNS relacionadas con ese dominio. Cuando alguien en Internet quiere visitar "ejemplo.com", su dispositivo envía una solicitud de resolución de nombres al servidor de nombres autoritativo para "ejemplo.com".

El servidor de nombres autoritativo verifica su base de datos de zona, que contiene información sobre los registros DNS para "ejemplo.com". Encuentra la dirección IP asociada con "ejemplo.com" y responde directamente al dispositivo que realizó la consulta con esta información.

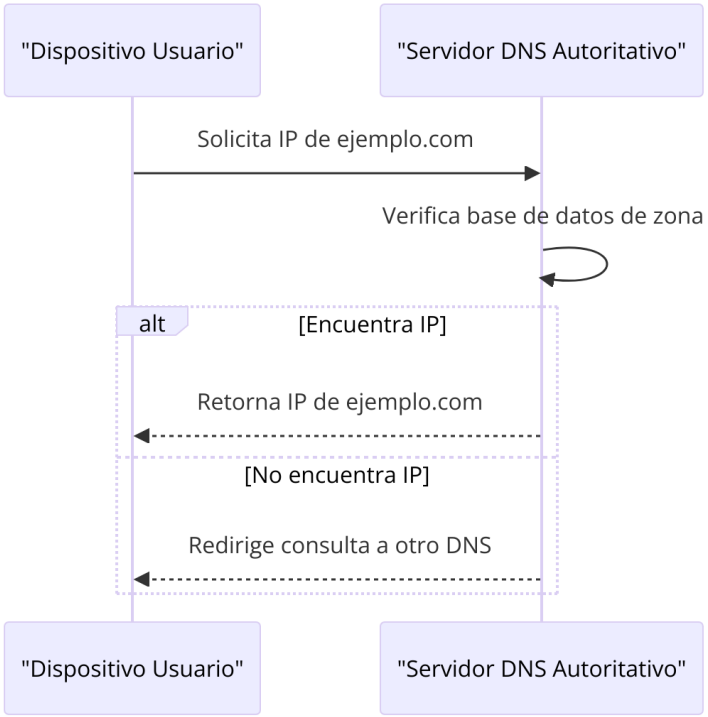


Ilustración 2: Servidor de nombres Autoritativo

2.1.3. Resolver (Caching Name Servers)

Un DNS Resolver, también conocido como servidor DNS recursivo o resolutor DNS, es un componente fundamental en el sistema de nombres de dominio (DNS). Su función principal es resolver las consultas DNS realizadas por los dispositivos conectados a una red.

Cuando un dispositivo, como una computadora o un teléfono, intenta acceder a un sitio web utilizando su nombre de dominio (por ejemplo, www.ejemplo.com), el DNS Resolver se encarga de traducir ese nombre de dominio a la dirección IP correspondiente que el dispositivo necesita para establecer la conexión.

El DNS Resolver realiza esta tarea de resolución de nombres de dominio de manera recursiva, lo que significa que puede buscar la respuesta a la consulta realizada en otros servidores DNS si no tiene la información en su propia caché. De esta manera, el DNS Resolver actúa como un intermediario entre el dispositivo que realiza la consulta y los servidores DNS autoritativos que tienen la información sobre los nombres de dominio y sus direcciones IP correspondientes.

2.1.4. Zona reversa

Una zona reversa en el sistema de nombres de dominio (DNS) es una configuración que permite la resolución inversa, es decir, convertir una dirección IP en un nombre de dominio. Esto es opuesto a la resolución directa, donde los nombres de dominio se convierten en direcciones IP.

Cómo Funciona:

- **PTR Records**: Las zonas reversas utilizan registros PTR (Pointer) para mapear direcciones IP a nombres de dominio.
- **Formato de Zona**: Para una dirección IP, la entrada en la zona reversa sigue un formato especial. Por ejemplo, la IP **192.0.2.1** se escribiría como **1.2.0.192.in-addr.arpa** en la zona reversa.
- **Consultas DNS**: Cuando se realiza una consulta DNS inversa, el servidor DNS busca en la zona reversa el registro PTR correspondiente a la dirección IP para devolver el nombre de dominio asociado.

Las zonas reversas se utilizan principalmente para la validación de direcciones IP en redes, como en la verificación de servidores de correo electrónico, para asegurar que una dirección IP corresponde a un dominio legítimo.

2.1.5. Load Balancing

El balanceo de carga en DNS es una técnica utilizada para distribuir la carga de tráfico entre varios servidores o recursos de red, de manera que se optimice el rendimiento y se evite la congestión en un único servidor. Funciona asignando diferentes direcciones IP a un mismo nombre de dominio y distribuyendo las solicitudes de los clientes entre esas direcciones IP.

El proceso funciona de la siguiente manera. En primer lugar, el administrador del sistema configura múltiples servidores o recursos de red y les asigna direcciones IP diferentes. Luego, se configura el servidor DNS para que responda con estas múltiples direcciones IP cuando recibe consultas de resolución de nombres para un dominio específico.

Cuando un cliente realiza una consulta DNS para resolver el nombre de dominio, el servidor DNS responde con una de las direcciones IP disponibles, alternando entre ellas o utilizando un método específico de balanceo de carga, como round-robin, ponderado, basado en geolocalización, etc.

Una vez que el cliente recibe la dirección IP del servidor DNS, utiliza esa información para establecer la conexión con el servidor correspondiente y acceder al recurso solicitado. Con el tiempo, las solicitudes de los clientes se distribuyen entre las diferentes direcciones IP asignadas, lo que ayuda a equilibrar la carga de tráfico entre los servidores o recursos de red y mejorar el rendimiento del sistema en su conjunto.

2.1.6. Ataques a través del protocolo DNS

El protocolo DNS (Domain Name System) es fundamental en la infraestructura de Internet ya que traduce los nombres de dominio legibles por humanos en direcciones IP utilizables por las máquinas. Sin embargo, su papel central en las comunicaciones lo convierten en una herramienta útil para los ciberdelincuentes

facilitando las actividades maliciosas. Alguno de los ataques más comunes donde el protocolo DNS se ve envuelto son:

- **Infección a través del Phishing:**
  En un ataque de phishing, los ciberdelincuentes intentan engañar a los usuarios para que revelen información confidencial, como contraseñas o información financiera. Pueden registrar dominios maliciosos que se asemejen a los legítimos (por ejemplo, "paypal-security.com" en lugar de "paypal.com") y luego enviar correos electrónicos falsificados con enlaces a estos sitios. Cuando un usuario ingresa al dominio primero ejecuta la resolución a través del protocolo DNS y luego accede a través de su navegador al sitio web falso en lugar del legítimo. Esto facilita la propagación de malware o la recolección de información confidencial.

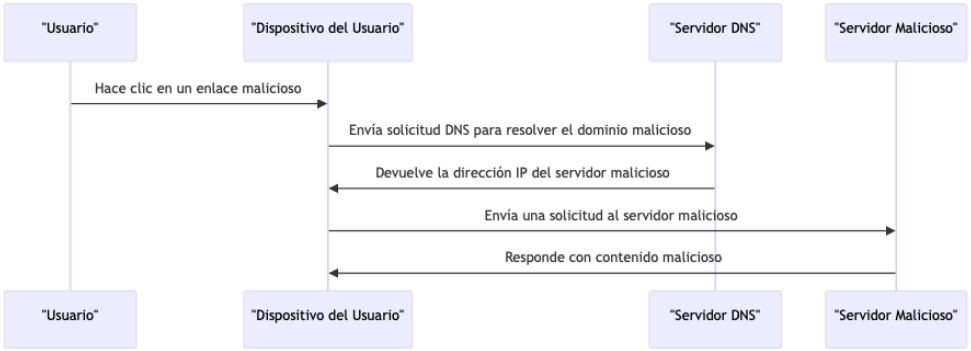


Ilustración 3: Infección Phishing

- **Comando y Control de un Malware:**
  Una vez que un malware ha infectado un dispositivo, a menudo necesita comunicarse con un servidor de comando y control (C&C) para recibir instrucciones o enviar datos robados. Los atacantes pueden utilizar dominios maliciosos para alojar estos servidores C&C y coordinar las actividades del malware. El DNS es utilizado por el malware para resolver los nombres de dominio de los servidores C&C en direcciones IP. Esto permite que el malware se conecte al servidor C&C y realice sus funciones maliciosas, como descargar actualizaciones, recibir comandos o exfiltrar datos.

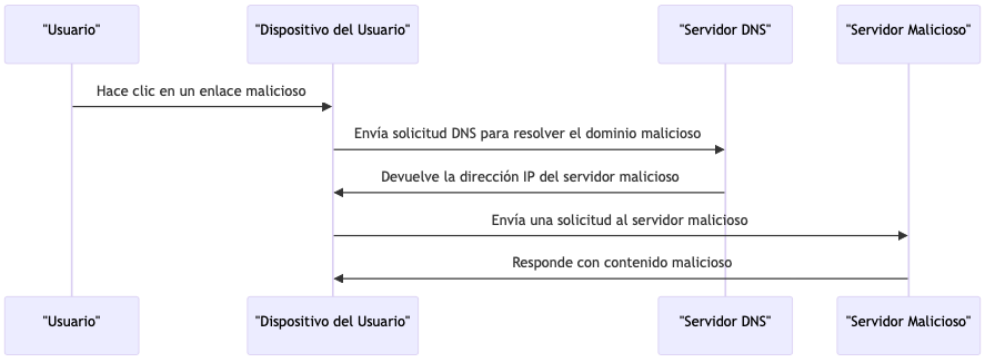


Ilustración 4: Comando y control

Además de los ataques de phishing y el comando y control de malware, el protocolo DNS puede ser explotado de otras maneras. Por ejemplo, los atacantes pueden llevar a cabo ataques de envenenamiento de caché DNS,

donde manipulan los datos almacenados en caché en los servidores DNS para redirigir a los usuarios a sitios maliciosos.

El protocolo DNS es un componente crítico de Internet que puede ser utilizado por los atacantes en una variedad de escenarios maliciosos, desde el phishing hasta el comando y control de malware. Por lo tanto, es crucial implementar medidas de seguridad adecuadas, como el monitoreo de tráfico DNS, el filtrado de contenido malicioso y la educación del usuario, para mitigar estos riesgos.

2.1.7. Como operan las zonas RPZ

Las zonas RPZ, o Zonas de Política de Respuesta en DNS, son una herramienta utilizada para mejorar la seguridad de la navegación por Internet. DNS es el sistema que traduce nombres de dominios comprensibles por humanos, como www.ejemplo.com, en direcciones IP que utilizan las computadoras para comunicarse entre sí en la red.

Las zonas RPZ funcionan como una capa de seguridad adicional en este proceso de traducción. Permiten a los administradores de redes configurar listas de sitios web sospechosos o maliciosos y bloquearlos antes de que los usuarios puedan acceder a ellos. Cuando un dispositivo en la red intenta acceder a un sitio web, el DNS verifica si el nombre del sitio está en la lista de bloqueo de la zona RPZ. Si es así, el DNS puede responder de varias maneras: negando el acceso, redirigiendo a una página segura que advierte al usuario, o simplemente ignorando la solicitud.

En conclusión, las zonas RPZ son una especie de filtro que ayuda a prevenir que los usuarios accedan a sitios peligrosos o no deseados, mejorando así la seguridad de la red.

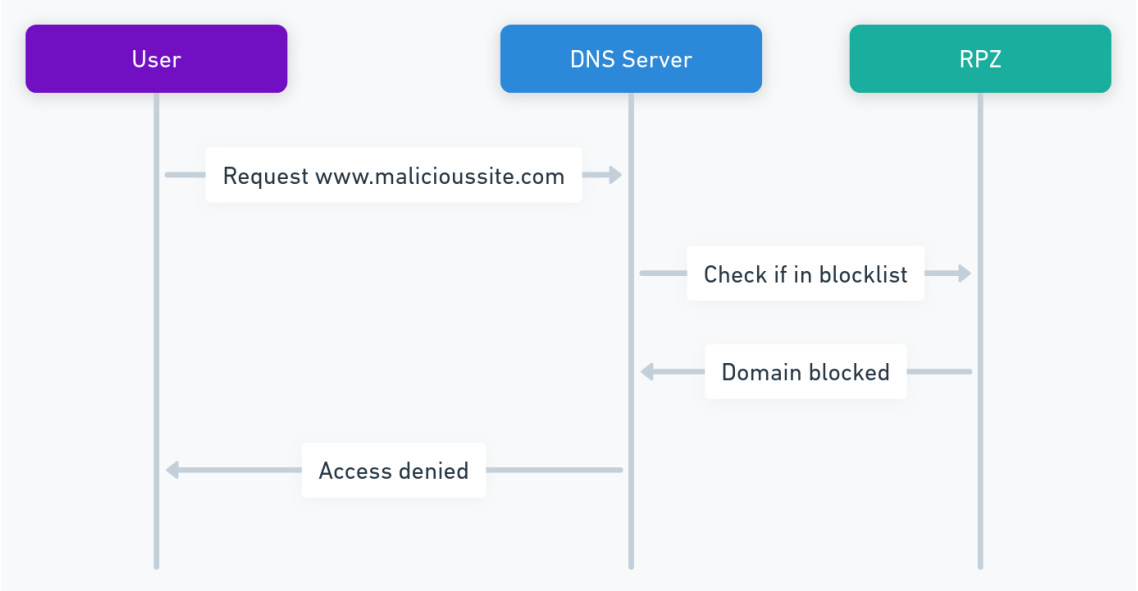


Ilustración 5 Operación de RPZ

2.1.8. Soluciones de protección sobre DNS

El Protocolo de Sistema de Nombres de Dominio (DNS) no solo es crucial para la resolución de nombres en Internet, sino que también se ha convertido en una herramienta poderosa para proteger a los usuarios y redes contra amenazas en línea. Un enfoque común es el uso de Firewalls DNS, que interceptan las consultas DNS y bloquean el acceso a sitios web maliciosos o no deseados al comparar las respuestas con una lista de dominios conocidos por su malicia. Asimismo, la implementación de Zonas de Política de Respuesta (RPZ) permite a los administradores configurar políticas de seguridad específicas para redirigir o bloquear el acceso a dominios peligrosos, proporcionando un control más granular sobre la navegación web.

Además, las técnicas como los DNS Sinkholes ofrecen una capa adicional de protección al redirigir el tráfico de dominios maliciosos hacia direcciones IP controladas, previniendo así la comunicación con servidores comprometidos. Por otro lado, servicios de filtrado de contenido DNS y servidores DNS seguros (DNSSEC) trabajan en conjunto para bloquear el acceso a sitios con contenido inapropiado y autenticar los datos DNS, respectivamente.

Estas herramientas y servicios aprovechan el poder del DNS para fortalecer la seguridad en línea y proteger a los usuarios contra una variedad de amenazas cibernéticas. Algunas de las herramientas utilizadas en el mercado son las siguientes:

- **Cisco Umbrella (anteriormente OpenDNS):** Cisco Umbrella es un servicio de seguridad en la nube que utiliza el DNS para bloquear el acceso a sitios web maliciosos y proteger a los usuarios contra amenazas en línea. Ofrece una versión de pago para empresas y una versión gratuita para uso personal.

- **Cloudflare Gateway:** Cloudflare Gateway es un servicio de seguridad de acceso a Internet que utiliza el DNS para filtrar y bloquear el tráfico hacia sitios web maliciosos, phishing y contenido no deseado. Ofrece una versión de pago con funciones avanzadas de seguridad.

- **Quad9:** Quad9 es un servicio de DNS público gratuito que utiliza una lista de dominios maliciosos conocidos para bloquear el acceso a sitios web peligrosos. Su objetivo es proteger a los usuarios contra malware, phishing y otros tipos de ataques en línea.

- **Pi-hole:** Pi-hole es un proyecto de software de código abierto que actúa como un servidor DNS local y bloquea anuncios y rastreadores en toda la red al filtrar las consultas DNS. Es gratuito y se puede instalar en dispositivos como Raspberry Pi.

- **DNSFilter:** DNSFilter es un servicio de filtrado de contenido DNS que utiliza el DNS para bloquear el acceso a sitios web maliciosos, phishing y contenido inapropiado. Ofrece una versión de pago con características adicionales y una versión de prueba gratuita.

- **OpenDNS Security Essentials:** OpenDNS Security Essentials es un servicio de seguridad en línea que utiliza el DNS para proteger a los usuarios contra amenazas en línea, como malware, ransomware y phishing. Ofrece una versión gratuita con funciones básicas de seguridad y una versión de pago con características avanzadas.

2.1.9. Sobre BIND 9

BIND 9, abreviatura de "Berkeley Internet Name Domain version 9", es un software de servidor de nombres de dominio (DNS) de código abierto y ampliamente utilizado. Su función principal es la de traducir nombres de dominio legibles por humanos a direcciones IP comprensibles por las máquinas, y viceversa. BIND 9 es mantenido por el Internet Systems Consortium, Inc. (ISC) la cual es una organización sin fines de lucro dedicada al desarrollo y mantenimiento de software y servicios de infraestructura de Internet. Fundada en 1994, ISC tiene una larga historia de contribución a la evolución de Internet mediante la creación y el soporte de software de código abierto que es fundamental para su funcionamiento.

Uno de los proyectos más conocidos y ampliamente utilizados desarrollados por ISC es BIND, que es uno de los servidores DNS más utilizados en Internet. BIND es crucial para implementar el DNS, proporcionando la capacidad de resolver nombres de dominio en direcciones IP y viceversa, lo que facilita la navegación en la red.

ISC también participa activamente en la investigación y el desarrollo relacionados con la seguridad y estabilidad de la infraestructura de Internet, ofreciendo asesoría y participando en varias iniciativas comunitarias para mejorar la robustez y la eficiencia de la red. Además, proporciona servicios comerciales y soporte para sus software, ayudando a organizaciones a implementar soluciones de red seguras y confiables.

2.2. Investigación sobre operación del SIEM

Un SIEM (Security Information and Event Management) es una solución de seguridad informática que proporciona funciones de recopilación, correlación, análisis y presentación de datos de seguridad en tiempo real. Su objetivo principal es ayudar a las organizaciones a detectar y responder a amenazas de seguridad cibernética de manera efectiva al proporcionar una visibilidad centralizada de los eventos de seguridad en toda la infraestructura de TI. Esto se logra integrando datos de registros de eventos de múltiples fuentes, como firewalls, sistemas de detección de intrusiones, sistemas de prevención de intrusiones, registros de aplicaciones y más, para identificar patrones sospechosos o actividades maliciosas. Además, los SIEM pueden incluir capacidades de gestión de incidentes y cumplimiento normativo para ayudar a las organizaciones a mantener la seguridad y cumplir con los requisitos

regulatorios. Dentro de los SIEM más reconocidos del mercado encontramos los siguientes:

- **LogRhythm**: Orientada especialmente a pequeñas y medianas empresas, LogRhythm utiliza técnicas avanzadas de análisis de comportamiento para detectar amenazas y minimizar los falsos positivos. Es valorada por ser económica y ofrecer un excelente servicio técnico, aunque su menor costo conlleva limitaciones en alcance comparado con otros SIEM del mercado.

- **AlienVault USM**: es una solución SIEM integral que incluye recopilación y análisis de datos, correlación de eventos, alertas y cumplimiento normativo. Destaca por ofrecer una amplia gama de herramientas de seguridad adicionales. Su interfaz gráfica puede resultar poco intuitiva, especialmente en la gestión de alertas.

- **Splunk Enterprise Security**: Esta herramienta lidera en personalización y escalabilidad, adecuada para organizaciones de cualquier tamaño. Ofrece abundante información en tiempo real y diversas características. Requiere de un conocimiento avanzado en redes para su adecuada configuración.

- **IBM QRadar**: Lidera el mercado con sus técnicas avanzadas de aprendizaje automático para la detección de amenazas y la reducción de falsos positivos. Es ideal para monitorizar infraestructuras en la nube pero se destaca por su complejidad de uso y su elevado costo.

- **Trellix Enterprise Security Manager**: Anteriormente conocido como McAfee ESM, Trellix se usa ampliamente en grandes empresas. Permite la gestión remota de sistemas conectados a la red, aunque no ofrece soporte óptimo para Linux (RICARDEV).

- **Elastic Security:** Es una solución de seguridad integral que incluye funciones de SIEM (Security Information and Event Management). Utiliza tecnología de Elastic Stack para recopilar, analizar y correlacionar datos de seguridad en tiempo real, lo que permite detectar y responder a amenazas de manera eficiente. Elastic Security SIEM proporciona capacidades avanzadas de detección de amenazas, visualización de datos y análisis forense, todo dentro de una plataforma unificada y escalable.

2.2.1. Sobre ELK Stack

El ELK Stack [3] es una plataforma de código abierto que combina Elasticsearch, Logstash y Kibana para proporcionar una solución integral de análisis de datos y visualización de registros. Elastic en soluciones impulsadas por la búsqueda, ayudando a acelerar los resultados importantes para organizaciones como Uber, Slack y Microsoft.

Elasticsearch es un motor de búsqueda y análisis distribuido, diseñado para buscar y analizar grandes volúmenes de datos en tiempo real. Logstash es un motor de procesamiento de registros que recopila, transforma y enriquece los datos de los registros antes de indexarlos en Elasticsearch. Kibana es una interfaz de usuario de visualización de datos que permite explorar y visualizar los datos indexados en Elasticsearch de manera intuitiva y efectiva.

Elastic en su sitio web destaca que la empresa y sus productos se basan en una filosofía de gratuidad y apertura, lo que fomenta la honestidad y la colaboración en la comunidad de usuarios y desarrolladores.[3]

ELK Stack proporciona una plataforma escalable y flexible para la búsqueda, análisis y visualización de datos de registros, lo que permite a las organizaciones obtener información valiosa de sus datos de forma rápida y eficiente.

Cada componente de Elastic Stack desempeña un papel único en el procesamiento y visualización de datos:

Elasticsearch:

- Elasticsearch es un motor de búsqueda y análisis distribuido.
- Su función principal es almacenar, indexar y buscar datos estructurados y no estructurados a gran escala.
- Permite realizar búsquedas complejas y análisis de texto completo en tiempo real.
- Es altamente escalable y se puede utilizar para diversos casos de uso, como búsqueda de texto, análisis de registros, análisis de métricas, etc.
- Es el corazón de Elastic Stack y proporciona la infraestructura subyacente para almacenar y consultar datos.

Kibana:

- Kibana es una interfaz de usuario de visualización de datos.
- Se utiliza para explorar, analizar y visualizar los datos almacenados en Elasticsearch.
- Proporciona una amplia variedad de herramientas de visualización, como gráficos, tablas, mapas y dashboards interactivos.
- Permite realizar consultas ad hoc, crear visualizaciones personalizadas y compartir informes con otros usuarios.
- Facilita la comprensión y el análisis de datos mediante una interfaz intuitiva y fácil de usar.

Logstash:

- Logstash es un motor de procesamiento de datos y logística.
- Se utiliza para recopilar, transformar y enriquecer datos de diferentes fuentes antes de enviarlos a Elasticsearch para su almacenamiento.
- Admite una amplia variedad de entradas, incluidos logs de archivos, bases de datos, métricas de sistemas y eventos en tiempo real.

- Proporciona filtros para manipular y estructurar los datos, como la extracción de campos, la eliminación de duplicados y la enriquecimiento con información adicional.
- Es altamente configurable y extensible, lo que permite adaptarse a diferentes tipos de fuentes de datos y requisitos de procesamiento.

2.2.2. Casos de uso de ELK Stack con DNS

La integración de Elasticsearch, Logstash y Kibana (ELK) con el sistema de nombres de dominio (DNS) ofrece una poderosa solución para monitorear, analizar y visualizar los registros de actividad del DNS en tiempo real. Al aprovechar la capacidad de recopilación de logs de Logstash, el almacenamiento y búsqueda escalables de Elasticsearch, y las capacidades de visualización de Kibana, los administradores de red pueden obtener una comprensión profunda de la actividad del DNS, identificar posibles amenazas de seguridad, y tomar medidas proactivas para proteger la infraestructura de red.

- **Monitoreo de actividad DNS:**

Es posible utilizar Logstash para recopilar registros de consultas DNS de servidores DNS y otros dispositivos de red.
Almacenar los datos en Elasticsearch y utiliza Kibana para crear dashboards interactivos que muestren estadísticas en tiempo real sobre consultas DNS, tipos de consultas, respuestas, etc.
Identificar patrones de actividad inusual que podrían indicar problemas de red, ataques o comportamiento malicioso.

- **Detección de anomalías en DNS:**

Podemos configurar alertas en Kibana para detectar patrones de actividad anómalos en los registros de DNS, como consultas inusuales, respuestas inesperadas, consultas a dominios sospechosos, etc.
Utilizar algoritmos de detección de anomalías para identificar comportamientos fuera de lo común que podrían indicar ataques de malware, exfiltración de datos u otros eventos de seguridad.

- **Investigación de incidentes de seguridad:**

Elasticsearch puede ser utilizada para buscar y analizar registros de DNS en busca de indicadores de compromiso (IOC) y patrones de ataque conocidos.
Por otra parte, podemos utilizar a Kibana para visualizar y analizar los datos de DNS relacionados con incidentes de seguridad, como consultas a dominios maliciosos, intentos de exfiltración de datos a través de DNS, comunicaciones de comando y control, etc.

- **Análisis de tráfico DNS malicioso:**

Es posible utilizar Logstash para analizar y filtrar registros de DNS en busca de consultas y respuestas maliciosas, como consultas a dominios generados al azar, respuestas NXDOMAIN, consultas de dominios de redireccionamiento, etc. Por otra parte, Kibana nos permite visualizar y explorar los datos de tráfico DNS malicioso, identificar patrones de ataque y tomar medidas correctivas para mitigar el riesgo de seguridad.

- **Auditoría y cumplimiento de políticas:**

Es posible utilizar Logstash para recopilar y normalizar registros de DNS de diferentes fuentes en un formato estándar.
También nos permite almacenar los datos en Elasticsearch y utiliza Kibana para generar informes y dashboards que muestren el cumplimiento de políticas de seguridad de DNS, como la resolución de nombres no autorizados, consultas bloqueadas, etc.
Por otra parte, podemos Identificar y aborda los problemas de cumplimiento y seguridad relacionados con el DNS para garantizar el cumplimiento de las políticas y estándares de seguridad de la organización.

2.3. Investigación de opciones de integración

2.3.1. DNS Firewall y Response Policy Zone

Un DNS firewall es una herramienta que examina el tráfico DNS y decide permitir que algunas respuestas pasen mientras bloquea otras. Esta inspección puede basarse en varios criterios, como el nombre solicitado, los datos asociados con ese nombre (como una dirección IP) o el nombre o dirección IP del servidor de nombres autoritativo para el nombre solicitado. Esto permite a los administradores tener más control sobre qué sistemas pueden acceder o ser accesibles desde sus redes.

Una forma de DNS firewall son las Zone Response Policy Zones (RPZ), donde las reglas del firewall se expresan dentro del propio DNS, codificadas en un formato abierto y neutral de proveedor como registros en zonas DNS especialmente construidas. Esto permite que las políticas se compartan fácilmente entre servidores utilizando el mecanismo estándar de transferencia de zona DNS.

En Linux el demonio, por ejemplo, named puede suscribirse a hasta 64 Response Policy Zones (RPZ), cada una de las cuales codifica un conjunto de reglas de política separadas. Cada regla se almacena en un conjunto de registros de recursos DNS (RRset) dentro del RPZ y consiste en un desencadenante y una acción. Hay cinco tipos de desencadenantes y seis tipos de acciones.[2]

Por ejemplo, una regla de política de respuesta en una RPZ DNS puede ser desencadenada por la dirección IP del cliente, el nombre de la consulta, una dirección que estaría presente en una respuesta verídica o el nombre o dirección de un servidor de nombres autoritativo responsable de publicar la respuesta original.

Las acciones de la política de respuesta pueden ser variadas, incluyendo sintetizar una respuesta de "dominio no existe" (NDOMAIN), sintetizar una respuesta de "el nombre existe pero no hay registros del tipo solicitado" (NODATA), descartar la respuesta, cambiar a TCP enviando una respuesta UDP truncada que requiere que el cliente DNS intente nuevamente con TCP, reemplazar/sobrescribir los datos de la respuesta con datos específicos proporcionados dentro de la zona de política de respuesta o eximir la respuesta de un procesamiento de política adicional.

El uso más común de un DNS firewall es "envenenar" un nombre de dominio, dirección IP, nombre del servidor de nombres o dirección IP del servidor de nombres. Esto se realiza generalmente forzando una respuesta sintética de "dominio no existe" (NDOMAIN). Por ejemplo, si un administrador mantiene una lista de dominios de "phishing" conocidos, estos nombres pueden hacerse inaccesibles para los clientes o usuarios finales simplemente agregando una política de firewall en el servidor DNS recursivo, con un desencadenante para cada dominio de "phishing" conocido, y una acción en cada caso forzando una respuesta sintética NDOMAIN. También es posible usar una acción de reemplazo de datos, como responder para estos dominios de "phishing" conocidos con el nombre de un servidor web local que puede mostrar una página de advertencia. Este servidor web se llamaría un "jardín amurallado".

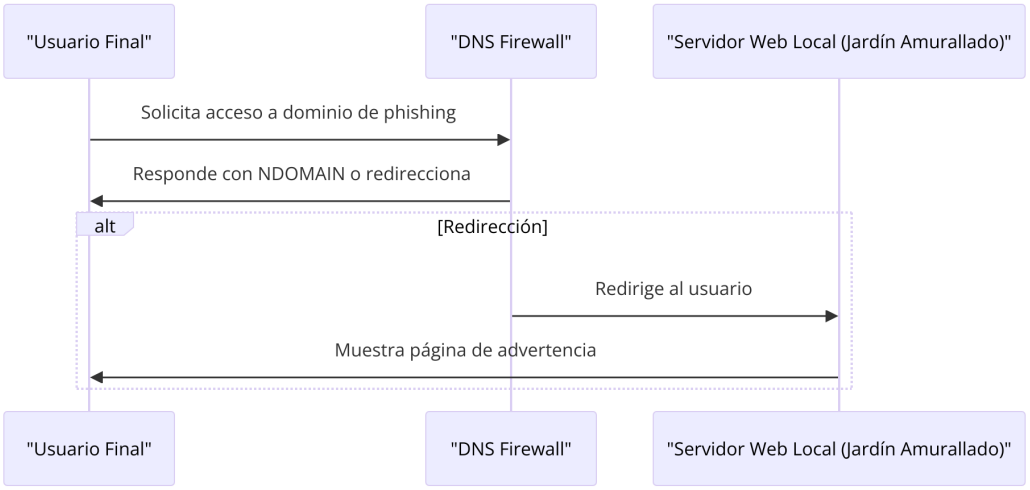


Ilustración 6: Operación DNS Firewall

2.3.2. Investigación sobre fuentes de terceros

Estos sitios son reconocidos en la comunidad de seguridad cibernética y son mantenidos por expertos en la materia, así como por comunidades de seguridad activas. Proporcionan información actualizada sobre amenazas en línea, como dominios maliciosos, direcciones IP sospechosas y URLs utilizadas en campañas de phishing, malware y botnets.

Al integrar estos feeds externos en la configuración de una RPZ, los administradores pueden implementar políticas de respuesta personalizadas que bloqueen el acceso a recursos maliciosos conocidos. Esto fortalece la seguridad

de la infraestructura DNS al proteger contra una amplia gama de amenazas en línea y ayuda a prevenir ataques cibernéticos dirigidos a la red. Los feeds más reconocidos del mercado son:

- **ioc2rpz:**
  ioc2rpz ofrece feeds gratuitos de DNS Firewall / RPZ basados en inteligencia de amenazas pública disponible. Estos feeds pueden fortalecer la seguridad de tu red contra diversas amenazas en línea.

- **PhishTank:**
  PhishTank proporciona feeds de URLs maliciosas utilizadas en campañas de phishing. Estos feeds son útiles para proteger tu red contra ataques de phishing y robo de credenciales.

- **Open Threat Exchange (OTX) de AlienVault:**
  OTX de AlienVault ofrece una variedad de feeds de amenazas, incluyendo listas de dominios maliciosos y direcciones IP sospechosas. Estos feeds pueden mejorar la seguridad de tu red al proporcionar inteligencia de amenazas actualizada.

- **Zeus Tracker:**
  Zeus Tracker ofrece feeds de dominios y direcciones IP asociados con botnets, malware y phishing. Estos feeds se actualizan regularmente y son útiles para proteger tu red contra diversas amenazas en línea.

- **Emerging Threats Open Ruleset:**
  Emerging Threats Open Ruleset proporciona feeds de amenazas en tiempo real, así como reglas de detección de intrusiones (IDS). Estos feeds son gratuitos y pueden fortalecer las defensas de tu red contra ataques cibernéticos.

- **URLhaus:**
  URLhaus ofrece información sobre URLs maliciosas utilizadas en campañas de malware y phishing. Estos feeds son útiles para proteger tu red contra diversas amenazas en línea, como descargas maliciosas y ataques de phishing.

2.3.3. Integrando ioc2rpz.net como RPZ en BIND 9

Ioc2rpz es una comunidad que proporciona feeds de DNS Firewall RPZ de código abierto. Estos feeds se basan en inteligencia de amenazas públicamente disponible y son mantenidos por comunidades o empresas de terceros. Sin embargo, solo un número limitado de indicadores ha sido incluido en una lista blanca. Es importante tener en cuenta que ioc2rpz no valida los feeds de TI en busca de falsos positivos.

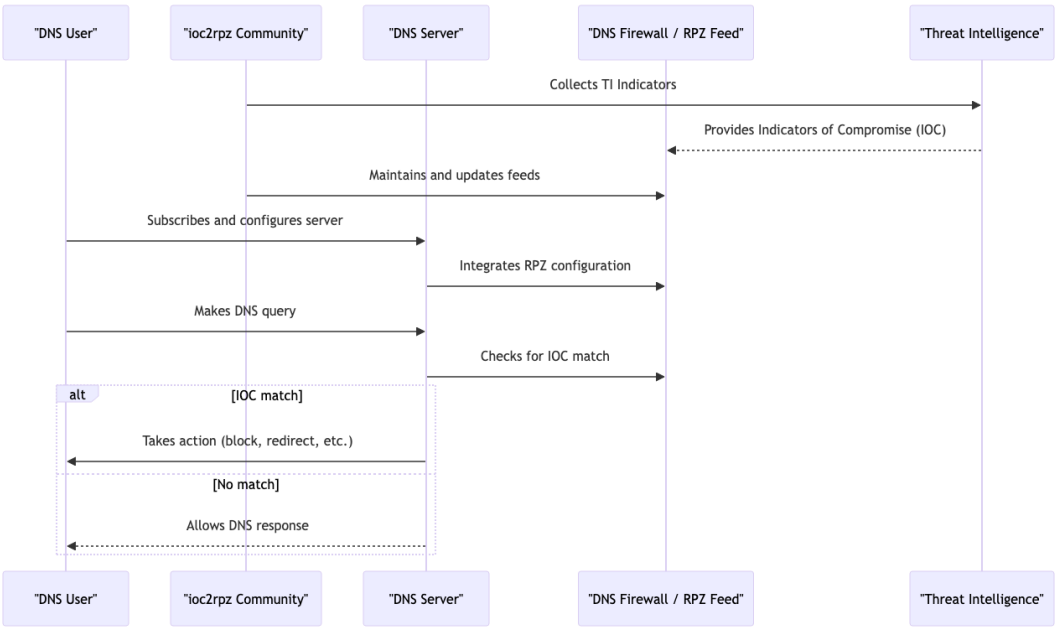
El funcionamiento de ioc2rpz se basa en la recopilación y distribución de feeds de DNS Firewall RPZ, los cuales contienen indicadores de compromiso (IOC) basados en inteligencia de amenazas públicamente disponible. Estos feeds son

mantenidos por la comunidad ioc2rpz y por terceros, y se actualizan periódicamente para incluir nuevos indicadores y eliminar los obsoletos.

Cuando un usuario se suscribe a los feeds de ioc2rpz, configura su servidor DNS, como BIND 9, para que utilice estos feeds como parte de su política de respuesta. Esto se logra mediante la integración de los feeds de ioc2rpz en la configuración de la zona de respuesta de política (RPZ) del servidor DNS.

Cuando un cliente realiza una consulta DNS, el servidor DNS verifica si la consulta coincide con alguno de los indicadores presentes en los feeds de ioc2rpz. Si se encuentra una coincidencia, el servidor DNS puede tomar una acción específica según la configuración de la política de respuesta. Por ejemplo, puede bloquear la consulta, redirigirla a un servidor específico, o aplicar cualquier otra acción definida por el administrador.

Ioc2rpz funciona proporcionando feeds de DNS Firewall RPZ que contienen indicadores de compromiso, los cuales son utilizados por los servidores DNS para proteger las redes contra amenazas conocidas, basadas en la inteligencia de amenazas pública disponible.



2.3.4. Integrando ELK y RPZ

Para integrar ELK (Elasticsearch, Logstash y Kibana) y recibir los logs de RPZ (Response Policy Zone) de BIND 9, primero debemos asegurarnos de configurar BIND 9 para registrar los eventos relevantes en los logs. Esto implica ajustar la configuración de registro de BIND 9 siguiendo las recomendaciones de la documentación oficial de BIND 9, como la definición de canales de registro y categorías específicas para eventos de RPZ. [3]

Una vez configurado BIND 9, procedemos a configurar Logstash para que escuche en el puerto donde BIND 9 enviará los logs. Esto se logra agregando un input en el archivo de configuración de Logstash y especificando el tipo de log (por ejemplo, syslog) y el puerto. Luego, definimos los filtros necesarios en Logstash para analizar y procesar los logs de RPZ entrantes, como la extracción de campos relevantes y la normalización de formatos.

Una vez procesados los logs de RPZ por Logstash, enviamos los datos estructurados a Elasticsearch para su almacenamiento y búsqueda. Configuramos Logstash para que utilice un output de Elasticsearch, especificando la dirección y el puerto del clúster de Elasticsearch. Con los datos almacenados en Elasticsearch, podemos utilizar Kibana para crear visualizaciones y dashboards personalizados que muestren información relevante sobre los logs de RPZ, como métricas de consultas bloqueadas, dominios más bloqueados, direcciones IP asociadas, entre otros.

A medida que revisamos los logs y exploramos los datos en Kibana, es posible que deseemos ajustar la configuración de registro de BIND 9, los filtros de Logstash o las visualizaciones en Kibana para adaptarse mejor a nuestras necesidades específicas y objetivos de monitoreo y análisis. En resumen, este proceso de integración nos permite gestionar eficazmente las políticas de respuesta de DNS en nuestra infraestructura, utilizando las potentes capacidades de ELK para el análisis y la visualización de datos.

3. Implantación

3.1. Descripción del laboratorio

Para la elaboración del laboratorio se utilizan los siguientes componentes físicos:

- Equipo Windows 10 con los siguientes recursos:
  - Memoria RAM de 16GB
  - Procesador Intel® Core™ I5-10400F CPU @ 2.90GHz
  - Disco SSD de 930GB
- Router Inalámbrico de Internet marca Mercusys

Desde el Equipo Windows 10 se procede a conectar a la red del Router inalámbrico. Adicionalmente, sobre el Equipo Windows 10 se despliega una máquina virtual con Ubuntu 24.04 utilizando el hipervisor VMware Workstation 16.

La arquitectura lógica del laboratorio se representa en la ilustración 7, donde se muestran los distintos componentes que serán desplegados en el ambiente.

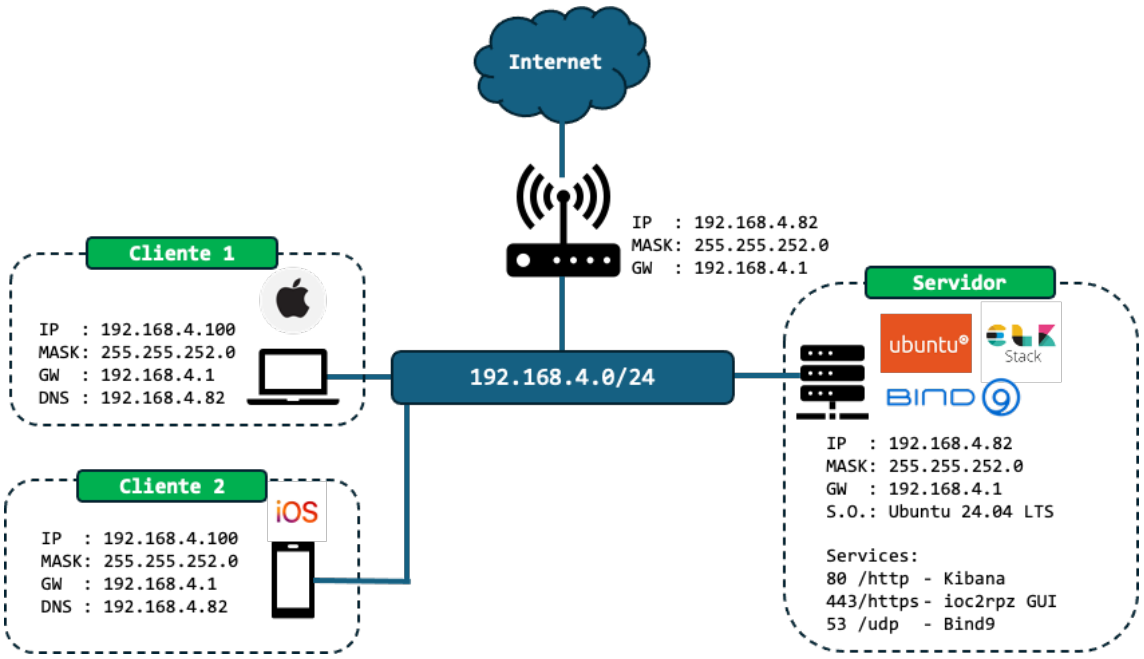


Ilustración 7: Diagrama lógico de red

3.1.1. Resultados esperados al finalizar el despliegue

Al finalizar el despliegue de las herramientas, se espera que cuando un equipo infectado (denominado "Cliente 1" en la ilustración 8) intente resolver el nombre de un dominio malicioso, el servidor DNS bloquee dicha resolución, como se muestra en la ilustración.

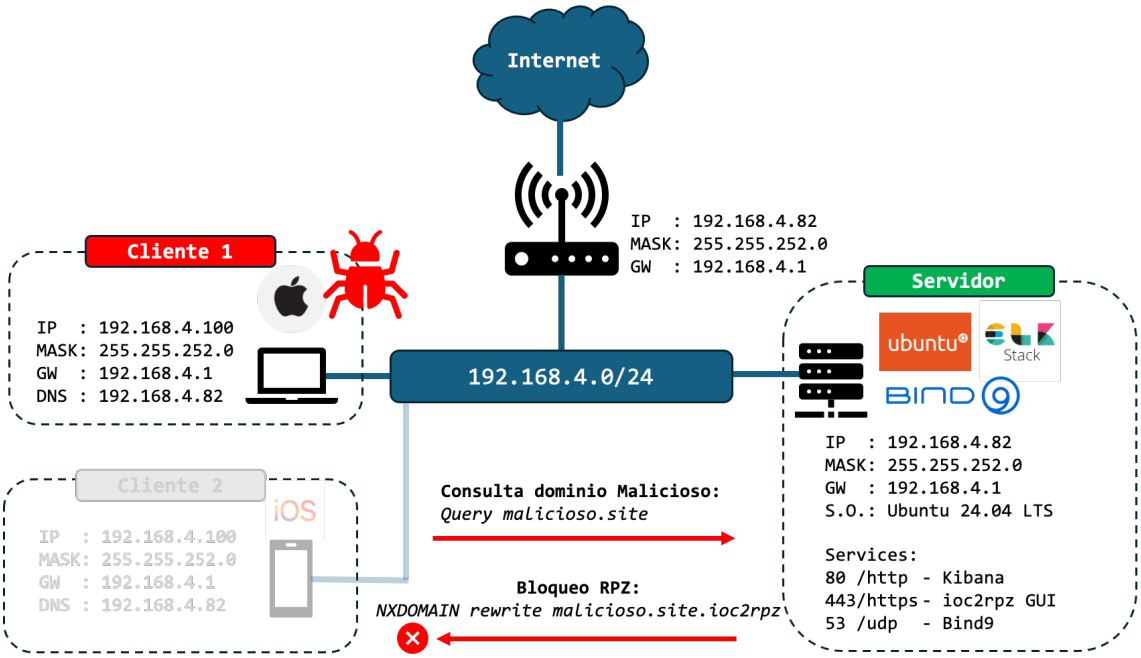


Ilustración 8: Operación de bloqueo

Por otra parte, este bloqueo generará un registro (log) en el servidor BIND9, el cual será utilizado para crear un panel de control web. Desde este panel, se podrá observar el TOP de sitios bloqueados y el TOP de máquinas que han intentado acceder a estos sitios, como se muestra en la Ilustración 9.

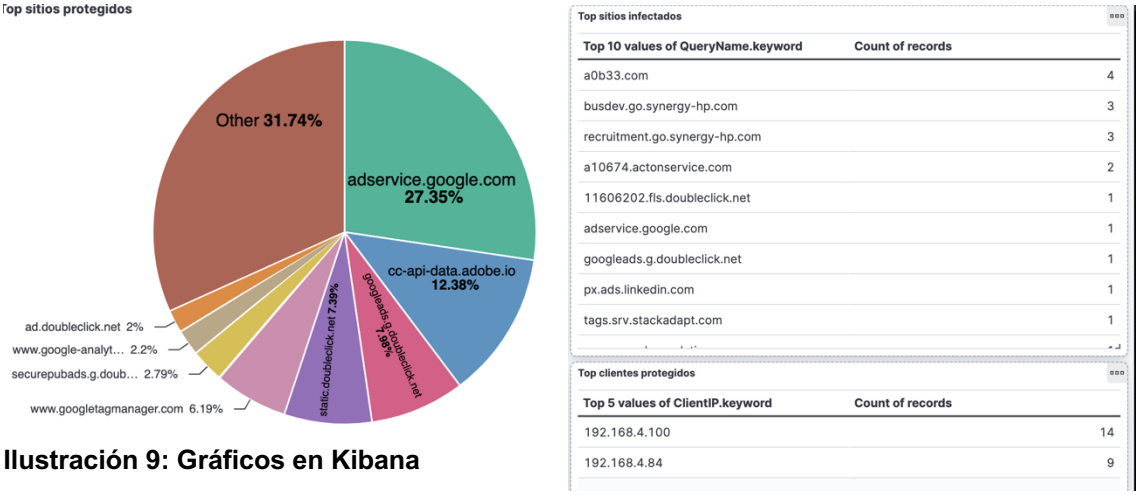


Ilustración 9: Gráficos en Kibana

3.2. Proceso de instalación a ejecutar

Para el despliegue del laboratorio se requerirán diversas herramientas, cada una con sus respectivas dependencias. Para asegurar un despliegue correcto, se propone ejecutar las instalaciones en el siguiente orden:

Ilustración 10: Proceso de instalación

1. Instalación de Ubuntu 24.04
    a. Se procede a realizar la instalación inicial de Ubuntu 24.04, aplicando la configuración de networking descrita en la Ilustración 7.
    b. Se procede a realizar la habilitación de SSH para la gestión remota.
2. Instalación de BIND9
    a. Se ejecuta la instalación del paquete de instalación BIND9.
    b. Se procede a ejecutar la creación de la zona appfirewall.cl, la cual utilizaremos para la gestión de la herramienta.
    c. Se procede a ejecutar la habilitación de consultas recursivas hacia 8.8.8.8, almacenando los logs de las queries.
3. Instalación de nginx
    a. Se procede ejecutar el paquete de instalación nginx en el servidor, para luego habilitar el proxy para publicar el servicio de Kibana en el puerto 80/http.
4. Instalación ELK Stack
    a. Se procede a realizar la instalación de los paquetes de instalación Elastic Search, Logstash y Kibana.
    b. Se realiza la configuración en nginx para habilitar el proxy que publica el servicio de Kibana en el puerto 80/http.
    c. Se realiza la configuración en ELK para visualizar los logs de las queries de BIND9 en forma recursiva.
5. Instalación de ioc2rpz
    a. Se procede a ejecutar el paquete de instalación de Docker, para luego ejecutar las imágenes ioc2rpz y ioc2rpz.gui.
    b. Se realiza la habilitación de RPZ con integración de Sources en ioc2rpz.
    c. Se crea en BIND9 una zona RPZ para integrarse con ioc2rpz con el objetivo de bloquear las consultas a sitios con malware o de seguimiento (tracking).
    d. Se almacenan los logs de RPZ y se visualizan en dashboard en Kibana.

3.3. Instalación de Ubuntu 24.04

El proyecto será montado sobre VMware Workstation utilizando la imagen Ubuntu 24.04 LTS. El procedimiento asociado a la instalación de la imagen se detalla en el Anexo I: Procedimiento de Instalación de Ubuntu sobre VMware.

Al finalizar el despliegue base obtendremos acceso a nuestro servidor como se observa en la Ilustración 11.

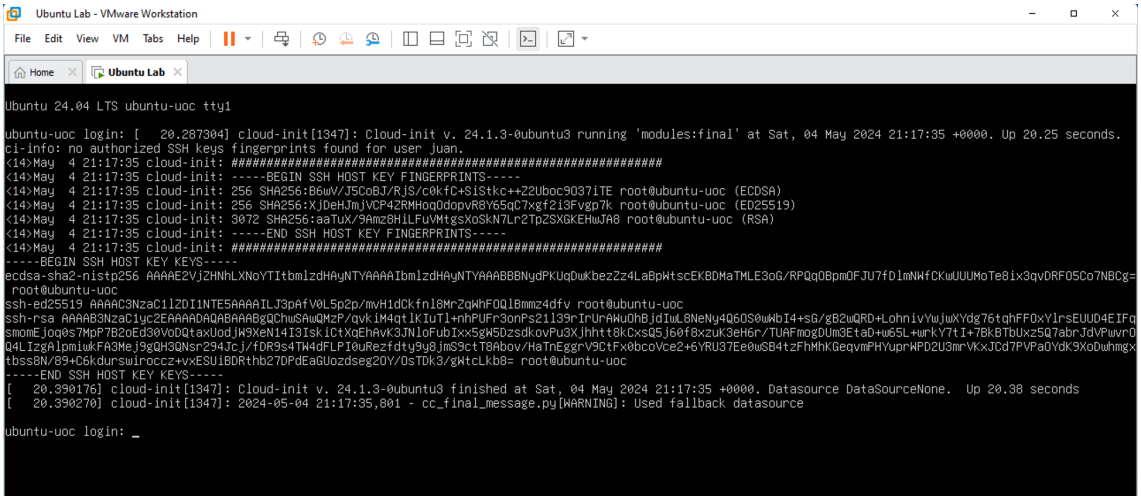


Ilustración 11: Primer Access Ubuntu

3.3.1. Instalación de herramientas de red

Para gestionar de forma mas sencilla la maquina, realizamos la instalación del paquete de instalación “net-tools” a través del comando “apt-get install net-tools”.

Ilustración 12: Instalación de net-tools

Validamos la configuración de red y la conectividad hacia Internet utilizando los comandos:

- Ifconfig
- Ping www.google.com

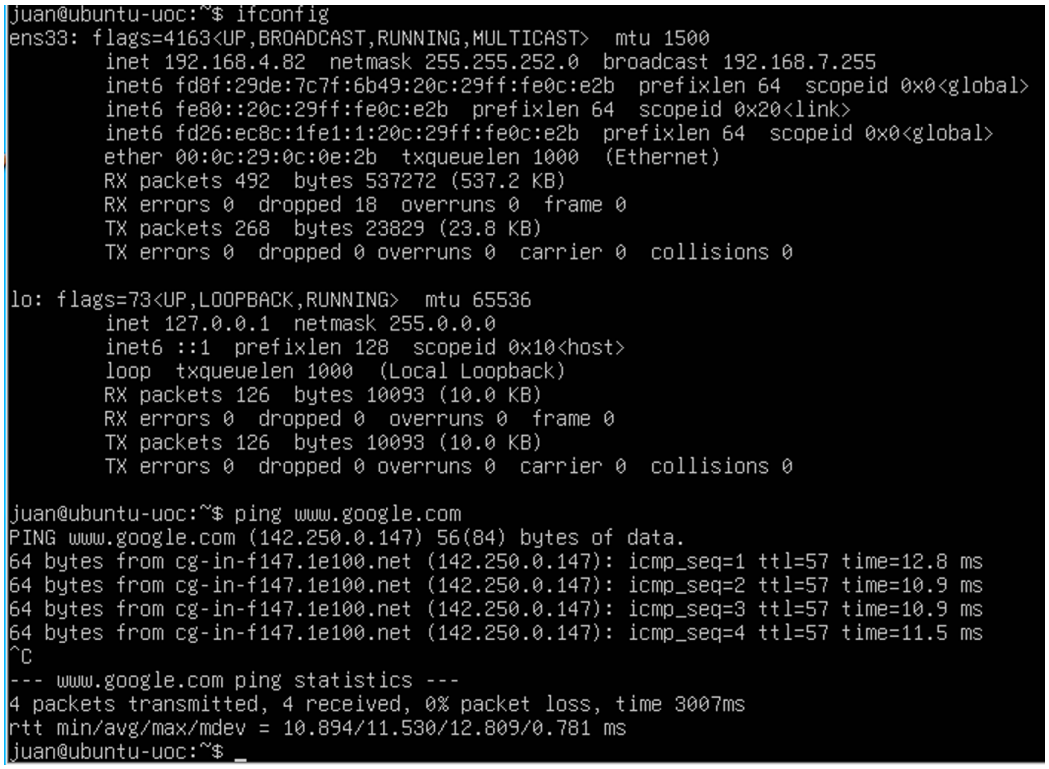


Ilustración 13: Configuración de red Ubuntu

Con la configuración de red ya operando de forma correcta, procedemos a tomar control del servidor ubuntu desde el cliente SSH a través del comando ssh juan@192.168.4.82.

Ilustración 14: Primer acceso SSH

3.4. Instalación de BIND9

En esta parte realizaremos el despliegue de BIND9 para lograr manejar la zona appfirewall.cl y gestionar las consultas DNS locales de la red, a través de función de DNS recursivo. Por otra parte, en el proceso se desarrollan configuraciones de seguridad para permitir que solo la resolución de nombres seán respondidas desde la red interna.

Para iniciar el proceso de instalación [5], primero realizamos la actualización de nuestro sistema utilizando el comando “apt update -y && apt upgrade -y”.

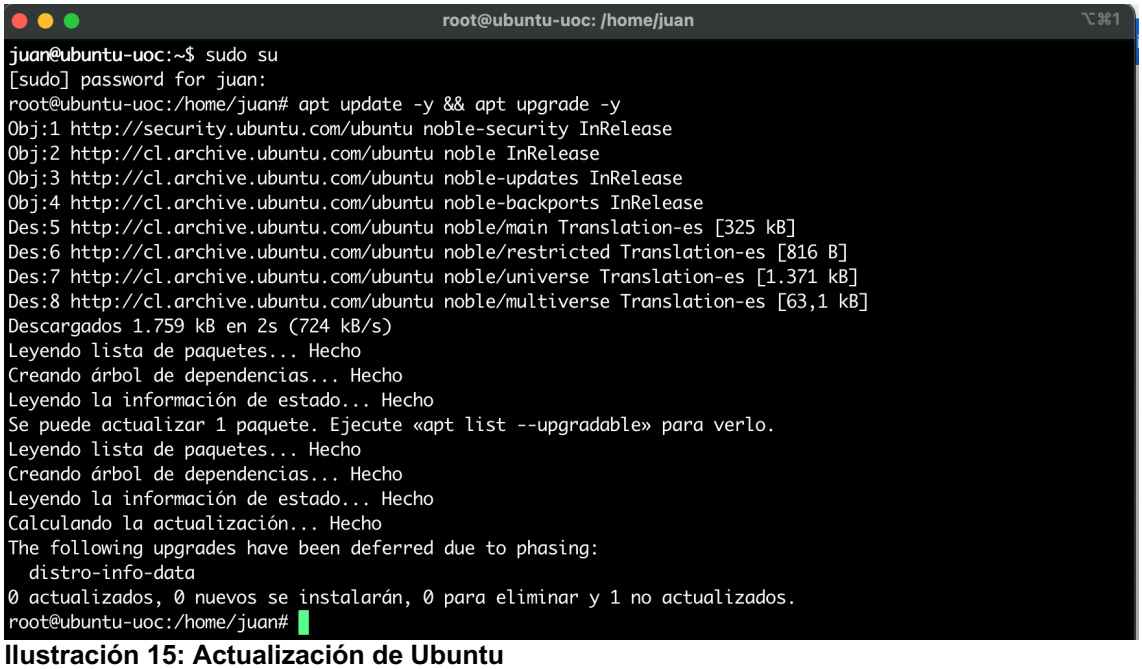


Ilustración 15: Actualización de Ubuntu

Con el sistema ya actualizado, se procede a instalar BIND 9 en el servidor. Para esto es necesario instalar los siguientes 3 paquetes:

- bind9 - The BIND 9 DNS server software.
- bind9utils - Utilities that make working with BIND 9 easier.
- bind9-doc - A documentation package for BIND 9.

Utilizamos el comando “apt install bind9 bind9utils bind9-doc -y” para ejecutar la respectiva instalación.

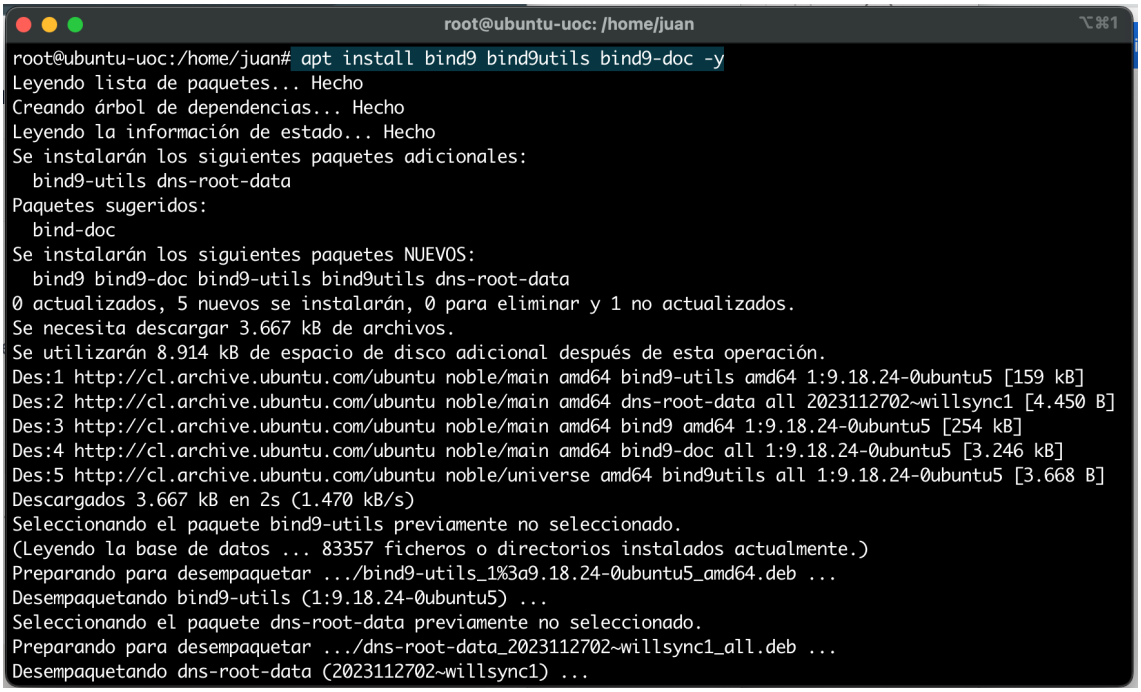


Ilustración 16: Instalación de bind9

Validamos la instalación chequeando que el servicio named de BIND 9 se encuentre operando. Para esto usamos el comando “systemctl status bind9”

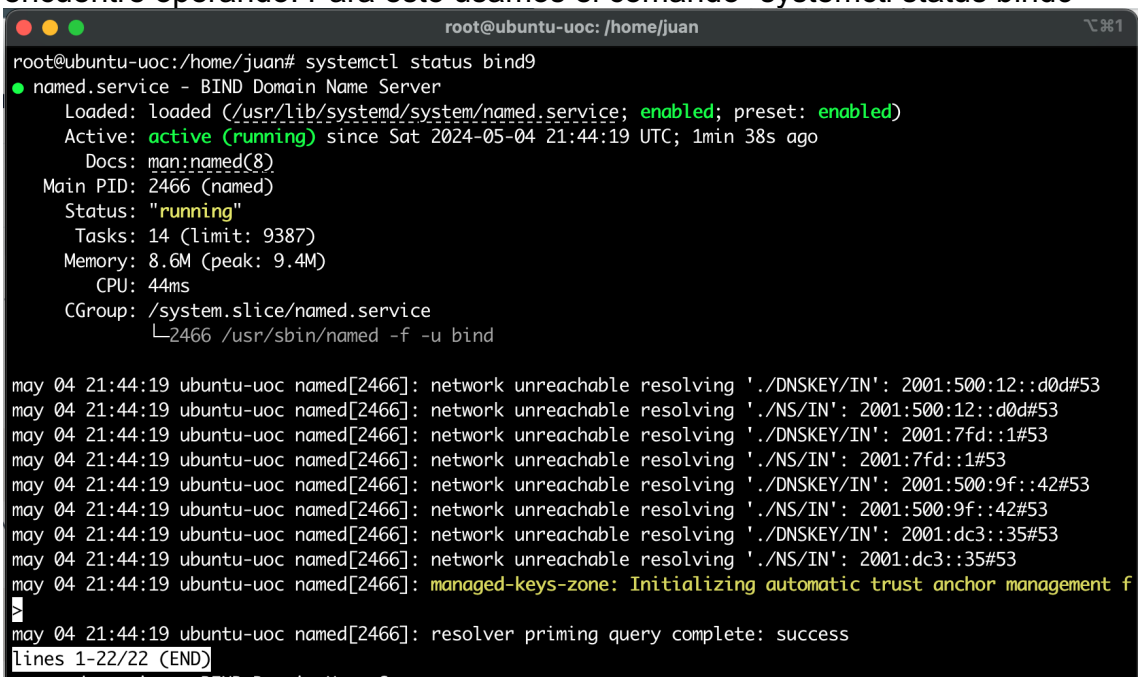


Ilustración 17: Servicio en estado running de bind9

A continuación se debe realizar las siguientes configuraciones en nuestro BIND9:

- Una directiva acl que define nuestra red de area local (LAN).
- Una directiva allow-query que define que direcciones IP pueden enviar consultas DNS al servidor.
- Una directiva forwarders que define a que servidores DNS reenviar este servidor las consultas recursivas.
- Una directiva recursion que permite las consultas DNS recursivas al servidor.

Estas configuraciones las realizamos en el archivo named.conf.options. Para esto editamos el archivo utilizando el comando “nano /etc/bind/named.conf.options”

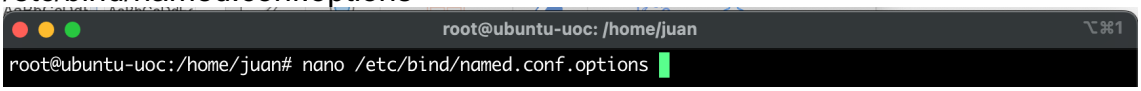


Ilustración 18: Archivo de configuración de bind 9

Editamos el archivo permitiendo el trafico desde la red local 192.168.4.0/24 definida en la acl LAN. Dentro de la configuración definimos el directorio que utilizamos por defecto “/var/cache/bind”. En la configuración utilizamos “allow-query” para permitir solo consultas desde las red LAN. Luego indicamos los servidores que serán forwarders (8.8.8.8) y habilitamos la recursión (yes).

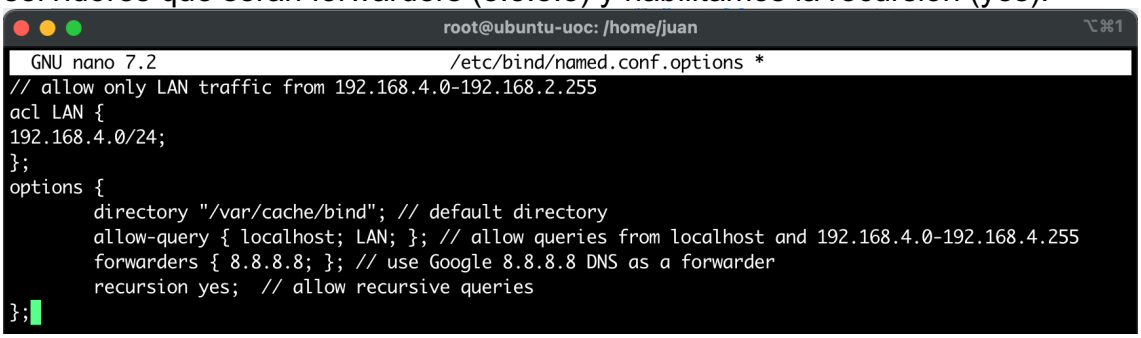


Ilustración 19: Configuración de archivo named.conf

Con las directivas ya creadas, iniciamos el proceso de configuración de la zona appfirewall.cl utilizando el archivo named.conf.local con el editor nano.

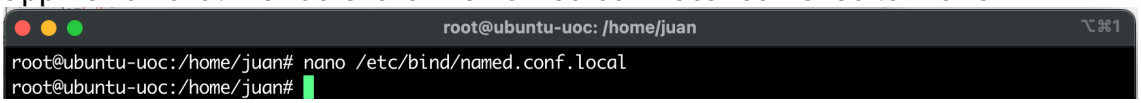


Ilustración 20: Ubicación de archivo named.conf.local

En este archivo se procede a crear la zona “appfirewall.cl” la cual será utilizada mas adelante para nuestros sistema de gestión web y el reverso de la zona. La configuración aplicada en el archivo de refleja en la siguiente imagen.

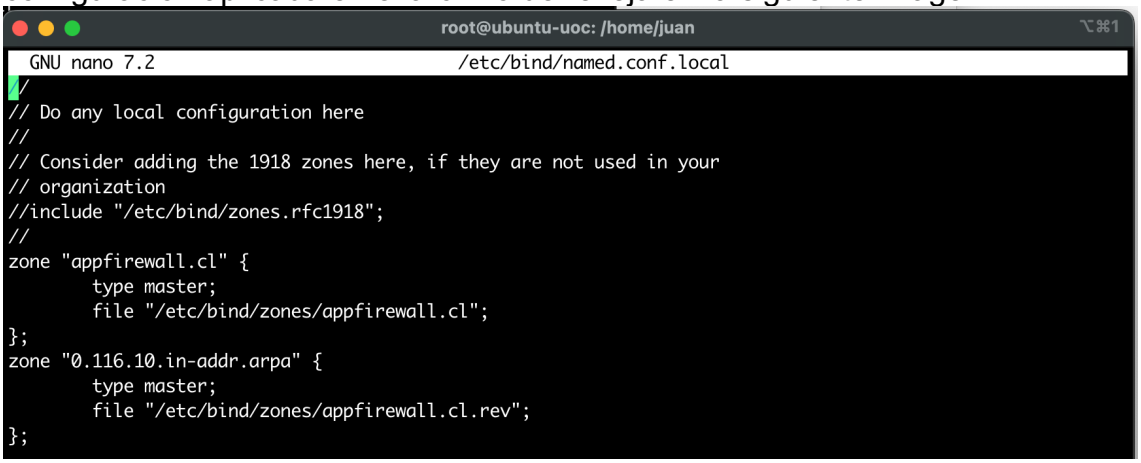


Ilustración 21: configuración de archivo named.conf.local

Utilizamos el comando “named-checkconf” para validar la configuración del archivo named.conf.options. Luego procedemos a crear un directorio zones utilizando el comando “mkdir /etc/bind/zones”. Copiamos el archivo “/etc/bind/db.local” en el nuevo directorio con el nombre “appfirewall.cl” para aplicar las configuraciones de la zona creada.

```
root@ubuntu-uoc: /home/juan
root@ubuntu-uoc:/home/juan# nano /etc/bind/named.conf.local
root@ubuntu-uoc:/home/juan# named-checkconf /etc/bind/named.conf.options
root@ubuntu-uoc:/home/juan# mkdir /etc/bind/zones
root@ubuntu-uoc:/home/juan# cp /etc/bind/db.local /etc/bind/zones/appfirewall.cl
root@ubuntu-uoc:/home/juan#
```

Ilustración 22: creación de zona appfirewall.cl

Se procede a editar el archivo utilizando el comando "nano /etc/bind/zones/appfirewall.cl"

```
root@ubuntu-uoc:/home/juan# nano /etc/bind/zones/appfirewall.cl
```

Modificamos los datos del archivo con la información de nuestra zona appfirewall.cl. como se visualiza en la Ilustriación. En el archivo se procede a crear dos registros:

- Un registro NS que por defecto redirija las consultas a www.appfirewall.cl
- Un registro A que resuelva www hacia el servidor ubuntu 192.168.4.82.

```
root@ubuntu-uoc: /home/juan
 GNU nano 7.2                          /etc/bind/zones/appfirewall.cl
$TTL    604800
; SOA record with MNAME and RNAME updated
@       IN      SOA     appfirewall.cl. root.appfirewall.cl. (
                              4         ; Serial Note: increment after each change
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
; Name server record
@       IN      NS      www.appfirewall.cl.
; A record for name server
www     IN      A       192.168.4.82
```

Ilustración 23: configuración de zona appfirewall.cl

Se procede a validar la configuración a traves del comando "named-checkzone appfirewall.cl /etc/bind/zones/appfirewall.cl"

```
root@ubuntu-uoc:/home/juan# named-checkzone appfirewall.cl /etc/bind/zones/appfirewall.cl
zone appfirewall.cl/IN: loaded serial 3
OK
root@ubuntu-uoc:/home/juan#
```

Ilustración 24: validación de zona

Con esta zona lista, se procede a crear la zona reversa editando el archivo a través del comando "nano /etc/bind/zone/appfirewall.cl.rev".

```
root@ubuntu-uoc:/home/juan# nano /etc/bind/zones/appfirewall.cl.rev
root@ubuntu-uoc:/home/juan#
```

Ilustración 25: validación de zona reversa

En este archivo se procede a crear los siguientes 3 rgistros tal como se presentan en la Ilustración:

- Un registro NS que por defecto redirija las consultas a www.appfirewall.cl
- Un registro A que resuelva www hacia el servidor ubuntu 192.168.4.82
- Un registro PTR hacia www.appfirewall.cl

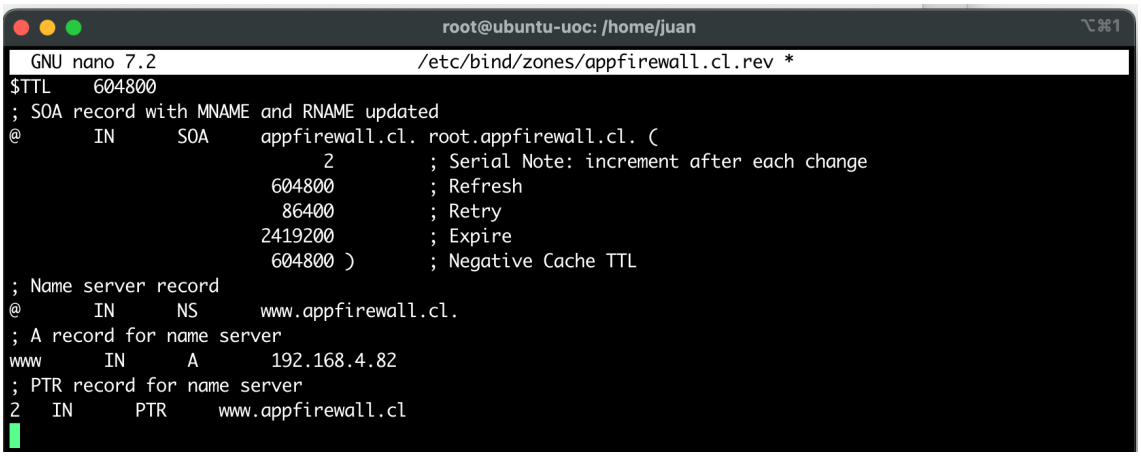


Ilustración 26: configuración de zona appfirewall.cl

Luego procede a validar la configuración de la zona a través del comando “named-checkzone appfirewall.cl /etc/bind/zones/appfirewall.cl.rev”

Ilustración 27: validación de zona reversa

Con la configuración ya finalizadas se procede a realizar un reinicio del servicio bind9 y luego se revisa que haya iniciado correctamente. Para este procedimiento se utilizan los siguientes comandos:

- systemctl restart bind9
- systemctl status bind9

Se valida que la consola nos muestre el sevicio named.service en estado “active (running)”. Con esto ya validado, se procede a ejecutar pruebas de resolución desde un cliente dentro de la red.

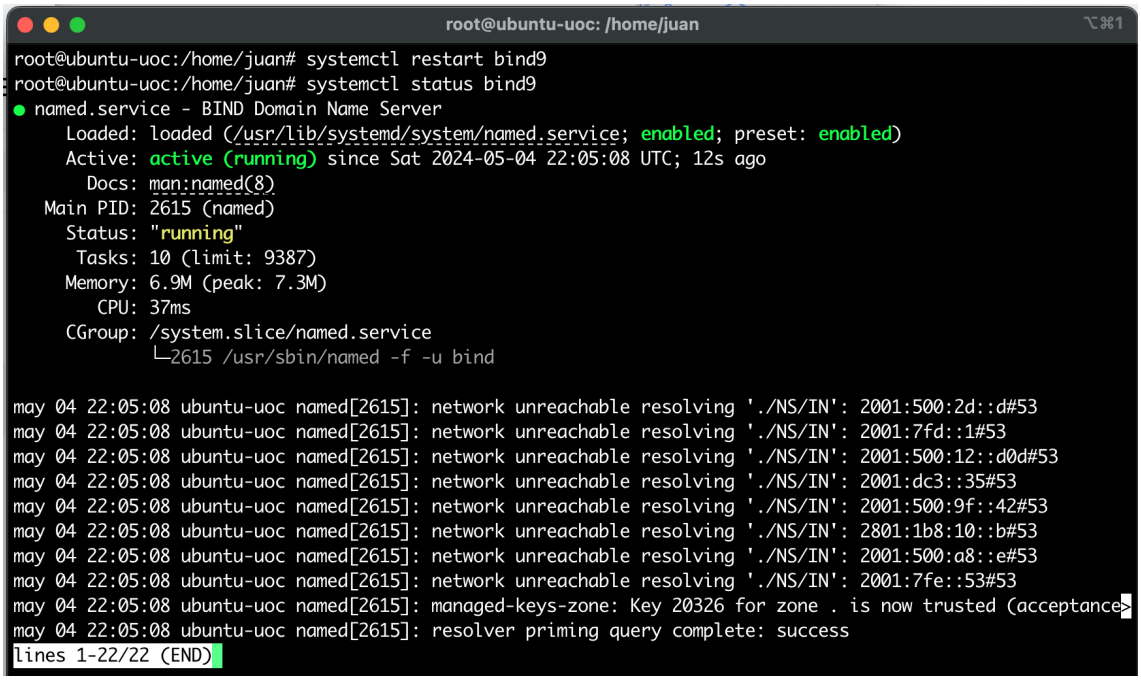


Ilustración 28: validación de estado running en bind 9

Para validar la resolución aplicamos consultas a la zona appfirewall.cl y al dominio www.google.com siempre apuntando como servidor la dirección 192.168.4.82. Los comandos utilizados son:

- nslookup www.appfirewall.cl 192.168.4.82
- nslookup www.google.com 192.168.4.82

Como resultado se espera obtener la resolución de la o las ips de los dominios como se observa en la Ilustración.

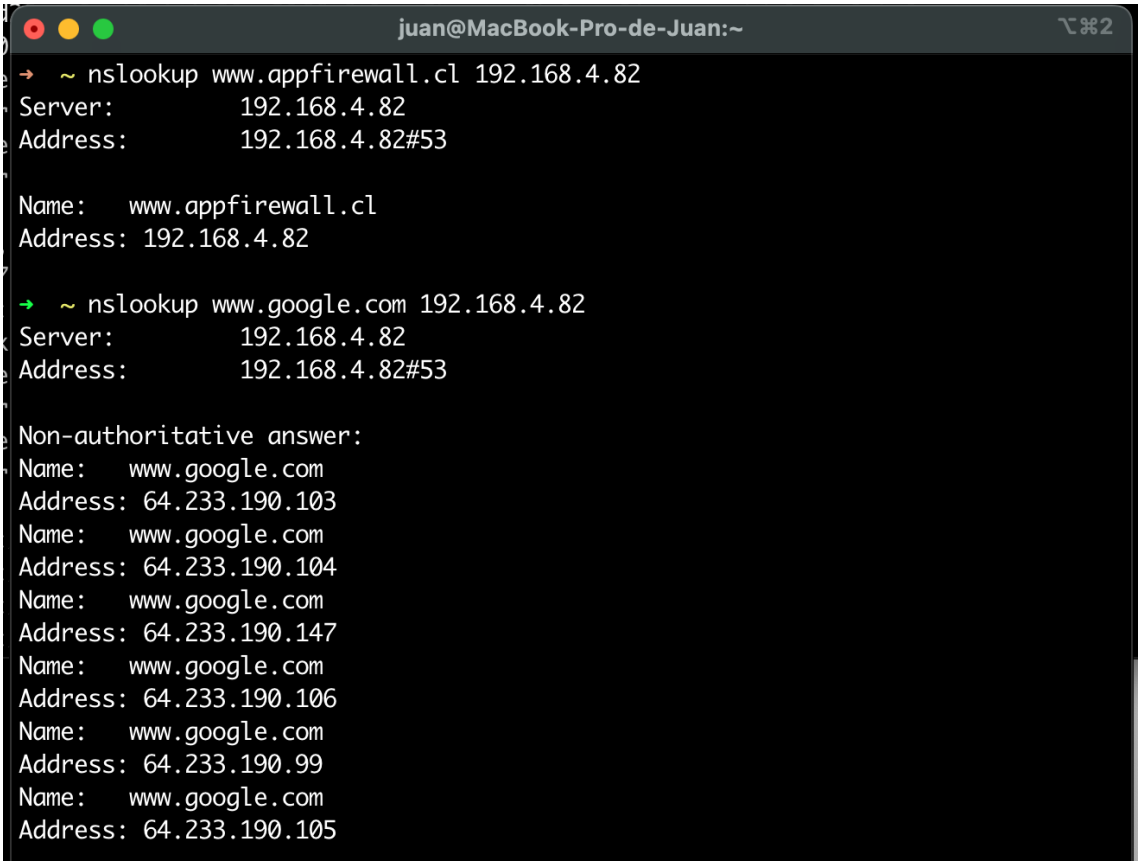


Ilustración 29: Prueba de validación utilizando nslookup

3.5. Instalación NGINX

Para el proyecto necesitaremos mantener un manejo del servicio web de Kibana, por lo que se utilizará NGINX dado su sencilla instalación en el ambiente y fácil configuración para gestionar el puerto y publicación del servicio web.

Se procede a con la instalación [6], y dado que nginx está disponible en los repositorios predeterminados de Ubuntu, es posible instalarlo desde estos repositorios utilizando el comando “sudo apt install nginx”.

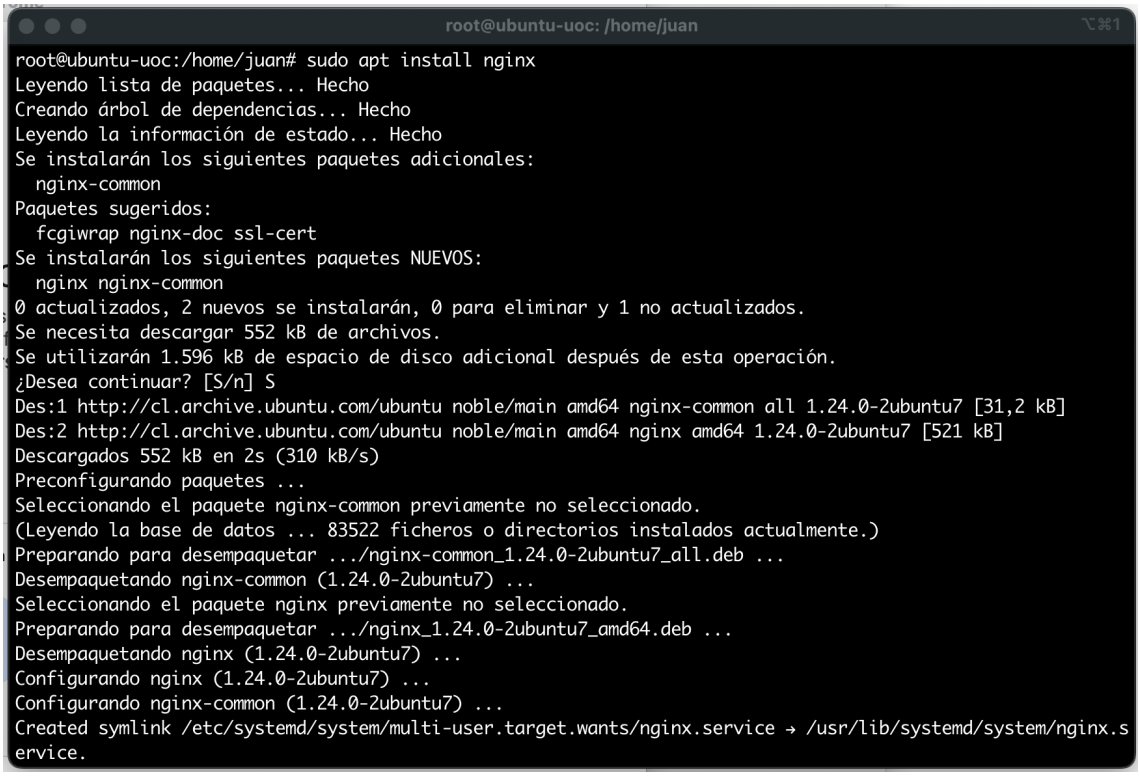


Ilustración 30: proceso de instalación nginx

Al finalizar la instalación, se debe revisar que el servicio nginx se encuentra activo. Se utiliza el comando “systemctl status nginx” y se valida que el servicio se encuentre en estado “active (running)”.

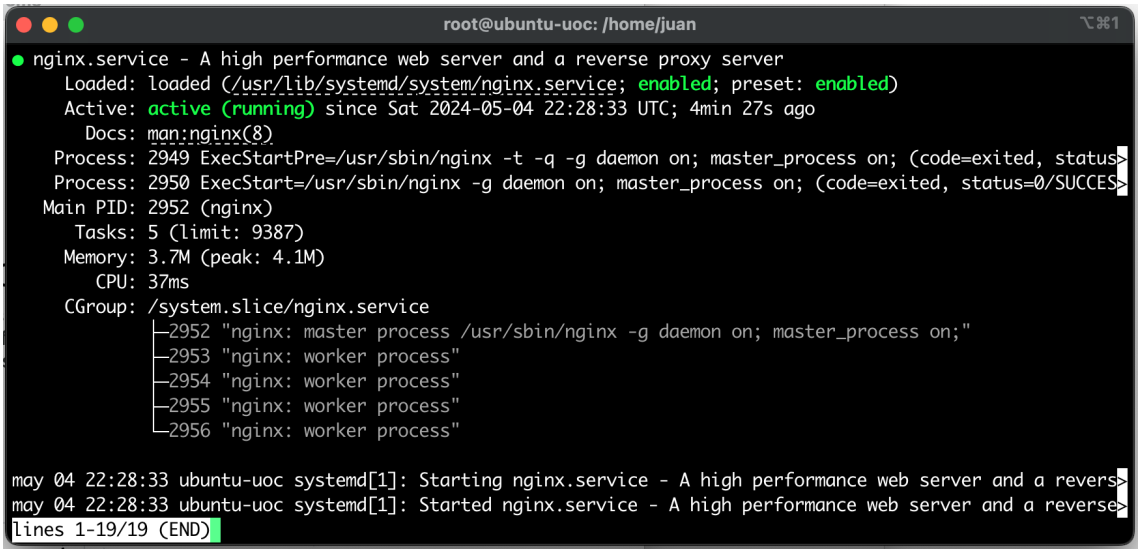


Ilustración 31: validación de estado servicio nginx

Si la configuración ha finalizado de forma correcta, podremos acceder al sitio por defecto que levanta nginx en el puerto 80/http. Para esto desde el navegador del cliente se accede a http://www.appfirewall.cl.

Ilustración 32: validación de acceso nginx

3.6. Instalación de Elasticsearch, Logstash, y Kibana

Como pre-requisito [7] ELK Stack necesita que OpenJDK11 se encuentre instalado, por lo que se procede a iniciar el paquete de instalación "default-jre" y "default-jdk". Iniciamos la instalación [8] con el comando "sudo apt install default-jre".

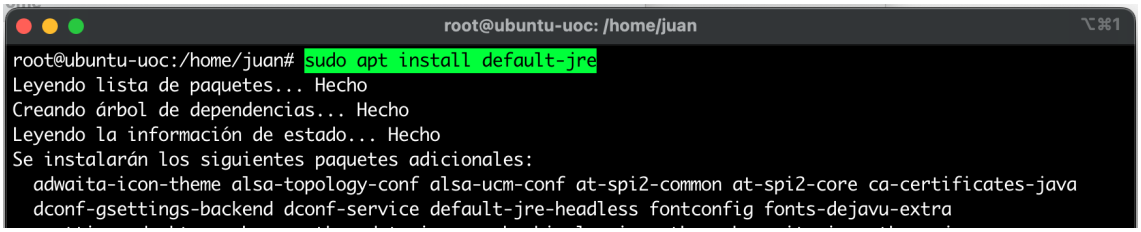


Ilustración 33: instalación de java

Iniciamos la instalación con el comando "sudo apt install default-jdk".

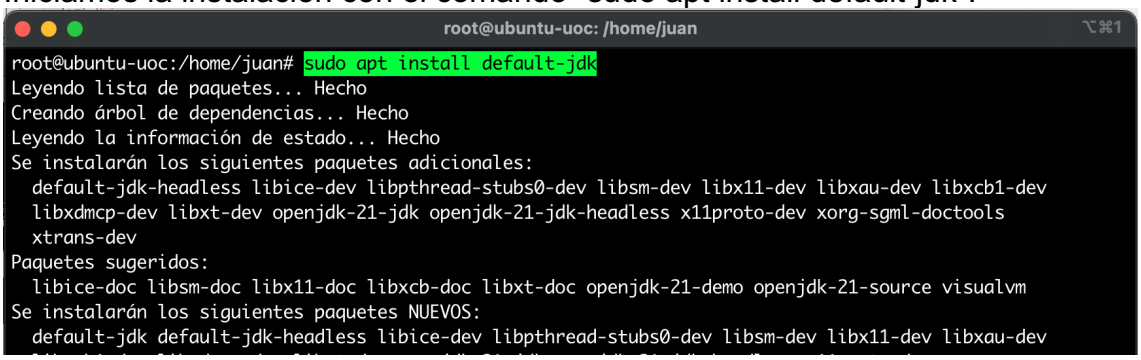


Ilustración 34: instalación jdk

Al finalizar la instalación, podemos revisar las versiones instalados con el comando "javac -version" y "java -version".

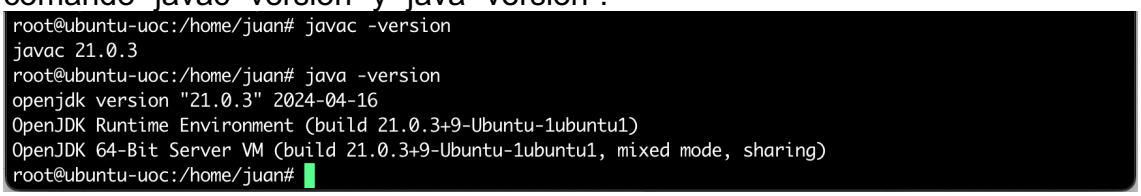


Ilustración 35: validación de versión de java

Con OpenJDK11 operando, se debe proceder a realizar la instalación de Elastic stack. Debido a que los componentes de Elasticsearch no se encuentran en los repositorios de Ubuntu, es que se debe añadir el siguiente repositorio. Para esto utilizamos los comandos:

- wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg
- echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-8.x.list

Luego actualizamos los repositorios con “sudo apt update”.

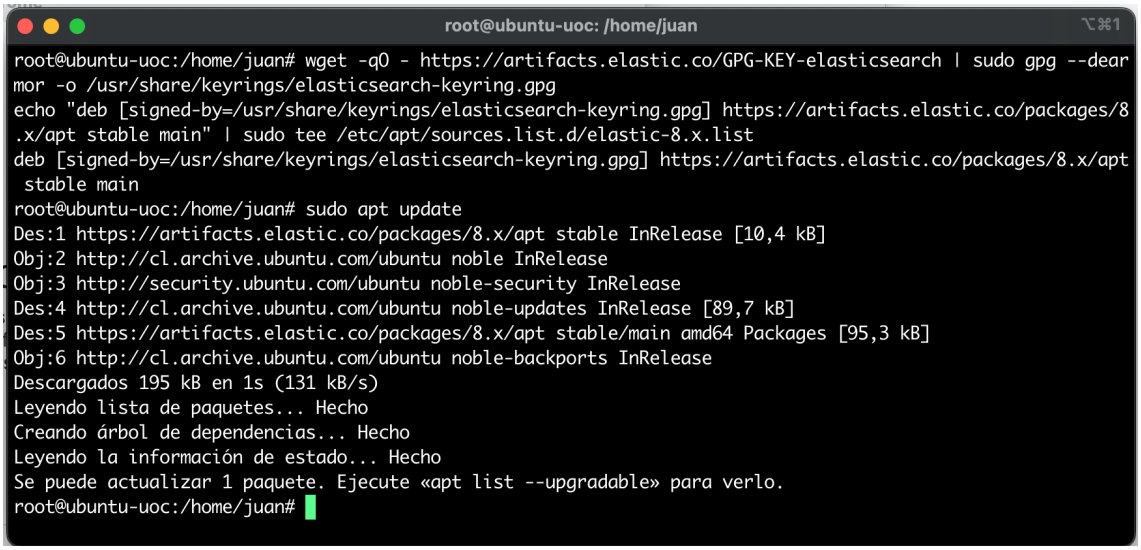


Ilustración 36: actualización de repositorios

3.6.1. Instalación de elasticsearch

Para la instalación de elasticsearch [9] utilizamos el comando “sudo apt install elasticsearch”

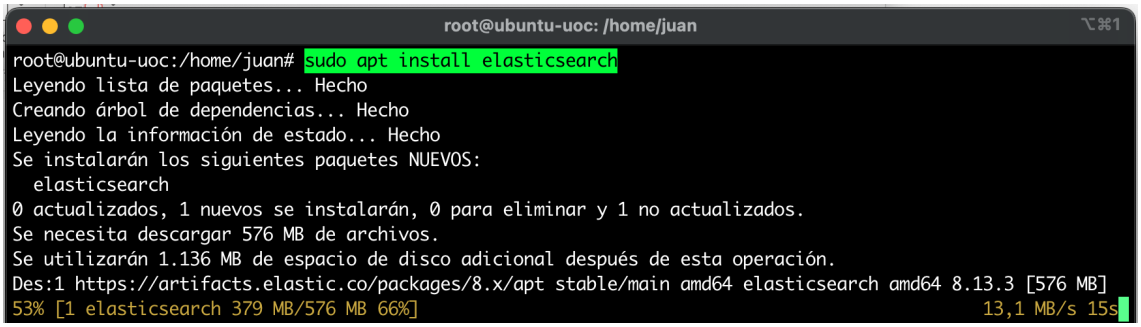


Ilustración 37: instalación de elasticsearch

Cuando finaliza la instalación, utilizando un editor de texto accedemos al archivo de configuración. Para este caso utilizamos “nano /etc/elasticsearch/elasticsearch.yml”

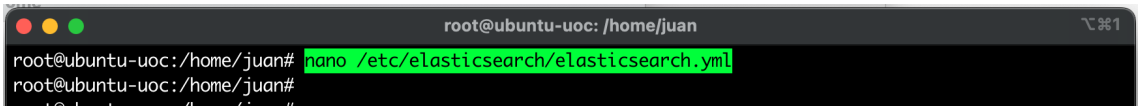


Ilustración 38: ubicación de archivo de configuración elasticsearch

Dentro de la configuración se debe descomentar la linea “network.host: 0.0.0.0” y “http.port: 9200”. Con esto elastic estará escuchando en cualquier ip del servidor en el puerto 9200.

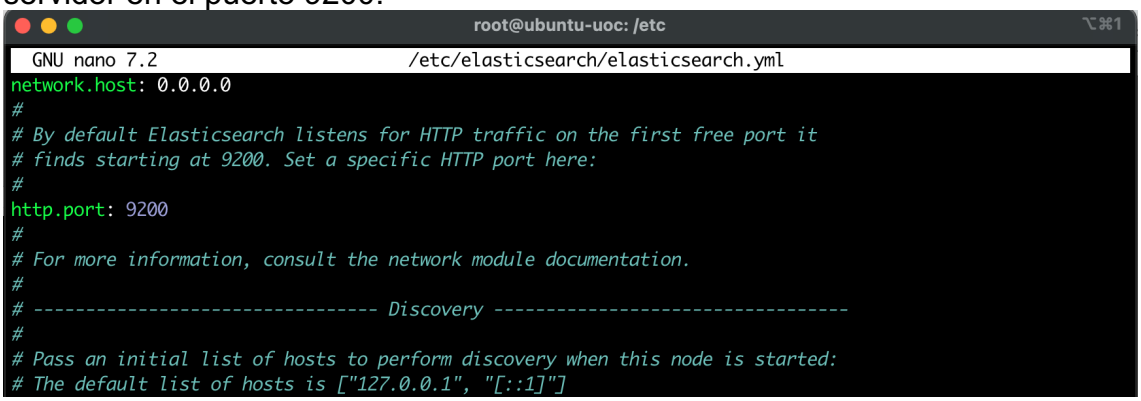


Ilustración 39: configuración de elasticsearch

Con la configuración ya aplicada, se debe proceder a habilitar e iniciar el proceso de elasticsearch. Si el servicio se encuentra operando según la configuración utilizando el comando netstat encontraremos el servicio operando los puertos 9300 y 9200.

Para esto utilizamos los comandos:

- systemctl enable elasticsearch
- systemctl start elasticsearch
- netstat -tulnp | grep 9300

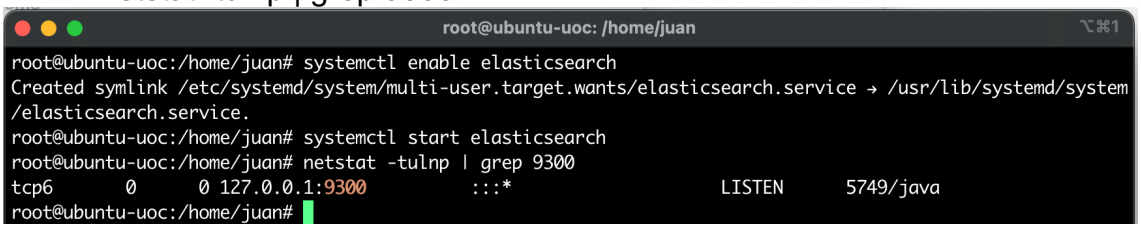


Ilustración 40: validación de puertas y servicios habilitados

3.6.2. Instalación de Kibana

La plataforma de visualización de datos Kibana se debe instalar a través del mismo repositorio que elasticsearch. Para esto se debe utilizar el comando “apt install kibana”

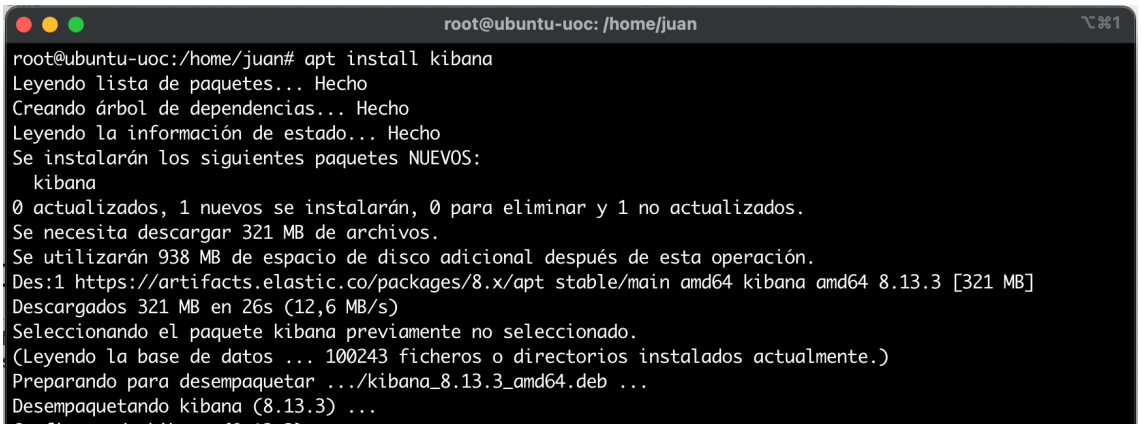


Ilustración 41: instalación de kibana

Al finalizar la instalación de Kibana, habilitamos e iniciamos el servicio de kibana a través de los comandos:

- systemctl enable kibana
- systemctl start kibana

Revisamos que la configuracipón se encuentre operando de forma correcta en el puerto 5601 a través del comando “netstat -tulpn | grep 5601

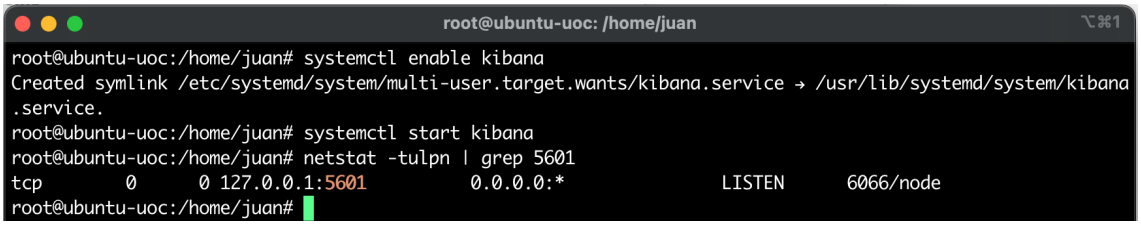


Ilustración 42: validación de servicios en operación

3.6.3. Instalación logstash

Para la instalación de logstash se debe utilizar el mismo repositorio, por lo que se procede a utilizar el comando “apt install logstash”.

Ilustración 43: instalación de logstash

Con la instalación finalizada se procede a crear un pipeline de Logstash, para esto se debe crear un nuevo archivo de configuración en el respectivo repositorio. Para esto se utiliza el comando "nano /etc/logstash/conf.d/basic.conf".

Ilustración 44: archivo de configuración logstash

En este archivo utilizamos el "input" para definir el archivo de log de origen. Para este archivo basico de evaluación se mantendrá el origen de logs "/var/log/*.log". En el "filter" estructuramos con grok los filtros que serán utilizados a través de la web de Kibana y que se utilizan en los eventos recolectados.
En el apartado del output configuramos el envío de logs a elasticsearch. Este apartado será modificado posteriomente para agregar las configuraciones de autenticación.

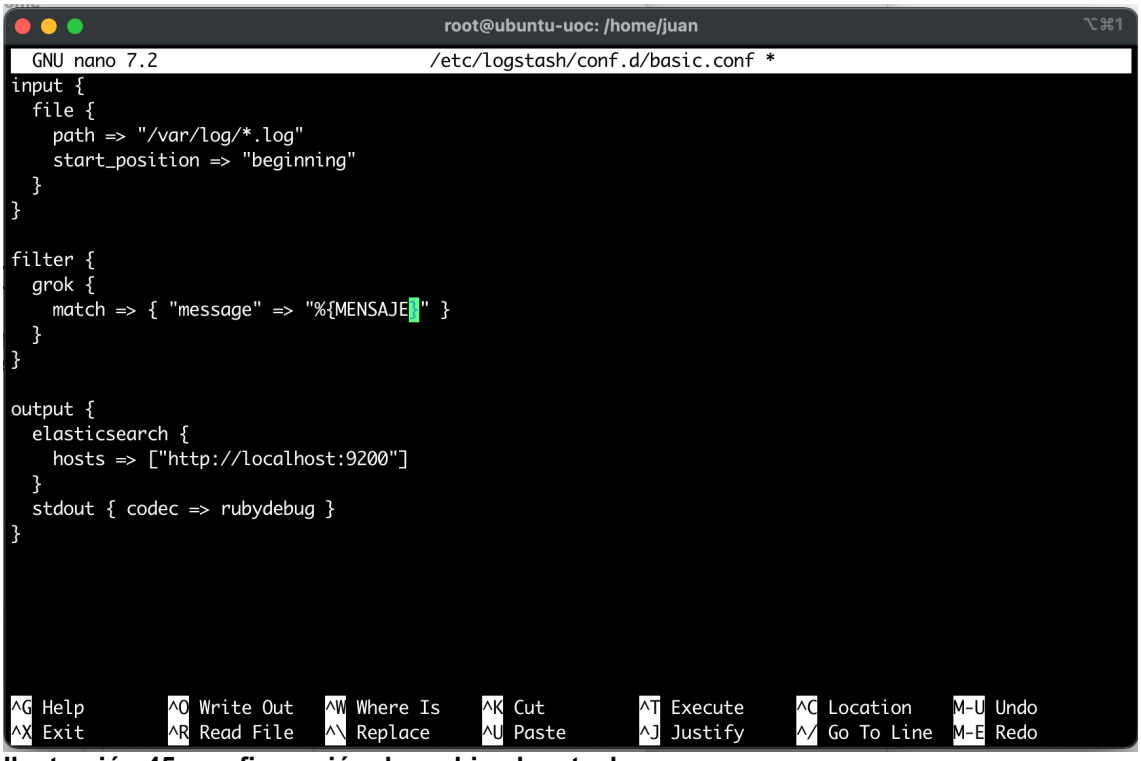


Ilustración 45: configuración de archivo logstash

Con el archivo ya creado, procedemos a habilitar e iniciar el servicio de logstash.

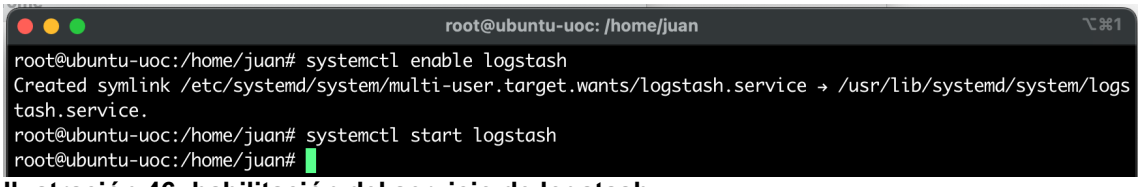


Ilustración 46: habilitación del servicio de logstash

3.6.4. Configuración de nginx para acceder a Kibana

Ejecutaremos una configuración en NGINX para publicar a través del puerto 80 el servicio de gestión web de Kibana. Para lograr esto se desarrolla una configuración de proxy hacia “http://localhost:5601”.

Para esto creamos el archivo www.appfirewall.cl utilizando el comando “nano /etc/nginx/sites-available/www.appfirewall.cl” y aplicamos la configuración de proxy como se observa en la Ilustración.

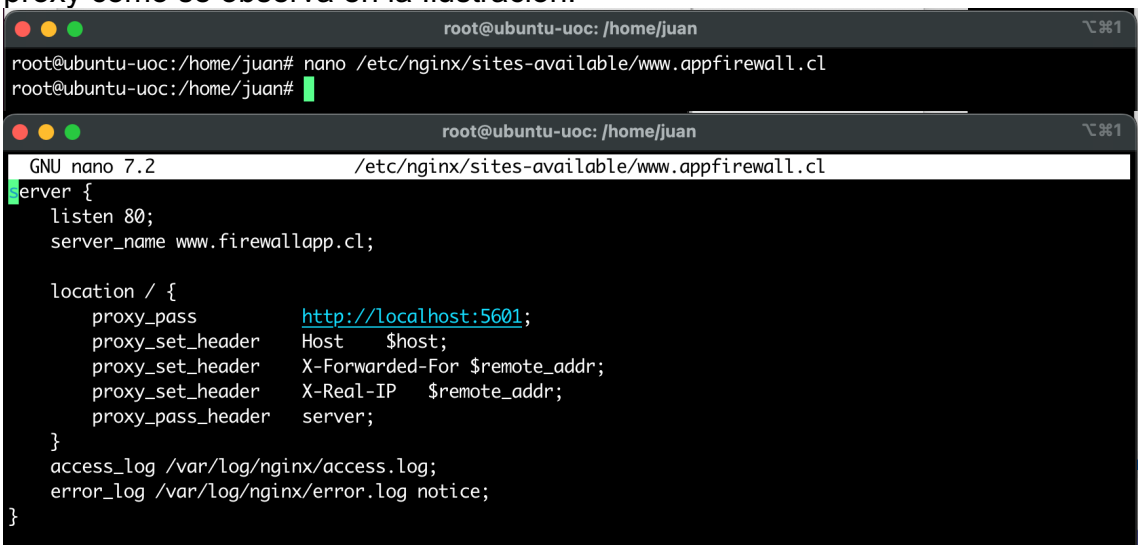


Ilustración 47: archivo de configuración para www.firewallapp.cl

Con el archivo creado, se procede quitar el enlace al sitio web default. Para esto utilizamos “unlink”. Luego enlazamos el nuevo sitio “www.appfirewall.cl”. Utilizamos los comandos:

- sudo unlink /etc/nginx/sites-enabled/default
- sudo ln -s /etc/nginx/sites-available/www.appfirewall.cl /etc/nginx/sites-enabled/

Ilustración 48: habilitación de configuración de nginx para appfirewall.cl

Validamos y cargamos la nueva configuración en nginx utilizando los comandos:

- sudo nginx -t
- systemctl restart nginx

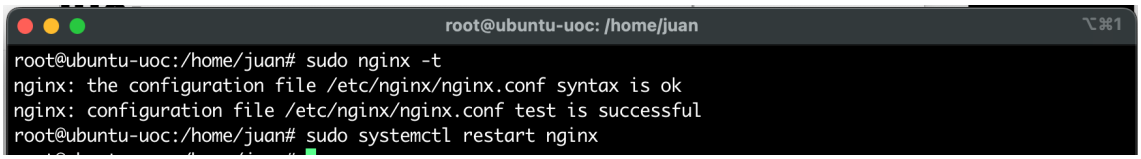


Ilustración 49: reinicio de servicio nginx

3.6.5. Primera configuración de Kibana

Con la configuración NGINX preparada, procedemos a realizar la configuración inicial de Kibana desde la web. Para ellos será necesario de seguir el procedimiento del “Anexo II: Primera configuración de Kibana”. En la configuración se accede al sitio “appfirewall.cl” y se procede a configurar la autenticación del sistema para la gestión de la plataforma Elastic Stack.

Al finalizar la instalación, se procederá a iniciar sesión utilizando el usuario “elastic”, permitiendo la gestión y control completa desde el ambiente.

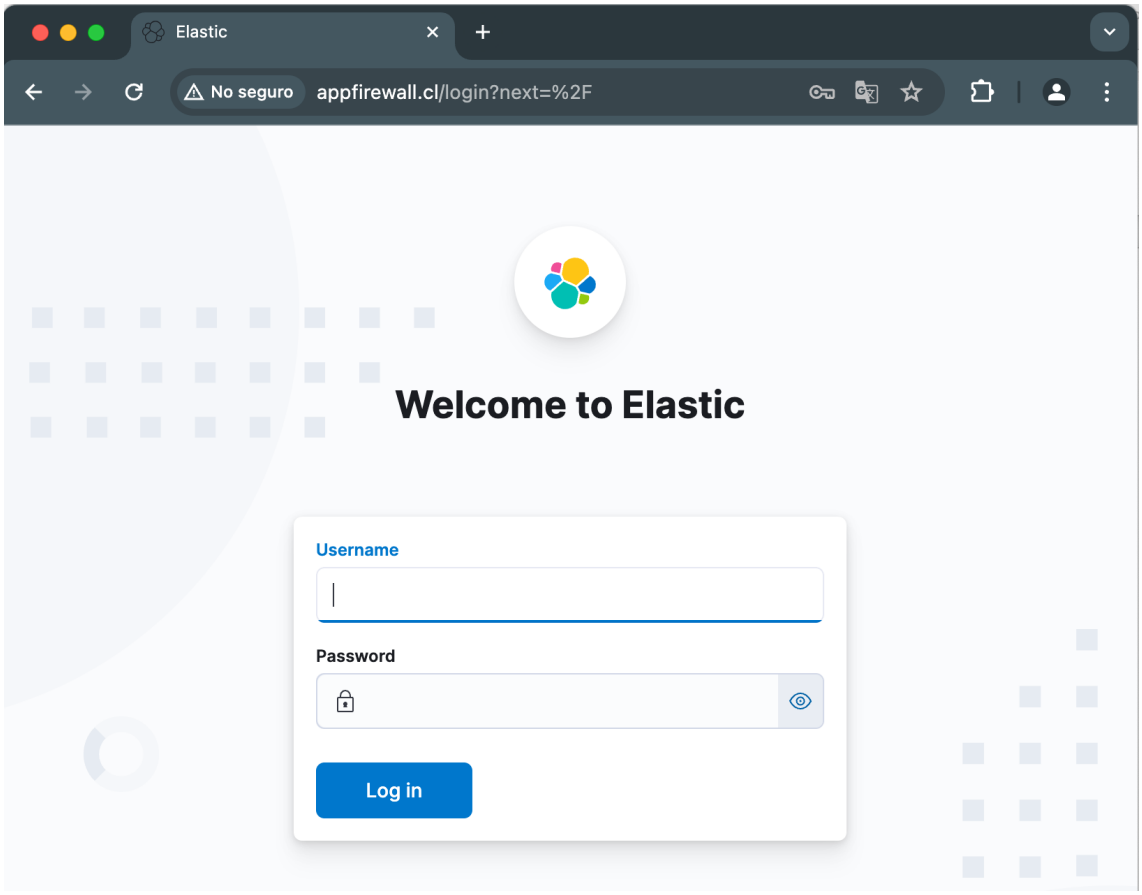


Ilustración 50: primer acceso elastic

3.7. Configuración de integración entre sistemas

Para construir una visualización y dashboard en Kibana, es necesario obtener los datos desde archivos logs. El objetivo inicial es poder observar los eventos de resolución de nombres de la red. Para la definición de que eventos y donde se almacenarán es que se procede a editar el archivo “named.conf.local”.

Ilustración 51: edición de archivo de configuración named.conf.local

En el archivo de configuración en el apartado “logging” se utiliza inicialmente los registros de eventos de las consultas o “queries” de los usuarios. Dentro de la configuración en el apartado “channel queries_log” se registran los eventos en el archivo “/var/log/named/queries”. Este archivo “queries” es el que se utilizará mas adelante como “input” en el pipeline de logstash.

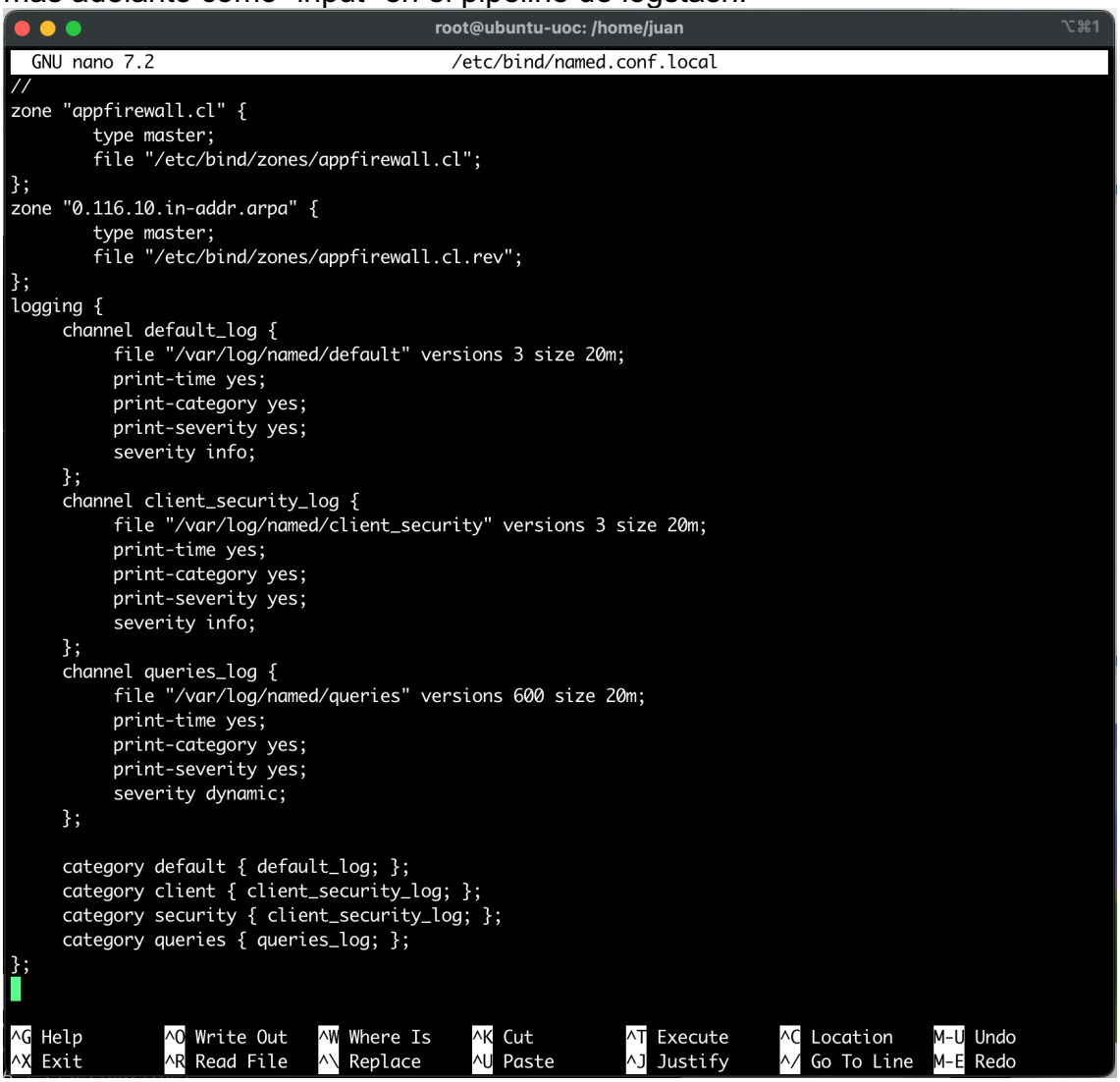


Ilustración 52: configuración de logs

Con la configuración del archivo y la definición de la ubicación de almacenamiento de los logs, es que se crea el respectivo directorio “/var/log/named/”.

Finalmente se reinicia el serivico bind9 utilizando el comando “sudo systemctl restart bind9”.

Ilustración 53: validación del servicio named

Para validar la configuración de almacenamiento de logs se generan consultas desde el servidor utilizando el comando “dig www.google.com @127.0.0.1”.
Con el comando dig se realiza una resolución del nombre www.google.com, la cual se muestra en pantalla.

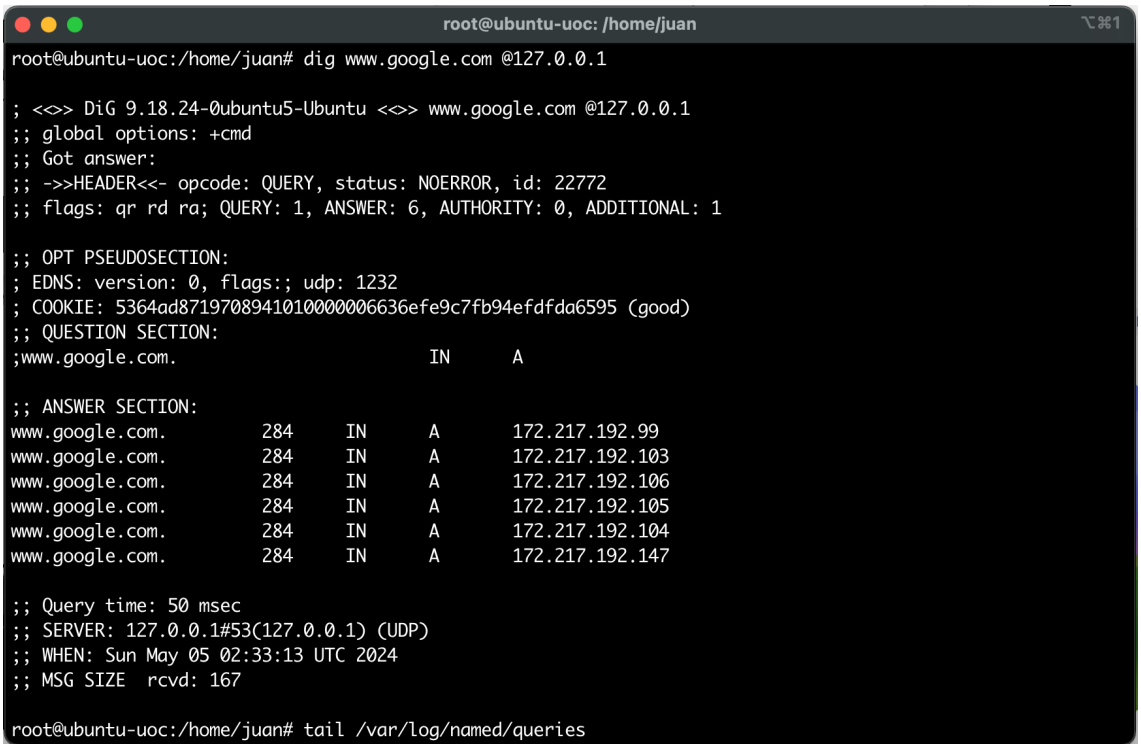


Ilustración 54: pruebas utilizando dig

Esta resolución de nombre genera un registro que es almacenada en el archivo “queries”. Si se consulta el ultimo contenido del archivo queries utilizando el comando “tail /var/log/named/queries” se puede observar la consulta ejecutada de www.google.com.

Ilustración 55: visualización de logs en el servidor

Con el archivo queries ya disponible, será necesario configurar en logstash la utilización del nuevo archivo. Como parte de la configuración, el envío de información hacia elasticsearch deberá ser autenticada en el archivo de

configuración de logstash. Para esto se utilizará la cuenta del usuario “elastic”, por lo que se procede a resetear la contraseña con el comando:

- /usr/share/elasticsearch/bin/elasticsearch-reset-password -u elastic

```
root@ubuntu-uoc:/etc# /usr/share/elasticsearch/bin/elasticsearch-reset-password  -u elastic
This tool will reset the password of the [elastic] user to an autogenerated value.
The password will be printed in the console.
Please confirm that you would like to continue [y/N]y


Password for the [elastic] user successfully reset.
New value: MjGR8Bg5n*gNv-Mw6q6a
root@ubuntu-uoc:/etc#
```

Ilustración 56: reinicio de password para usuarios en elasticsearch

Estas credenciales serán utilizadas en el archivo de configuración “basic.conf”. Para esto se utiliza el comando “nano /etc/logstash/conf.d/basic.conf”.

```
root@ubuntu-uoc: /etc
root@ubuntu-uoc:/etc# nano /etc/logstash/conf.d/basic.conf
```

En la configuración del archivo, se modifica el path apuntando al archivo queries (/var/log/named/queries). Por otra parte, se modifica la configuración para apuntar al hosts “https://192.168.4.82:9200”. Al trabajar con el protocolo https es necesario trabajar con certificados, por lo que en la configuración para el caso puntual del laboratorio se deshabilita la verificación del certificado utilizando “false”. Como parte de la configuración, se define la cuenta de acceso con las credenciales previamente generadas.

```
root@ubuntu-uoc: /etc
  GNU nano 7.2                          /etc/logstash/conf.d/basic.conf
input {
  file {
    path => "/var/log/named/queries"
    }
  }
output {
  stdout {
    }
 elasticsearch {
    hosts => ["https://192.168.4.82:9200"]
    ssl_certificate_verification => false
    user => "elastic"
    password => "MjGR8Bg5n*gNv-Mw6q6a"
    index => "raw-bind9-%{+YYYY.MM.dd}"
    }
  }
```

Ilustración 57: edición de archivo de configuración basic.conf

Con la configuración finalizada del archivo “basic.conf”, se procede a acceder al sitio web de administración y se utilizan las credenciales del usuario “elastic”.

Como nota, se deja constancia que la deshabilitación de la validación del certificado se aplica solo al ambiente laboratorio. En el caso de que se defina pasar a un ambiente productivo, por motivos de seguridad se deben mantener habilitada la validación de certificados validos.

Con la etapa de configuración de logstash finalizada, se procede a configurar en Kibana la recepción de la información para visualizarla de forma grafica.

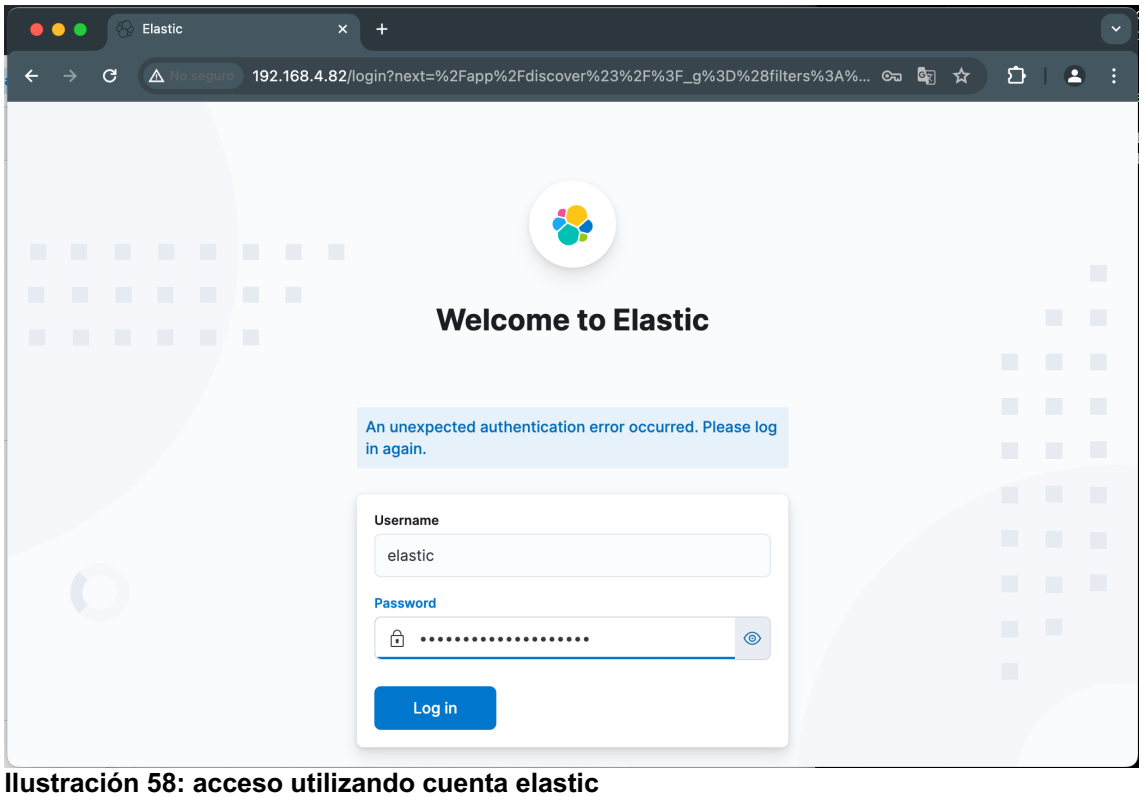


Ilustración 58: acceso utilizando cuenta elastic

En el portal de acceso se procede a dar clic en el "menu" y en el apartado "Analytics" se da clic en "Discover".

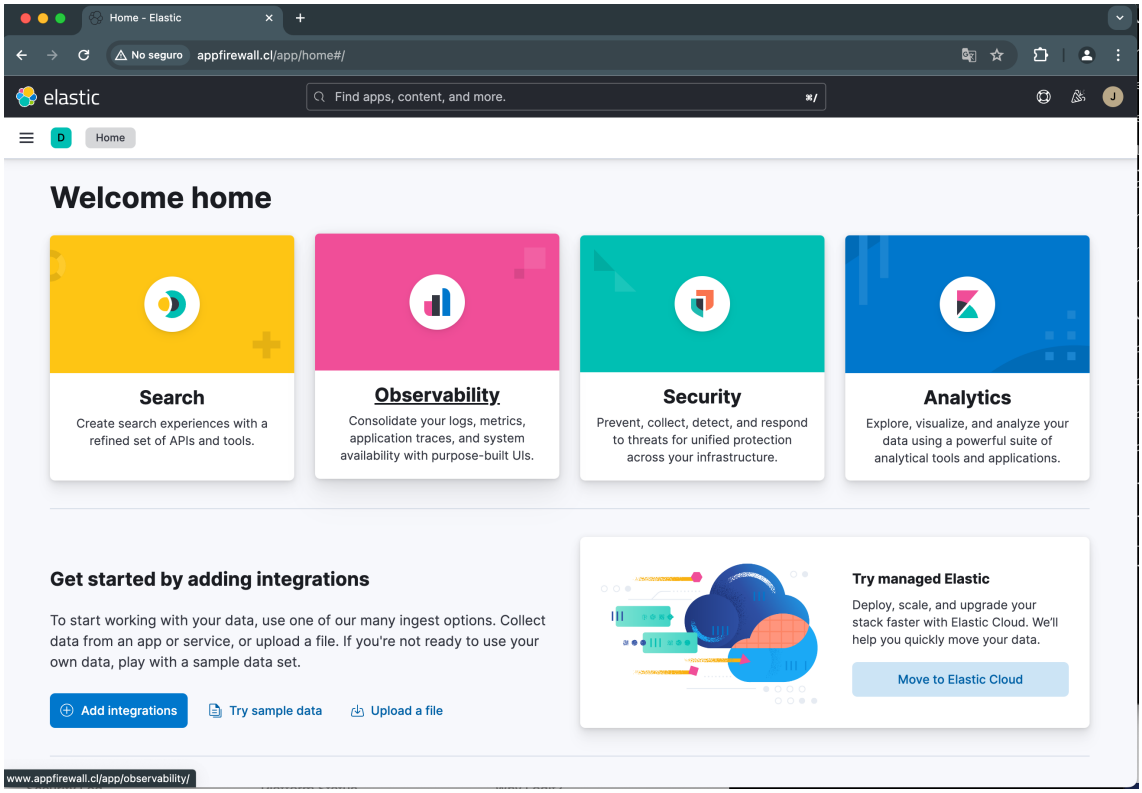


Ilustración 59: primer acceso a elastic

En el apartado se procede a crear un nuevo “data view”. Se define como nombre “DNS QUERIES”, y en index pattern se utiliza el inicio del nombre del archivo “raw”.

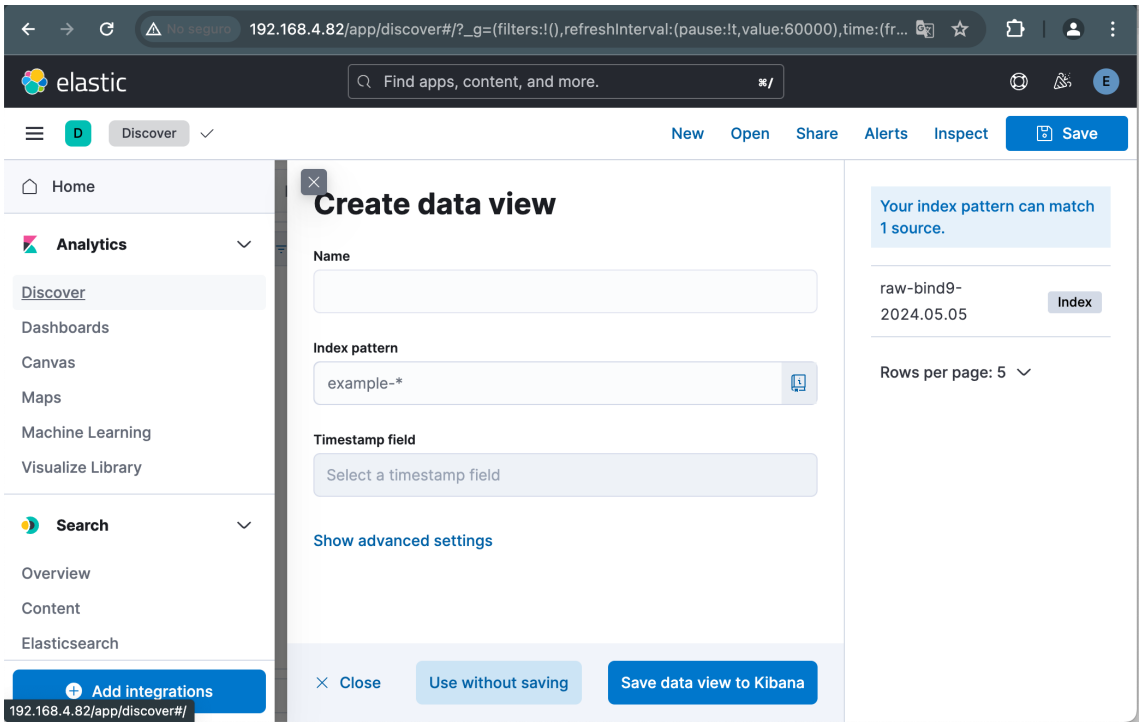


Ilustración 60: creación de data view

Para avanzar con la configuración, se da clic en “Save data view to Kibana” para almacenar la información.

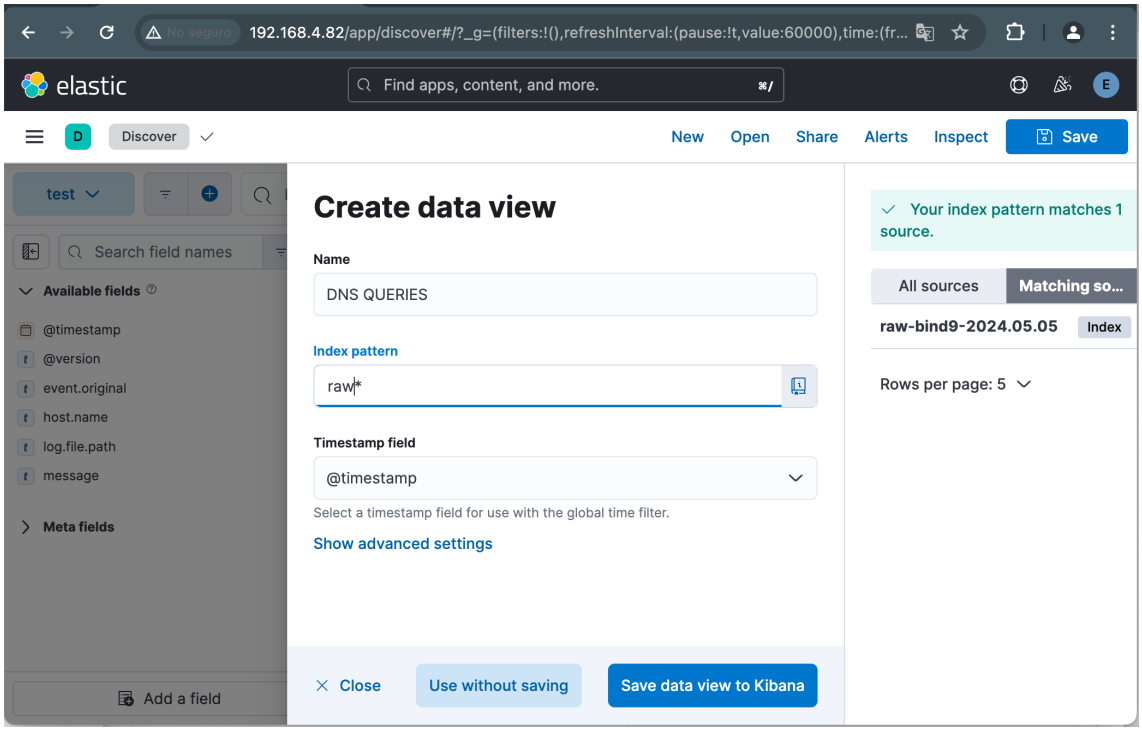


Ilustración 61: configuración para procesamiento de logs

En este apartado ya es posible ver el log almacenado en bruto, con ciertos “fields” disponibles generados de forma automatica.

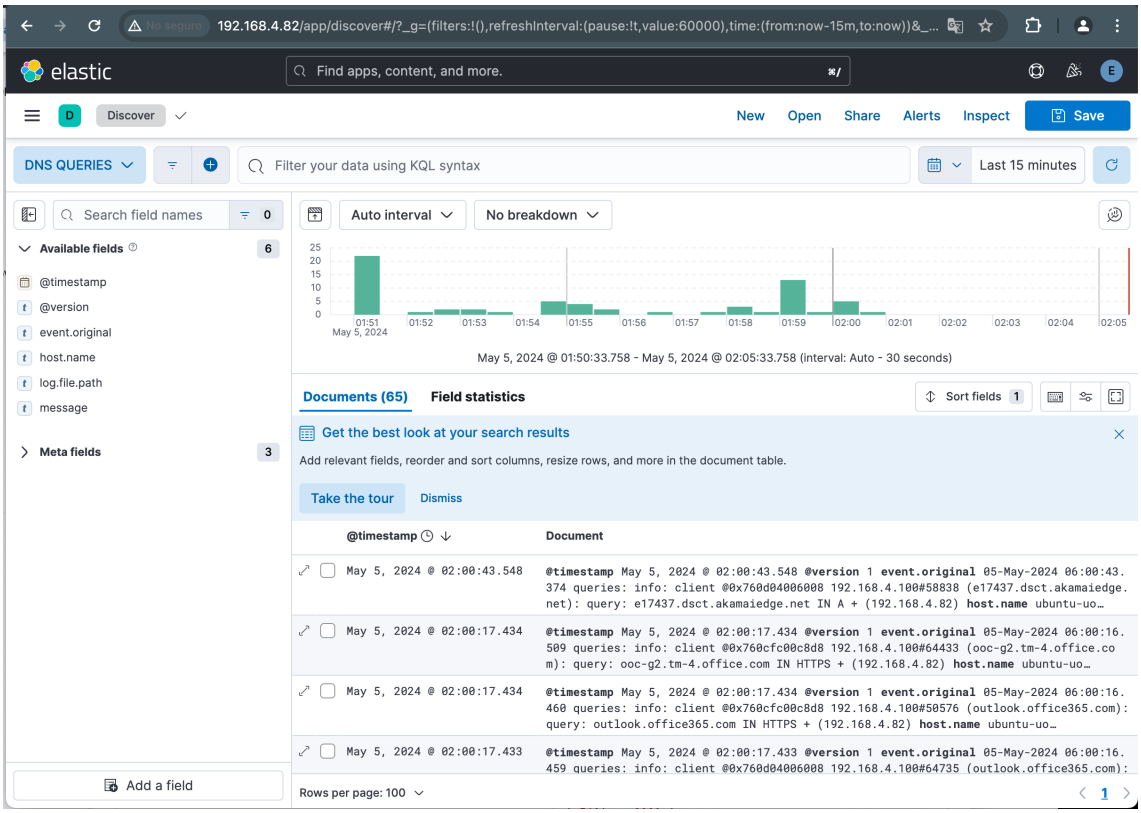


Ilustración 62: visualización de queries dns en elastic

Para graficar y filtrar de forma correcta las queries, es necesario definir correctamente los “patrones” o “fields”. Para esto dentro de la configuración se utiliza “grok”. Para mejorar los filtros es que se procede a crear el directorio “patterns” dentro de “/etc/logstash/”.

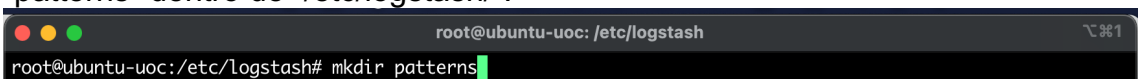


Ilustración 63: creación de directorio de patrones

En el directorio se crea un nuevo archivo llamado “custom_patterns” y se define el patron para analizar correctamente las fechas de fecha, hora y día.

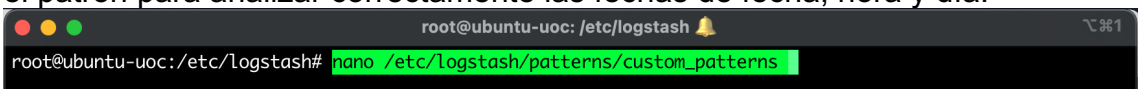


Ilustración 64: configuración de archivo de patrones

El contenido del archivo se agrega el siguiente formato:

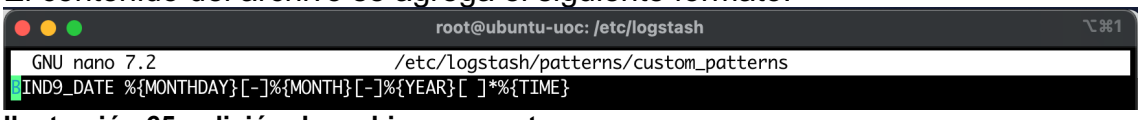


Ilustración 65: edición de archivo para patrones

Luego de la creación de este archivo, se procede a editar el archivo “/etc/logstash/conf.d/basic.conf”

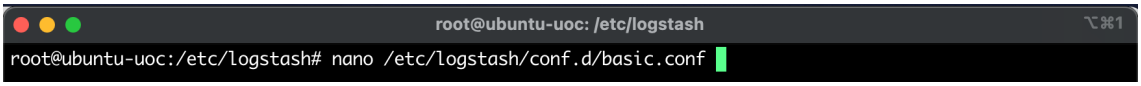


Ilustración 66: edición de archivo basic.conf

En el archivo se modifica el apartado “filter” y se utiliza grok junto con el directorio “patterns_dir” y se ajusta el “match” para que los “fields” trabajen correctamente en Kibana.

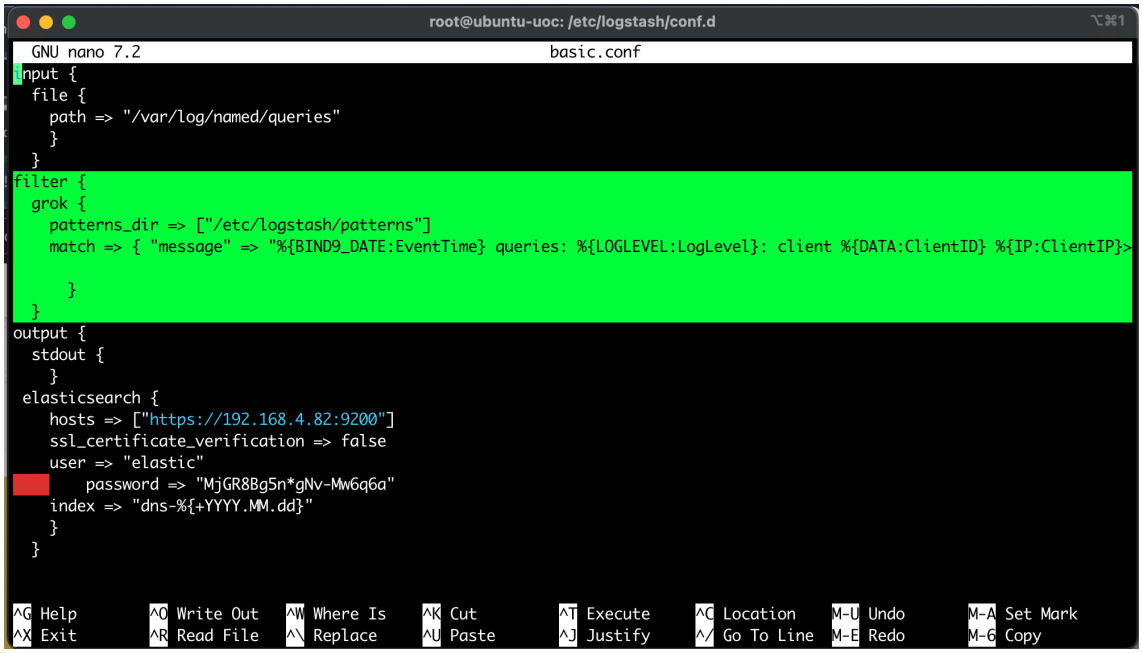


Ilustración 67: habilitación de archivo para la utilización patterns

Con el archivo de configuración finalizado, se procede a reiniciar el servicio de logstash para lograr visualizar las queries en Kibana.

Ilustración 68: reinicio de servicio logstash

Se procede a crear un nuevo “data view” utilizando el nuevo archivo source que ahora el patron es “dns*”.

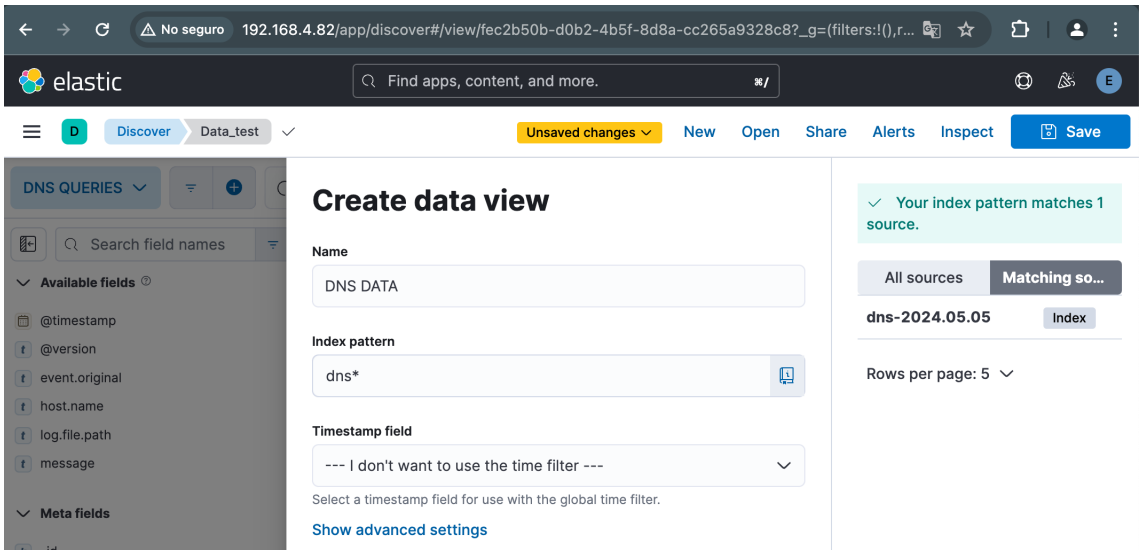


Ilustración 69: configuración de logs de dns en un nuevo data view

Con el nuevo data view podemos filtrar y visualizar y graficar las distintas “queries”. Se procede a guardar el data view dando clic en “Save”.

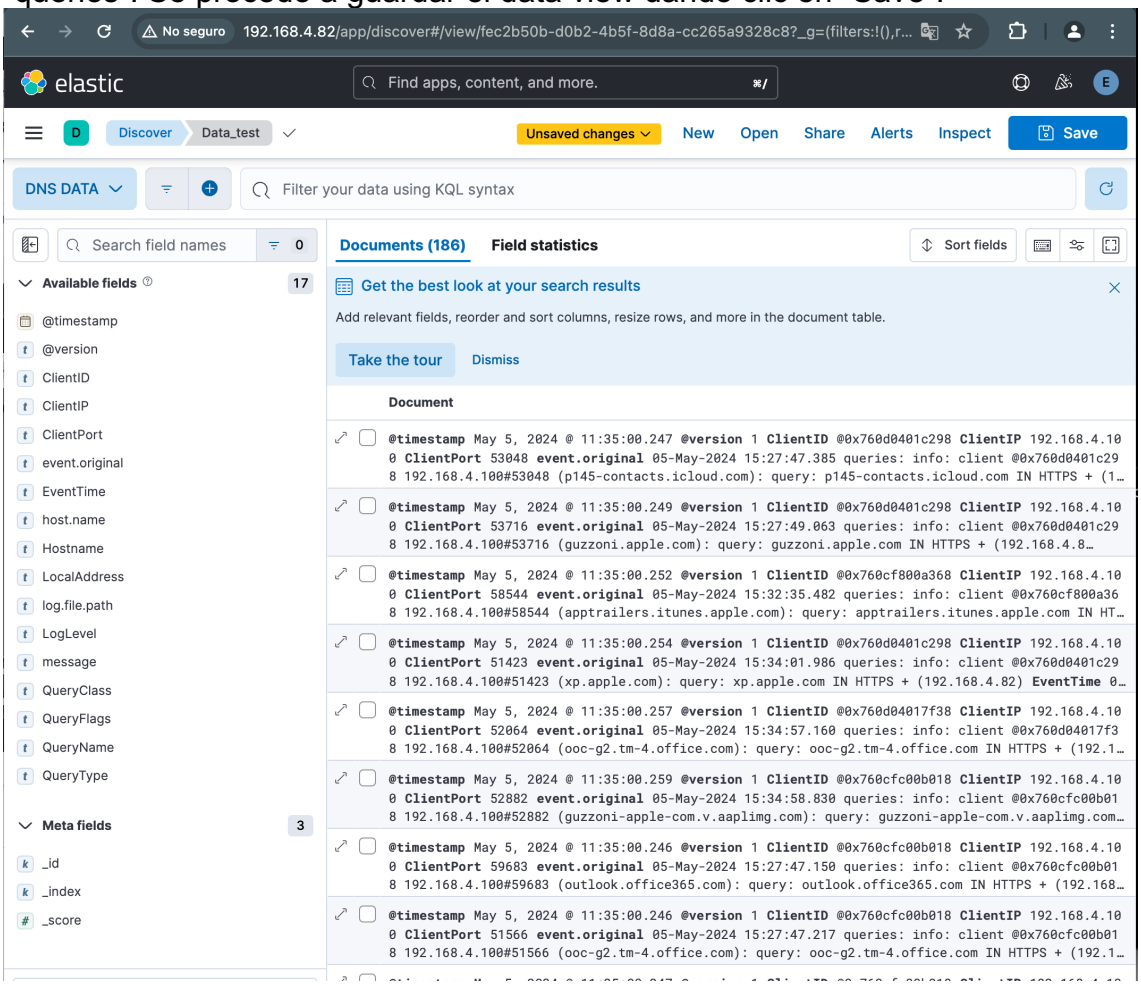


Ilustración 70: visualización de logs utilizando los nuevos patrones

En el menu se procede a dar clic en dashboard para crear un nuevo Dashboard dando clic en “Create a dashboard”.

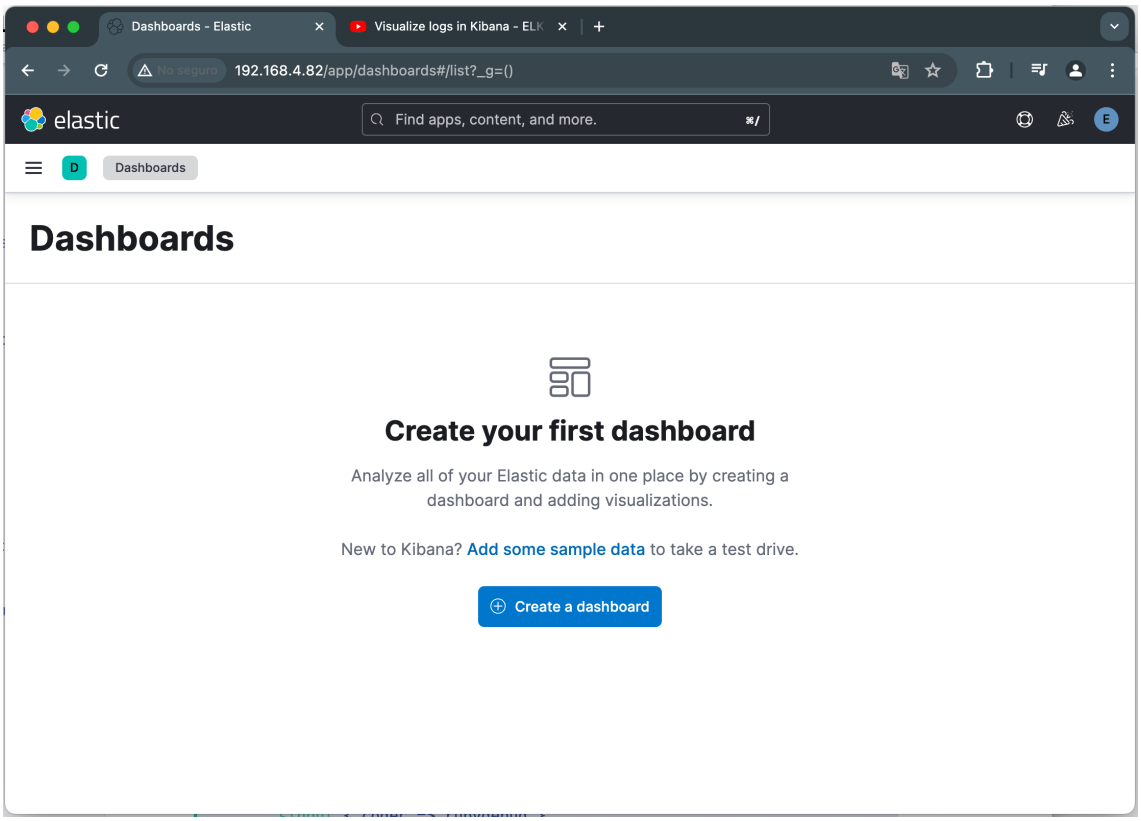


Ilustración 71: configuración de nuevo dashboard

En el nuevo Dashboard se procede a dar clic en “Crate visualization”.

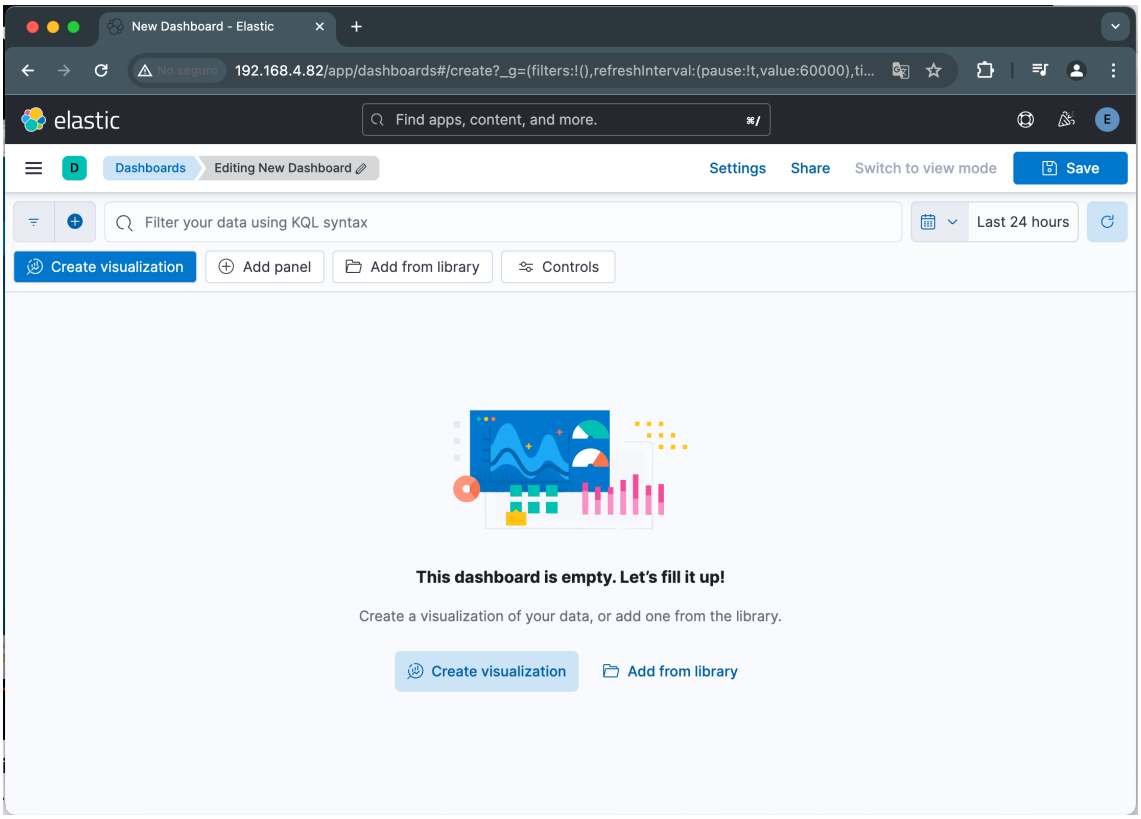


Ilustración 72: configuración de una visualización

En la nueva visualización se tiene la posibilidad de graficar las consultas. Para este caso se procede a utilizar el campo QueryName para identificar el top de consultas almacenadas en el archivo “queries”. Se procede a guardar el dashboard dando clic en “Save and return”.

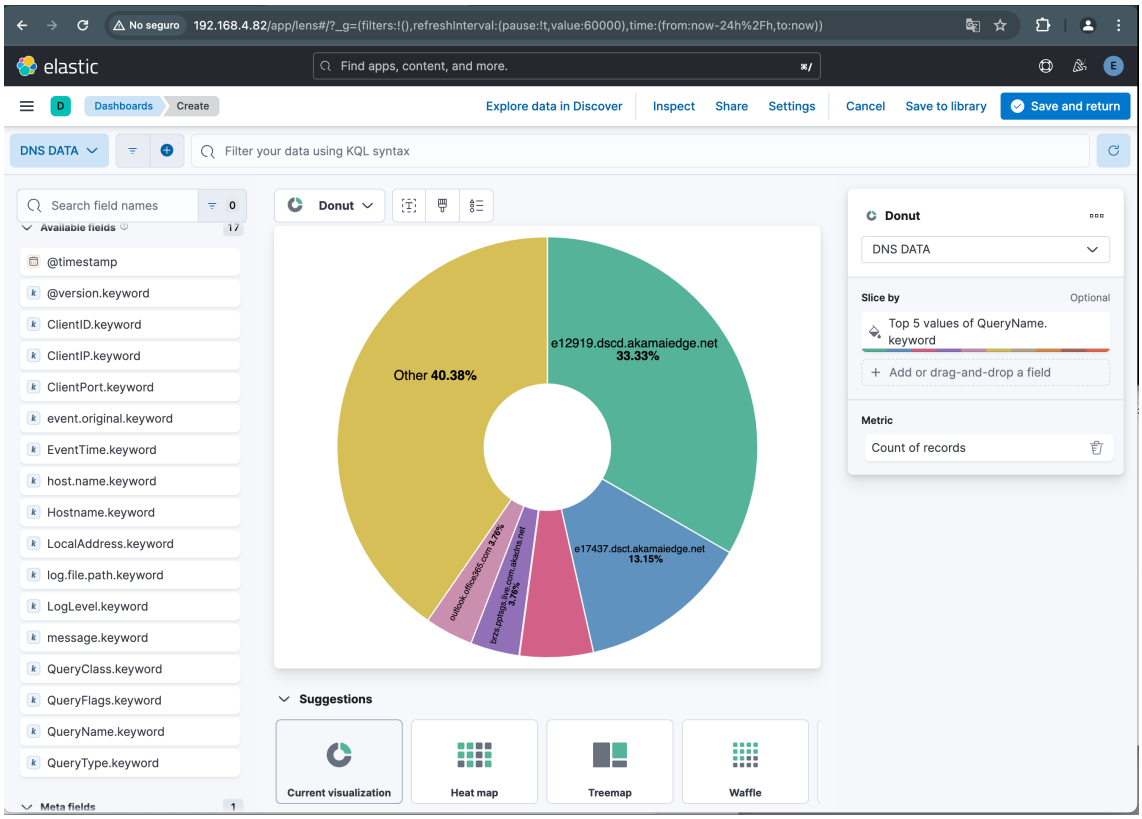


Ilustración 73: nuevo grafico

Se procede a guardar el Dashboard dando clic en “Save”.

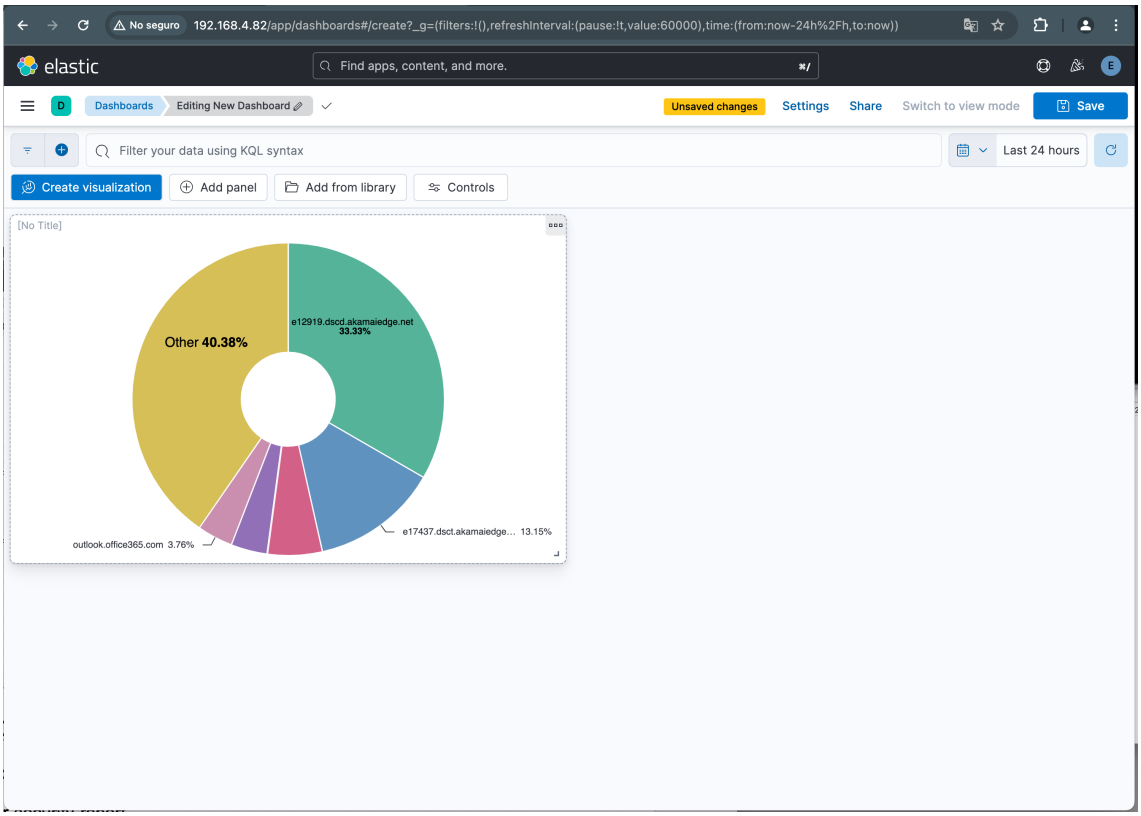


Ilustración 74: grafico de top

3.8. Instalación de ioc2rpz

En esta etapa del proyecto nuestro sistema permite solo visualizar las consultas de DNS, por lo que para lograr proteger nuestro ambiente es que avanzamos con la instalación de ioc2rpz. La utilización de ioc2rpz permitirá habilitar una "RPZ" que integra una lista de indicadores de compromisos de dominios. Cada vez que un usuario intente acceder a un dominio que esta en la lista, dejara un "log" o registro y se procederá a bloquear. Cada registro será presentado en el cuadro de mando.

La instalación de ioc2rpz [11] será desarrollada a través de Docker. Por lo que para la instalación de Docker utilizamos el siguiente comando para instalar los paquetes ca-certificates, curl, gnupg, lsb-release.

```
root@ubuntu-uoc: /home/juan                                    ⌥⌘1
root@ubuntu-uoc:/home/juan# sudo apt-get install \
    ca-certificates \
    curl \
    gnupg \
    lsb-release
Leyendo lista de paquetes... Hecho

Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
ca-certificates ya está en su versión más reciente (20240203).
curl ya está en su versión más reciente (8.5.0-2ubuntu10.1).
gnupg ya está en su versión más reciente (2.4.4-2ubuntu17).
lsb-release ya está en su versión más reciente (12.0-2).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 1 no actualizados.
root@ubuntu-uoc:/home/juan#
```

Ilustración 75: instalación de paquetes ca-certificates

A continuación, se añade al sistema la clave GPG de Docker:

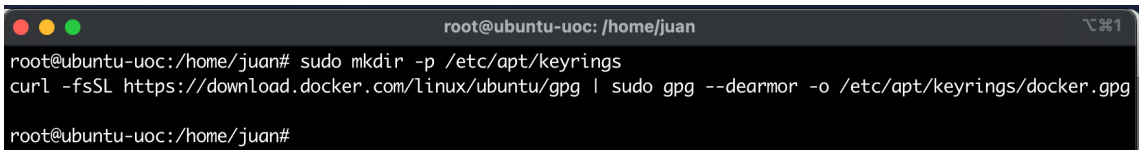


Ilustración 76: instalación de docker

El repositorio de Docker también se configura en la línea de comandos. Para ello, también basta con aplicar el comando de terminal adecuado.

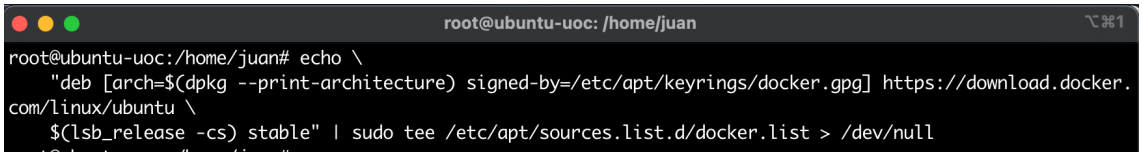


Ilustración 77: lista de versiones de docker

Para descargar una versión en concreto de Docker se puede utilizar el siguiente comando que muestra una lista de todas las versiones disponibles.

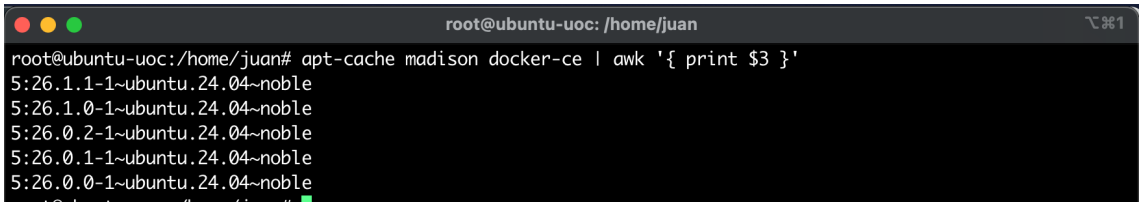


Ilustración 78: lista de versiones disponibles

Para esta instalación se procede a utilizar la versión 5:26.1.1 para el despliegue.

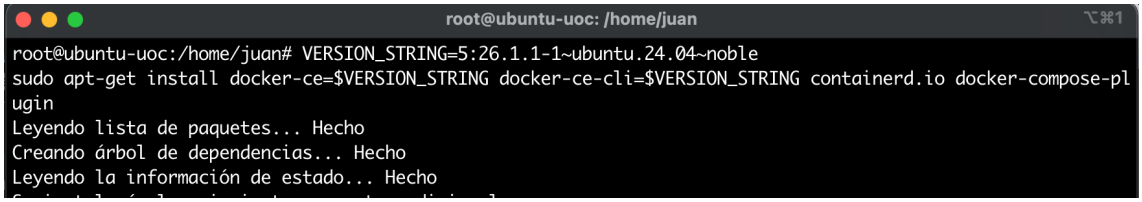


Ilustración 79: instalación de versión de docker

Para validar la instalación y correcta operación de docker, se utiliza el comando “sudo docker run hello-world”.

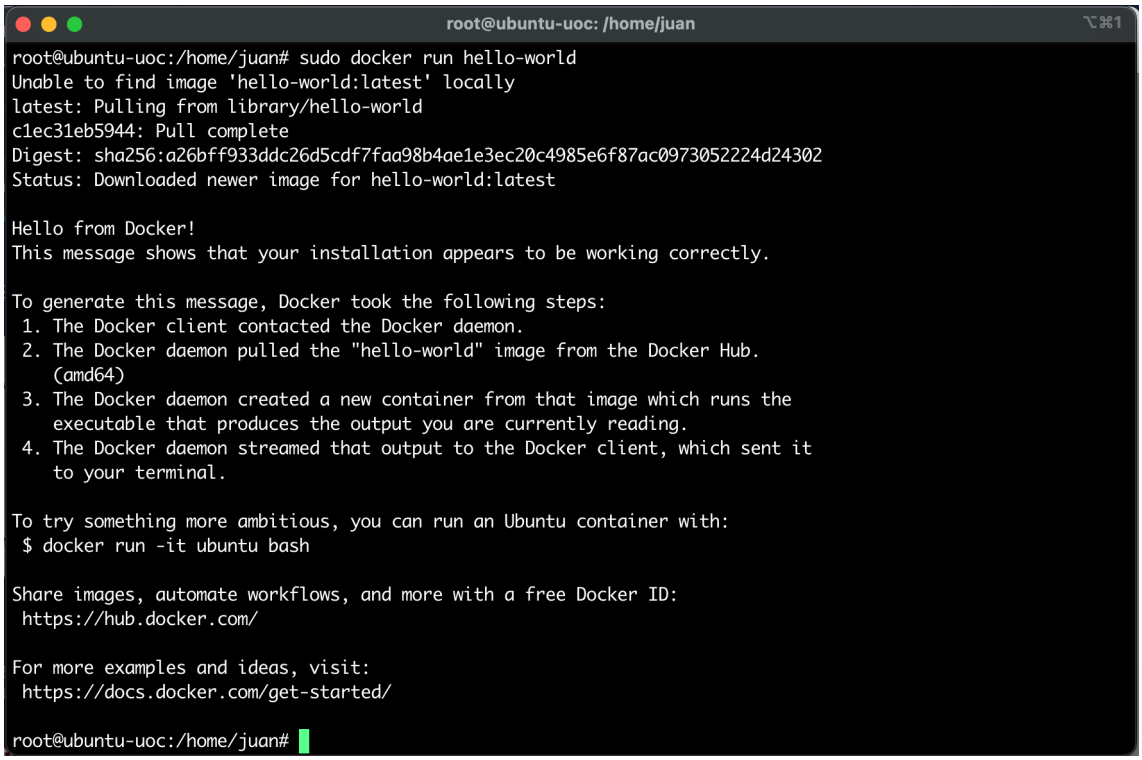


Ilustración 80: prueba de docker

Para el despliegue de las imágenes asociadas a la herramienta de ioc2rpz se procede a realizar un pull de las imágenes de “pvmdel/ioc2rpz” y “pvmdel/ioc2rpz.gui”.

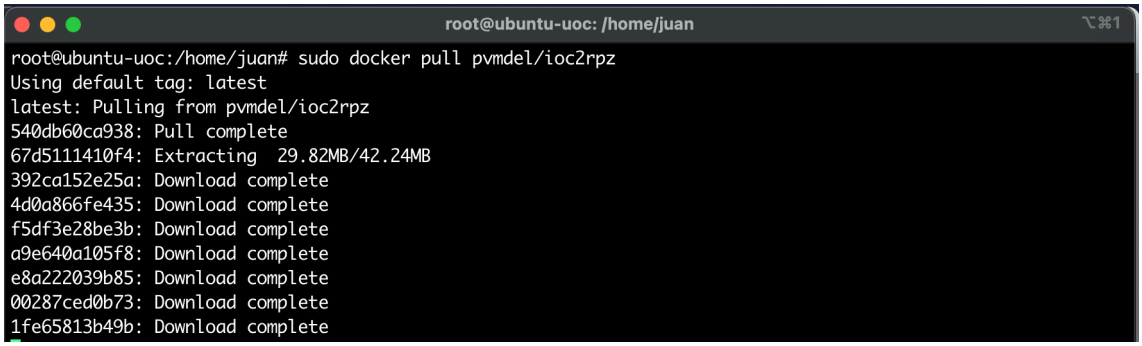


Ilustración 81: descarga de imagen

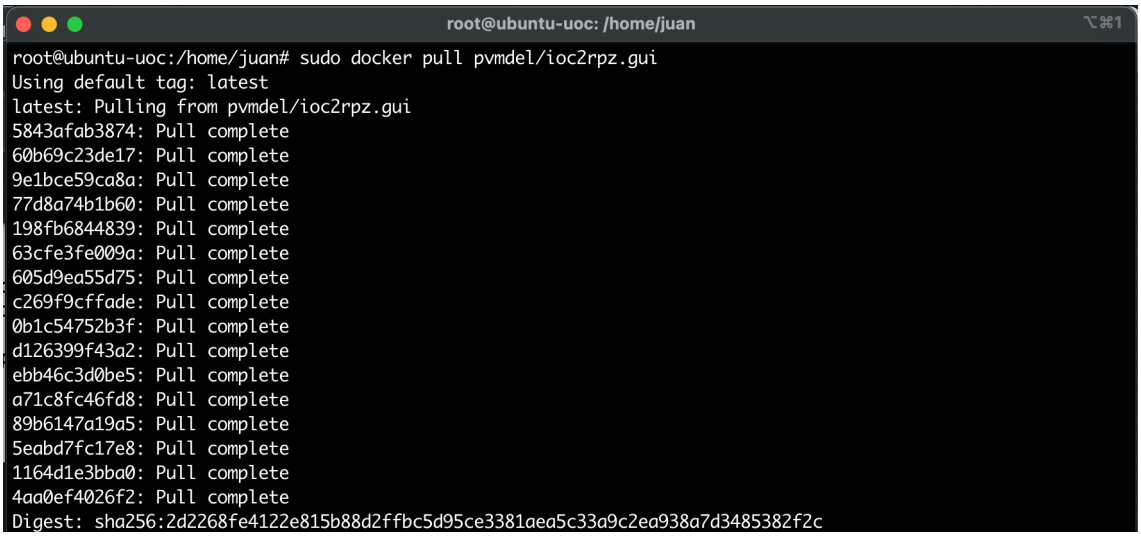


Ilustración 82: instalación de imagen

Se procede a crear los 3 nuevos directorios:

- /opt/ioc2rpz/cfg
- /opt/ioc2rpz/db
- /opt/ioc2rpz/ssl

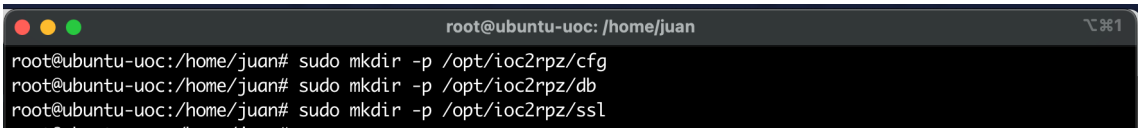


Ilustración 83: creación de directorios para ioc2rpz

Para el despliegue de la imagen, forzamos que utilice la dirección ip 127.0.0.1 en el puerto 53, ya que la dirección ip 192.168.4.82 en el puerto 53 es utilizado por e servicio bind9.

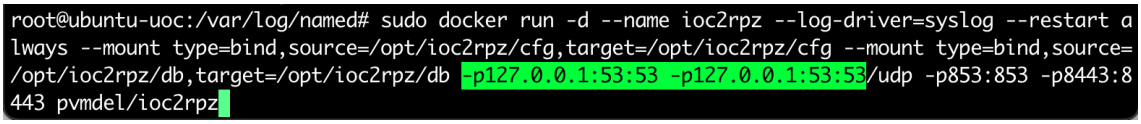


Ilustración 84: configuración de imagen de docker

Para el caso de la administración GUI de ioc2rpz se utiliza el puerto 8080 y el puerto 443.

Ilustración 85: habilitación en puerto 8080

Con la imagen trabajando, iniciaremos el primer acceso a través del sitio appfirewall.cl en el puerto 443/https, desde el cual nos pidirá crear un nuevo nombre de usuario y contraseña.

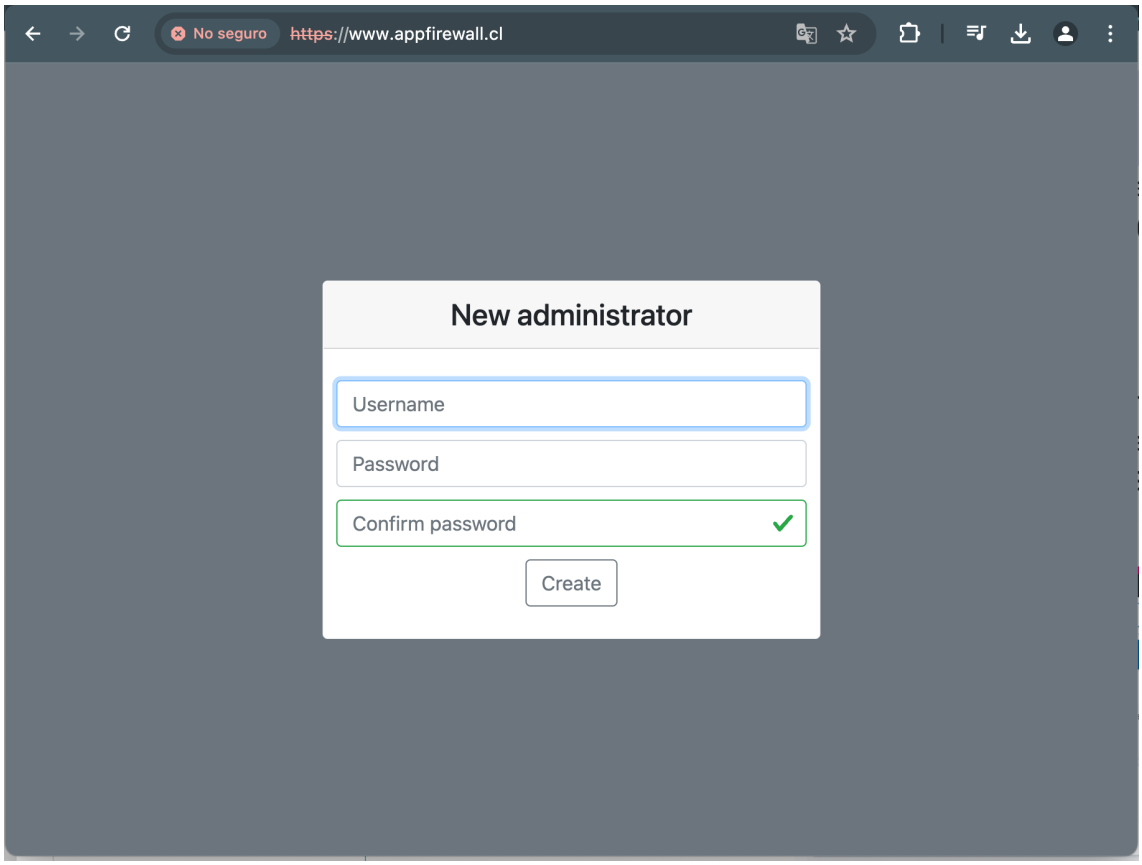


Ilustración 86: primer acceso de ioc2rpz

Con la administración web ya disponible de ioc2rpz, se procede a crear la RPZ. Para esto, al ingresar a ioc2rpz.gui en el apartado “Servers” se procede a modificar el registro ya creado de “server_1”. [11]

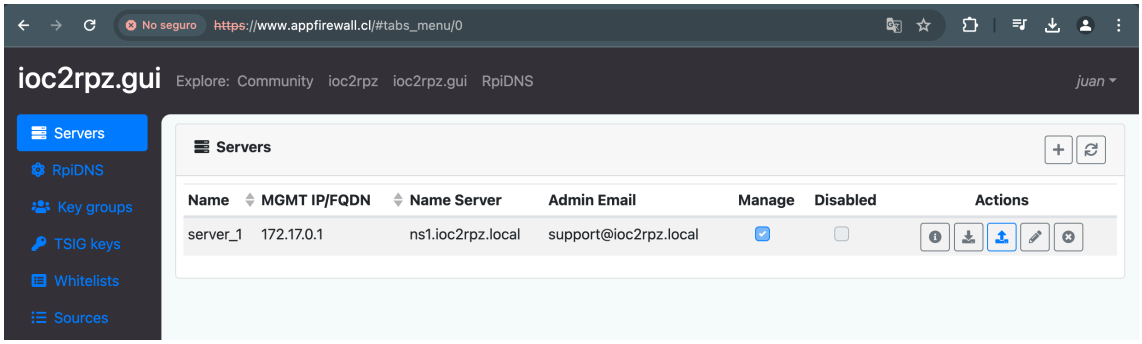


Ilustración 87: configuración de server en ioc2rpz

Para la configuración se utilizan las siguientes direcciones IP del ambiente.

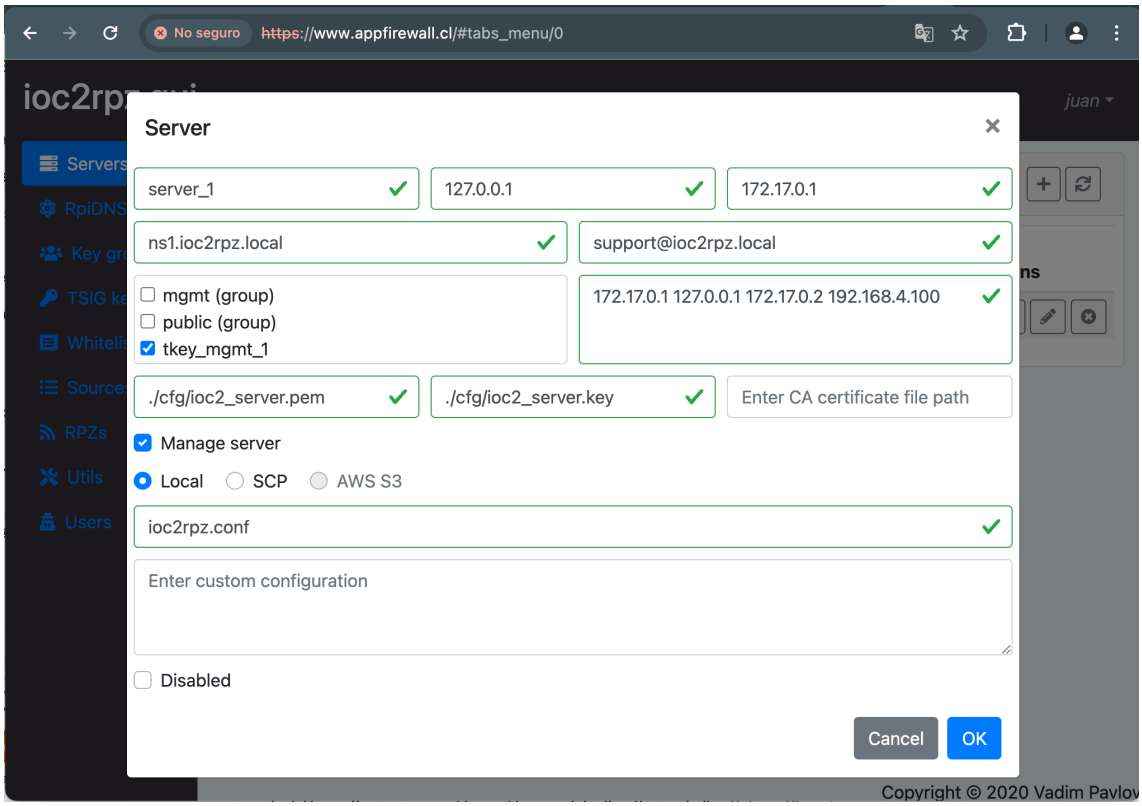


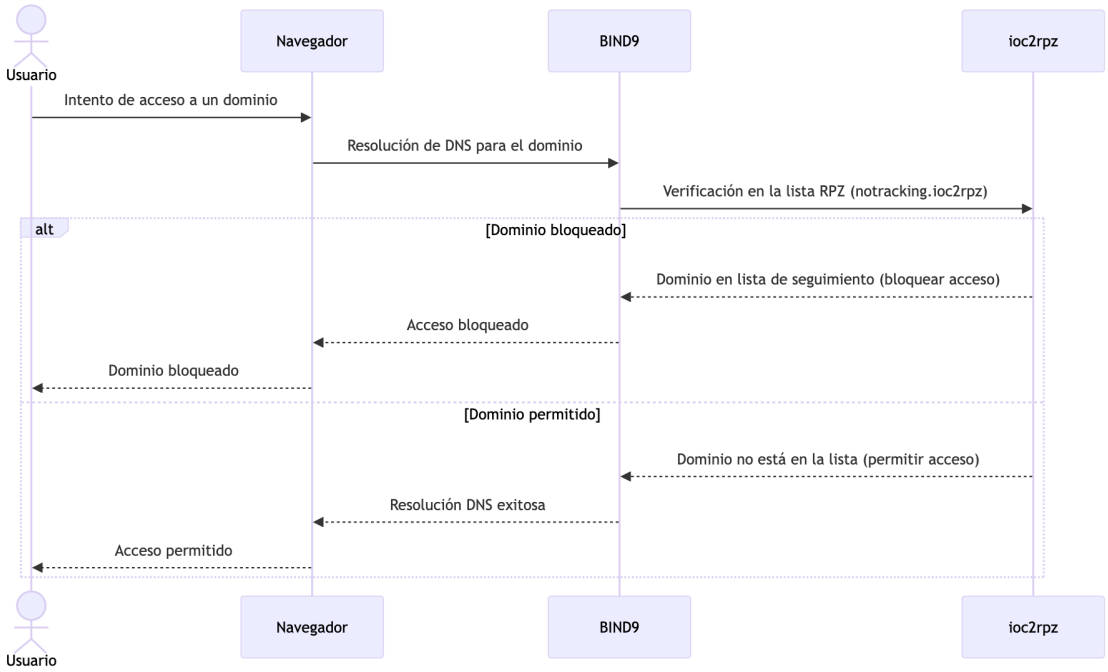
Ilustración 88: configuración de server_1 en ioc2rpz

3.9. Configuración y funcionamiento de la Zona notracking.ioc2rpz

En el sistema ioc2rpz se configurará una lista de indicadores de compromiso llamada "notracking". Esta lista contiene distintos dominios conocidos por recopilar información de los usuarios para rastreo y, en algunos casos, para mal uso de esta información.

En la configuración de ioc2rpz, se crea la zona notracking.ioc2rpz, que funciona como una Zona de Protección de Respuesta (RPZ) para bloquear el acceso a sitios categorizados como de seguimiento (tracking).

Cuando un usuario intenta resolver un dominio a través del navegador, esta consulta se envía al servidor BIND9. El servidor BIND9, al tener configurada la zona notracking.ioc2rpz como RPZ y apuntando al servidor "ioc2rpz", procederá a validar que el dominio no sea malicioso o categorizado como de seguimiento. Si el dominio no está en la lista de bloqueo, el usuario podrá acceder al sitio; de lo contrario, el acceso será bloqueado.



Para lograr la configuración primero se procede a validar la correcta operación de la RPZ utilizando en el menu “RPZ”. En la lista se utilizamos “notracking.ioc2rpz” cliqueandolo y en el registro se procede a dar clic en “info” y en la pestaña “Provision info” se procede a copiar el comando de prueba.

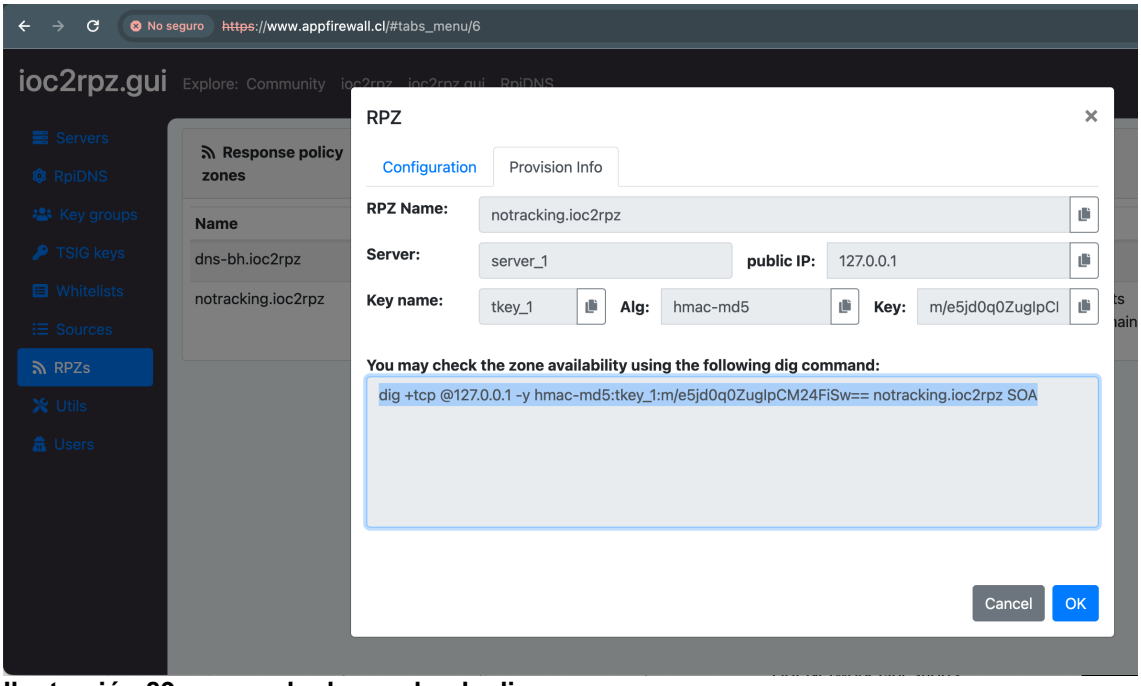


Ilustración 89: comando de prueba de dig

Al ejecutar el comando de prueba validamos que en el status responda como un mensaje de “NOERROR”.

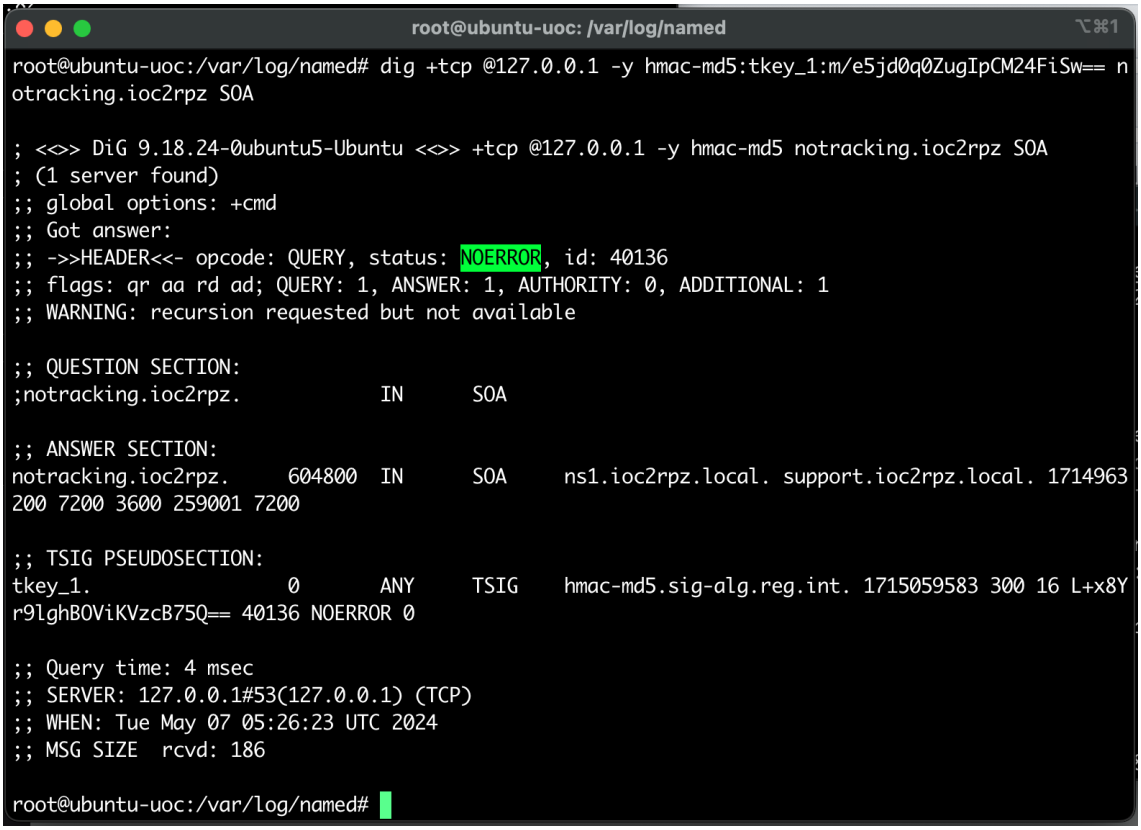


Ilustración 90: ejecución de prueba de dig

Si variamos el comando con la finalización de AXFR podremos listas los distintos dominios que serán bloqueados en la RPZ.

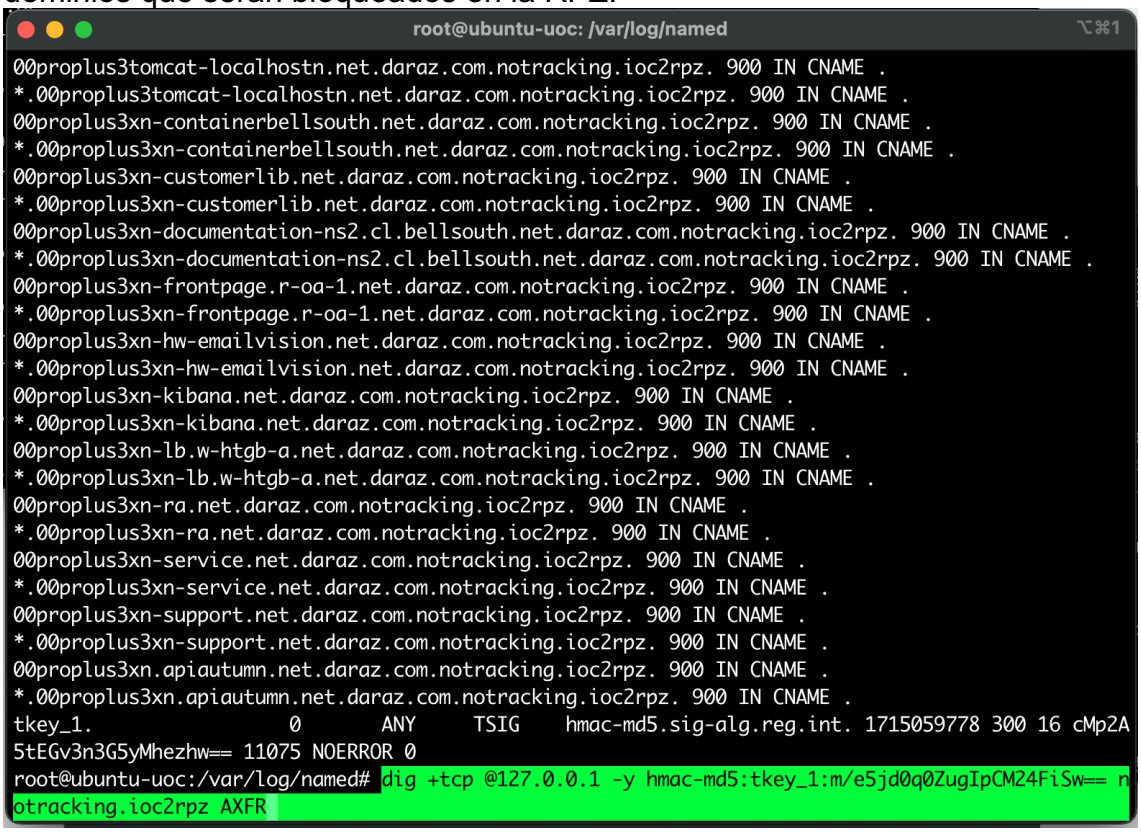


Ilustración 91: dominios bloqueados en la rpz

La lista de dominios o host bloqueados son listados en un archivo .txt que se puede visualizar en el “Source”.

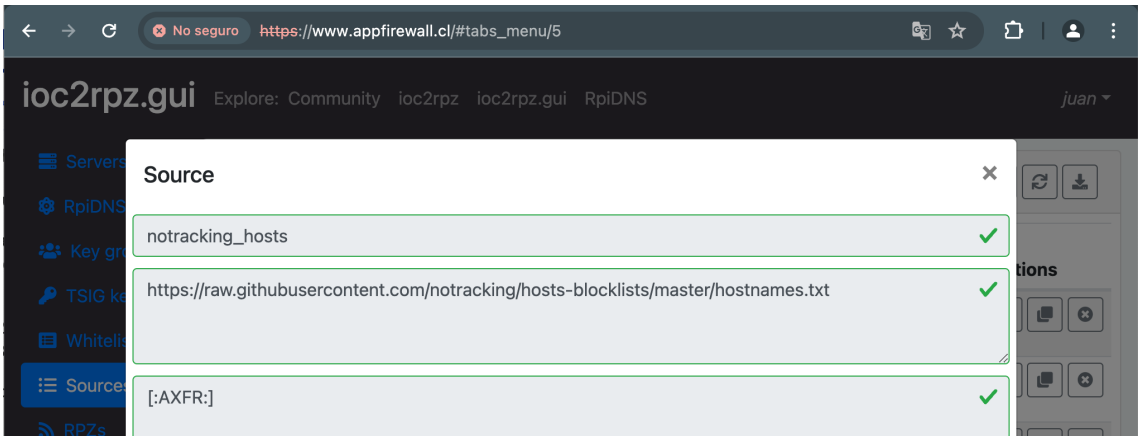


Ilustración 92: lista de dominios bloqueados

Si se accede directo al archivo txt de la url es posible visualizar la lista con los hosts o dominios a bloquear.

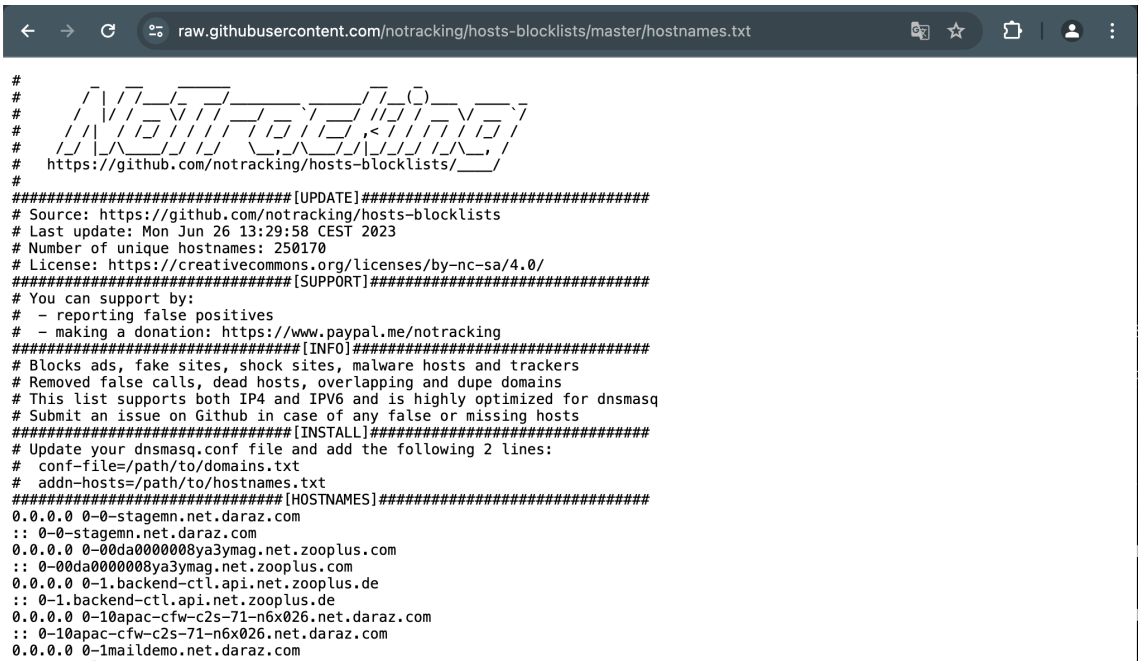


Ilustración 93: detalle de lista de bloqueo

Con la configuración ya operando de ioc2rpz, es necesario configurar el servicio BIND9 para que utilice la RPZ definida en ioc2rpz. Para esto, vamos al apartado Utils y exportamos la configuración de ISC Bind.

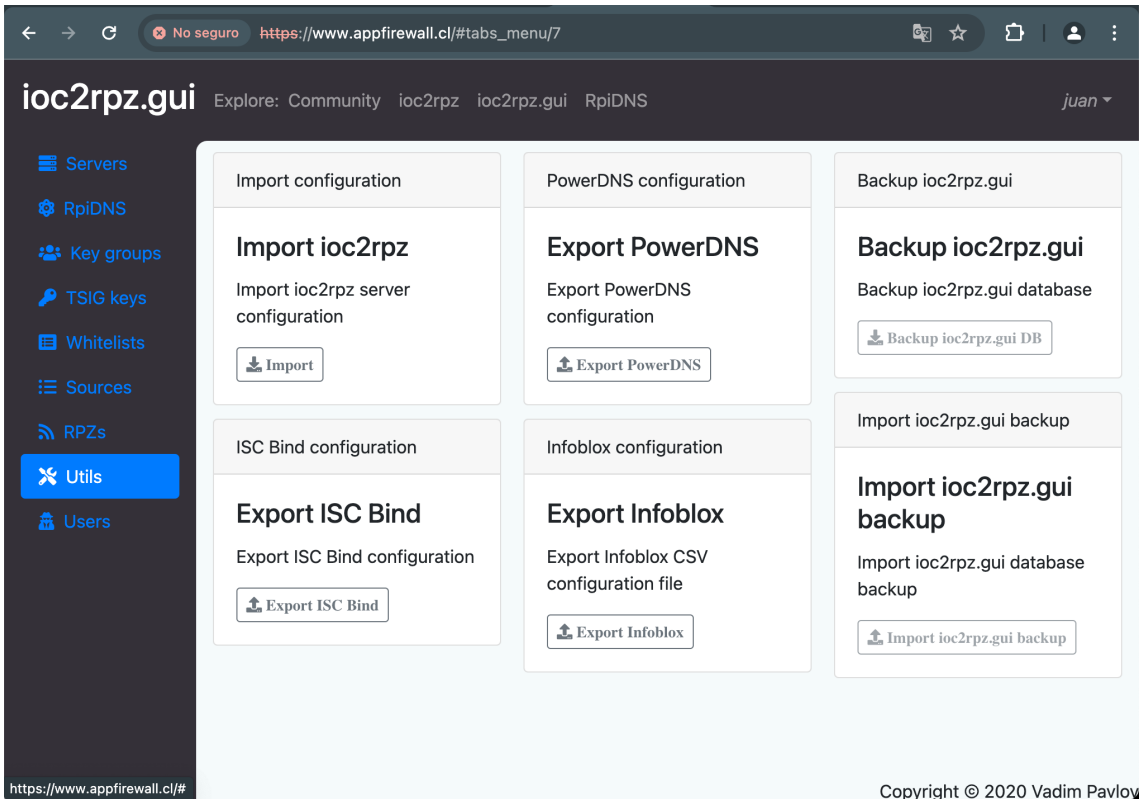


Ilustración 94: exportar ISC Bind

En el servidor es necesario modificar el archivo de configuración de BIND9 con el comando “nano /etc/bind/named.conf.options”.

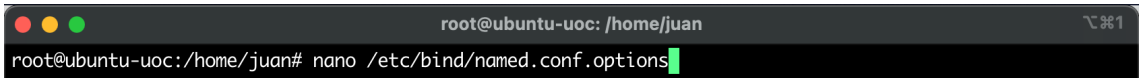


Ilustración 95: edición e archivo named.conf.options

La configuración exportada se carga en el archivo agregando los parametros como se observa en la configuración.

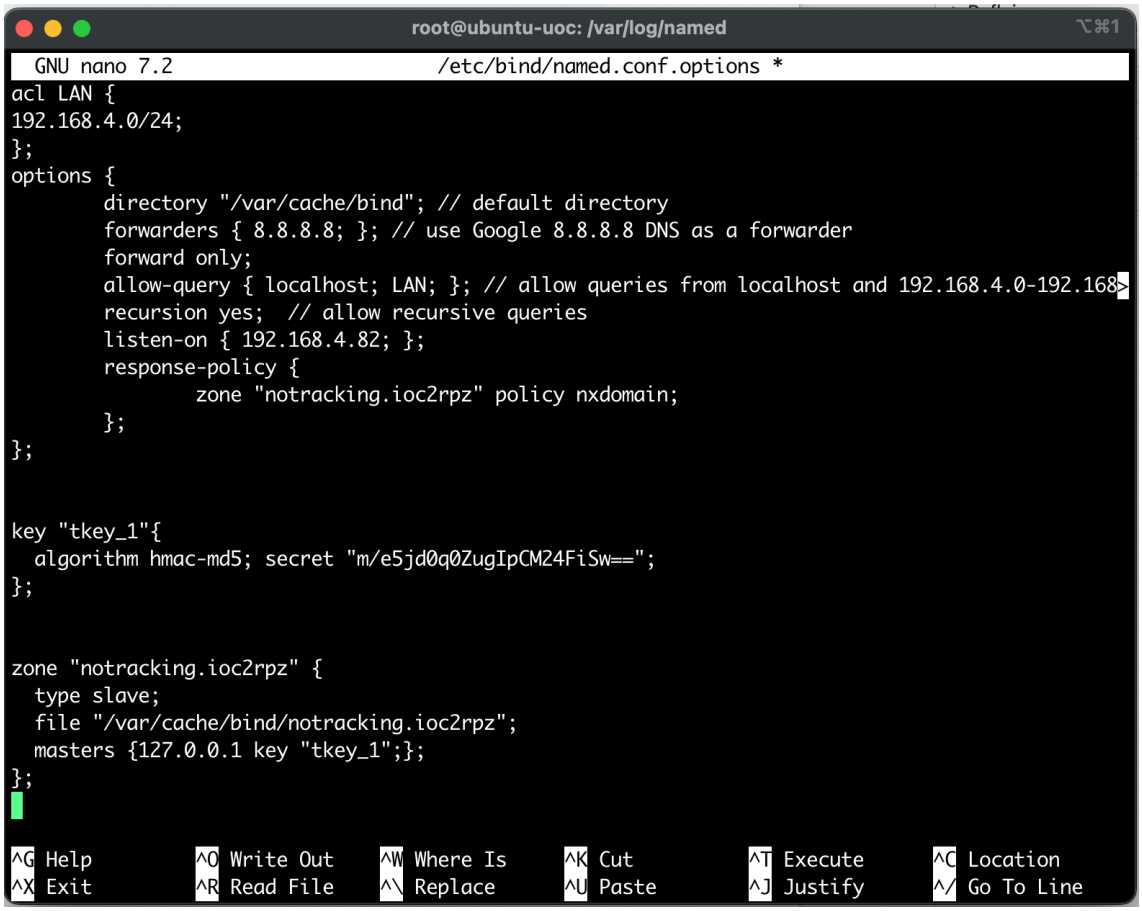


Ilustración 96: configuración de archivo named.conf.options

Dado que los logs que se están registrando son solo de “queries”, es que se requiere modificar el archivo “named.conf.local” y se ajusta la configuración de logs para registrar la nueva zona RPZ.

Editamos el archivo a través del comando “nano /etc/bind/named.conf.local”.

```
root@ubuntu-uoc:/var/log/named# nano /etc/bind/named.conf.local
```

En el archivo de configuración [4], se almacenan los logs en el archivo “/var/log/named/rpz”. La configuración que se aplica es la siguiente.

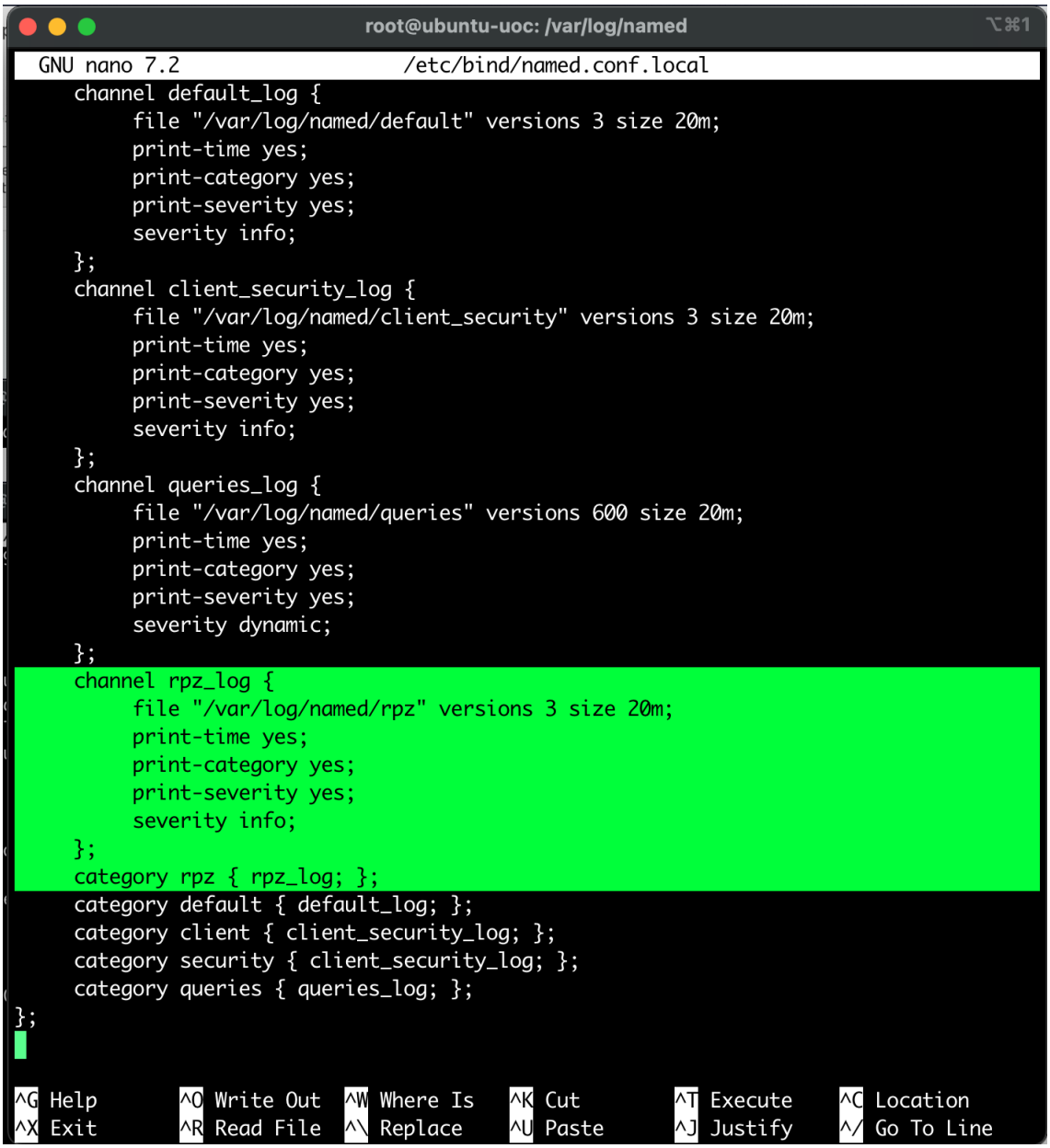


Ilustración 97: configuración de logs en archivo named

Se procede a reiniciar el servicio de bind9 utilizando el comando “systemctl restart bind9”. Posteriormente se realiza la prueba de resolución de unos de los host listados en la lista “no-traking” de la RPZ. Como respuesta se obtiene un mensaje “server can’t fin a0b33.com: NXDOMAIN” validando la correcta operación.

Ilustración 98: nslookup como prueba de resolución

Al ejecutar la evaluación directamente desde el navegador se obtiene el mismo mensaje de bloqueo del sitio por resolución.

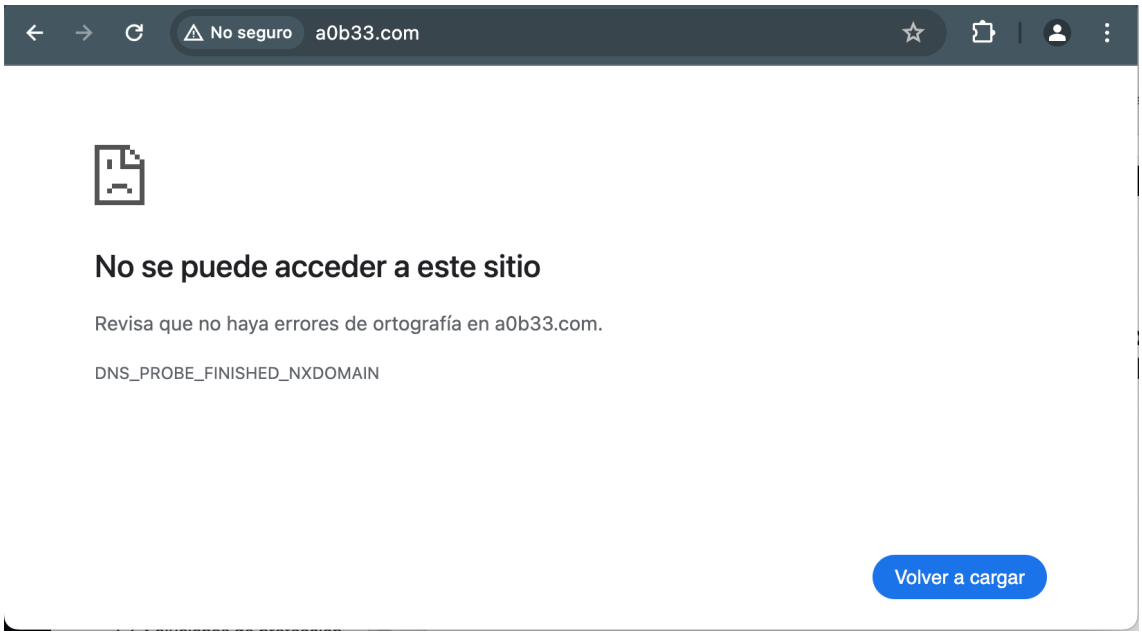


Ilustración 99: validación de bloqueo

Se realiza la revisión del archivo de logs de RPZ en "/var/log/named/rpz" para validar que se guarde el registro, con el objetivo de utilizarlo en Kibana.

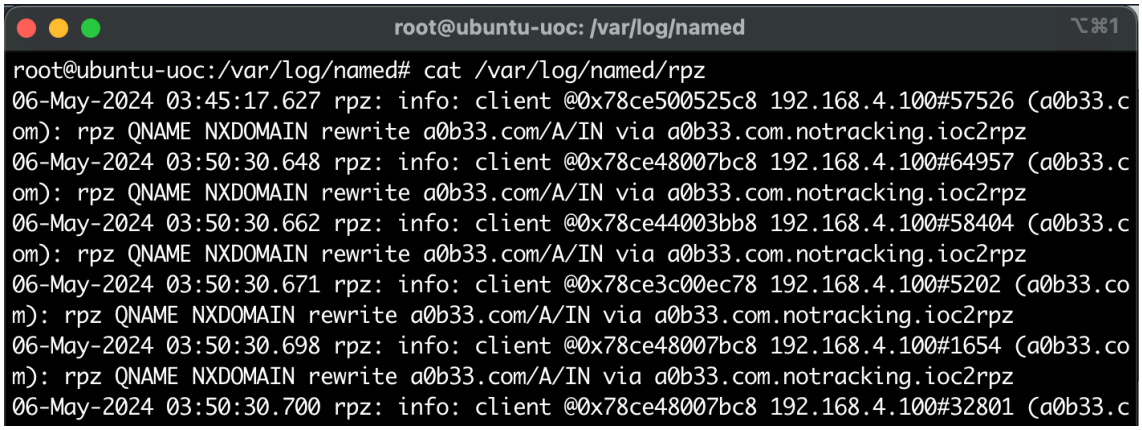


Ilustración 100: registro de dominios bloqueados

Con los registros ya validados se crea un nuevo archivo de configuración de pipeline en "/etc/logstash/conf.d/rpz.conf". Se utiliza el mismo contenido del archivo basic.conf y se realiza una modificación en el "path" apuntando al nuevo archivo de logs "/var/log/named/rpz".

En el apartado "match", se crea el siguiente filtro con estructura para el log de RPZ:

- %{BIND9_DATE:EventTime} rpz: %{LOGLEVEL:LogLevel}: client %{DATA:ClientID} %{IP:ClientIP}#%{NUMBER:ClientPort} \(%{HOSTNAME:QueryName}\): rpz QNAME %{DATA:Action} %{DATA:RuleAction} %{DATA:Rule} via %{DATA:Zone}

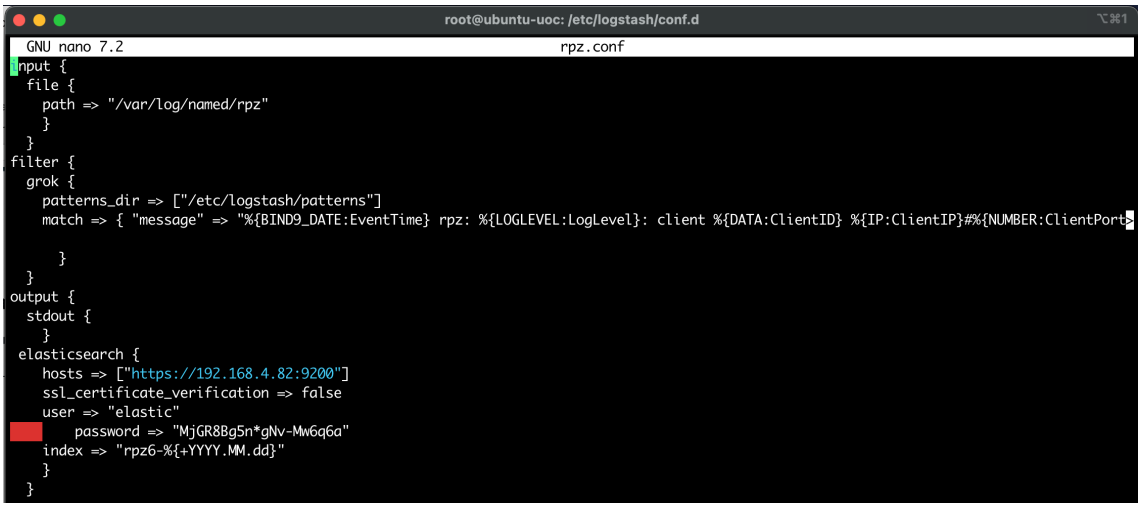


Ilustración 101: configuración de archivo rpz.conf

Se procede a reiniciar el servicio de logstash a través del comando “systemctl restart logstash”.

Con esto desde elastic creamos un nuevo “data view” utilizando como source el nuevo contenido de “rpz”. Damos clic en “Save data view to Kibana”.

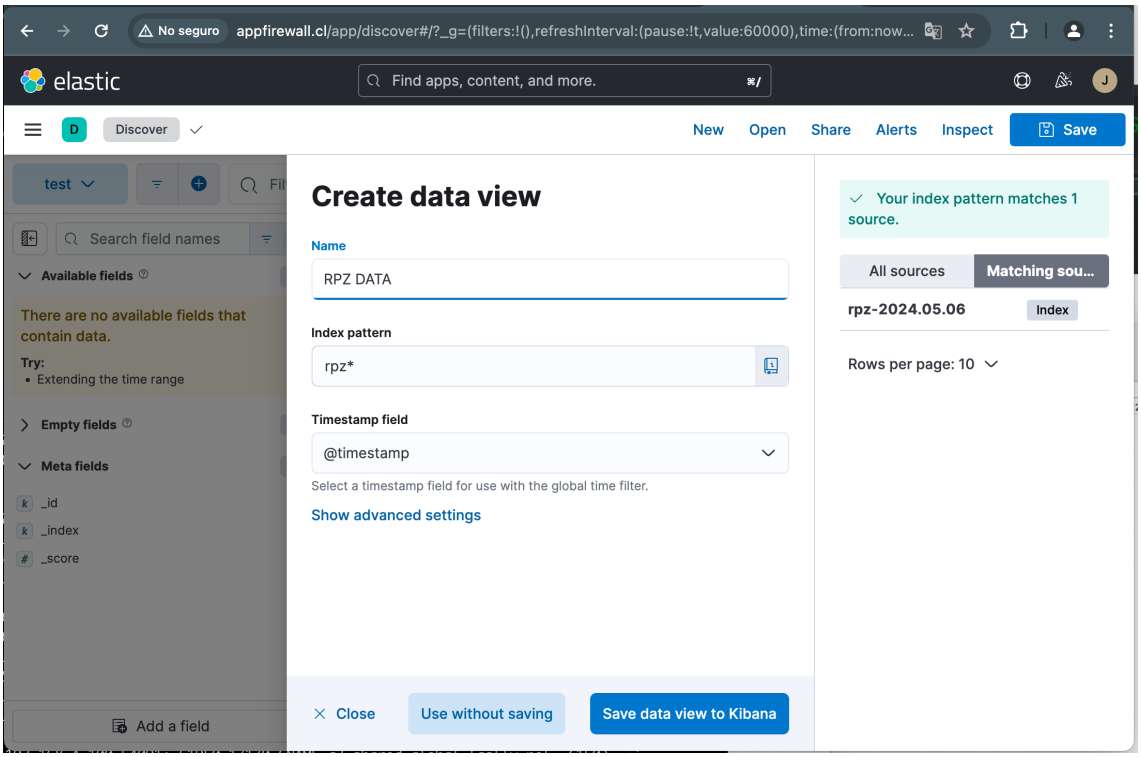


Ilustración 102: data view de archivo rpz

En el “data view” se utiliza el campo “Hostname.keyword” para filtrar los dominios o host que se han respondido a través de la RPZ.

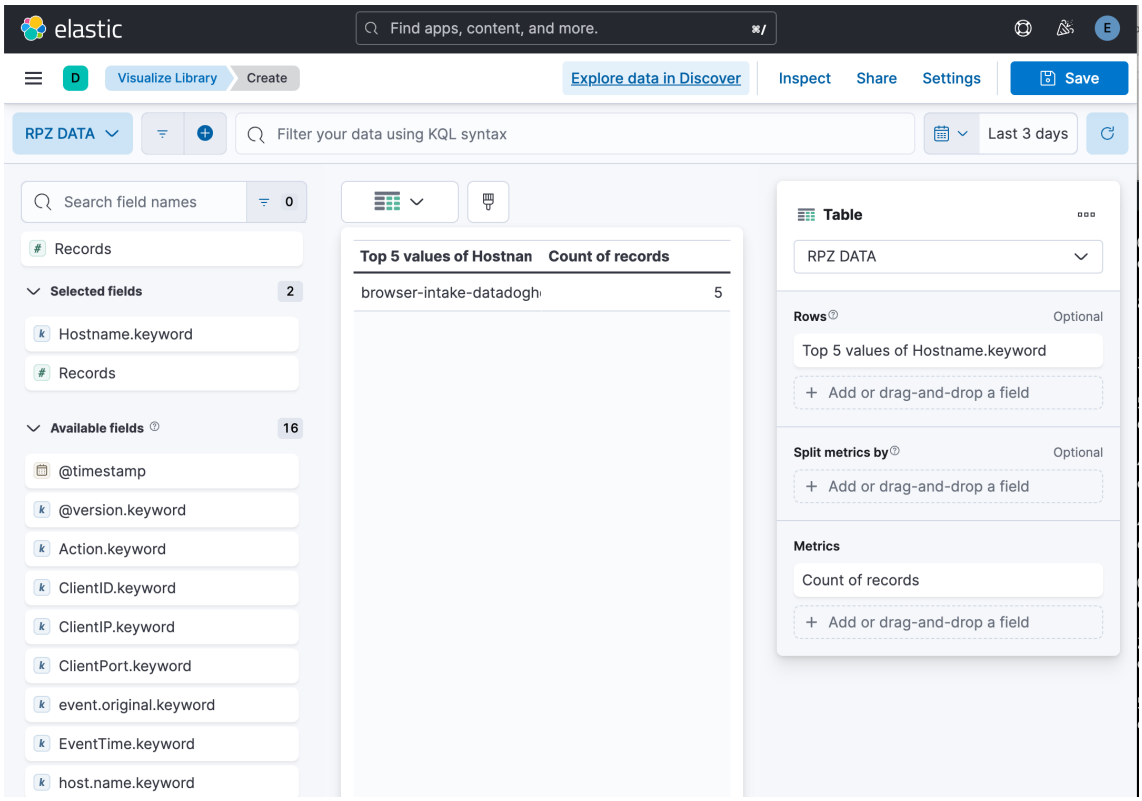


Ilustración 103: configuración de data view rpz

Se guarda como dashboard y se ejecutan pruebas de conectividad para generar registros desde uno de los clientes.

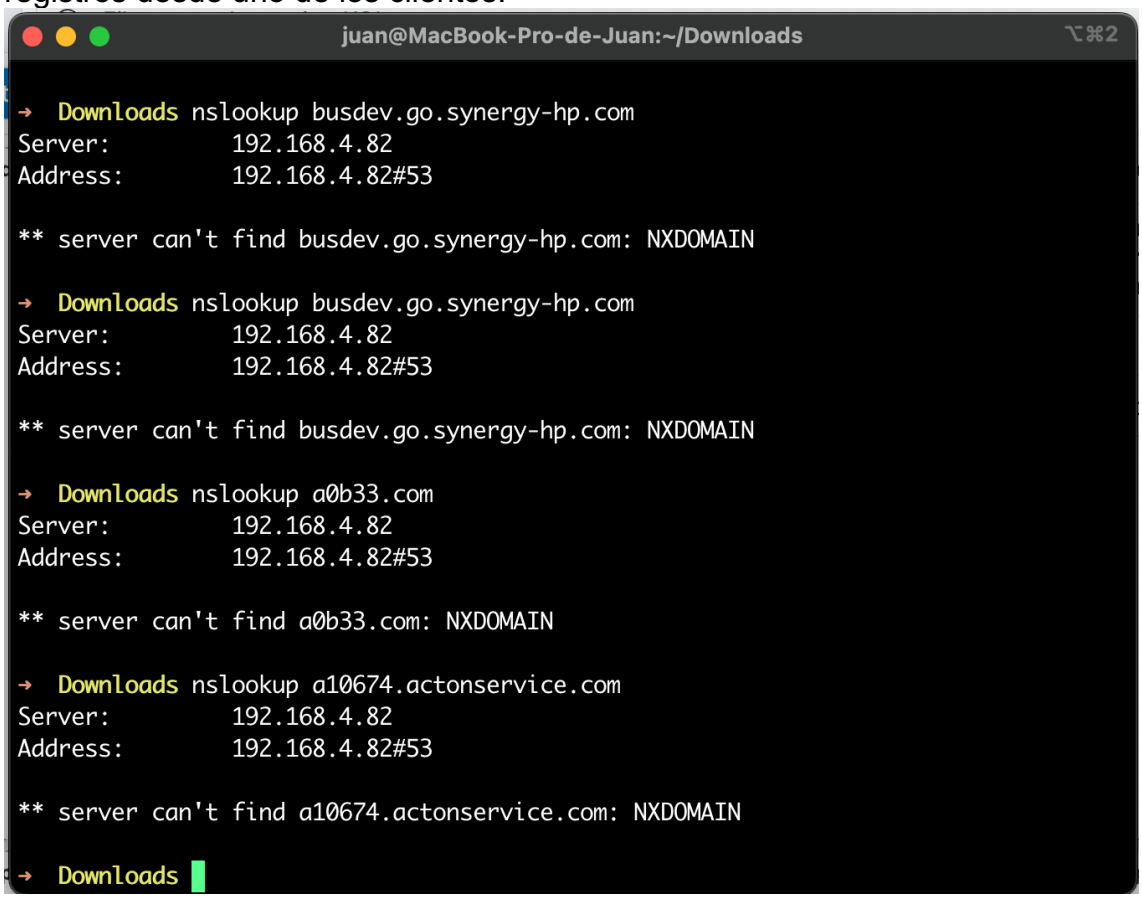


Ilustración 104: prueba de acceso a sitio bloqueado por rpz

Al finalizar, estos logs se empiezan a registrar en el dashboard. Para visualizar la información, se crean los siguientes Dashboard.

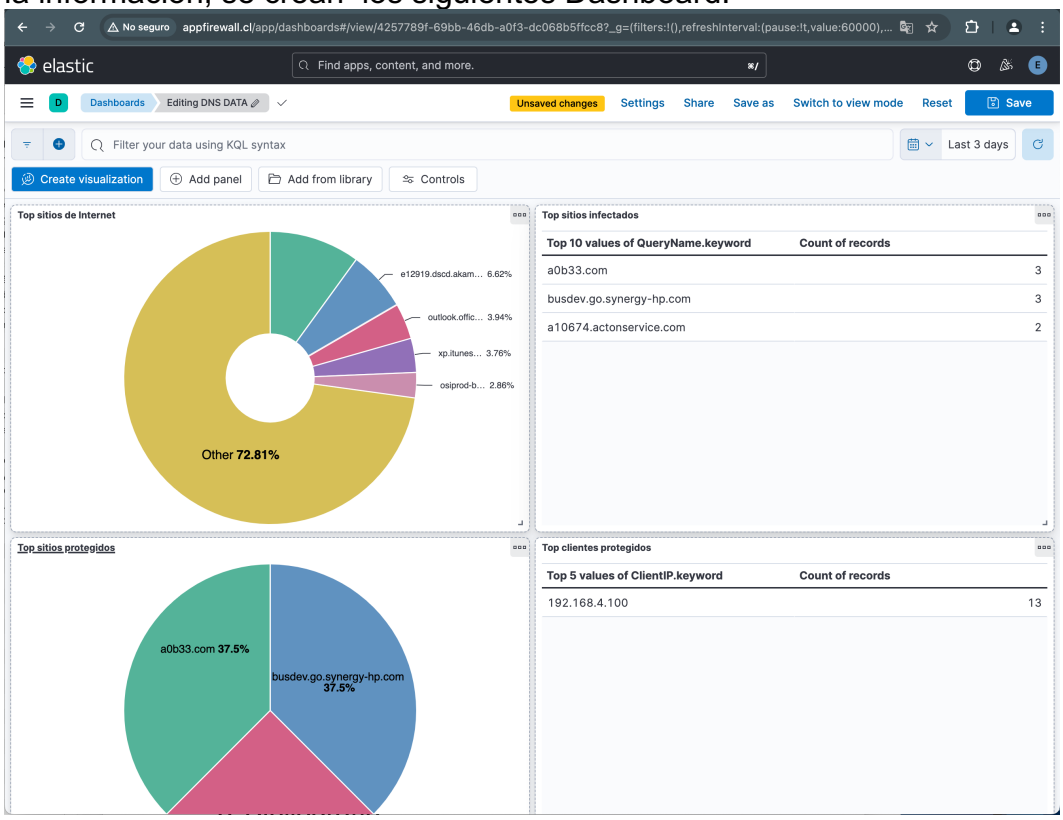


Ilustración 105: dashboard de visualización de bloqueos

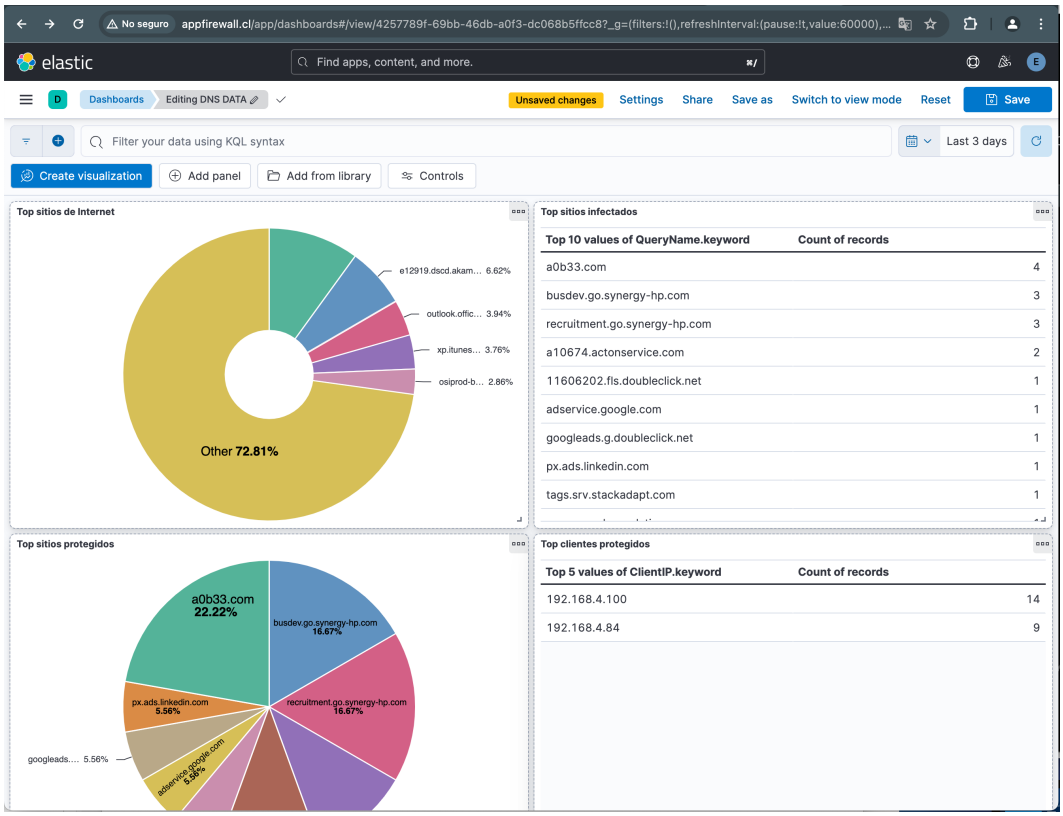


Ilustración 106: dashboard de visualización de bloqueos

4. Conclusiones

A continuación, se describen las conclusiones del trabajo, así como algunas lecciones aprendidas:

- En la puesta en marcha de la herramienta se ha logrado validar como utilizar un Firewall DNS para ejecutar las funciones de prevención de amenazas durante la navegación web en pequeña y medianas empresas.
- Se ha podido comprobar que es posible detectar dispositivos infectados dentro de la red utilizando el registro (log) de comunicación de un dispositivo hacia el sitio categorizado como malicioso.
- Por otra parte, se ha logrado demostrar la factibilidad técnica de utilizar herramientas de código abierto para crear una protección funcional para PYMES e integrar fuentes de terceros que proveen información de sitios categorizados como maliciosos.
- Se ha podido comprobar la factibilidad técnica de utilizar BIND9 en un contexto empresarial como Firewall DNS.
- Por otra parte, se ha logrado demostrar la operación de protección con la implantación de solo herramientas Opensource, tales como Ubuntu, ELK Stack, BIND9 y ioc2rpz
- También se ha logrado demostrar que es posible configurar dashboard para visualizar el comportamiento de seguridad de la red a través del protocolo DNS.

4.1. Seguimiento de la planificación establecida

En base a la planificación generada en el inicio del proyecto es posible concluir que se ha cumplido exitosamente con los compromisos de avance establecidos en cada una de las etapas basados en la carta Gantt.

Gran parte del éxito del proyecto y el cumplimiento del objetivo se logra gracias a la correcta elección de las herramientas opensource, las cuales poseen una amplia documentación y una facilidad de despliegue para el ambiente base seleccionado. También, la correcta planificación de los tiempos asociados a las tareas esenciales ha permitido lograr este éxito.

4.2. Evaluación de objetivos alcanzados

A continuación, se realiza una valoración del cumplimiento de los objetivos establecidos al inicio del proyecto.

Objetivos de investigación:

- **Investigación sobre cómo utilizar un Firewall DNS para funciones de prevención de amenazas durante la navegación web.**
  Se ha investigado y estudiado como utilizar un Firewalls DNS y se ha llegado a la conclusión que es factible funcionalmente desarrollar el despliegue para proteger la navegación web en un ambiente empresarial. Por otra parte, se han encontrado herramientas de pago que son incluidas en productos de seguridad que ya incluyen la protección DNS. Por otra

parte, sigue siendo conveniente para una PYME considerar una solución opensource para aumentar su protección con el mejor costo.

- **Investigación sobre como detectar dispositivos infectados dentro la red.**
  Se desarrolla la investigación, donde se valida que es posible detectar los dispositivos infectados, basado en la cantidad de veces (contadores) que un equipo intenta acceder a dominios infectados, pudiendo inferir que los equipos que tienen una cantidad de contadores “alto” deberían estar infectados y se pueden satisfactoriamente mostrar en el dashboard.

- **Investigación sobre las posibilidades que ofrecen las herramientas de código abierto para pymes y su integración con terceros.**
  Basado en el despliegue satisfactorio y funcional del laboratorio, se afirma la posibilidad de incluir un despliegue de un servidor DNS con todas las capacidad de protección, utilizando solo herramientas del tipo opensource.

- **Estudio del funcionamiento y configuración de BIND 9 en el contexto de un Firewall DNS.**
  Se logra estudiar y comprender el funcionamiento de BIND9 basado en el contexto de una PYME, la cual podría poseer una ZONA y estaría requiriendo adicionalmente proteger la navegación de los usuarios a través de consultas recursivas.

Objetivos de implantación:

- **Instalación y configuración de un Firewall DNS basado en BIND 9 montado en un Ubuntu Server.**
  Este punto se ha logrado ejecutar de forma satisfactoriamente en un ambiente con Ubuntu 24.04 y BIND9.

- **Aprender a configurar reglas y políticas de seguridad en el Firewall DNS para prevenir el acceso a sitios maliciosos y filtrar contenido inapropiado.**
  Se ha logrado configurar la funcionalidad de prevenir el acceso a sitios maliciosos y filtrar contenido inapropiado, sin embargo no directamente basado en “reglas”, sino que basado en listas “sources” seleccionadas por el administrador.

- **Integrar un SIEM ELK con el Firewall DNS para mejorar la detección de amenazas y realizar pruebas de detección.**
  Se ha logrado integrar el Stack ELK a un ambiente que trabaja como Firewall DNS, pudiendo visualizar los “equipos infectados” y el top de “sitios maliciosos”.

- **Configurar un sistema de monitorización para el Firewall DNS y facilitando su gestión y visualización.**
  Se ha logrado construir una herramienta que permita visualizar los eventos de tráfico y de bloqueos asociados al Firewall DNS utilizando los eventos de la zona RPZ.

4.3. Trabajo futuro

Una vez conseguidos los objetivos didácticos establecidos en este trabajo, se enumeran a continuación varias líneas de trabajo futuro para mejorar el proyecto:

- Se ha quedado pendiente integrar más fuentes de origen de indicadores de compromisos que permitan detectar una mayor cantidad de sitios maliciosos.
- Existe la posibilidad de generar casos de uso basados en correlación de eventos de seguridad de los dispositivos que se conectan a los sitios infectados y el tráfico de red de los firewalls disponibles de PYME donde se implementa la solución.
- Existe la posibilidad de crear una herramienta que gestione todas las infraestructuras a través de una GUI, permitiendo:
    - Controlar el acceso a través de un login web.
    - Permite visualizar rápidamente los dashboard obtenidos desde KIBANA.
    - Crear de forma manual y grafica listas de IOC cargadas en el mismo servidor.
    - Permite bloquear y aislar un equipo (host) infectado dentro de la red.

5. Glosario

ARPANET: La ARPANET fue la precursora de Internet, una red de computadoras desarrollada por la Agencia de Proyectos de Investigación Avanzada del Departamento de Defensa de los Estados Unidos (ARPA) a finales de la década de 1960 y principios de la década de 1970. Fue la primera red en implementar el protocolo de comunicación TCP/IP, sobre el cual se basa Internet.

BIND9: BIND (Berkeley Internet Name Domain) es el software de servidor de nombres de dominio (DNS) más utilizado en Internet. BIND9 es la versión más reciente del software BIND.

C&C: Command and Control (Comando y Control), se refiere a la infraestructura utilizada por los atacantes para enviar instrucciones y controlar malware, botnets u otras actividades maliciosas.

CACHE: Un caché es un lugar donde se almacenan datos temporalmente para que las solicitudes futuras de esos datos puedan ser atendidas de manera más rápida.

CNAME: Canonical Name (Nombre Canónico), es un tipo de registro en un servidor de nombres de dominio (DNS) que se utiliza para establecer una alias o apodo para un nombre de dominio.

DASHBOARD: Un panel de control, es una interfaz gráfica que muestra información de manera visual y fácilmente comprensible. Generalmente, se utiliza para monitorear datos en tiempo real o para mostrar estadísticas y métricas importantes.

DHCP: Dynamic Host Configuration Protocol (Protocolo de Configuración Dinámica de Host), es un protocolo de red que se utiliza para asignar de forma automática direcciones IP y otros parámetros de configuración de red a dispositivos en una red.

DNS: Domain Name System (Sistema de Nombres de Dominio), es un sistema que asocia nombres de dominio legibles por humanos con direcciones IP numéricas que identifican recursos en Internet.

DOMINIO: Un dominio es una identificación asociada a una dirección IP en Internet. Se utiliza en las URL para identificar páginas web, correos electrónicos, etc.

ELASTICSEARCH: Elasticsearch es un motor de búsqueda y análisis de código abierto utilizado para buscar, analizar y visualizar datos en tiempo real.

ELK STACK: ELK es un acrónimo que se refiere a Elasticsearch, Logstash y Kibana, tres herramientas utilizadas juntas para recopilar, almacenar, buscar y visualizar datos de registros en tiempo real.

FEEDS: En el contexto de seguridad informática, feeds se refiere a fuentes de información que proporcionan datos sobre amenazas, vulnerabilidades u otros eventos relevantes para la seguridad.

FIREWALL: Un firewall es un dispositivo o software que se utiliza para controlar el tráfico de red permitiendo o bloqueando ciertas comunicaciones basadas en un conjunto de reglas de seguridad.

HTTP: Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto), es un protocolo de comunicación utilizado para la transferencia de datos en la World Wide Web.

HTTPS: Hypertext Transfer Protocol Secure (Protocolo de Transferencia de Hipertexto Seguro), es una versión segura del protocolo HTTP que utiliza cifrado SSL/TLS para proteger la comunicación.

IOC: Indicators of Compromise (Indicadores de Compromiso), son evidencias que sugieren que un sistema puede haber sido comprometido por una actividad maliciosa.

IOC2RPZ: Una herramienta que convierte indicadores de compromiso (IOC) en políticas de zona de lista de represión (RPZ) para su uso en sistemas de nombres de dominio (DNS) para bloquear tráfico malicioso.

IP: Internet Protocol (Protocolo de Internet), es un protocolo de comunicación que se utiliza para transmitir datos a través de una red, asignando direcciones únicas a cada dispositivo conectado a la red.

IPv6: Internet Protocol version 6 (Protocolo de Internet versión 6), es la versión más reciente del protocolo IP que utiliza direcciones de 128 bits, lo que permite un mayor número de direcciones IP disponibles en comparación con IPv4.

LOGS: Registros, son archivos que contienen información sobre eventos que ocurren en un sistema, aplicación o red.

NXDOMAIN: Non-existent Domain (Dominio No Existe), es un código de error devuelto por un servidor de nombres de dominio (DNS) cuando no puede resolver un nombre de dominio dado.

OPENSOURCE: Código abierto, se refiere a un software cuyo código fuente es accesible públicamente y puede ser modificado y distribuido por cualquier persona.

PHISHING: Phishing es una técnica utilizada por ciberdelincuentes para engañar a las personas haciéndose pasar por entidades confiables con el fin de obtener información confidencial, como contraseñas, números de tarjetas de crédito, etc.

PYMES: Pequeñas y Medianas Empresas, son empresas que tienen un número limitado de empleados y recursos financieros en comparación con las grandes corporaciones.

ROUND-ROBIN: Round-robin es un algoritmo utilizado para balancear la carga de trabajo entre varios servidores o recursos de manera equitativa, asignando solicitudes en secuencia a cada uno.

RPZ: Response Policy Zone (Zona de Política de Respuesta), es un mecanismo utilizado en servidores de nombres de dominio (DNS) para aplicar políticas de seguridad, como bloquear el acceso a ciertos dominios conocidos por ser maliciosos.

SIEM: Security Information and Event Management (Gestión de Información y Eventos de Seguridad), es un enfoque de seguridad que combina la gestión de registros (logs) y la correlación de eventos para proporcionar visibilidad y protección contra amenazas en tiempo real.

SINKHOLE: Sinkhole (pozo absorbente), es un término utilizado en seguridad informática para describir un servidor o dispositivo configurado para redirigir el tráfico malicioso lejos de su destino previsto hacia un destino seguro o controlado.

SSH: Secure Shell (Shell Seguro), es un protocolo de red que permite a los usuarios acceder de forma segura a un servidor remoto a través de una conexión encriptada.

6. Bibliografía

[1] Check Point Software Technologies. (2023). *Cyber Security Report*. Obtenido de https://resources.checkpoint.com/report/2023-check-point-cyber-security-report

[2] Internet Systems Consortium (ISC). (12 de Marzo de 2024). *BIND 9 Administrator Reference*. Obtenido de BIND 9 Administrator Reference: https://downloads.isc.org/isc/bind9/9.18.25/doc/arm/Bv9ARM.pdf

[3] Elastic. (2024). *About Elastic*. Obtenido de About Elastic: https://www.elastic.co/es/about/

[4] Internet System Consortium. (2024). *Base de Conocimiento (KB)*. Obtenido de BIND Logging - some basic recommendations: https://kb.isc.org/docs/aa-01526

[5] https://www.cherryservers.com/blog/how-to-install-and-configure-a-private-bind-dns-server-on-ubuntu-22-04

[6] https://www.digitalocean.com/community/tutorials/how-to-install-nginx-on-ubuntu-20-04

[7] https://www.digitalocean.com/community/tutorials/how-to-install-elasticsearch-logstash-and-kibana-elastic-stack-on-ubuntu-22-04

[8] https://www.digitalocean.com/community/tutorials/how-to-install-java-with-apt-on-ubuntu-22-04#installing-the-default-jrejdk

[9] https://voidnull.es/instalacion-kibana-elasticsearch-logstash-ubuntu-22-04/

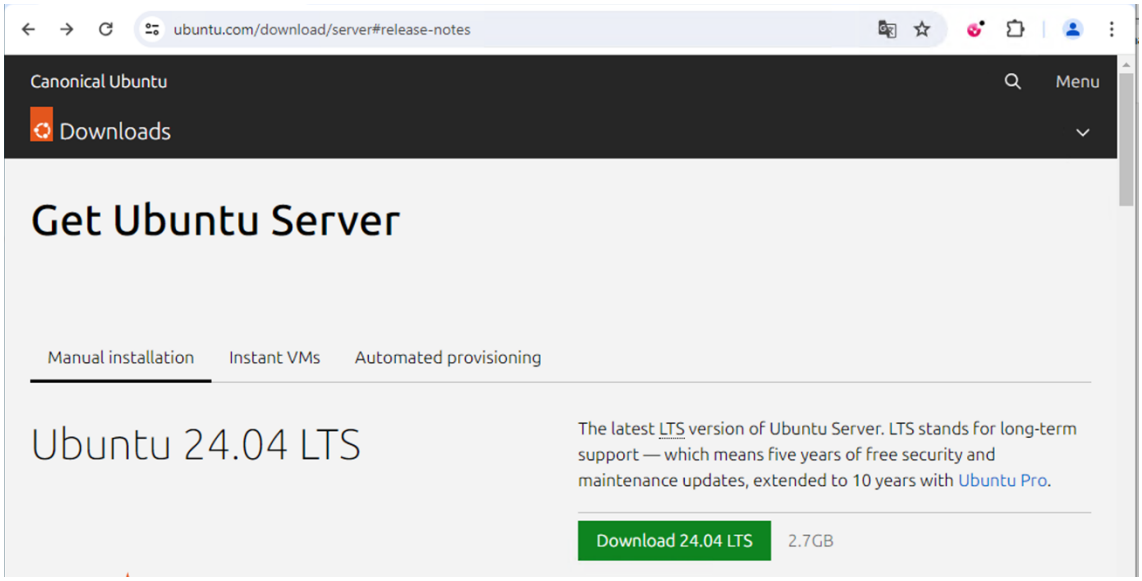
[10] https://github.com/Homas/ioc2rpz

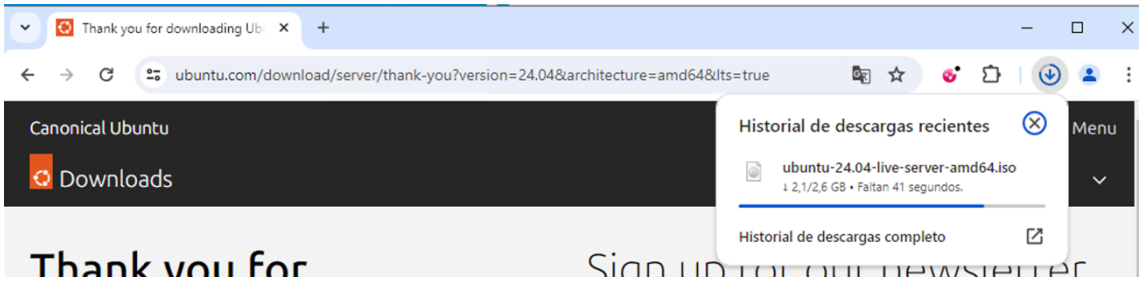
[11] https://www.ionos.es/digitalguide/servidores/configuracion/docker-ubuntu-2204/

7. Anexos

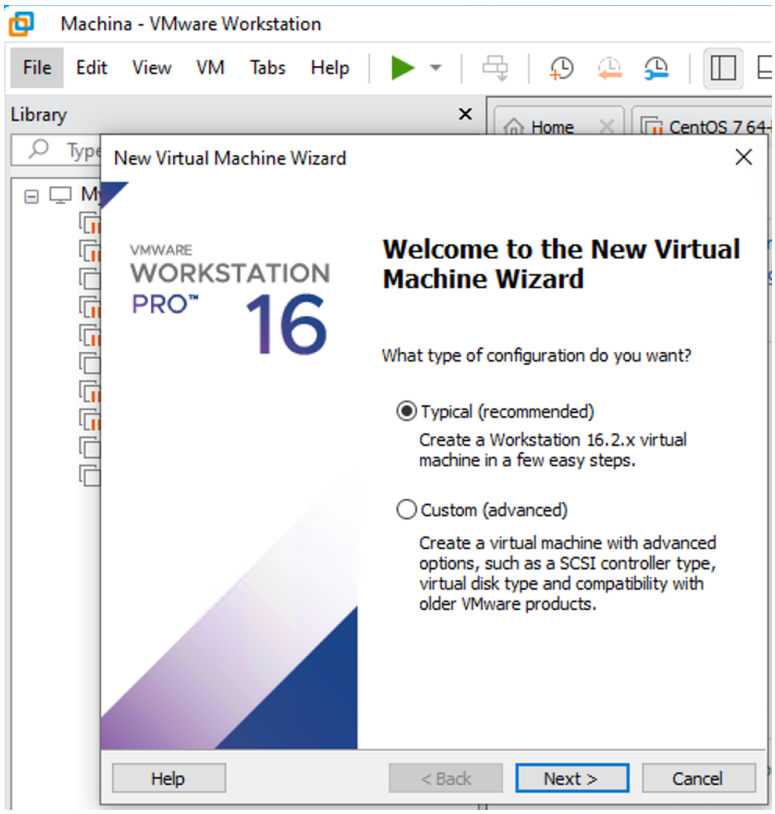
I. Anexo I: Procedimiento de Instalación Ubuntu sobre VMware

Para ejecutar la instalación de Ubuntu, procedemos primero a descargar la imagen (Ubuntu-24.04-live-server-amd64.iso) desde el sitio web oficial Ubuntu.com. Para el caso particular del laboratorio descargamos la versión Ubuntu Server 24.04 LTS.

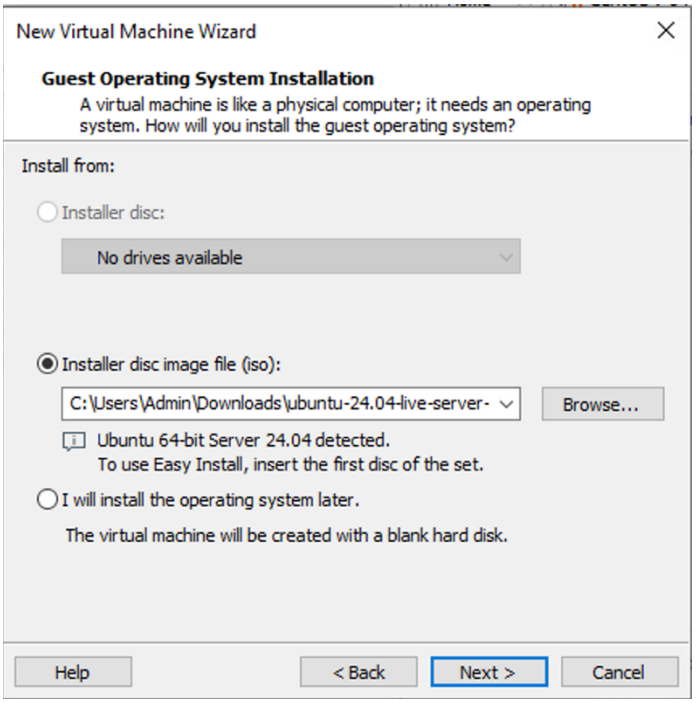




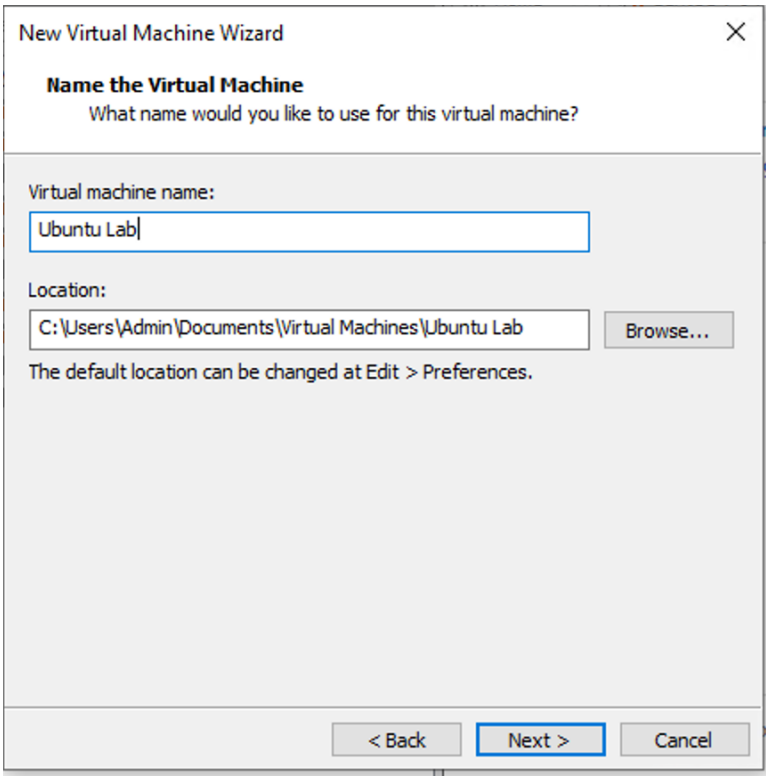
Cuando la descarga a finalizado procedemos a crear una nueva máquina virtual en VMware Workstation. Para esto damos clic en el menú *File* y seleccionamos la opción *New virtual machine.* Desde aquí daremos clic en “Next” para iniciar.



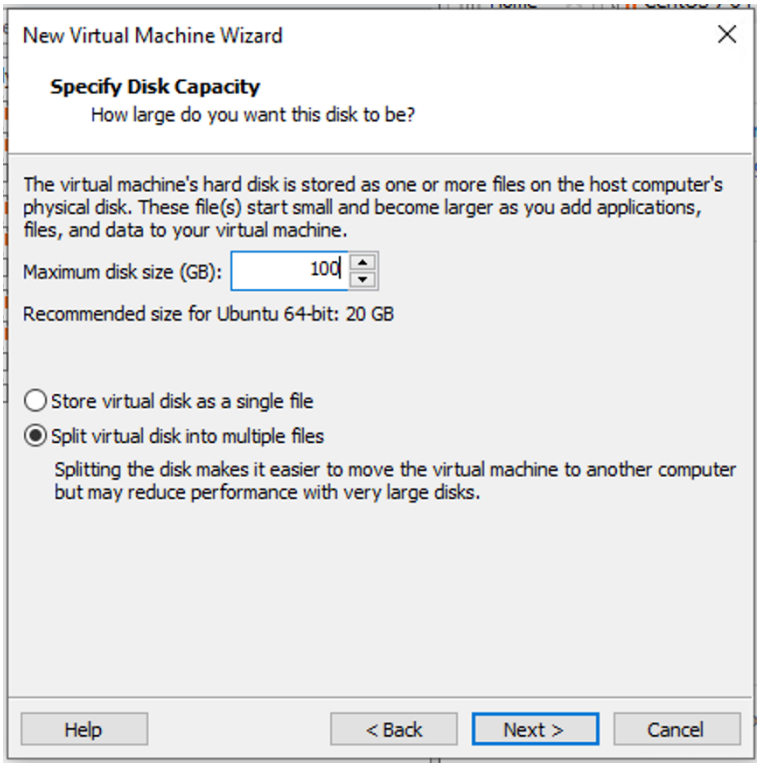
Al iniciar el proceso de instalación debemos seleccionar dando clic en *Browse* la imagen *Ubuntu-24.04-live-server-amd64.iso* descargada anteriormente.



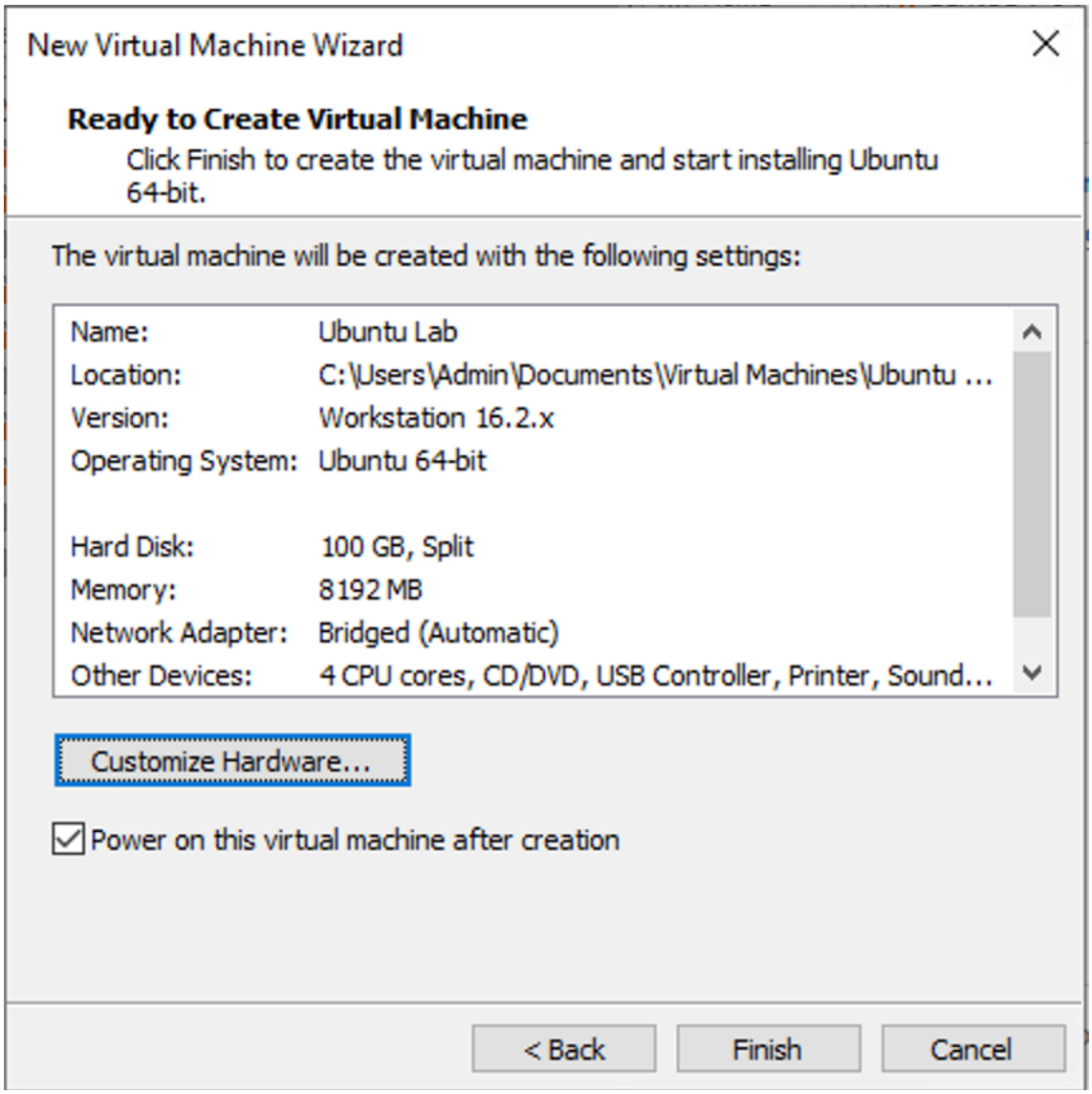
Luego procedemos a realizar la configuración del nombre de la máquina virtual, que para este caso será *Ubuntu Lab*. Finalizamos dando clic en *Next*.



En el siguiente cuadro asignamos 100GB de espacio de disco virtual y damos clic en *Next*.

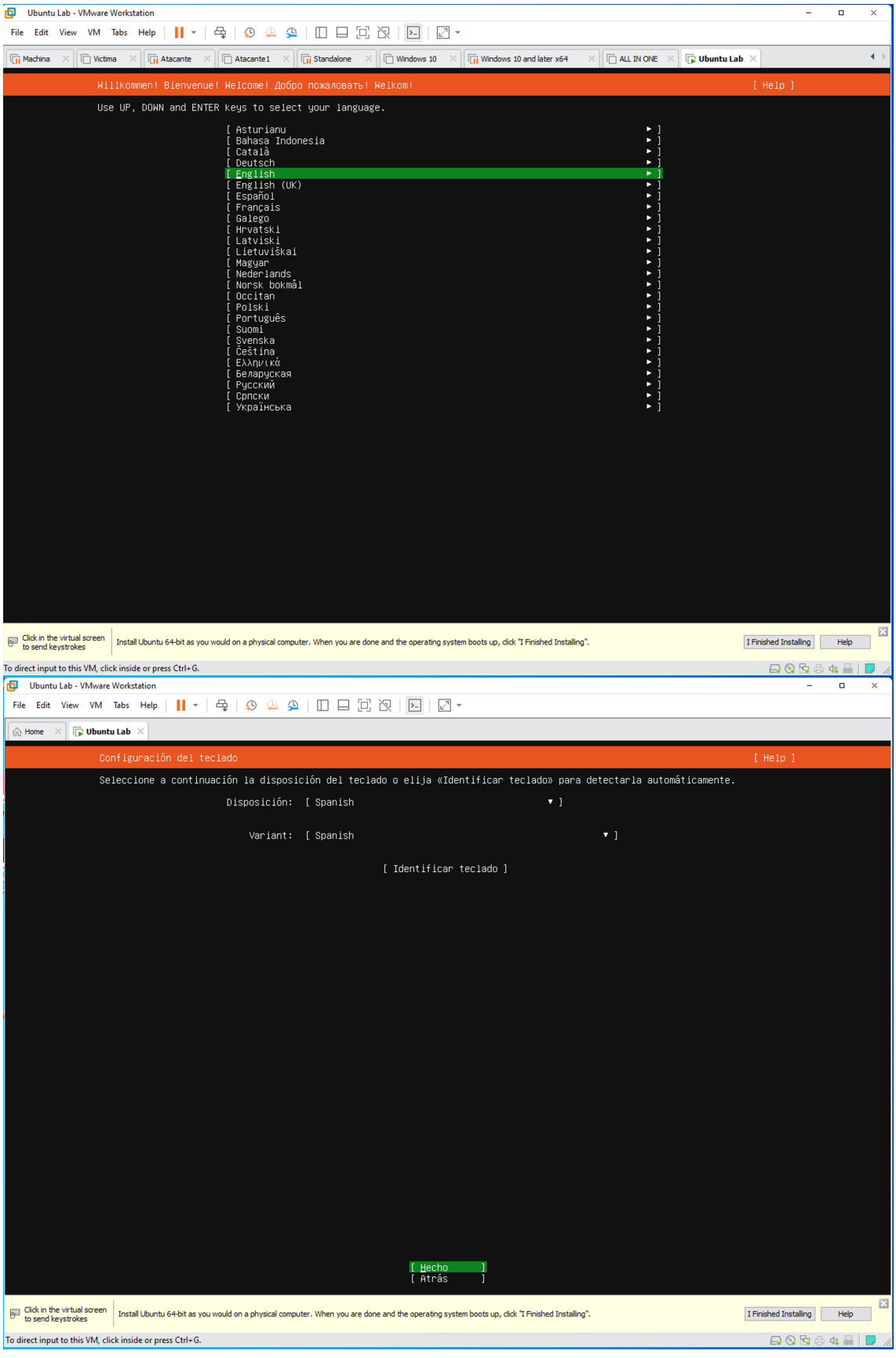


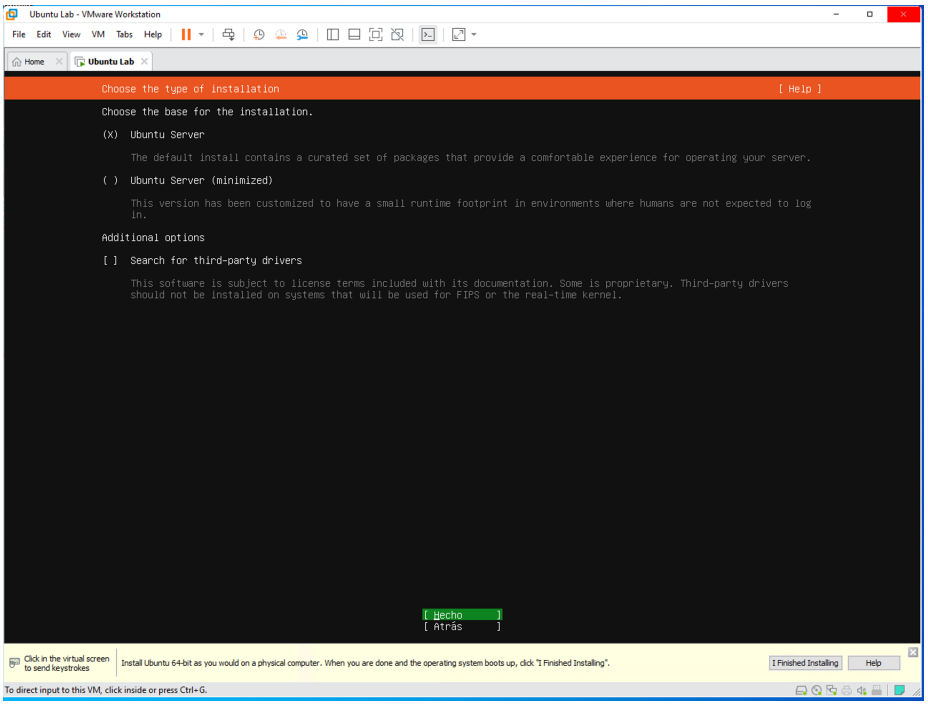
Para finalizar el proceso de carga, damos clic en Customize Hardware y asignamos 8GB de RAM y 4 CPU cores. Finalizamos dando clic en *Finish*.



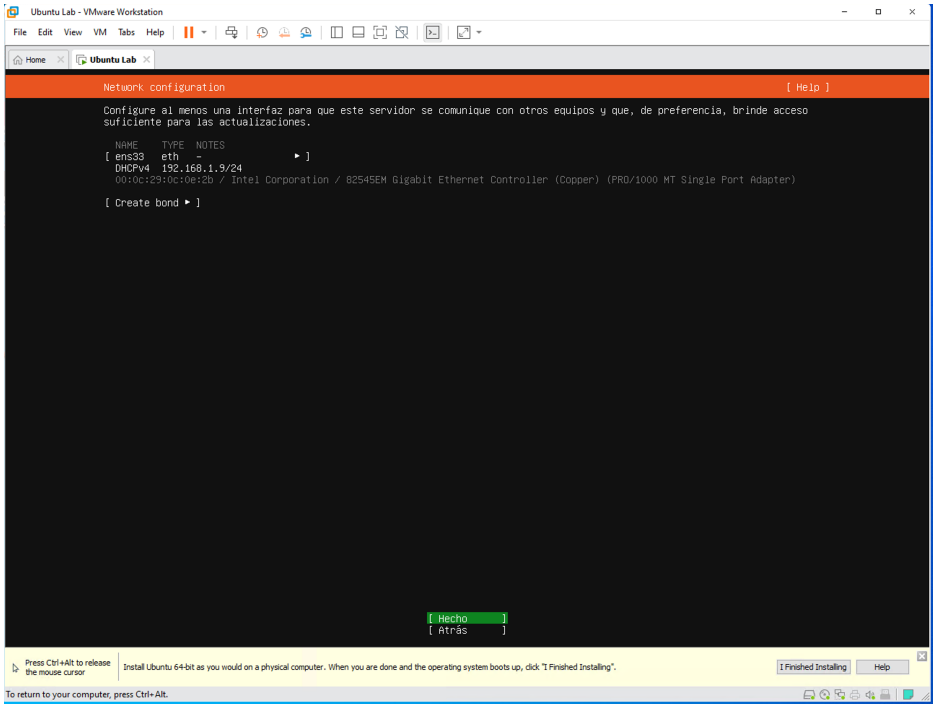
Durante el proceso de boot cargará la imagen de Ubuntu e iniciará el proceso de instalación. En este proceso debemos seleccionar las siguientes opciones:

1. Lenguage: Español
2. Configuración de teclado: Spanish
3. Tipo de instalación: Ubuntu Server

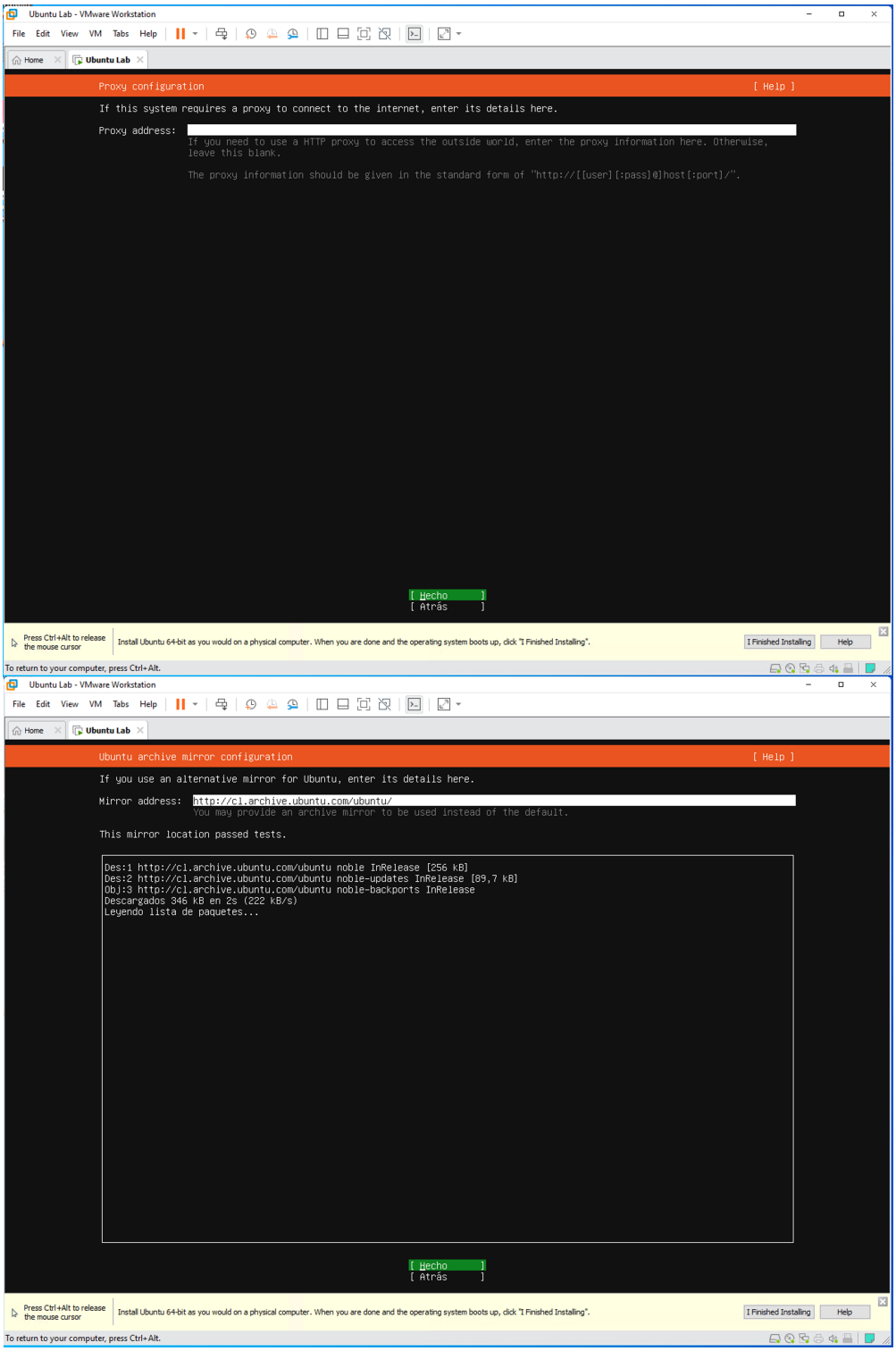

Ubuntu Lab - VMware Workstation
File Edit View VM Tabs Help
Machina Victima Atacante Atacante1 Standalone Windows 10 Windows 10 and later x64 ALL IN ONE Ubuntu Lab
Willkommen! Bienvenue! Welcome! Добро пожаловать! Welkom!
[Help]
Use UP, DOWN and ENTER keys to select your language.
[Asturianu ▸]
[Bahasa Indonesia ▸]
[Català ▸]
[Deutsch ▸]
[English ▸]
[English (UK) ▸]
[Español ▸]
[Français ▸]
[Galego ▸]
[Hrvatski ▸]
[Latviski ▸]
[Lietuviškai ▸]
[Magyar ▸]
[Nederlands ▸]
[Norsk bokmål ▸]
[Occitan ▸]
[Polski ▸]
[Português ▸]
[Suomi ▸]
[Svenska ▸]
[Čeština ▸]
[Ελληνικά ▸]
[Беларуская ▸]
[Русский ▸]
[Српски ▸]
[Українська ▸]
Click in the virtual screen to send keystrokes
Install Ubuntu 64-bit as you would on a physical computer. When you are done and the operating system boots up, click "I Finished Installing".
I Finished Installing
Help
To direct input to this VM, click inside or press Ctrl+G.
Ubuntu Lab - VMware Workstation
File Edit View VM Tabs Help
Home Ubuntu Lab
Configuración del teclado
[Help]
Seleccione a continuación la disposición del teclado o elija «Identificar teclado» para detectarla automáticamente.
Disposición: [Spanish ▾]
Variant: [Spanish ▾]
[Identificar teclado]
[Hecho]
[Atrás]
Click in the virtual screen to send keystrokes
Install Ubuntu 64-bit as you would on a physical computer. When you are done and the operating system boots up, click "I Finished Installing".
I Finished Installing
Help
To direct input to this VM, click inside or press Ctrl+G.



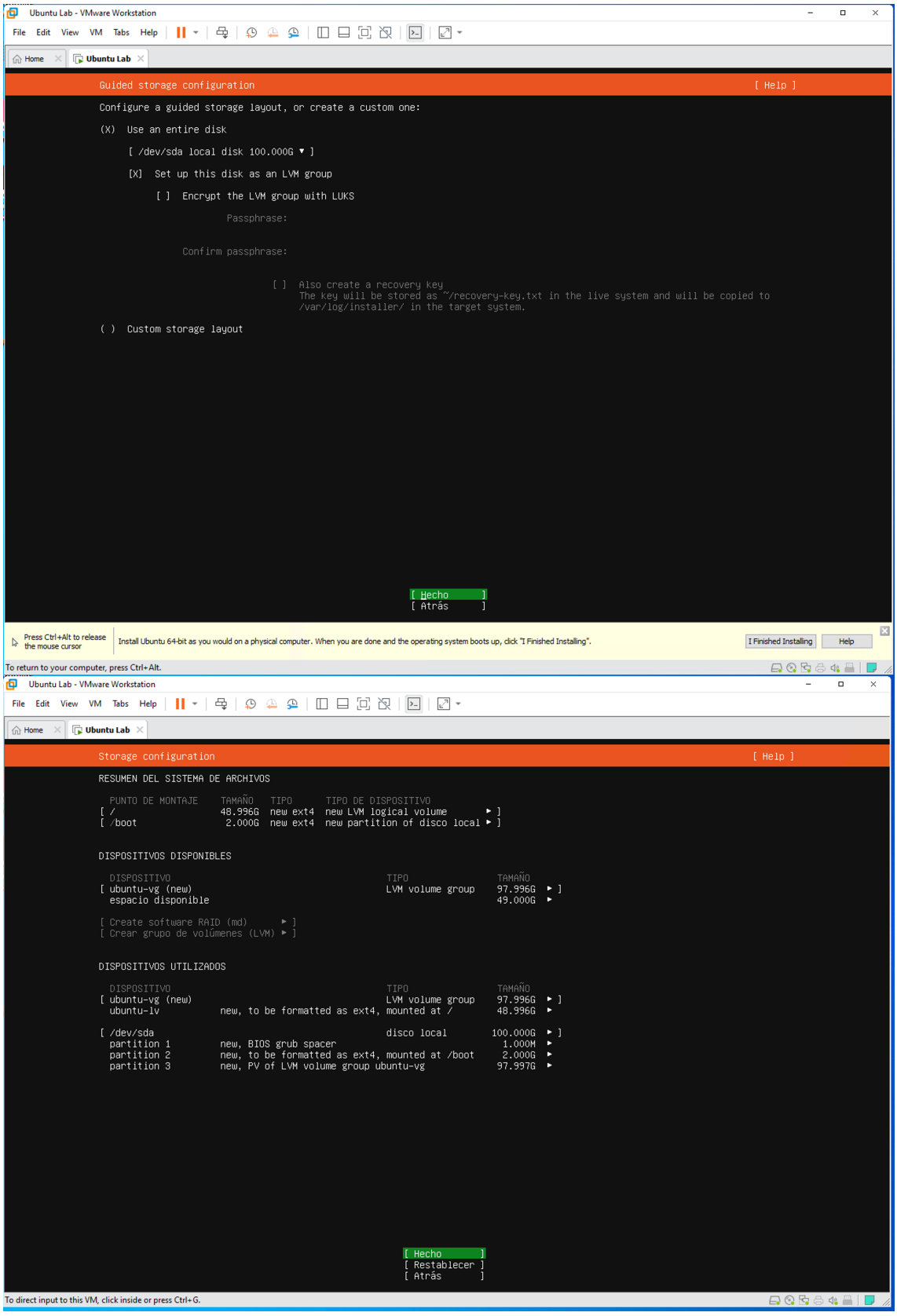
Para la configuración de red, por defecto la herramienta intentará obtener dhcp en la puerta conectada. En este apartado se configura la puerta de la máquina virtual en modo “Bridge” y se obtiene la dirección IP 192.168.4.82/22. Avanzamos dando enter en “Hecho”.



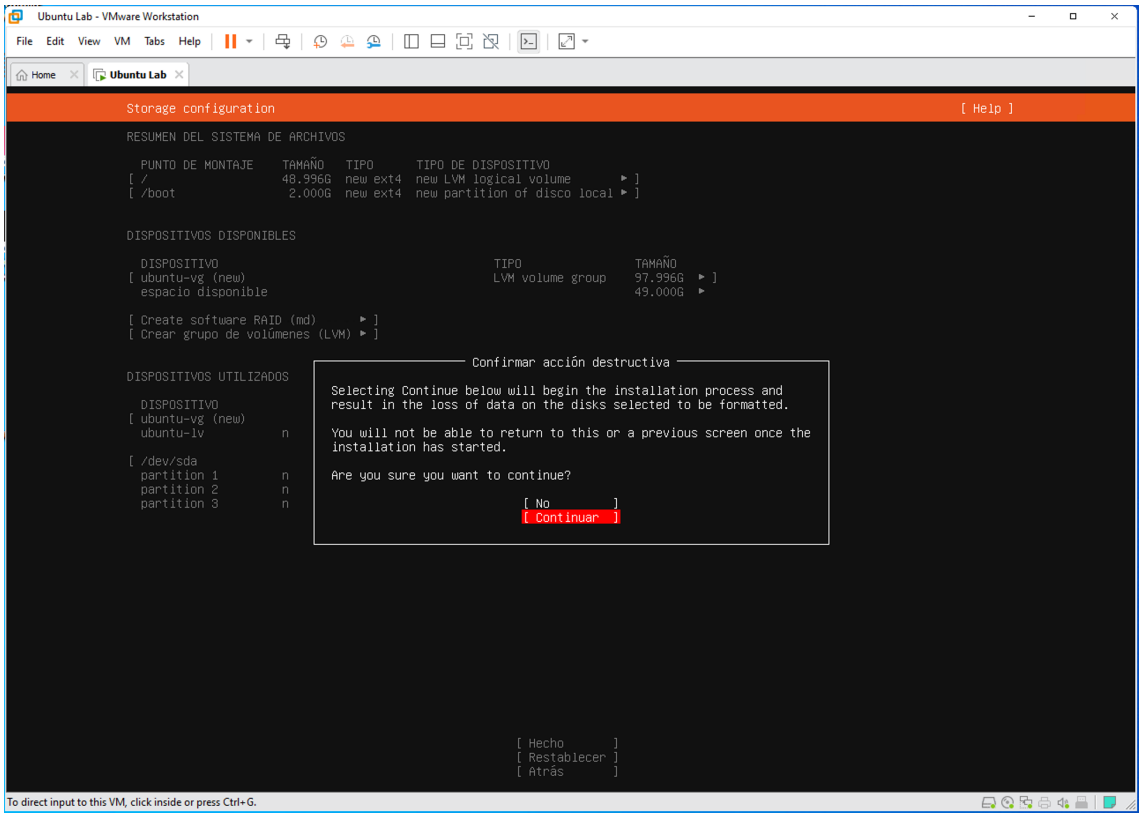
Para la configuración de este laboratorio la máquina virtual navega directo hacia internet, sin necesidad de usar un Proxy, por lo que se mantiene en blanco y se avanza seleccionando “Hecho”. Luego se iniciará el proceso de update de los repositorios.



Al finalizar el proceso de instalación y actualización de los repositorios iniciamos la configuración del almacenamiento. Para esto seleccionamos “Usar el disco entero” y luego avanzamos con “Hecho”. Antes de iniciar la instalación nos presentará un resumen del sistema de archivo. Avanzamos dando “Hecho”.



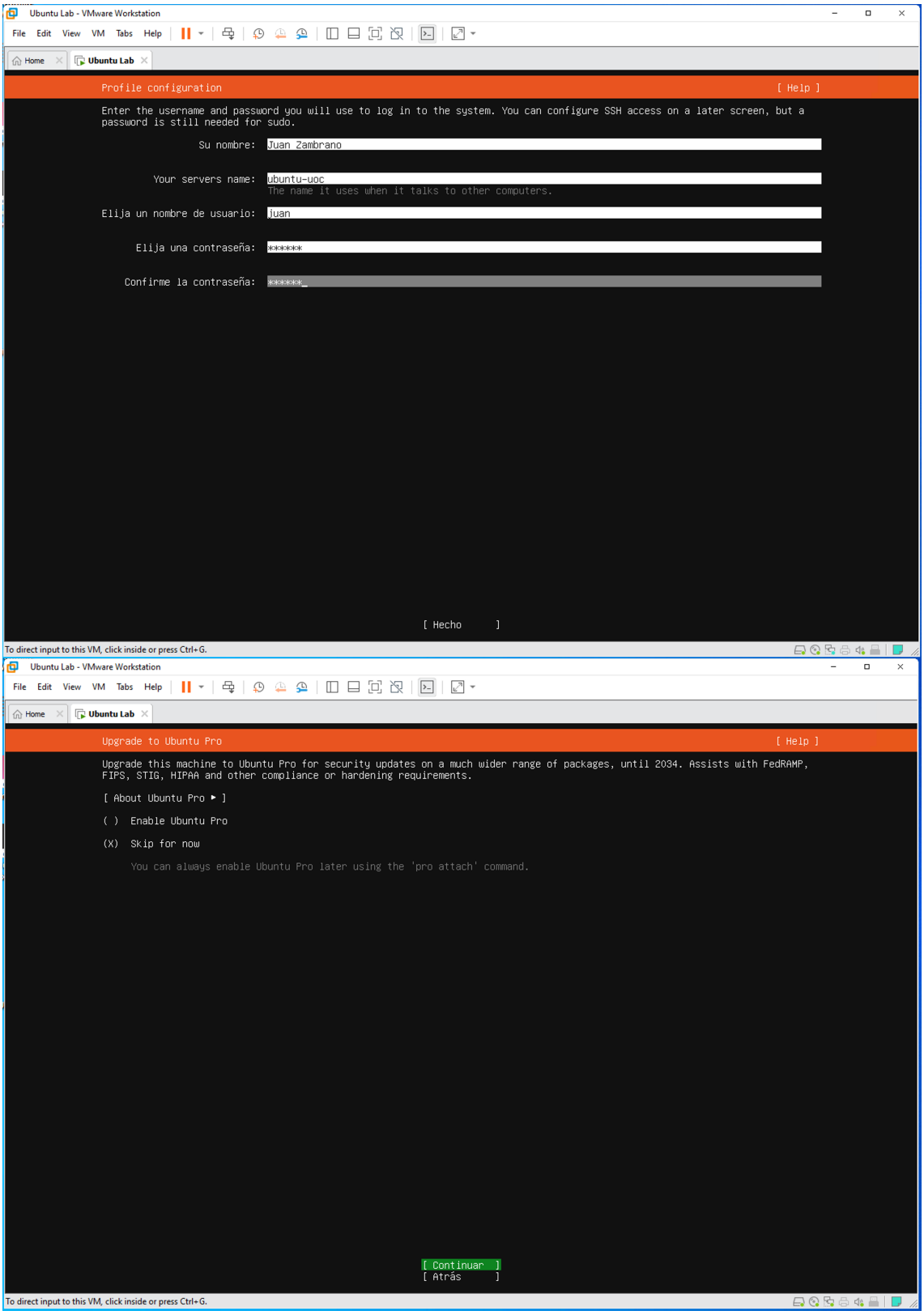
Confirmamos el inicio de instalación dando enter en “Continuar”.



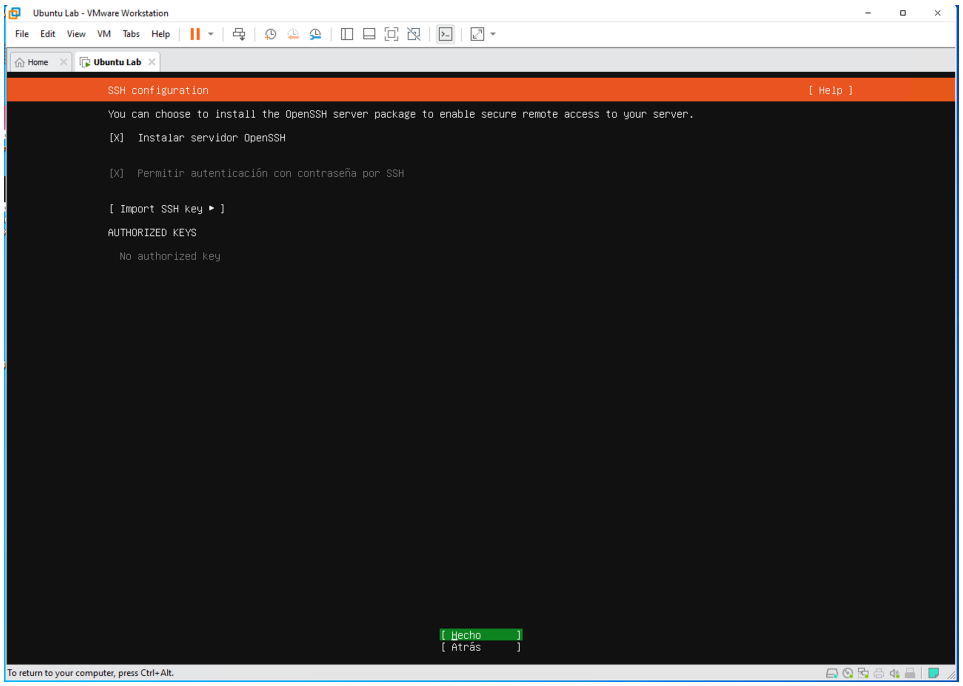
En el siguiente cuadro realizaremos la configuración del perfil del administrador. Agregamos los parámetros:

- Nombre: Juan Zambrano
- Nombre del servidor: Ubuntu-uoc
- Nombre de usuario: juan
- Contraseña: <contraseña>

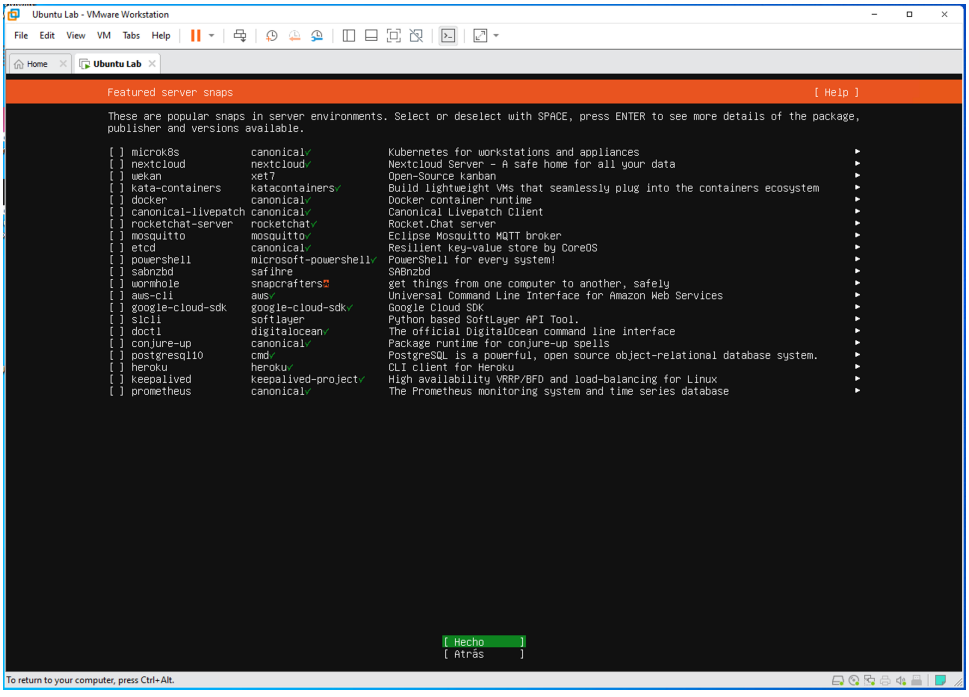
Finalizamos dando enter en “Hecho”. En el siguiente cuadro nos consultará si deseamos actualizar a “Ubuntu Pro”, el cual saltaremos por el momento seleccionando “Skip for now” y dando enter en “Continuar”.



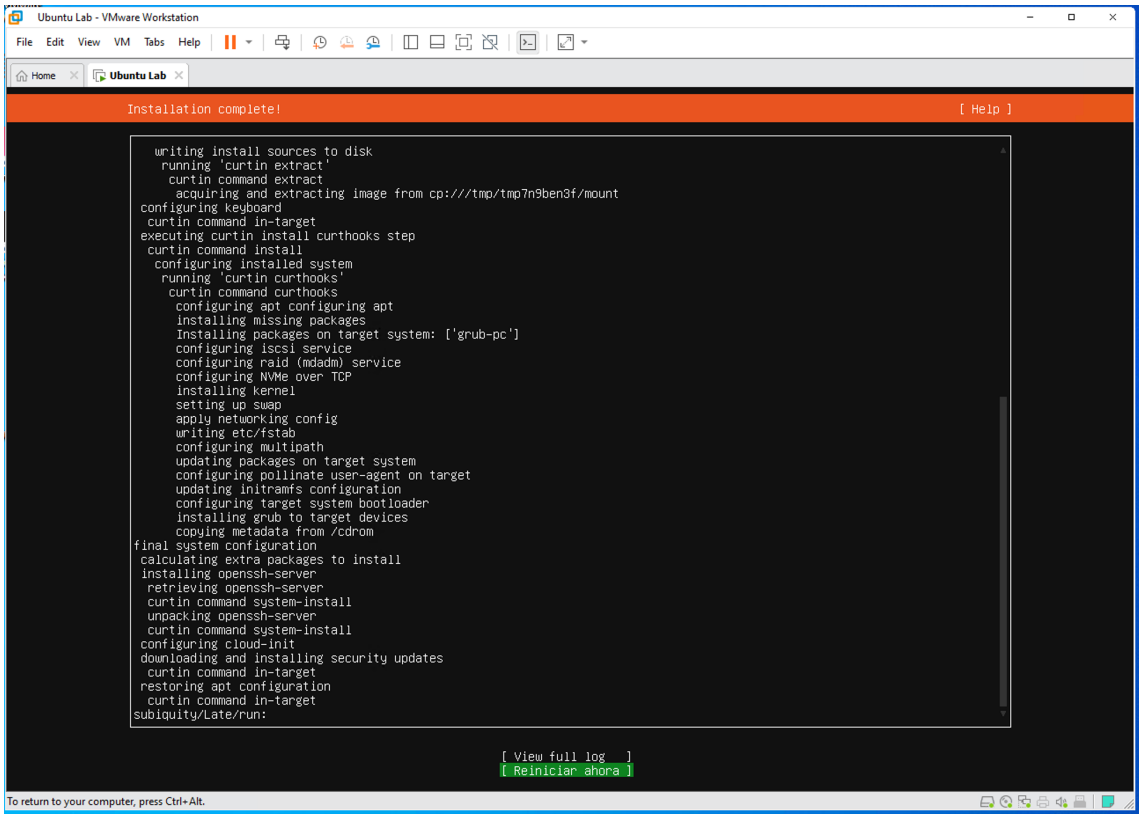
En el siguiente cuadro nos consultará por la configuración de SSH, sobre la cual mantendremos habilitadas las opciones “Instalar servidor OpenSSH” y “permitir autenticación con contraseña por SSH”. Al finalizar damos enter en “Hecho”.



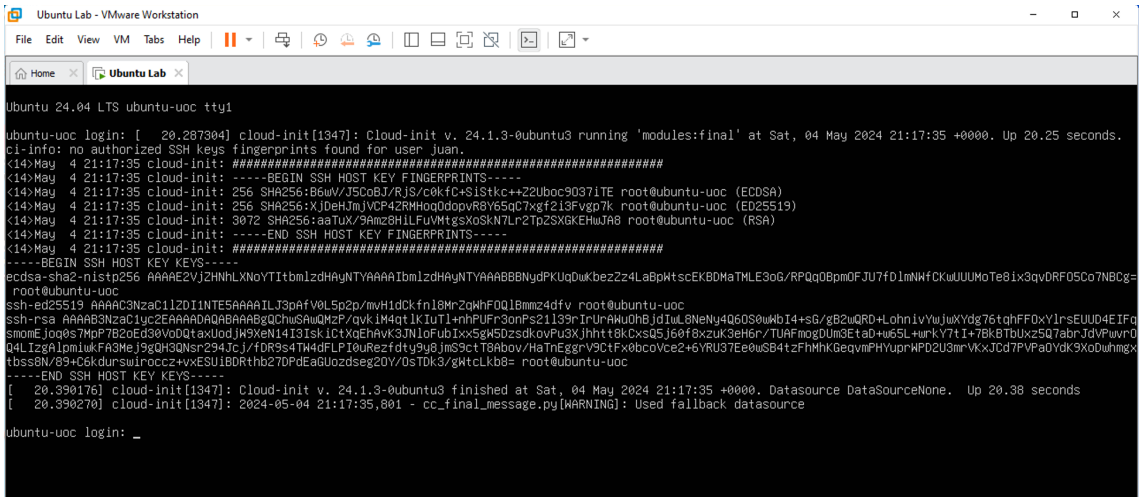
En el siguiente cuadro nos consultará sobre las características que podemos habilitar o deshabilitar. En este aparado no se utilizarán, por lo que se mantienen y se da enter en “Hecho”.



Al finalizar el proceso de instalación seleccionamos “Reiniciar ahora” para empezar a trabajar sobre nuestro sistema operativo Ubuntu.



Al iniciar la maquina accedemos utilizando las credenciales creadas anteriormente.



En este primer login podremos observar las configuraciones aplicadas de la maquina, tales como la dirección IP de cada una de las interfaces y el uso de recursos del sistema.

```
ubuntu-uoc login: juan
Password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of sáb 04 may 2024 21:19:35 UTC

  System load:           0.11
  Usage of /:            12.9% of 47.93GB
  Memory usage:          3%
  Swap usage:            0%
  Processes:             276
  Users logged in:       0
  IPv4 address for ens33: 192.168.4.82
  IPv6 address for ens33: fd8f:29de:7c7f:6b49:20c:29ff:fe0c:e2b
```

II. Anexo II: Primera configuración de Kibana

Con la configuración de nginx preparada, procedemos a acceder a través del navegador al sitio “appfirewall.cl”, desde donde es posible configurar un token o crear la configuración de autenticación manual. Para esto se debe dar clic en “Configure manually”.

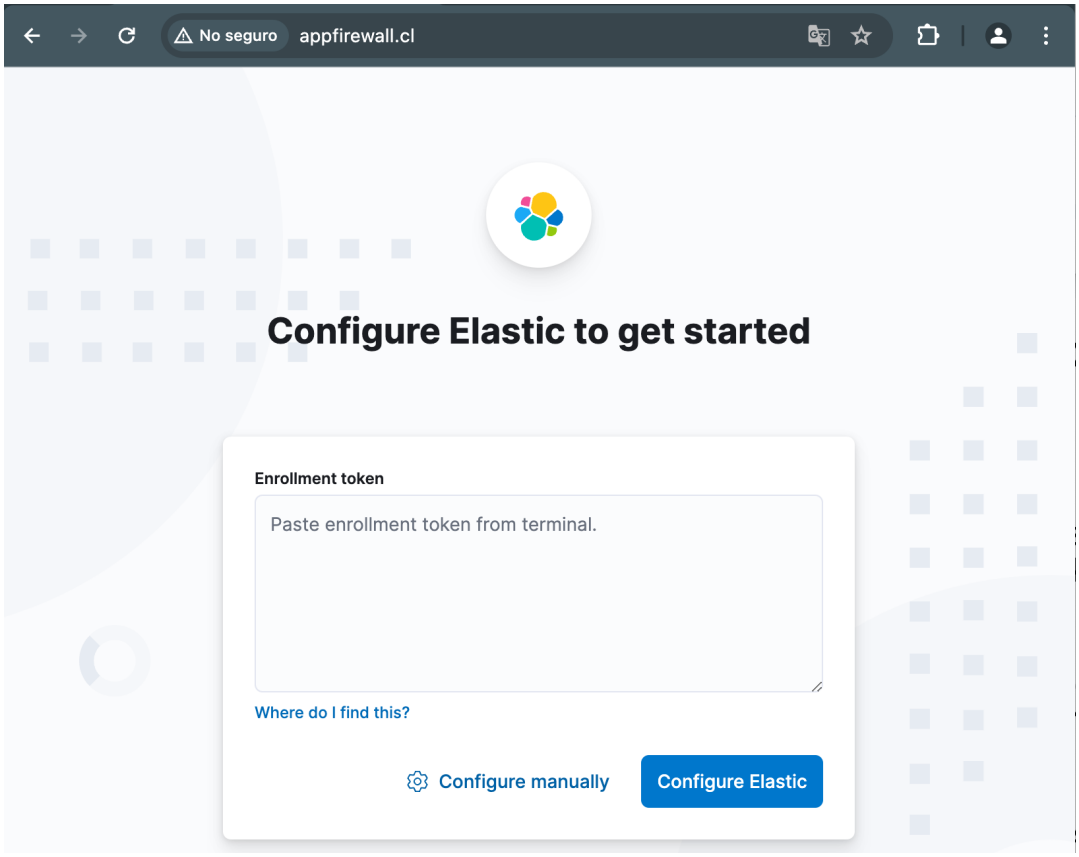


Ilustración 107: Configuración inicial Elastic

Luego en la configuración será necesario indicar la dirección de escucha del servicio Elastic. Para este caso se mantiene el servicio, debe dar clic en “Check address” para validar la conectividad.

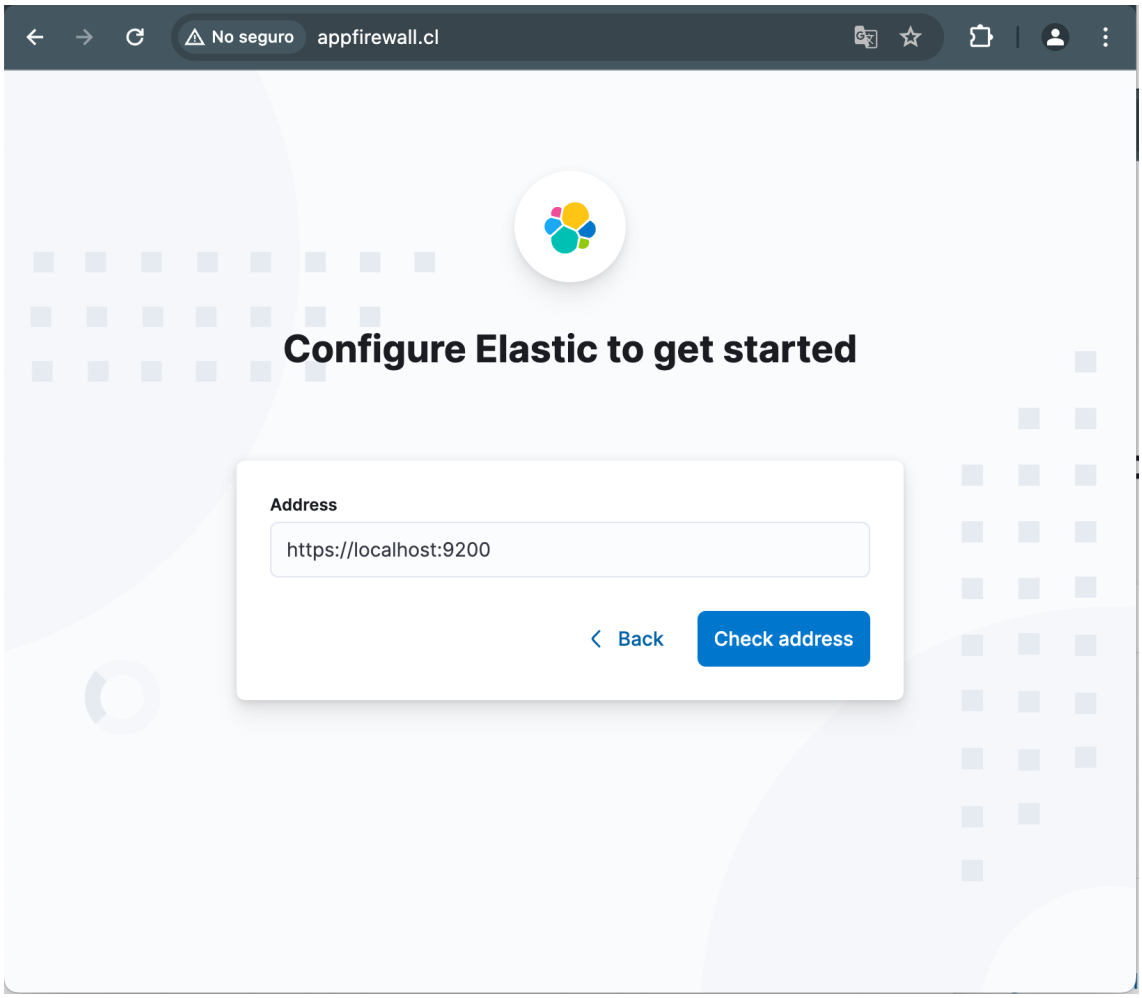


Ilustración 108: Puerto de configuración Elastic

En el siguiente cuadro se debe usar el nombre de usuario kibana_system. Para mostrar la contraseña deberemos de ejecutar el siguiente comando:

- /usr/share/elasticsearch/bin/elasticsearch-reset-password -u kibana_system

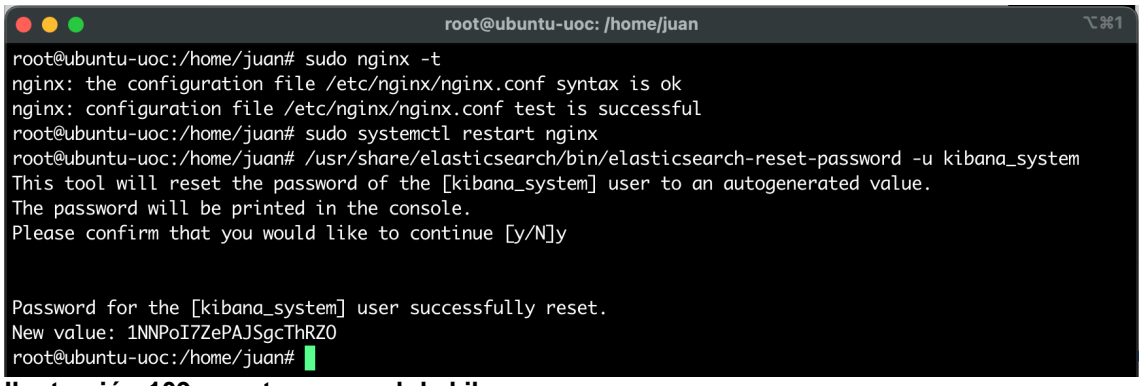


Ilustración 109: reset password de kibana

Con esto se debe utilizar la credencial generada para la cuenta de kibana_system y seleccionamos confiar en el certificado del servidor. Se finaliza la configuración dando clic en “Configure Elastic”.

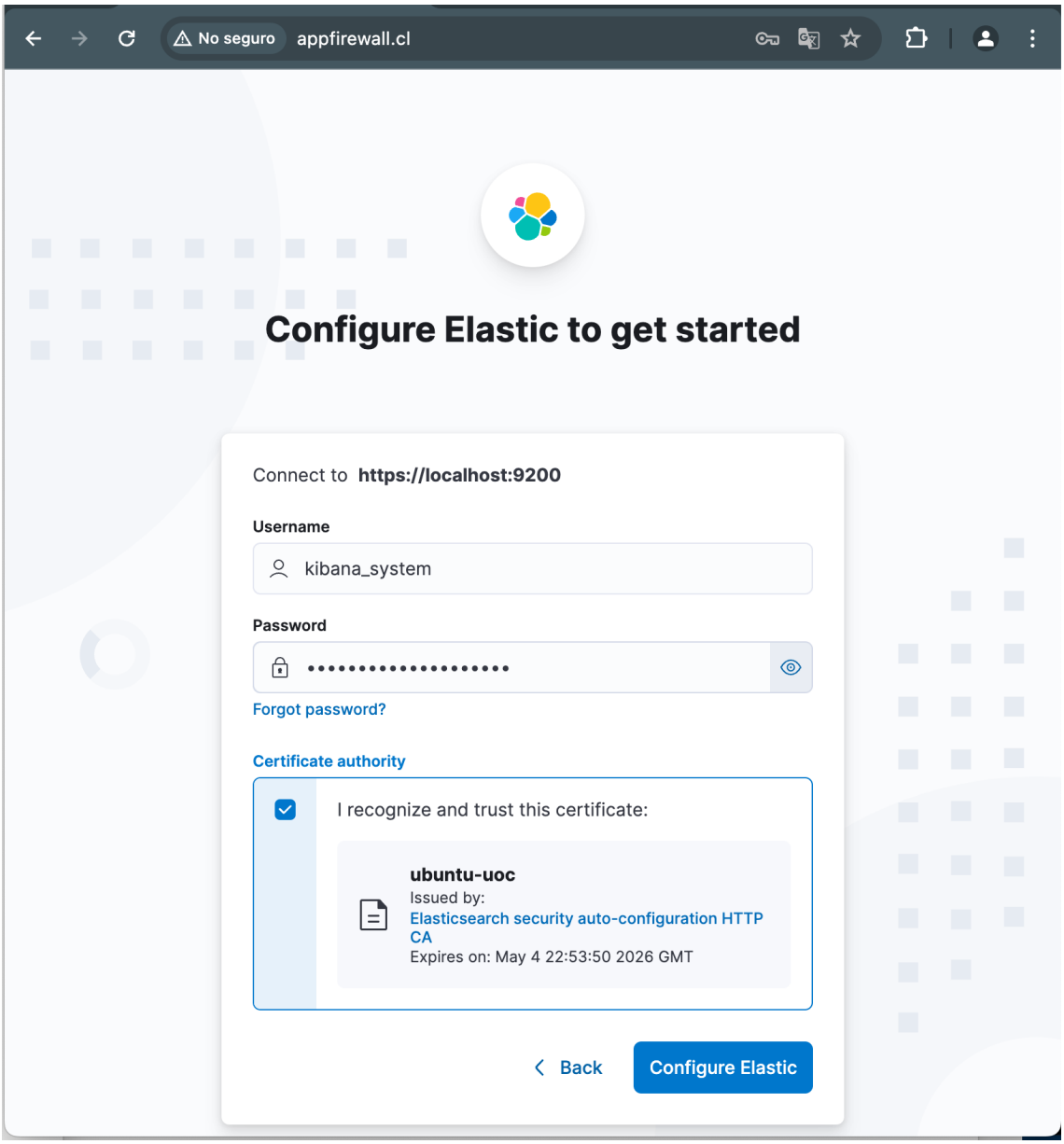


Ilustración 110: Carga de configuración para elastic

Luego para validar la configuración se debe solicitar la generación de un codigo de verificación. Para esto se utiliza el comando "/usr/share/kibana/bin/kibana-verification-code"

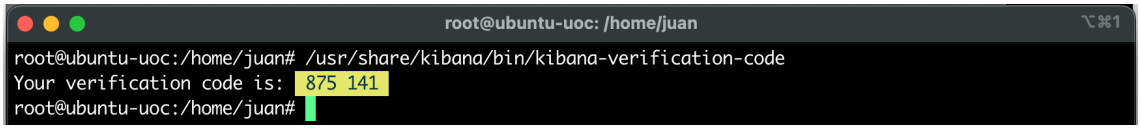


Ilustración 111: código de verificación de kibana

Se procede a utilizar el codigo de verificación y dar clic en "Verify".

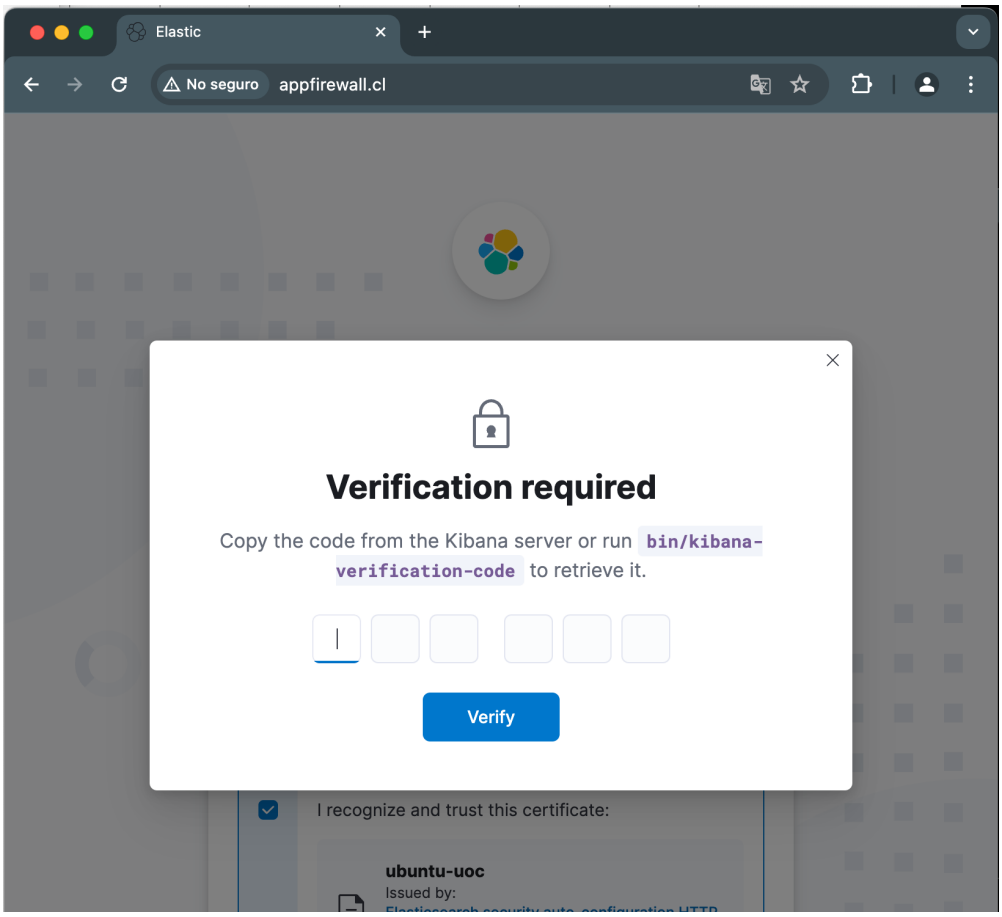


Ilustración 112: ingreso de código de verificación

Con esto se inicia el proceso de instalación y primer inicio como se observa en la imagen.

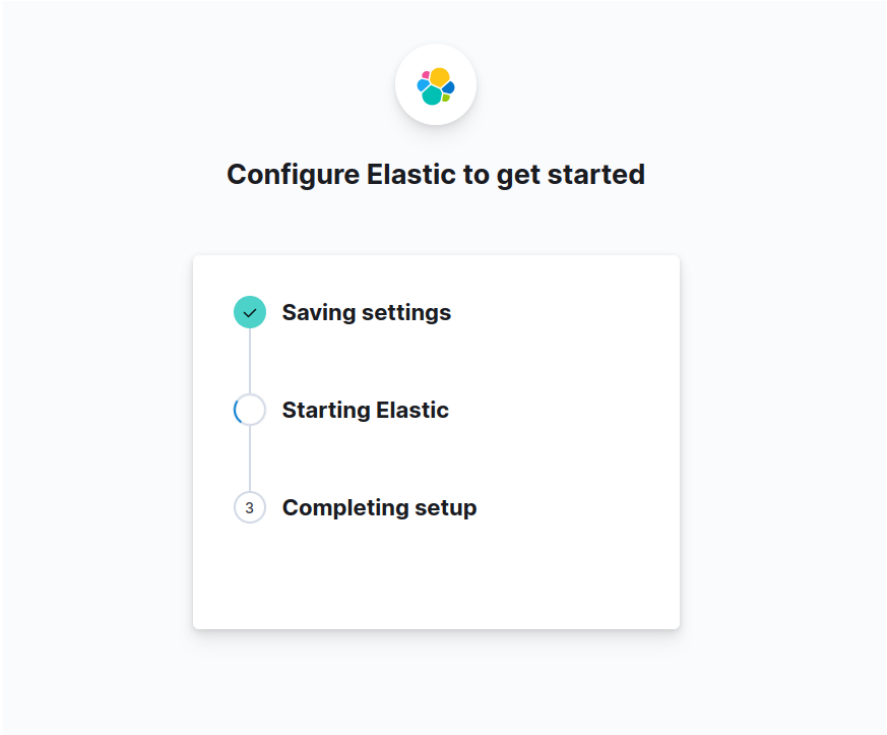


Ilustración 113: inicio de configuración

Al finalizar la instalación, necesitaremos utilizar las credenciales de usuario o la cuenta principal bajo el usuario “elastic”.

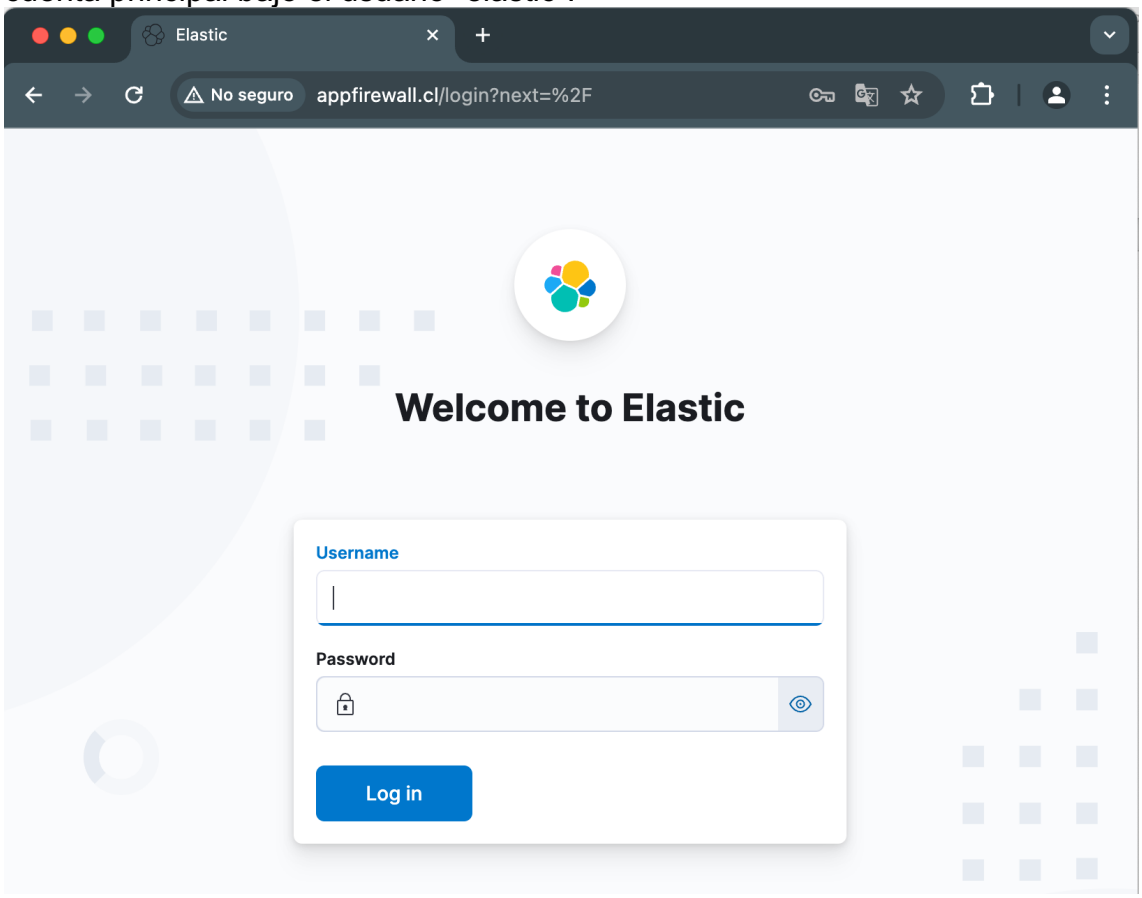


Ilustración 114: Primer login en Elastic

Para crear una cuenta de usuario se debe utilizar el siguiente comando:

- /usr/share/elasticsearch/bin/elasticsearch-users useradd juan -r kibana_user

```
root@ubuntu-uoc:/home/juan# /usr/share/elasticsearch/bin/elasticsearch-users useradd juan -r kibana_user
Enter new password:
Retype new password:
root@ubuntu-uoc:/home/juan#
```

Ilustración 115: Creación de usuario de Kibana

Con la cuenta de usuario nueva creada se utilizan estas credenciales para el acceso.