Acercamiento al *Edge Computing* a través del despliegue de un SIEM sobre Raspberry Pi

Jesús Moreno Carrillo
Máster Universitario en Ciberseguridad y Privacidad
Security of Internet of the Things (IoT)

Víctor Méndez Muñoz
Helena Rifá Pous

01/2021

Agradecimientos

A **mis padres**, por su apoyo incondicional.

A la mejor compañera de viaje, **Verónica**, por su apoyo, templanza e inestimable ayuda.

A **Víctor Méndez Muñoz** por sus consejos aportados y su ayuda durante este proyecto.

A **Helena Rifá Pous** por darme la opción de investigar sobre esta temática.

A **Juan José Murgui**, por su constante asesoramiento.

FICHA DEL TRABAJO FINAL

Título del trabajo:	Acercamiento al Edge Computing a través del despliegue de un SIEM sobre Raspberry Pi
Nombre del autor:	Jesús Moreno Carrillo
Nombre del consultor/a:	Víctor Méndez Muñoz
Nombre del PRA:	Helena Rifá Pous
Fecha de entrega (mm/aaaa):	01/2021
Titulación:	Máster Universitario en Ciberseguridad y Privacidad
Área del Trabajo Final:	Seguridad en la Internet of Things
Idioma del trabajo:	Español
Palabras clave	SIEM, Raspberry, ciberseguridad, monitorización, evento, vulnerabilidad, *log*, amenaza

Resumen del Trabajo

Este proyecto tiene como objetivo tratar de resolver una problemática cada vez más creciente a la cual los usuarios nos enfrentamos: los ataques a dispositivos a conectados a internet

Para dar respuesta a la misma se realizará una aproximación al componente principal de un Centro de Operaciones de Seguridad (SOC) sobre el cual tratará el proyecto: el SIEM. Un elemento fundamental para la detectar y responder ante las amenazas.

Contaremos con Wazuh, SIEM distribuido bajo licencia Open Source, el cual, apoyado en la suite de Elastic (Elasticsearch, *Logs*tash, Filebeat y Kibana) será desplegado bajo una arquitectura unificada en el que los servicios correrán sobre un minicomputador de bajo coste. Este proyecto será implantado a cabo una Raspberry Pi 4. Ello nos permitirá tener un acercamiento hacia el paradigma del *Edge Computing.*

Este sistema contará con dos servicios de protección adicionales para asegurar las comunicaciones (implantación de VPN mediante OpenVPN) y de un sistema de DNS, monitorización de tráfico en red y mejora de la privacidad en la navegación (Pi-hole).

Mediante el desarrollo de una prueba de concepto abordaremos la problemática del tratamiento de *logs* de diversas fuentes, así como se recrearán algunos de los principales ataques a sistemas. Se comprobará cómo el despliegue realizado permitirá dar cumplimiento a aspectos de Compliance de normativas como la LOPDGDD, RGPD o ISO 27001.

Finalmente se trazarán una serie de líneas para abordar una futura ampliación de las funcionalidades del sistema base implementado.

Abstract

The aim of this project is aimed to solve a growing problem that users are facing: attacks to devices through communication networks.

In order to provide an answer to this problem, an approach will be made to the main component of a Security Operations Centre (SOC) on which the project will be focused: the SIEM. A fundamental element for detecting and responding to threats.

We will have Wazuh, SIEM distributed under an Open Source license, which, supported by the Elastic suite (Elasticsearch, *Log*stash, Filebeat and Kibana) will be deployed under a unified architecture in which the services will run on a low-cost minicomputer. This project will be implemented with a Raspberry Pi 4. This will allow us to have an approach towards the Edge Computing paradigm.

This system will have two additional protection services to ensure communications (OpenVPN VPN implementation) and a DNS system, network traffic monitoring and improvement of privacy in navigation (Pi-hole).

Through the development of a proof of concept we will address the problem of processing *log*s from various sources, as well as recreating some of the main attacks on systems. We will check how the deployment will allow to comply with aspects of compliance with regulations such as LOPDGDD, RGPD or ISO 27001.

Finally, a series of lines will be drawn up to address a future extension of the functionalities of the implemented base system.

Índice

1. Introducción 1
1.1. Contexto y justificación del Trabajo 1
1.2. Objetivos del Trabajo 3
1.3. Enfoque y metodología seguida 4
1.4. Definición y planificación de las tareas 4
1.5. Estado del arte 7
1.6. Descripción de los recursos necesarios 8
2. Análisis 9
2.1. Amenazas en la red 9
2.2. Introducción al *Edge Computing* 10
2.3. Soluciones de seguridad para tratar la problemática etudiada 11
2.4. ¿Qué es un SOC? 11
2.4.1. ¿Qué funciones tiene un SOC? 11
2.4.2. Arquitectura o componentes de un SOC 12
2.5. SIEM. Concepto, componentes y funcionalidades 13
2.6. ¿Por qué es importante contar con un SIEM? 13
2.7. ¿Cómo puede un SIEM ayudar a una organización a cumplir con aspectos regulatorios como el GDPR, ISO 27001? 13
3. Desarrollo 15
3.1. Wazuh 15
3.2. Diseño de la solución 16
3.2.1. Requisitos necesarios para llevar a cabo el despliegue 17
3.2.1.1. Requisitos de software 17
3.2.1.2. Requisitos de hardware 18
3.3. Instalación de Wazuh 3.13 19
3.3.1. Adecuación del sistema 19
3.3.2. Instalación de Wazuh Server 19
3.3.3. Instalación de Wazuh API 20
3.4. Instalación Suite Elastic 20
3.4.1. Instalación de Filebeat 21
3.4.2. Instalación de Elasticsearch 22
3.4.3. Instalación de Kibana 23
3.5. Actualización de versión 25
3.6. Despliegue de los agentes 26
3.6.1. Agente para Windows 27
3.6.2. Agente para Linux (Debian) 27
3.6.3. Agente Macintosh OS 27
3.7. Securización de la interfaz gráfica 28
3.8. Panel de administración Kibana / Wazuh 28
3.9. Instalación de OpenVPN 29
3.9.1. Instalación del servidor 30
3.9.2. Creación del cliente 30
3.10. Pi-hole 31
3.10.1. Instalación 31
3.10.2. *Log*s 32

4. Casos de uso 33
4.1. Detección de archivo malicioso alojado en cloud a través de VirusTotal 33
4.2. Prevención y detección ante un ataque de Ransomware 35
4.3. Detección de ataque de fuerza bruta por SSH 37
4.4. Detección de conexiones a la VPN desplegada 38
4.5. Detección de vulnerabilidades 40
4.6. Notificación de alertas vía correo electrónico 41
4.7. Notificación de alertas a través de un bot de Telegram 43
4.8. Generación de informes 46
4.9. Cumplimiento normativo: GDPR / RGPD 47
5. Conclusiones 50
6. Líneas de trabajo futuro 53
7. Bibliografía 55
8. Anexos 57
I. Securización de la Suite Elastic 57
II. Instalación de Postfix 62
III. Instalación de Pi-hole 64
IV. Instalación de PiVPN - OpenVPN 70
V. Script para simulación de ataque Ransomware 77
VI. Creación de Bot de Telegram 79
VII. Informe ampliado Compliance: RGPD 82

Índice de ilustraciones

Ilustración 1. Planificación del proyecto. Elaboración propia. 6
Ilustración 2. Threat Landscape 2020 - List of top 15 threats. ENISA. 9
Ilustración 3. Cloud Computing vs Edge Computing. ThinkBig. 10
Ilustración 4: Modelo de organización de un SOC. Van Os. 12
Ilustración 5. Root causes to determine best remediations. Filkins. 14
Ilustración 6. Esquema del funcionamiento de los sistemas a implementar. Wazuh. 16
Ilustración 7. All in one deployment. Wazuh. 17
Ilustración 8. Diagrama básico de red. Elaboración propia. 18
Ilustración 9. Configuración fichero Filebeat.yml. Elasticsearch. 22
Ilustración 10. Estado Elasticsearch. Raspberry Pi OS. 23
Ilustración 11. Revisión estado Elasticsearch. Elasticsearch. 23
Ilustración 12. Estado del servicio Kibana. Raspberry Pi OS. 24
Ilustración 13. Matriz de compatibilidad de versiones Wazuh 3.13.X. Wazuh. 25
Ilustración 14. Matriz de compatibilidad Wazuh 4.0.0. Wazuh. 25
Ilustración 15. Comprobación versión Wazuh. Raspberry Pi OS. 26
Ilustración 16. Despliegue de agentes. Wazuh. 26
Ilustración 17. Comprobación funcionamiento del agente en Task Manager. Windows 10. 27
Ilustración 18. Panel de agentes Wazuh. Kibana. 27
Ilustración 19. Autenticación portal web Elastic. Elastic. 28
Ilustración 20. Panel de administración de Elastic. Kibana. 29
Ilustración 21. Panel de administración Wazuh. Kibana / Wazuh. 29
Ilustración 22. Confirmación creación del perfil OpenVPN. OpenVPN. 30
Ilustración 23. Conexión mediante VPN. OpenVPN. 30
Ilustración 24. Panel de administración Pi-hole. Pi-hole. 31
Ilustración 25. Listado de peticiones de un dispositivo inteligente. Pi-hole 32
Ilustración 26. Proceso escaneo VirusTotal. Wazuh. 33
Ilustración 27. Generación de API key en VirusTotal. VirusTotal. 33
Ilustración 28. Alojamiento archivo EICAR en Nextcloud. Nextcloud. 34
Ilustración 29. Análisis de archivos alojado en Nextcloud. Kibana / Wazuh. 35
Ilustración 30. Configuración FIM en agente Wazuh. Wazuh. 35
Ilustración 31. Ejecución de script. Creación de escenario. Powerhsell. 35
Ilustración 32. Comprobación del funcionamiento del FIM en directorio objetivo. Kibana. 36
Ilustración 33. Simulación ataque Ransomware. Encriptación de archivos. Powershell 36
Ilustración 34. Simulación ataque Ransomware. Comprobación archivos encriptados. Windows 10. 36
Ilustración 35. Simulación ataque Ransomware. Revisión de eventos en FIM. Kibana / Wazuh. 37
Ilustración 36.Intento de validación SSH usuario admin. Putty. 37
Ilustración 37. Simulación ataque fuerza bruta SSH. Wazuh. 38
Ilustración 38. Detección de conexión por VPN. Wazuh. 39
Ilustración 39. Notificación de alerta usuario conectado por VPN. Outlook 2016. 39

Ilustración 40. Notificación de intento de conexión no identificada. Outlook 2016. 40
Ilustración 41. Identificación vulnerabilidades Workstation. Kibana / Wazuh. 41
Ilustración 42. Notificación de alerta nivel 5. Error en la autenticación de Windows. Outlook 2016. 42
Ilustración 43. Notificación de bloqueo de cuenta. Outlook 2016. 43
Ilustración 44. Cuenta BotFather. Telegram. 44
Ilustración 45. Notificación alerta Bot SIEM. Telegram. 45
Ilustración 46. Inventory data report agente. Wazuh. 46
Ilustración 47. Módulo compliance. Kibana / Wazuh. 47
Ilustración 48. Controles RGPD. Fuente: Kibana / Wazuh. 47
Ilustración 49. Eventos asociados al requerimiento IV_35.7. Kibana / Wazuh 48
Ilustración 50. Dashboard GDPR. Kibana / Wazuh. 48
Ilustración 51. Evento de seguridad asociado a art IV 32.3. Kibana / Wazuh. 49
Ilustración 52. Generación de certificados. Raspberry Pi OS. 57
Ilustración 53. Alojamiento de certificados. Raspberry Pi OS. 58
Ilustración 54. Configuración de certificados Filebeat. Raspberry Pi OS. 59
Ilustración 55. Configuración de certificados Kibana 1. Raspberry Pi OS. 60
Ilustración 56. Configuración de certificados Kibana 2. Kibana. 61
Ilustración 57. Generación de contraseñas de acceso Elastic. Elasticsearch. 61
Ilustración 58. Postfix: Instalación por consola. Raspberry Pi OS. 62
Ilustración 59. Postfix: Selección de tipo de correo Postfix. 62
Ilustración 60. Postfix. Especificación del dominio. Postfix. 62
Ilustración 61. Postfix: Configuración de archivo main.cf. Postfix. 63
Ilustración 62. Postfix: Configuración archivo sasl_password y asignación de persmisos. Postfix. 63
Ilustración 63. Postfix. Prueba de envío. Postfix. 63
Ilustración 64. Postfix: Recepción del correo de prueba. Outlook 2016. 63
Ilustración 65. Pi-hole: Inicio del asistente. Pi-hole. 64
Ilustración 66.Pi-hole:Selección interfaz de red. Pi-hole. 64
Ilustración 67.Pi-hole:Inclusión de listas de bloqueo. Pi-hole 65
Ilustración 68. Pi-hole:Selección de protocolos. Pi-hole. 65
Ilustración 69. Pi-hole: Confirmación de la configuración de red. Pi-hole. 66
Ilustración 70. Pi-hole: Notificación de conflicto IP. Pi-hole 66
Ilustración 71. Pi-hole:Confirmación de instalación de interfaz web. Pi-hole 67
Ilustración 72. Pi-hole: Instalación de servidor web. Pi-hole. 67
Ilustración 73.Pi-hole: Confirmación almacenado de logs. Pi-hole 68
Ilustración 74. Pi-hole:Selección del nivel de privacidad. Pi-hole 68
Ilustración 75. Pi-hole: Confirmación de la configuración. Pi-hole. 69
Ilustración 76. Pi-hole: Panel de administración Pi-hole en funcionamiento. Pi-hole 69
Ilustración 77. PiVPN: Instalación de IPtables. PiVPN. 70
Ilustración 78.PiVPN: Información tipo de direccionamiento IP necesario. PiVPN. 70
Ilustración 79.PiVPN:Elección de servicio VPN a instalar. PiVPN. 71
Ilustración 80.PiVPN: Selección de protocolo de conexión. PiVPN. 71
Ilustración 81. PiVPN: Selección de puerto de conexión. PiVPN. 71
Ilustración 82.PiVPN: Detección de servidor DNS Pi-hole. PiVPN. 72
Ilustración 83.PiVPN:Configuración dirección IP Pública . PiVPN. 72
Ilustración 84.PiVPN: Generación Token Duck DNS. Duck DNS. 73

Ilustración 85.PiVPN: Confirmación configuración. PiVPN. 73
Ilustración 86.PiVPN: Configuración para mejora en sobre cifrado RSA. PiVPN. 74
Ilustración 87.PiVPN: Selección del tipo de cifrado pra el certificado a generar. PiVPN. 74
Ilustración 88.PiVPN:Configuración de actualizaciones de seguridad desatendidas. PiVPN. 75
Ilustración 89.PiVPN: Confirmación de instalación. PiVPN. 75
Ilustración 90.PiVPN: Creación de cliente VPN . PiVPN. 76
Ilustración 91. Búsqueda de BotFather. Telegram. 79
Ilustración 92. Información BotFather. Telegram. 80
Ilustración 93. Creación de bot y generación de token. Telegram. 80
Ilustración 94. ID Chat Bot Telegram. Telegram. 81
Ilustración 95. Informe Compliance 1/6. Wazuh. 82
Ilustración 96 Informe Compliance 2/6. Wazuh. 83
Ilustración 97 Informe Compliance 3/6. Wazuh. 84
Ilustración 98. Informe Compliance 4/6. Wazuh. 85
Ilustración 99.Informe Compliance 5/6. Wazuh. 86
Ilustración 100. Informe Compliance 6/6. Wazuh. 87

Índice de tablas

Tabla 1. Funciones SIEM y artículos y controles cubiertos. Elaboración propia. 14
Tabla 2 Estimación del volumen generado por los dispositivos en 90 días. Wazuh. 18

Glosario de acrónimos y términos

Acrónimos

API. Application Programming Interface. Se trata de un conjunto de instrucciones y normas que permiten a aplicaciones creadas por distintos desarrolladores y/o en distintos lenguajes comunicarse entre ellas.

GDPR/ RGPD. General Data Protection Regulation. Se trata del Reglamento Europeo actual que rige en materia de protección de datos.

GUI (*Grafical User Interface*). Se trata de la interfaz gráfica de un programa o aplicación.

IDS. Intrusion Detection System. Es un sistema de detección de intrusiones el cual se emplea, como su propio nombre indica para detectar y controlar accesos no autorizados a determinados sistemas.

IoT. Internet of Things. Término para definir el Internet de las Cosas, es decir, la conexión de múltiples dispositivos que hasta hace un tiempo atrás no disponían de conexión a internet tales como relojes, bombillas o electrodomésticos entre otros.

IPS. Intrusion Protection System. Es un sistema de seguridad que permite el control del acceso a un determinado sistema.

ISO/IEC 27001. Es un estándar internacional encargado de la regulación de seguridad de la información de las organizaciones.

IT. Information Technology. Siglas que hacen referencia a las tecno*logí*as de la información.

JSON. JavaScript Object Notation. Se trata de un formato que permite el intercambio sencillo de datos.

LOPDGDD. Ley Orgánica de Protección de Datos y Garantía de los Derechos Digitales. Se trata de la actual normativa en protección de datos y que, junto el RGPD, conforman el marco legislativo en materia de protección de datos que rige en España.

PCI-DSS. Payment Card Industry Data Security Standard. Se trata del marco de control que sirve de referencia para la industria de los medios de pago.

SIEM. System Information and Event Management. Se trata de un sistema que permite la recopilación y el tratamiento de eventos de seguridad que se producen en un determinado entorno.

SOC. Security Operations Center. Es un departamento encargado de la monitorización y mejora continua del nivel de madurez de seguridad de una determinada organización.

SSL. **Secure Socket Layer**. Protocolo empleado para la creación, por ejemplo, de certificados de seguridad.

NIST. National Institute of Standard and Techho*log*y. Se trata de un marco de gobierno de ciberseguridad muy extendido entre empresas privadas de Estados Unidos.

TFM. Trabajo Fin de Máster.

VPN. Virtual Private Network o Red Virtual Privada.

Términos

Agente. Se trata del software instalado en un sistema determinado y que permite el envío de información a un nodo central que se encarga de recopilar y tratar la información suministrada periódicamente por éste.

Amenaza. Es la posibilidad de ocurrencia de cualquier tipo de evento o acción que puede producir un daño (material o inmaterial) sobre los elementos de un sistema.

Bot. Es un programa que permite la realización de tareas de forma automatizada a través del desarrollo de funciones previamente establecidas**.**

Correlación. Es una técnica que permite enlazar varios eventos con el fin de identificar ciertos patrones de comportamiento que permitan

Evento de seguridad. Suceso que informa de una situación relevante para la seguridad de un determinado sistema.

Log. Es un archivo en el que se registra información acerca de eventos ocurridos en un determinado proceso o aplicación y que se almacena de forma sistematizada.

Malware. Es una aplicación que contiene código malicioso y que tiene como fin causar daños en el dispositivo en el que se instala.

Open Source. De código abierto. Es un modelo de desarrollo de software basado en la colaboración abierta. Se enfoca más en los beneficios prácticos que en cuestiones éticas o de libertad que tanto se destacan en el software libre.

Riesgo. Es la probabilidad de que una vulnerabilidad sea explotada en el sistema.

Ransomware. Tipo de amenaza a través de la cual el atacante secuestra los datos (ransom en inglés) para pedir un rescate por la liberación de los mismos.

Vulnerabilidad. Se trata de un fallo en un dispositivo que permite violar el sistema de seguridad.

1. Introducción

1.1. Contexto y justificación del Trabajo

El creciente e intensivo uso de las nuevas tecnol*ogí*as junto con la irrupción del Internet de las Cosas – *Internet of Things, IoT* - ha conllevado un aumento considerable del número de dispositivos conectados a la red ha ocasionado que la superficie de exposición ante ciberamenazas haya incrementado en consonancia.

El IoT ha supuesto una revolución en muchos sentidos, pero sobre todo en la dotación de una conexión a internet de dispositivos convencionales. ¿Podíamos imaginar hace 15 años que íbamos a estar dándole órdenes a las bombillas para que se apagaran? ¿o al aspirador? Las funcionalidades que nos brindan en muchos casos no llevan asociadas unas medidas de seguridad adecuadas. Esta situación actúa en contra de nuestra privacidad y la de un activo muy valioso: la información.

A este nuevo paradigma debemos añadir todos aquellos equipos que ya disponían de una conexión a la red. Hablamos de estaciones de trabajo, servidores, routers o equipos de impresión vulnerables debido a no conocer el estado de seguridad de los mismos, así como desconocer los diversos vectores de amenazas a los que se encuentran expuestos.

Esta situación es de sobra conocida por los ciberdelincuentes los cuales han ampliado sus *targets* siendo cada vez más comunes los ataques a usuarios y a pequeñas o medianas instituciones. *"¿Por qué me atacan a mí si no tengo nada de valor?"* Seguramente hayamos oído esta cuestión en más de una ocasión. Los atacantes son perfectamente conocedores de la ausencia de unas medidas de seguridad mínimas en entornos en las que la ciberseguridad relegada a un segundo plano.

La ausencia de una dotación de recursos destinados a implantar -o mejorar- las medidas de seguridad en el caso de las empresas suponen un factor de riesgo importante pues en juego está, en muchos casos, la viabilidad de la empresa. Cada vez es más frecuente encontrarse con casos en los que compañías se ven obligadas a paralizar su proceso productivo por un ataque Ransomware por ejemplo, recibir multas por no cumplir con normativas como son las de Protección de Datos o lo que es más grave a cerrar debido a los daños causados por ataques cibernéticos.

Este proyecto pretender aportar soluciones que permitan la mejora del nivel de seguridad de un determinado entorno. Uno de los principales problemas ante el que nos encontramos es el desconocimiento de qué sucede en nuestra red y en nuestros dispositivos. ¿Mejoraría nuestra seguridad si pudiéramos analizar el tráfico de nuestra red? ¿y si detectáramos vulnerabilidades en el sistema operativo de varios dispositivos? ¿y si automáticamente se bloquearan intentos maliciosos de conexión? Si tuviéramos información clave, ¿podríamos proteger mejor nuestros activos? La respuesta a todas

estas cuestiones está en el despliegue de soluciones de fácil y rápida implantación a través del empleo de minicomputadores desde la perspectiva del *Edge Computing* [1].

Tras definir las principales amenazas se definirá uno de los servicios más relevante de las empresas que se dedican a la ciberseguridad, un Centro de Operaciones de Seguridad (*Security Operation Center, SOC*) y más concretamente la herramienta sobre la que giran sus procesos: el SIEM.

Mediante la implantación de un SIEM se podrá obtener un mayor conocimiento de lo que sucede en una determinada red, detectar vulnerabilidades y realizar acciones para evitar de detección, prevención y respuesta rápida que eviten las amenazas exploten vulnerabilidades asociadas a cualquiera de los actores intervinientes.

Este sistema también permitirá, gracias a las diversas funcionalidades que lo componen, resolver la problemática del tratamiento de la diversidad de *log*s proporcionados por los distintos complementos de los que la solución será dotada.

El proyecto parte de la premisa de encontrar una solución base de código abierto que se pueda desplegar sobre un hardware *lowcost* y que sea escalable y fácilmente replicable.

Se definirán como continuación a este proyecto nuevas posibilidades, que, debido al alcance de este proyecto, no ha sido posible abordar.

[1] Es un paradigma de computación distribuida que acerca computación y almacenamiento de datos a la ubicación donde se necesita, para mejorar los tiempos de respuesta y ahorrar ancho de banda.

1.2. Objetivos del Trabajo

Dentro de este trabajo se pretende dar respuesta a las siguientes cuestiones:

Realizar un acercamiento a la problemática de seguridad a la que nos encontramos en las redes y cómo, a través de la tecnología *Edge Computing*, dar respuesta a ésta a través de la validación de soluciones de seguridad.

Definir e identificar uno de los componentes principales de un Centro de Operaciones de Seguridad (SOC), el SIEM. A través de esta herramienta se pretende, mediante el desarrollo de una prueba de concepto, abordar las siguientes cuestiones:

- Validar su utilidad en una red local y comprobar qué problemática trata de solventar.
- Comprobar si es viable desplegarlo de hardware lowcost, concretamente sobre un minicomputador (Raspberry Pi 4), así como sus posibilidades de escalabilidad.
- Permite solventar la problemática de los *logs*.
- Monitorizar la actividad de dispositivos.
- La integración de fuentes de herramientas complementarias como una Red Virtual Privada (VPN) con el que permita acceder a una red.
- ¿Se puede analizar y bloquear el tráfico de red?
- Disponer de un panel para conocer qué sucede, en tiempo real, en una red.
- Soporte para la definición a la definición de un cuadro de mandos de ciberseguridad.
- Permite este desarrollo cumplir con aspectos regulatorios en materia de seguridad de la información (RGPD, LOPDGDD, ISO 27K, NIST, PCI-DSS).
- Verificar si permite prevenir y detectar amenazas que se propagan a través de las redes.
- Comprobar las posibilidades de integración de Wazuh con terceros a través de APIs.

Por último, se pretende una serie de líneas de trabajo futuro que permitan ampliar el proyecto base desarrollado.

1.3. Enfoque y metodología seguida

Para dar respuesta tanto a los objetivos planteados se realizará una aproximación para conocer la problemática y las amenazas más conocidas de la red, y se realizará un acercamiento al Centro de Operaciones de Seguridad y su componente principal, el SIEM.

Se avanzará hacia un contexto más específico y práctica en el que se desarrollará una implantación de un sistema base mediante el estudio y la configuración de los distintos componentes que integrarán el proyecto.

Para ello se emplearán técnicas de investigación tanto cuantitativas como cualitativas que nos permitan aterrizar nuestro estudio. Tras ello se pondrán a disposición de los interesados las herramientas empleadas y se definirá un aprueba de concepto a través de la cual se abordarán la problemática planteada y servirá, además, para ver el potencial de las herramientas empleadas.

1.4. Definición y planificación de las tareas

Se detallan, a continuación, las principales tareas que componen el proyecto, así como un calendario en el que se muestran los plazos para la realización de las mismas. Para el cálculo se ha tenido en cuenta el número total de créditos y su equivalencia en horas siendo ésta del ratio 1 crédito = 25 horas.

El proyecto llevará un total de 300 horas. Teniendo en cuenta este dimensionamiento se establecerá una jornada de 4 horas / día. Las tareas a desarrollar son las siguientes.

1. PLANIFICACIÓN DEL PROYECTO

- Revisión guías proporcionadas
- Definición de la problemática a resolver
- Definición de los objetivos a alcanzar
- Elección de la metodología a seguir
- Definición de tareas a realizar
- Planificación temporal del proyecto
- Revisión del estado del arte

Entrega 1

2. INVESTIGACIÓN

- Definición de la problemática de seguridad en dispositivos. Identificación de amenazas
- Acercamiento al Edge Computing
- Qué es un SOC. Definiciones y componentes
- SIEM. Definición y características
- Establecimiento de los requisitos de HW para acometer el desarrollo
- Diseño de la solución y aprovisionamiento

Entrega 2

3. IMPLANTACIÓN DE LOS SISTEMAS

- Implantación de los componentes
- Instalación y Configuración del SO base
- SIEM: Wazuh
- Configuración del servidor Wazuh
- Instalación de clientes Wazuh
- VPN: OpenVPN
- Configuración del servidor
- Instalación de cliente
- Bloqueador de contenidos y servidor DNS
- Configuración base
- Recopilación y procesamiento de *logs*
- Búsqueda y analítica
- Presentación gráfica de eventos

Entrega 3

4. DESARROLLO DE LA PRUEBA DE CONCEPTO

- Detección de archivo malicioso alojado en cloud privada
- Prevención y detección ante un ataque de Ransomware
- Detección de ataque de fuerza bruta por SSH
- Detección de conexiones a la VPN desplegada
- Detección de vulnerabilidades
- Notificación de alertas vía correo electrónico
- Notificación de alertas a través de bot de Telegram
- Generación de informes
- Cumplimiento normativo: GDPR/RGPD

5. CONCLUSIONES

- Conclusiones

6. LÍNEAS DE TRABAJO FUTURO

- Líneas de trabajo futuro

Entrega 4

7. ENTREGA FINAL

- Entrega de documento
- Presentación audiovisual
- Resolución de cuestiones

La planificación se detalla a continuación.

Id	Nombre de tarea	Duración	Comienzo	Fin
1	**1. PLANIFICACIÓN DEL PROYECTO**			
2	Revisión guías proporcionadas	9 horas	mié 16/09/20	jue 17/09/20
3	Definición de la problemática a resolver	16 horas	jue 17/09/20	lun 21/09/20
4	Definición de los objetivos a alcanzar	8 horas	lun 21/09/20	mar 22/09/20
5	Elección de la metodología a seguir	11 horas	mar 22/09/20	mié 23/09/20
6	Definición de tareas a realizar	9 horas	jue 24/09/20	vie 25/09/20
7	Planificación temporal del proyecto	7 horas	vie 25/09/20	vie 25/09/20
8	Revisión del estado del arte	9 horas	lun 28/09/20	mar 29/09/20
9	***Entrega 1***	1 hora	mar 29/09/20	mar 29/09/20
10	**2. INVESTIGACIÓN**			
11	Definición de la problemática de seguridad en dispositivos. Identificación de amenazas	10 horas	jue 01/10/20	vie 02/10/20
12	Acercamiento al Edge Computing	8 horas	lun 05/10/20	lun 05/10/20
13	Qué es un SOC. Definiciones y componentes	11 horas	jue 08/10/20	vie 09/10/20
14	SIEM. Definición y características	10 horas	mié 14/10/20	jue 15/10/20
15	Establecimiento de los requisitos de HW para acometer el desarrollo	11 horas	lun 19/10/20	mar 20/10/20
16	Diseño de la solución	15 horas	mié 21/10/20	jue 22/10/20
17	Aprovisionamiento	10 horas	vie 23/10/20	lun 26/10/20
18	***Entrega 2***	1 hora	lun 26/10/20	lun 26/10/20
19	**3. IMPLANTACIÓN DE LOS SISTEMAS**			
20	Implantación de los componentes	6 horas	mié 28/10/20	mié 28/10/20
21	Instalación y Configuración del SO base	5 horas	mié 28/10/20	jue 29/10/20
22	SIEM: Wazuh	7 horas	jue 29/10/20	vie 30/10/20
23	Configuración del servidor Wazuh	8 horas	lun 02/11/20	lun 02/11/20
24	Instalación de clientes Wazuh	5 horas	mié 04/11/20	mié 04/11/20
25	VPN: openVPN	6 horas	jue 05/11/20	jue 05/11/20
26	Configuración del servidor	3 horas	jue 05/11/20	vie 06/11/20
27	Instalación de cliente	3 horas	lun 09/11/20	lun 09/11/20
28	Bloqueador de contenidos y servidor DNS	4 horas	lun 09/11/20	lun 09/11/20
29	Configuración base	6 horas	mar 10/11/20	mar 10/11/20
30	Recopilación y procesamiento de logs	5 horas	lun 16/11/20	lun 16/11/20
31	Búsqueda y analítica	8 horas	mar 17/11/20	mar 17/11/20
32	Presentación gráfica de eventos	6 horas	mié 18/11/20	mié 18/11/20
33	***Entrega 3***	1 hora	mié 18/11/20	mié 18/11/20
34	**4. DESARROLLO DE LA PRUEBA DE CONCEPTO**			
35	Detección de archivo malicioso alojado en cloud privada	7 horas	mié 25/11/20	mié 25/11/20
36	Prevención y detección ante un ataque de Ransomware	7 horas	jue 26/11/20	jue 26/11/20
37	Detección de ataque de fuerza bruta por SSH	6 horas	vie 27/11/20	vie 27/11/20
38	Detección de conexiones a la VPN desplegada	6 horas	lun 30/11/20	lun 30/11/20
39	Detección de vulnerabilidades	8 horas	mar 01/12/20	mar 01/12/20
40	Notificación de alertas vía correo electrónico	5 horas	mié 02/12/20	mié 02/12/20
41	Notificación de alertas a través de bot de Telegram	5 horas	lun 14/12/20	lun 14/12/20
42	Generación de informes	9 horas	mar 15/12/20	mié 16/12/20
43	Cumplimiento normativo : GDPR/RGPD	7 horas	vie 18/12/20	vie 18/12/20
44	**5. CONCLUSIONES**			
45	Conclusiones	4 horas	mié 23/12/20	mié 23/12/20
46	**6. LÍNEAS DE TRABAJO FUTURO**			
47	Líneas de trabajo futuro	3 horas	lun 28/12/20	lun 28/12/20
48	***Entrega 4***	1 hora	lun 28/12/20	lun 28/12/20
49	**7. ENTREGA FINAL**			
50	Entrega de documento	1 hora	mar 29/12/20	mar 29/12/20
51	Presentación audiovisual	2 horas	lun 04/01/21	lun 04/01/21
52	Resolución de cuestiones	2 horas	vie 15/01/21	vie 15/01/21

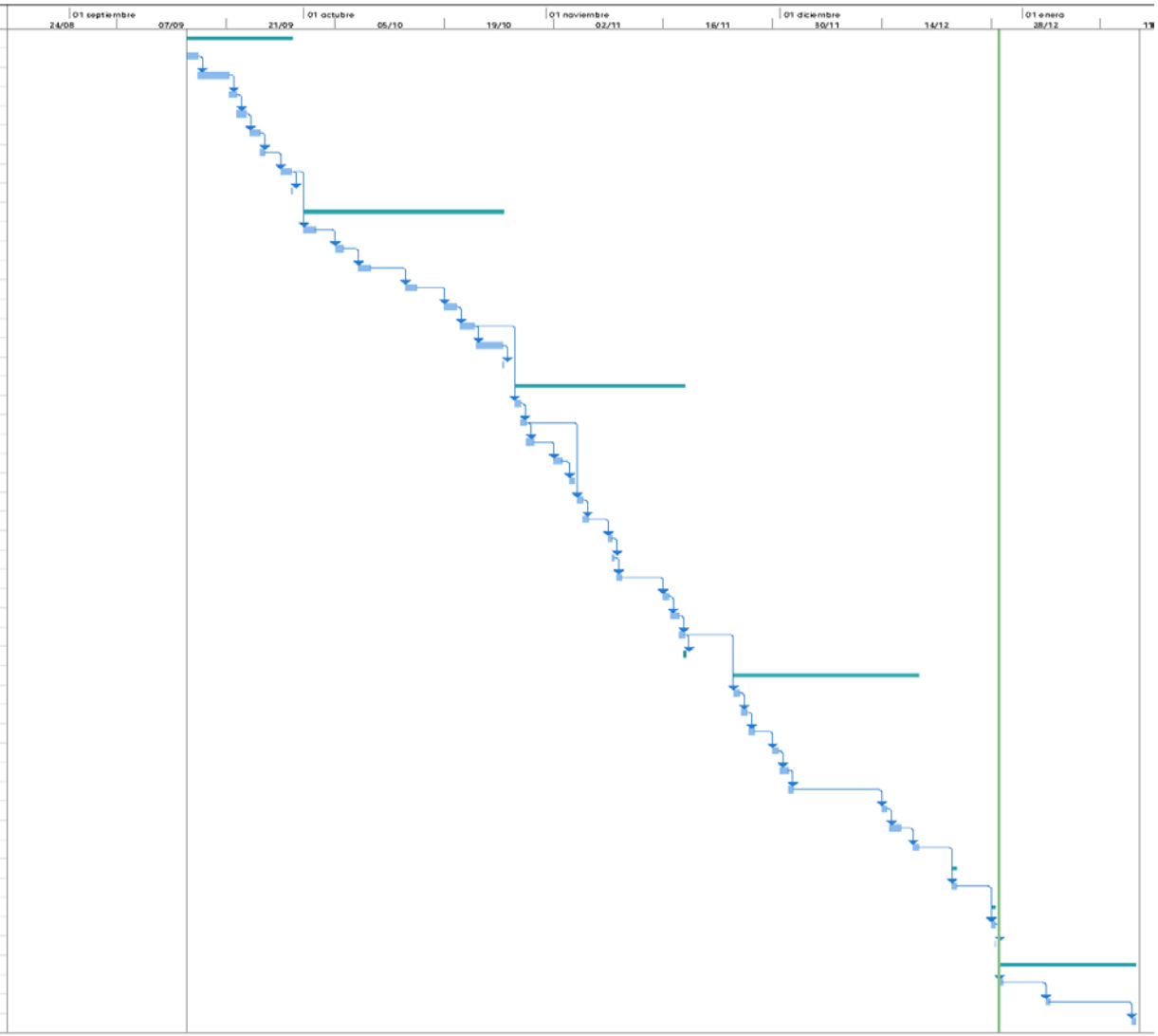


Ilustración 1. Planificación del proyecto. Elaboración propia.

1.5. Estado del arte

El análisis del estado del arte presentado aborda la problemática asociada a la interconexión de los dispositivos que ha visto agravada su situación con la llegada del Internet de las Cosas (IoT).

La ciberdelincuencia se ha convertido en los últimos años como uno de los negocios ilícitos más rentables superando, por ejemplo, al narcotráfico. Según un informe publicado por McAfee en colaboración con el CSIS (Centro de Estudios Estratégicos e Internacionales) el cibercrimen cuesta, a nivel global, alrededor de 600.000 millones de dólares al año, es decir, lo equivalente al 0,8% del PIB global.

Gran parte de estos eventos se materializan debido a la ausencia de medidas de seguridad en las redes que utilizamos tanto a nivel particular como empresarial. En el caso de estos últimos por la ausencia de una partida presupuestaria que asegure la implantación de medidas de seguridad que permitan actuar de forma proactiva. Aquí entran juego los minicomputadores como Raspberry Pi que permiten el despliegue de SIEM ayudan a mejorar el nivel de seguridad de nuestras redes y dispositivos gracias a sus interesantes capacidades.

Bhatt et al. (2012) "*define un Centro de Operaciones de Seguridad (SOC) como unidad centralizada de la supervisión en tiempo real y la identificación de incidentes de seguridad. Los sistemas de gestión de eventos e información de seguridad (SIEM) son una herramienta importante utilizada en los SOC; recopilan eventos de seguridad de diversas fuentes en las redes empresariales, normalizan los eventos a un formato común, almacenan los eventos normalizados para análisis forenses y correlacionan los eventos para identificar actividades maliciosas en tiempo real*".

El SIEM es un buen punto de partida para mejorar el nivel de madurez de seguridad de una organización o de, yendo a ámbitos más personales, un hogar en el cual, adaptándolo a las necesidades del usuario es una opción a tener en cuenta para prevenir y detectar amenazas.

A través del despliegue del sistema y de la definición de la prueba de concepto se pondrán de manifiesto todas aquellas carencias que, gracias a estas herramientas.

Se creará un modelo base, *lowcost* y perfectamente replicable que nos permita conocer qué pasa en nuestra red, cuántas vulnerabilidades se encuentran en los dispositivos, analizar y tratar vulnerabilidades, ayudar a cumplir con normativas de seguridad de la información tales como RGPD / LOPDGDD, PCI-DSS, NIST o ISO 27000 entre otros.

Con respecto al cumplimiento del compliance, Montesino et al. (2012) afirma que gracias a este tipo de herramientas la automatización de los controles permite llegar hasta el 30% del total. Una cifra interesante.

1.6. Descripción de los recursos necesarios

A lo largo del presente proyecto se mostrarán los recursos empleados para dar solución a la problemática planteada.

Con el fin de hacer accesible la replicación del proyecto y de comprobar la versatilidad de los sistemas empleados emplearemos soluciones Open Source para que sea asequible a los distintos grupos de interés.

Los componentes principales utilizados son los siguientes:

Hardware

- Ordenador portátil MSI PS42 Modern (I7, 16 GB RAM y 512 GB de almacenamiento).
- Raspberry Pi 4 Modelo B 8 GB.
- Adaptador de corriente usb C 5V 3 A.
- Tarjeta microsd Sandisk 64 gb.
- Router Asus AC56u.
- HP Proliant Microserver G8 200 gb de almacenamiento y 16 gb de RAM.

Software

Dentro de las principales aplicaciones que emplearemos se encuentran:

Wazuh. Sistema de detección de intrusos SIEM	Elastic ELK. Motor de analíticas y análisis distribuido.	Debian / Raspbian 10 para el sistema operativo base.
OpenVPN para crear red virtual privada. Virtual Private Network (VPN).	Bloqueador de anuncios, rastreadores y controlador de tráfico en red.	Herramienta para creación de máquinas virtuales.

2. Análisis

Para este apartado se realizarán diversas aproximaciones a conceptos que nos ayudarán a contextualizar y definr el alcance del trabajo.

2.1. Amenazas en la red

Mokalled et al. (2019) afirma que:

> la tecnología de la información y la comunicación (TIC) ha tenido un impacto notable en la sociedad. Las empresas hoy en día dependen de la tecnología de la información y la comunicación que pone sus activos bajo ciertos riesgos, especialmente los cibernéticos, por lo que deben mantenerse bajo control por medio de contramedidas de seguridad que generan confianza en el uso de estos bienes. Las empresas de todo el mundo necesitan asegurar activos valiosos, una operación comercial ininterrumpida (procesos), datos fiables y calidad de servicio (QoS) a diversos grupos de usuarios. Necesitan proteger a sus clientes y empleados tanto dentro como fuera la organización. Según Gartner, para 2020, el 30% de las empresas mundiales habrá sido directamente comprometida por un grupo independiente de ciberdelincuentes o activistas cibernéticos. Además, en el 60% de las violaciones de la red, los hackers comprometen la red en cuestión de minutos.

La European Union Agency for Cybersecurity (ENISA), organismo de referencia europeo en materia de ciberseguridad, ha publicado recientemente un reporte en el que muestra las tendencias de este año en cuanto a amenazas en la red se refiere.

Ilustración 2. Threat Landscape 2020 - List of top 15 threats. ENISA.

Algunos de ellos serán abordados a lo largo del trabajo para conocer con un nivel de detalle mayor cómo funcionan y cómo afectan a nuestros dispositivos.

2.2. Introducción al *Edge Computing*

En los últimos años, ha incrementado notablemente el número de datos generados por el IoT (Premsankar, G., et. al, 2018). Cada vez son más los dispositivos utilizados en nuestro día a día con motivo de mejorar nuestra comodidad, como es el caso de los altavoces inteligentes que hacen efectivas órdenes mediante el IoT, o la ejecución de acciones de forma remota, como sería el hecho de encender el robot aspirador que tenemos en casa desde el trabajo. Ante este auge, y los problemas que le supone al tradicional y convencional *Cloud Computing*, surge el *Edge Computing*, el cual presenta menores latencias y consumo energético, mayor eficiencia espectral y densidad de dispositivos, y, en definitiva, una red mucho más ágil (Millán, 2017).

El *Edge Computing*, o computación de borde de la red, consiste en un concepto novedoso para entornos de IoT que supone un complemento al *cloud computing* consistente en el que el procesamiento de los datos se realiza de forma descentralizada, en tiempo real y de forma más próxima a los dispositivos y/o sensores que generan dichos datos.

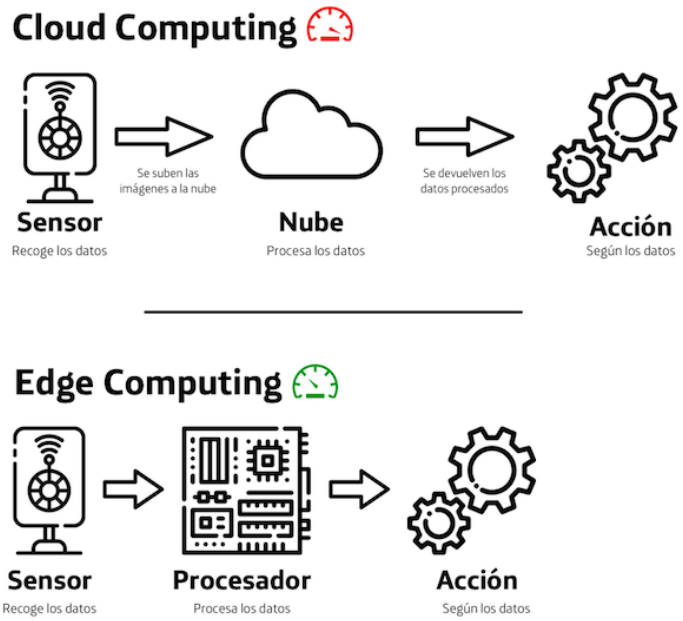


Ilustración 3. Cloud Computing vs Edge Computing. ThinkBig.

Uno de los factores fundamentales y ventajas que esta tecnología pone a nuestra disposición, la baja latencia. En casos como puede ser el de un coche autónomo la velocidad en la recepción y el procesamiento de la información es vital en la toma de decisiones que pueden llegar a costar vidas. Extrapolándolo a nuestro ámbito de actuación los tiempos de respuesta ante incidentes de seguridad en muchos casos son críticos para minorar el impacto que ataques como el Ransomware provocan.

Para este proyecto se desarrollará un sistema en el extremo partiendo desde la perspectiva de la seguridad que ayude a procesar el volumen de información que generan los dispositivos que tenemos conectados a una red.

2.3. Soluciones de seguridad para tratar la problemática estudiada

El incremento de los ciberataques es un hecho constatable ya que todos estamos en el punto de mira debido a que no hemos implementado una estrategia y los componentes de seguridad para las redes de comunicación.

Dekten (2015), afirma que:

> Tablets, los teléfonos inteligentes y los netbooks cambiaron rápidamente los requisitos de seguridad de la tecnología de la información. Hoy en día, existen varios componentes de seguridad (por ejemplo, sistema antivirus, cortafuegos y sistema de detección de intrusos) disponibles para proteger las redes. Lamentablemente, funcionan de forma independiente, es decir, aislados. Pero muchos ataques sólo pueden reconocerse si se combinan y correlacionan entre sí los registros y eventos de los diferentes componentes de seguridad. Esta posibilidad la ofrece un Sistema de Gestión de Información y Eventos de Seguridad (SIEM).

En los siguientes capítulos se pondrá de manifiesto las posibilidades que ofrecen este tipo de soluciones.

2.4. ¿Qué es un SOC?

Antes de realizar con más nivel de detalle qué es un SIEM es necesario destacar de que estos sistemas dentro del ámbito empresarial están integrados dentro de un ***Centro de Operaciones de Seguridad* (*Security Operations Center,* SOC). Un SOC** es un departamento encargado de la monitorización y mejora continua del nivel de madurez de seguridad de una determinada organización.

Este departamento está conformado por personas con un alto conocimiento en materia de seguridad de la información y ramas afines a ésta. Disponen de herramientas que permiten la detección, el análisis y la respuesta frente a incidentes que se puedan producir en los sistemas en los cuales están implantadas.

Es habitual que la información generada en los distintos casos encontrados permita la creación de **casos de uso** los cuales, mediante el desarrollo de procedimientos, ser uno de los puntos fundamentales de estos servicios.

2.4.1. ¿Qué funciones tiene un SOC?

Entre las funciones más relevantes de un Servicio de Operaciones de Seguridad podemos encontrar:

- ✓ Gestión de amenazas y vulnerabilidades.
- ✓ Monitorización y auditoría de seguridad para la prevención y detección de amenazas.
- ✓ Mantenimiento y organización de dispositivos de seguridad.

- ✓ Gestión de incidentes de ciberseguridad para análisis, mitigación y respuesta ante los riesgos.
- ✓ Apoyo al cumplimiento en materia de normativa de seguridad.
- ✓ Creación de casos y generación de inteligencia de servicio.

2.4.2. Arquitectura o componentes de un SOC

Múltiples son las fuentes que podríamos citar en para identificar un modelo de arquitectura válida para describir un SOC. Por operatividad, Van Os (2016), define un modelo en el que desglosa un SOC en cinco dominios principales 20 aspectos dependientes de éstos para recoger de forma organizada la estructura de este tipo de servicios.

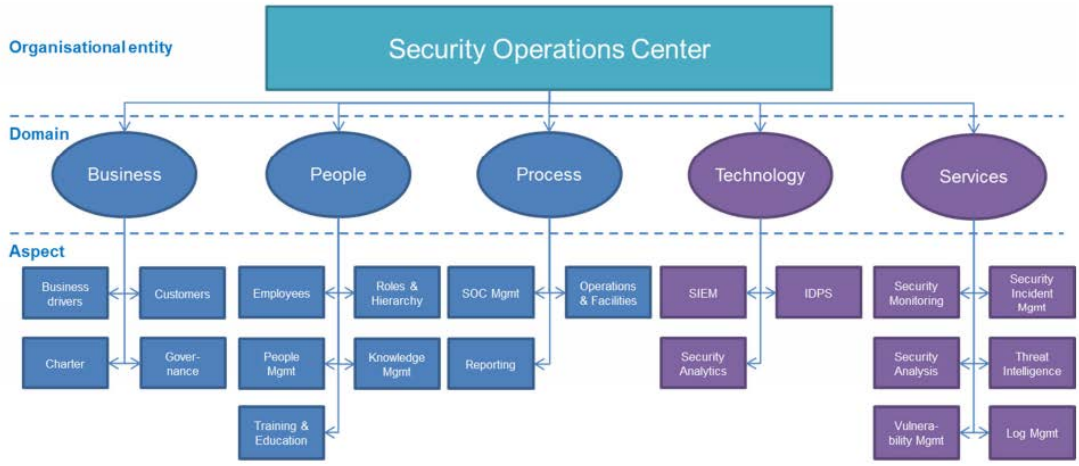


Ilustración 4: Modelo de organización de un SOC. Van Os.

- **Negocio**. Enfocado desde el punto de vista del negocio en el que se reportará información a los diferentes grupos de interés y a los requerimientos de estos

- **Personas**. Basado tanto en los componentes que integran el SOC, así como su conocimiento, formación y roles dentro del equipo de trabajo.

- **Procesos**. Bajo este dominio se abarcan las cuestiones relativas la gestión del propio servicio, reporte y operaciones del día a día.

- **Tecnología**. Engloban los sistemas que conforman el SOC entre los que podemos destacar el SIEM (objeto de este estudio) IDPS (*Intrusion Detection and Prevention System*) y el software de analítica de seguridad.

- **Servicios**. Actividades entre las que destacan la monitorización global y la analítica de seguridad, la gestión de los *log*s y de las vulnerabilidades.

2.5. SIEM. Concepto, componentes y funcionalidades

El término SIEM (**Security Information and Event Management**), fue asignado por Mark Nicolett y Amrit Williams of Gartner in 2005 , y describe las posbilidades de este sistema para recopilar, analizar y presentar información de una red y de los dispositivos de seguridad.

Esta tecnología nace al combinar por un lado la **Gestión de Eventos de Seguridad** (*Security Event Manager* – SEM) con la **Gestión de Seguridad de la Información** (*Security Information Managemen*t – SIM).

El componente SEM permite la centralización del almacenaje de los datos para su análisis en tiempo real mientras que el SIM actúa a largo plazo mediante la recopilación, en un repositorio central, de toda la información recabada durante un determinado tiempo para su posterior análisis.

2.6. ¿Por qué es importante contar con un SIEM?

Como hemos podido apreciar en el punto anterior son varios los componentes que forman un SOC y pese a que todos son importantes será el SIEM el objeto de estudio en este ensayo. Este elemento nos va a permitir, como veremos más adelante abarcar varios de los servicios planteados entre los que destacan:

- Gestión de amenazas y vulnerabilidades que afectan a los sistemas.
- Monitorización y auditoría de seguridad en seguridad en redes y cloud para detectar amenazas ya sean externas o internas.
- Centralización de amenazas potenciales.
- Analítica y gestión de *log*s.
- Detección de vulnerabilidades.
- Respuesta ante incidentes.
- Detección de intrusiones.
- Cumplimiento normativo.

2.7. ¿Cómo puede un SIEM ayudar a una organización a cumplir con aspectos regulatorios como el GDPR, ISO 27001?

Filkins (2017), a través de su contribución en el SANS (SysAdmin Audit, Networking and Security Institute), señala como la principal fuente los *log*s recopilados por el SIEM para proporcionar evidencias que puedan ser válidas para una auditoría o para la presentación de las mismas ante una posible brecha de seguridad.

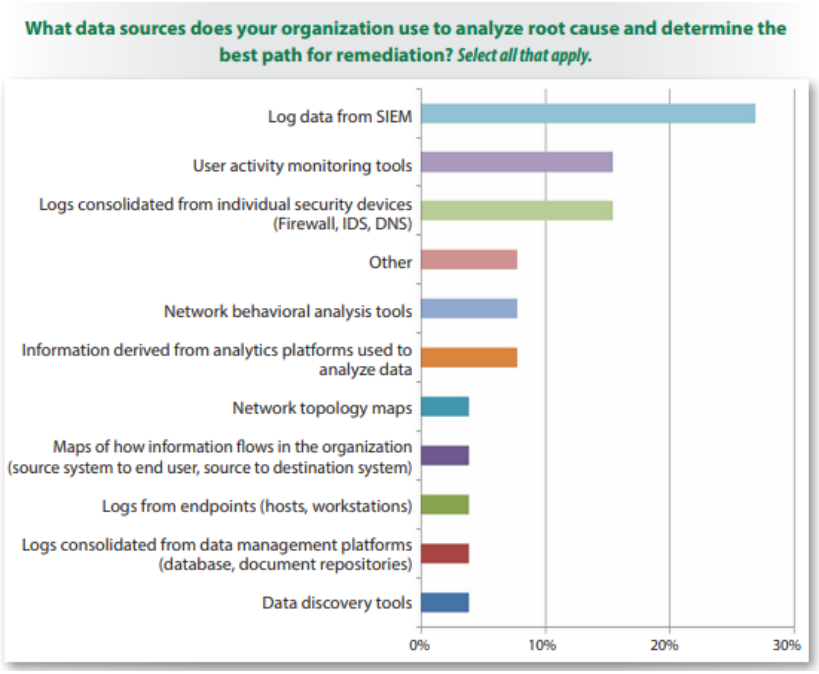


Ilustración 5. Root causes to determine best remediations. Filkins.

Este tipo de sistemas permiten a las organizaciones cumplir de forma parcial o total con los requerimientos que normativas como el RGPD[2] o la LOPDGDD[3] o estándares de seguridad de la Información como ISO 27001 exigen. En la siguiente tabla podemos ver cuáles son los aspectos que este tipo de sistemas mapean.

Función SIEM	art. RGPD	Control ISO 27001: 2013
Procesamiento de datos de carácter personal	4, 14.2	18.1.4
Borrado de datos de carácter personal de forma permanente	17	18.1.4
Notificación de los eventos de Seguridad de la información	12, 22	16.1.2
Inventariado de activos	30	8.1.1
Uso aceptable de los equipos	40	8.1.3
Revisión del control de acceso	32.1.d	9.1.1
Registro de eventos	32.2	12.4.1
Control de eventos en la red	32.1.d	13.1.1
Respuesta ante incidentes	33.3.d	16.1.5
Aprendizaje de los incidentes de Seguridad de la información	83	16.1.6
Recopilación de evidencias	82	16.1.7
Evaluación y decisión sobre eventos de Seguridad de la información	82	16.1.4
Control de vulnerabilidades técnicas	32.1.d	12.6.1

Tabla 1. Funciones SIEM y artículos y controles cubiertos. Elaboración propia.

[2] Reglamento General de Protección de Datos (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016.
[3] Ley Orgánica de Protección de Datos y Garantía de los Derechos Digitales, (LO 3/2018 de 8 de diciembre).

Este tipo de herramientas, como afirman Montesinos et al. (2012) ayudan a que "alrededor del 30% de los controles de seguridad de la información pueden automatizarse y agruparse en una lista de diez controles de seguridad automatizables. Se puede utilizar un marco basado en el SIEM para la gestión centralizada e integrada de los diez controles de seguridad automatizables".

Bajo mi experiencia como auditor de protección de datos he podido constatar a lo largo de los últimos años como gran cantidad de organizaciones, principalmente pequeñas y medianas, no cuentan con suficientes medidas de seguridad para dar cumplimiento a ciertos aspectos críticos de la vigente normativa de protección de datos.

Consideramos que el tipo de solución que se propone está al alcance de todas las organizaciones en cuanto a costes de implantación y mantenimiento se refiere y permite garantizar, en un alto grado, los pilares básicos de seguridad de la información.

En la prueba de concepto (*Proof of Concept* - PoC) que llevaremos a cabo pondremos de manifiesto cómo este tipo de sistemas nos ayuda a dar cumplimiento a la vigente normativa europea en materia de protección de datos.

3. Desarrollo

Tras comprobar cómo un SIEM ayuda a la seguridad de una red procederemos a realizar el despliegue de la herramienta Wazuh, así como la definición de los casos de uso para analizar las principales funcionalidades.

3.1. Wazuh

Teniendo como objetivo el cumplimiento de la premisa de solución Open Source hemos elegido Wazuh, que cuenta con una gran comunidad y unas posibilidades de escalabilidad bastante grandes.

Como características más relevantes destacamos:

- **Analítica de seguridad**. Gracias al tratamiento de evento y la correlación de los mismos, generación de auditorías, etc.
- **Detección de intrusos (IDS**). Cuenta con una gran colección de reglas predefinidas que permiten controlar, por ejemplo, los intentos de conexión a un determinado sistema.
- **Analítica de *logs***. Permite la detección, mediante reglas, de divergencias en los sucesos ocurridos en un determinado sistema.
- **Monitorización de la integridad de ficheros**. Comprueba y alerta de posibles alteraciones en directorios y ficheros.
- **Detección de vulnerabilidades**. Cuenta con un módulo encargado de detectar versiones obsoletas en aplicaciones que pueden ser comprometidos. Eta información es correlada con el MITRE.
- **Respuesta ante incidentes**. Mediante la creación de alertas y ejecución de scripts pueden hacer frente a las amenazas.

- **Evaluación de la configuración de los equipos.** Realizando auditorías a los equipos que están bajo su dominio.
- **Cumplimiento normativo**. Dan respuesta a cuestiones técnicas y que son reguladas por normativas de seguridad de la información.
- **Seguridad en la nube**. Monitorizando servidores de Amazon (AWS) por ejemplo.
- **Seguridad en contenedores**. Monitorización específica para Kubernetes de Docker.

Como complemento a esta herramienta contaremos con la suite de Elastic para el tratamiento de la información, la presentación y el tratamiento de la misma y que abordaremos en los siguientes puntos.

3.2. Diseño de la solución

A partir de la siguiente ilustración podemos encontrar los agentes que intervienen en el sistema que pretendemos implementar.

Agente Wazuh. Se trata de la aplicación que, siendo instalada en nuestros equipos (pcs, servidores, etc…), recopilará la información necesaria para que posteriormente el SIEM. La procese.

Servidor Wazuh. Se trata del core de la aplicación el cual recibirá y procesará la información obtenida por parte de los diferentes agentes desplegados y que servirá para realizar el análisis de amenazas, la detección de las vulnerabilidades y el análisis compliance.

Pila Elastic (ELK). Recibirá los datos que previamente ha tratado el servidor Wazuh y se encargará del procesamiento de los mismos, el envío de alertas y la presentación de los datos que se han obtenido por los diferentes sistemas.

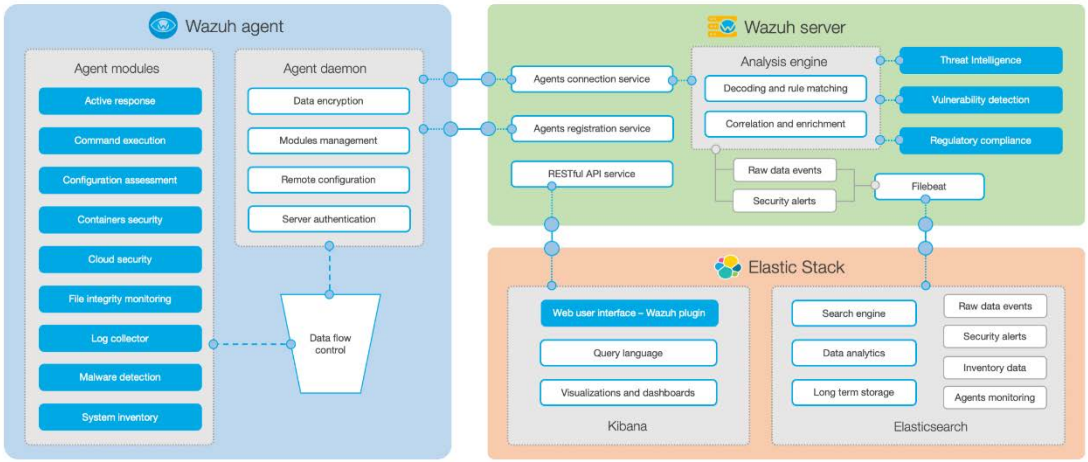


Ilustración 6. Esquema del funcionamiento de los sistemas a implementar. Wazuh.

3.2.1. Requisitos necesarios para llevar a cabo el despliegue

El caso que a continuación mostrado está pensado para un entorno de producción pequeño, de entorno a unos 100 agentes conectados. Como se puede apreciar, sería un volumen suficiente y que se podría dar en una pequeña empresa o en un hogar.

Tras analizar las distintas opciones y partiendo de la premisa del uso de hardware a bajo coste optaremos por la instalación unificada en la que todos los servicios Wazuh y la suite ELK estarán implantadas en un mismo servidor.

Gracias al plugin de Kibana para Wazuh podremos integrar el contenido del mismo para visualizarlo de forma gráfica a través de su interfaz web.

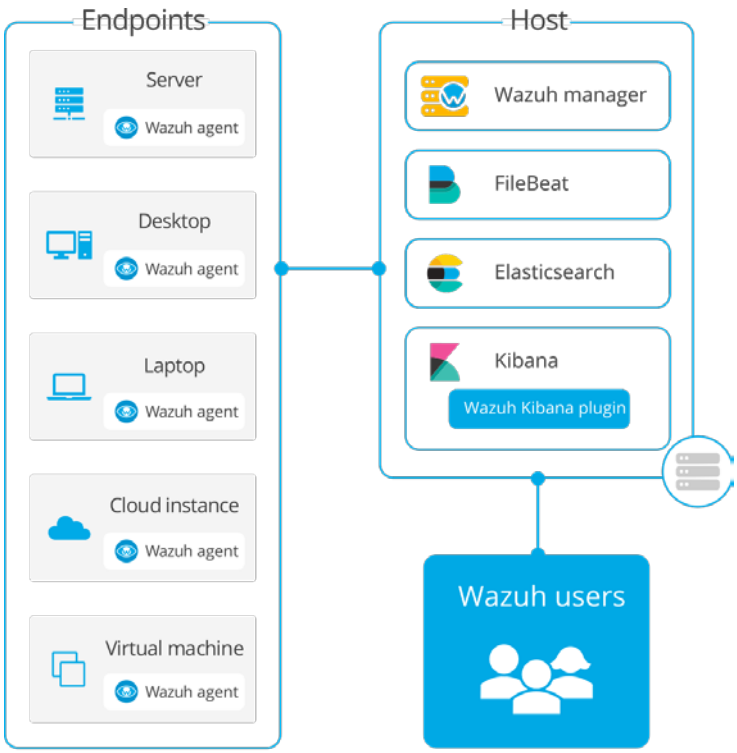


Ilustración 7. All in one deployment. Wazuh.

3.2.1.1. Requisitos de software

Tras la documentación técnica de Wazuh y de Elastic se puede apreciar la necesidad de implementar los sistemas bajo un equipo que corra con Linux 64 bits siendo las versiones mínimas aceptadas las siguientes:

- Amazon Linux 1 y 2.
- CentOS 6 o superior.
- Debian 7 o superior.
- Fedora 31 o superior.
- Oracle Linux 6 o superior.
- Red Hat Enterprise Linux 6 o superior.
- Ubuntu 12 o superior.

Para esta prueba se optará por Raspberry Pi OS de 64 bits, (con Debian 10 como base) y que está adaptada al dispositivo que vamos a emplear como servidor.

3.2.1.2. Requisitos de hardware

En lo referente al hardware nos indican que para el tipo de desarrollo planteado serían necesarios 4 GB de memoria RAM y un procesador de 2 núcleos.

En este caso se empleará una Raspberry Pi 4 con un procesador 4 núcleos a 1,50 Ghz y 8 GB de RAM.

En cuanto al almacenamiento dependerá del número de alertas por segundo que reciba cada dispositivo que sea necesario una capacidad mayor u otra. La siguiente tabla muestra una estimación del peso de los *log*s generados por los distintos dispositivos.

Puntos finales supervisados	APS	Almacenamiento (GB / 90 días)
Servidores	0,25	3.8
Estaciones de trabajo	0,1	1,5
Dispositivos de red	0,5	7,6

Tabla 2 Estimación del volumen generado por los dispositivos en 90 días. Wazuh.

Para el almacenamiento de tal cantidad de información se optará por almacenar la información generada en un disco externo de 1 tb. Se valorará a posteriori, como medida de seguridad adicional, replicar esta información en un repositorio externo.

El diagrama básico sobre el que basaremos nuestra solución intenta replicar los componentes más relevantes que se pueden encontrar en una red (routers, equipos de sobremesa, portátiles), así como añadidos con sistemas objeto de monitorización (web o nube privada).

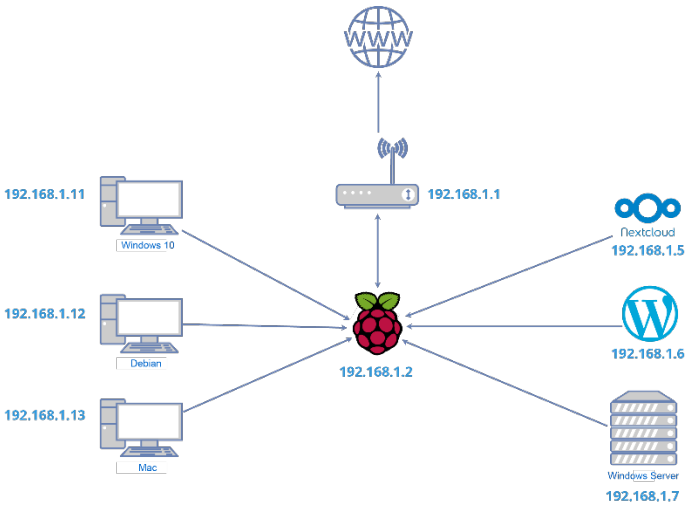


Ilustración 8. Diagrama básico de red. Elaboración propia.

3.3. Instalación de Wazuh 3.13

Como se ha indicado en el diseño de la solución procederemos a realizar la instalación de todos los componentes en la misma máquina (single host). En el momento de iniciar el presente estudio la última versión disponible es la 3.13.

Todas las instrucciones que se detallarán en el proceso serán ejecutadas como superusuario a través del terminal.

3.3.1. Adecuación del sistema

Para garantizar que la instalación se llevará a cabo con éxito será necesario la instalación de los siguientes paquetes.

```
# apt-get install curl apt-transport-https lsb-release gnupg2
```

Tras ello se instala la clave GPG del repositorio de Wazuh con el siguiente comando:

```
# curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | apt-key add -
```

Una vez esté instalada la clave se añade el repositorio de Wazuh para proceder a la instalación de los paquetes necesarios.

```
# echo "deb https://packages.wazuh.com/3.x/apt/ stable main" | tee -a /etc/apt/sources.list.d/wazuh.list
```

Para finalizar se actualiza la lista de repositorios para asegurarnos que a la hora de realizar la petición los paquetes requeridos sean encontrados dentro de los mismos:

```
# apt-get update
```

3.3.2. Instalación de Wazuh Server

En primer lugar, se instala el manager de Wazuh el cual será el core del sistema.

```
# apt-get install wazuh-manager
```

Tras ello se procede al inicio del servicio con el siguiente comando:

```
# systemctl status wazuh-manager
```

3.3.3. Instalación de Wazuh API

Para que el sistema funcione correctamente y Wazuh se pueda integrar con la suite Elastic será necesario instalar una API para realizar la interconexión. Los pasos a seguir son los siguientes:

Dado que será necesario la instalación de código JavaScript en el servidor se debe instalar Nodejs. Para ello:

```
curl -sL https://deb.nodesource.com/setup_10.x | bash -
```

Se instala Nodejs.

```
apt-get install nodejs
```

Se instala la API de Wazuh.

```
apt-get install wazuh-api
```

Se procede a la iniciación del servicio.

```
service wazuh-api startEmpleo de Elastic Stack (ELK)
```

3.4. Instalación Suite Elastic

Elastic Stack es una suite que permite la recopilación, ingesta, enriquecimiento, almacenamiento, análisis y visualización de los datos a través de las distintas aplicaciones que la componen.

Estas aplicaciones son:

Elasticsearch. Será el motor que permita el indexado de toda la información que llegue de las diferentes fuentes.

Log*stash*. Permitirá agregar, procesar y transformar los datos recopilados de las distintas fuentes para posteriormente enviarlos a Elasticsearch.

Kibana. Es la herramienta que permitirá la visualización y la gestión de los datos de forma gráfica. En este caso concreto, servirá para disponer de un cuadro de mandos de ciberseguridad el cual será complementado con la aplicación de Wazuh para mostrar todos los datos que, a través de esta fuente, se van a recoger.

Filebeat. Es la aplicación que permite el reenvío de alertas. En nuestro caso, proporcionará a Elasticsearch las alertas y eventos de forma segura.

3.4.1. Instalación de Filebeat

En primer lugar, se procede a la instalación de la aplicación Filebeat.

Para ello se añade la clave y el repositorio de Elastic lo cual servirá para el resto de componentes. Una vez agregados los datos se procederá a la actualización de la lista.

```
# curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | apt-key add -
# echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | tee /etc/apt/sources.list.d/elastic-7.x.list
# apt-get update
```

Se instala la última versión en el momento de la prueba.

```
apt-get install filebeat=7.9.3
```

Se procede a la descarga del archivo de configuración de Filebeat que permitirá el reenvío de las alertas de Wazuh a Elasticsearch.

```
curl -so /etc/filebeat/filebeat.yml https://raw.githubusercontent.com/wazuh/wazuh/v3.13.2/extensions/filebeat/7.x/filebeat.yml
```

Se descarga e instala la plantilla de alertas para Elasticsearch.

```
curl -so /etc/filebeat/wazuh-template.json https://raw.githubusercontent.com/wazuh/wazuh/v3.13.2/extensions/elasticsearch/7.x/wazuh-template.json
```

Por último, se descarga el módulo específico de Wazuh para Filebeat.

```
curl -s https://packages.wazuh.com/3.x/filebeat/wazuh-filebeat-0.1.tar.gz | sudo tar -xvz -C /usr/share/filebeat/module
```

Para que las alertas lleguen correctamente al host será necesario indicar a Filebeat la dirección en la cual estará alojado el host que aloje Elasticsearch. Se aplica la siguiente configuración sobre el archivo filebeat.yml

```
output.elasticsearch.hosts: ['http://192.168.1.2:9200']
```

Se muestra, a continuación, la configuración del archivo filebeat.yml

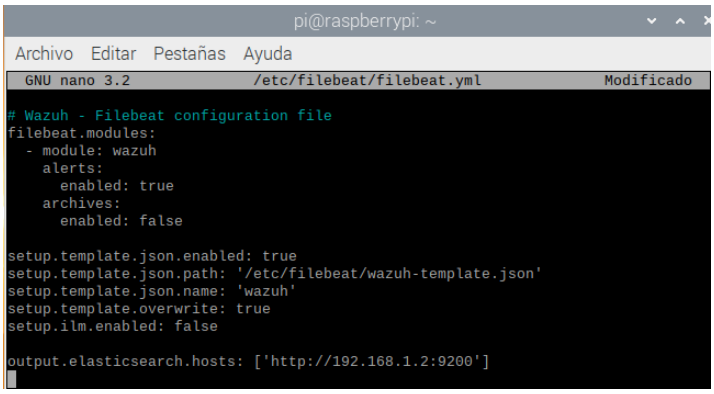


Ilustración 9. Configuración fichero Filebeat.yml. Elasticsearch.

Se activa e inicia el servicio de Filebeat

```
update-rc.d filebeat defaults 95 10
# service filebeat start
```

3.4.2. Instalación de Elasticsearch

Una vez instalado Filebeat, se procede a instalar, configurar y revisar que el servicio de Elasticsearch está activo.

Para instalarlo, se lanza el comando siguiente:

```
apt-get install elasticsearch=7.9.3
```

Una vez comprobado que se ha instalado con éxito se procede a cambiar la configuración de /etc/elasticsearch/elasticsearch.yml agregando la siguiente información:

Se indica el nodo en el que está instalado el sistema.

```
network.host: 192.168.1.2
http.port: 9200
```

Se especifica el nombre del nodo y se indica el clúster. En este caso, al emplear un único dispositivo y al no estar replicado, no se informará de ningún nodo adicional.

```
node.name: siem
cluster.initial_master_nodes: ["siem"]
```

Por último, se habilita el servicio y se inicia.

```
update-rc.d elasticsearch defaults 95 10
# service elasticsearch start
```

```
root@raspberrypi:/home/pi# service elasticsearch status
● elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; disabled; vend
   Active: active (running) since Sun 2020-11-08 19:30:38 CET; 26s ago
     Docs: https://www.elastic.co
 Main PID: 5050 (java)
    Tasks: 64 (limit: 4249)
   CGroup: /system.slice/elasticsearch.service
           ├─5050 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des.networ
           └─6806 /usr/share/elasticsearch/modules/x-pack-ml/platform/linux-aarc

nov 08 19:29:49 raspberrypi systemd[1]: Starting Elasticsearch...
nov 08 19:30:38 raspberrypi systemd[1]: Started Elasticsearch.
lines 1-12/12 (END)
```

Ilustración 10. Estado Elasticsearch. Raspberry Pi OS.

Para tener la certeza de que no habrá incidencia en las comunicaciones, se realiza un test sobre el puerto y se comprueba que el servicio se está ejecutando correctamente y lista para escuchar peticiones.

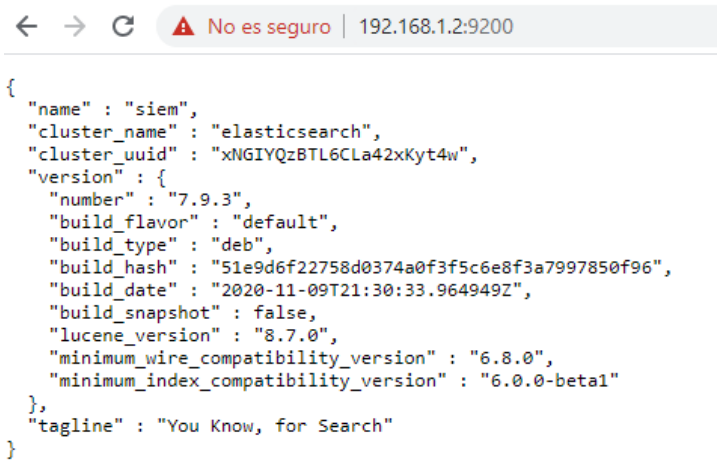


Ilustración 11. Revisión estado Elasticsearch. Elasticsearch.

Una vez se ha comprobado que el servicio se ha iniciado correctamente, se aplica la plantilla descargada en el punto anterior.

```
filebeat setup --index-management -E setup.template.json.enabled=false
```

3.4.3. Instalación de Kibana

Por último, se procede a la instalación de Kibana, que permitirá visualizar gráficamente el contenido que se recopilará de las diferentes fuentes que se han ido integrando a lo largo del proceso.

```
apt-get install kibana=7.9.3
```

Incidencia instalación Kibana

Al intentar realizar el proceso de instalación se observa que no se localiza el paquete para instalarlo debido, en parte, a la arquitectura del sistema operativo que tenemos instalado aarch64.

Para poder solventar esta incidencia se ha procedido a descargar el paquete facilitado por los desarrolladores de Elastic (https://github.com/elastic/kibana/issues/41460) y se ha instalado con la siguiente instrucción.

```
curl -Ls https://artifacts.elastic.co/downloads/kibana/kibana-7.9.3-linux-aarch64.tar.gz | tar zx
```

Tras ello se ha procedido a asignar los permisos al usuario kibana a los siguientes directorios.

```
chown -R kibana:kibana /usr/share/kibana/optimize
# chown -R kibana:kibana /usr/share/kibana/plugins
```

A continuación, se ha cambiado la configuración de kibana.yml añadiendo la siguiente información

```
server.host: 192.168.1.2
http.port: 5555
elasticsearch.hosts: ["http://192.168.1.2:9200"]
```

Por último, se habilita, inicia y comprueba que el servicio está activo.

```
update-rc.d kibana defaults 95 10
# service kibana start
```

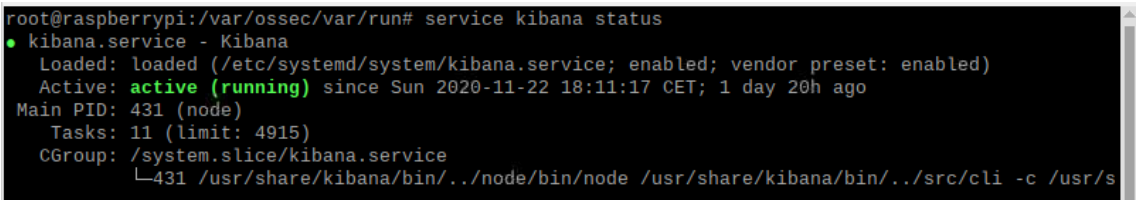


Ilustración 12. Estado del servicio Kibana. Raspberry Pi OS.

Incidencia en la instalación plugin Wazuh - Kibana

Finalmente, se procede a instalar el plugin que permitirá disponer de un apartado específico de Wazuh dentro de Kibana.

```
# sudo -u kibana bin/kibana-plugin install
https://packages.wazuh.com/wazuhapp/wazuhapp-3.13.2_7.9.2.zip
```

Al lanzar el siguiente comando se aprecia que la versión del plugin estaba obsoleta (7.9.2) y no existiendo un repositorio alternativo para poder realizar la instalación manualmente como en la incidencia anterior.

Versión de la aplicación API y Wazuh	Versión Elastic Stack
3.13.2	7.8.0, 7.8.1, 7.9.0, 7.9.1, 7.9.2
3.13.1	7.8.0, 7.8.1, 7.9.0, 7.9.1
3.13.0	7.7.0, 7.7.1, 7.8.0

Ilustración 13. Matriz de compatibilidad de versiones Wazuh 3.13.X. Wazuh.

Esta cuestión, se entiende, fue propiciada por el lanzamiento de la nueva versión de Wazuh (4.0.) en el que se puede apreciar que la versión requerida del plugin sí estaba disponible.

Wazuh version	Elastic Stack version	Open Distro version
4.0.0	7.9.1	1.10.1
4.0.0	7.9.3	

Ilustración 14. Matriz de compatibilidad Wazuh 4.0.0. Wazuh.

Para solucionarlo finalmente se procede a la actualización a las versiones más actualizadas de tanto de Wazuh como de Elasticsearch. Esta cuestión se tratará en el punto 3.4. Para la instalación del plugin se emplea la siguiente instrucción:

```
sudo -u kibana /usr/share/kibana/bin/kibana-plugin install https://packages.wazuh.com/4.x/ui/kibana/wazuh_kibana-4.0.1_7.9.3-1.zip
```

3.5. Actualización de versión

Las incidencias ocurridas en el proceso, así como las novedades ofrecidas en las últimas versiones de Wazuh (4.0.3), como ELK (7.9.3), han suscitado la actualización de la versión de los componentes instalados hasta ahora.

Entre las novedades más relevantes de Wazuh se encuentran las siguientes:

- Mejora en la inicialización de los agentes.
- Integración de la API dentro del administrador de Wazuh.
- Cambio en el protocolo de comunicación de los agentes de UDP a TCP.
- Instalador del agente para Windows pasa a estar firmado por Digicert.
- Cambios en la gestión de almacenamiento para la asignación de espacio a los diversos módulos que componen el sistema.
- Mejora en la precisión de usuarios, roles y políticas para administrar los privilegios de acceso. Permite un mayor control granular sobre el acceso a los recursos.

- Incorporación de un sistema de bloqueo de ataques DOS.
- Incorporación de un sistema de bloqueo contra ataques de fuerza bruta.
- Mejora en la monitorización de las rutas de variables en entorno.

Mediante la instrucción siguiente ha sido posible conseguir que se actualicen los paquetes de Wazuh a la última versión disponible.

```
sudo apt-get update && upgrade
```

Tras ello se comprueba que la última versión disponible es la instalada, v.4.0.3.

```
root@raspberrypi:/var/ossec/bin# /var/ossec/bin/manage_agents -V

Wazuh v4.0.3 - Wazuh Inc.
```

Ilustración 15. Comprobación versión Wazuh. Raspberry Pi OS.

3.6. Despliegue de los agentes

Con el fin de poder recopilar los eventos que se producen en los sistemas que se van a monitorizar, Wazuh pone a nuestra disposición una serie de agentes para los principales sistemas operativos.

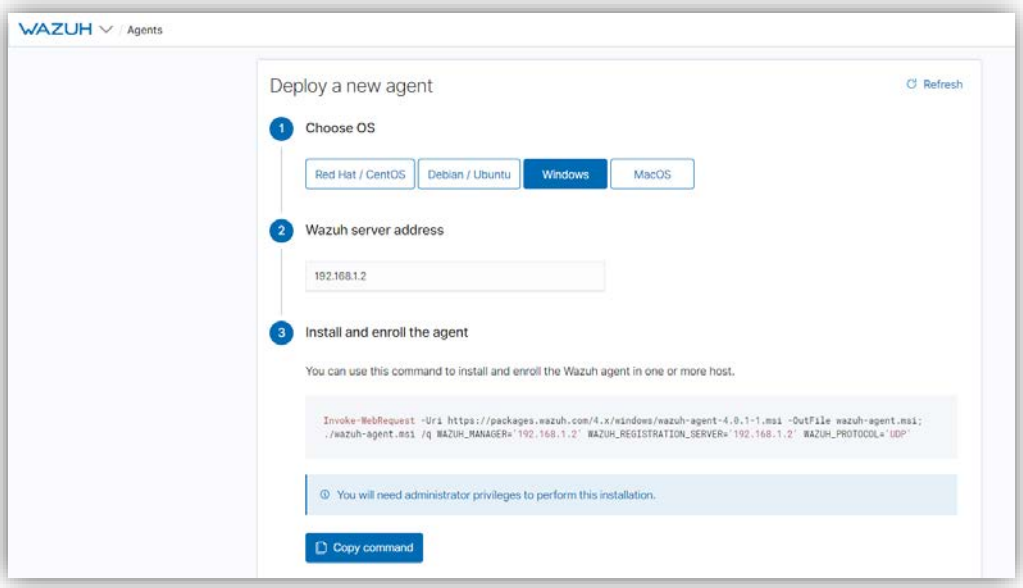


Ilustración 16. Despliegue de agentes. Wazuh.

En línea con el diseño propuesto se procederán a realizar el despliegue tanto en equipos cliente como en servidores que tenemos virtualizados.

Como se ha comentado anteriormente, el servidor Wazuh, para nuestro entorno, tendrá asignada la IP **192.168.1.2**.

3.6.1. Agente para Windows

En el caso de los sistemas Windows ya sea la versión para estación de trabajo o servidor será necesario ejecutar el siguiente comando con permisos de administrador a través de la consola de PowerShell:

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.0.1-1.msi -OutFile wazuh-agent.msi; ./wazuh-agent.msi /q WAZUH_MANAGER='192.168.1.2' WAZUH_REGISTRATION_SERVER='192.168.1.2' WAZUH_PROTOCOL='UDP'
```

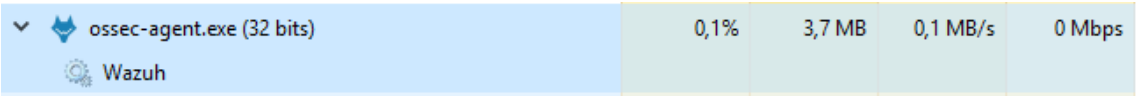


Ilustración 17. Comprobación funcionamiento del agente en Task Manager. Windows 10.

3.6.2. Agente para Linux (Debian)

Al igual que en el punto anterior, se procede a realizar la ejecución mediante comando para proceder a la descarga e instalación del agente para nuestras máquinas, que corren el sistema operativo Debian, concretamente en las VM destinadas a almacenar por un lado nuestra nube privada (Nextcloud) y nuestra web personal (WordPress).

```
curl -so wazuh-agent.deb https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.0.1-1_amd64.deb && sudo WAZUH_MANAGER='192.168.1.2' WAZUH_PROTOCOL='UDP' dpkg -i ./wazuh-agent.deb
```

3.6.3. Agente Macintosh OS

Para MAC OS se ejecuta la siguiente línea en el terminal.

```
curl -so wazuh-agent.pkg https://packages.wazuh.com/4.x/macos/wazuh-agent-4.0.1-1.pkg && sudo launchctl setenv WAZUH_MANAGER '192.168.1.2' WAZUH_PROTOCOL 'UDP' && sudo installer -pkg ./wazuh-agent.pkg -target /
```

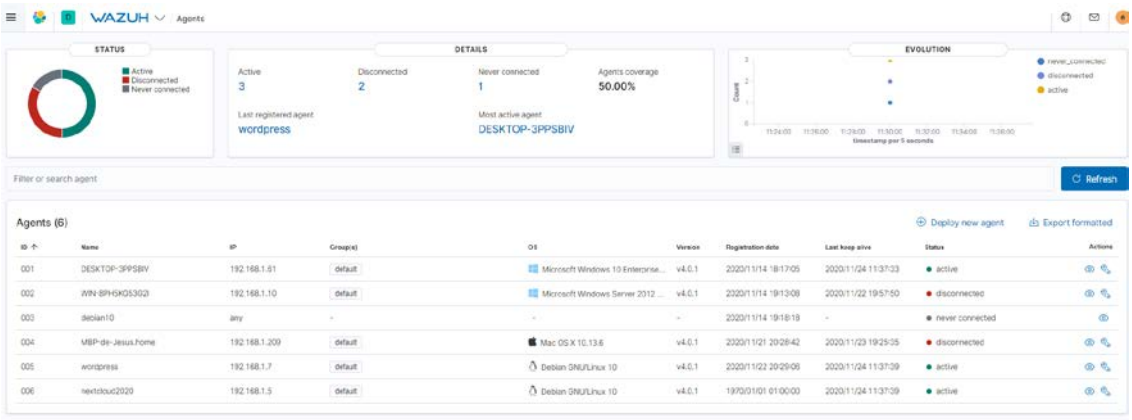


Ilustración 18. Panel de agentes Wazuh. Kibana.

3.7. Securización de la interfaz gráfica

Por defecto, Elastic no implementa ningún sistema de autenticación para acceder a la interfaz web por lo que cualquier usuario que hiciera una búsqueda de servicios en la red podría descubrir y acceder al panel de administración del sistema que se está implemenentando.

Para solventar esta incidencia se procede a crear e implantar certificados de seguridad SSL para los diversos sistemas implementados. Debido a la extensión del proceso, dejamos las instrucciones seguidas para su consulta en el Anexo I. Securización de Elasticsearch.

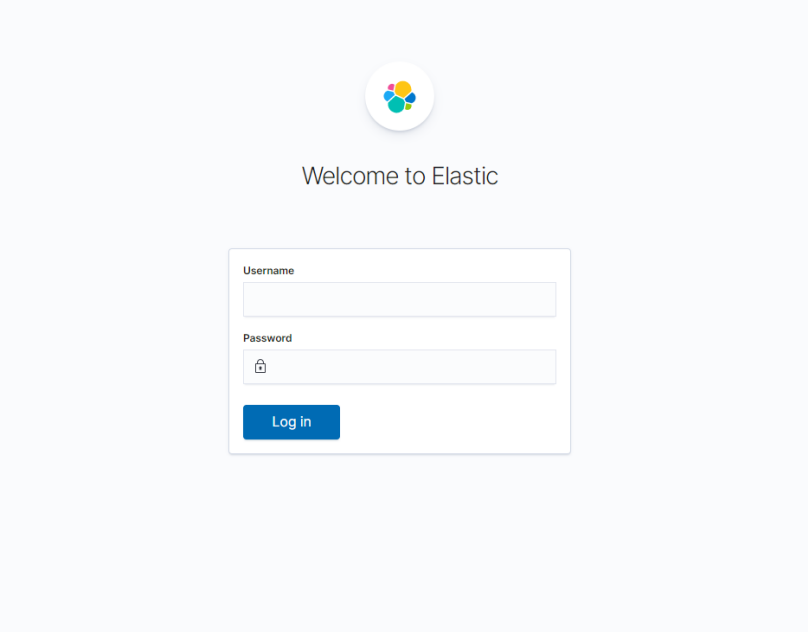


Ilustración 19. Autenticación portal web Elastic. Elastic.

3.8. Panel de administración Kibana / Wazuh

Tras realizar la instalación de todos los módulos, es posible acceder al panel de visualización y administración de datos a través de la IP del servidor, 192.168.1.2 en nuestro caso.

Inicialmente muestra el dashboard integral de Elastic el cual se puede configurar posteriormente.

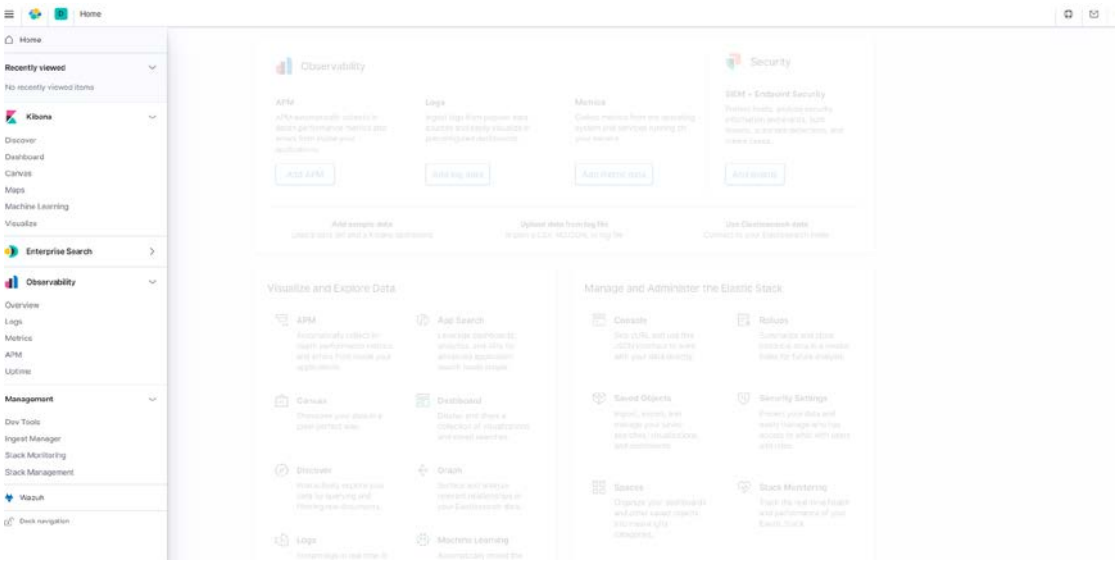


Ilustración 20. Panel de administración de Elastic. Kibana.

Gracias a la instalación del plugin de Wazuh, se integra un apartado que permite configurar las opciones de Wazuh entre las que destacan las comentadas al inicio de este apartado, y que se desarrollarán en los distintos casos de uso que serán planteados en los próximos capítulos.

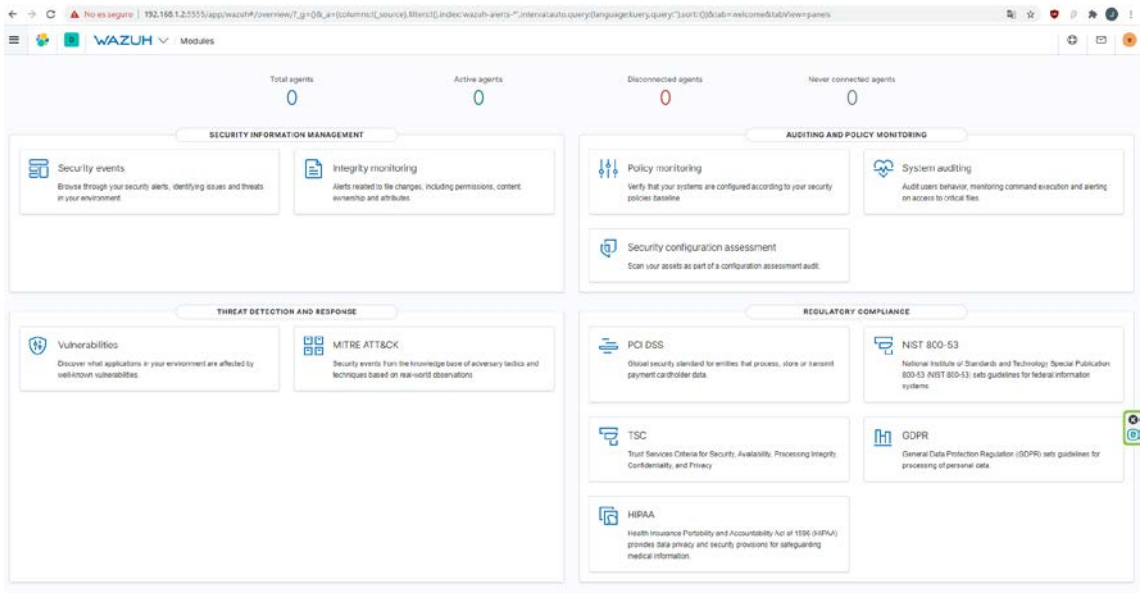


Ilustración 21. Panel de administración Wazuh. Kibana / Wazuh.

3.9. Instalación de OpenVPN

Con el fin de mejorar la seguridad de la gestión de estas aplicaciones, incluir equipos de red que estén conectados en diferentes localizaciones y realizar el tratamiento de *logs* de otras aplicaciones se implementará una red virtual privada (VPN) a través de la herramienta PiVPN, un sistema adaptado al sistema de Raspberry.

3.9.1. Instalación del servidor

Mediante el comando siguiente, será posible descargar y lanzar el asistente que servirá de guía en el proceso de instalación de la herramienta.

```
curl -L https://install.pivpn.io | bash
```

Esta aplicación permitirá elegir entre Wireguard o OpenVPN. En este caso, se opta por el último. El proceso resumido es el siguiente (se amplía información de la instalación completa en el Anexo IV):

- Elección del servicio VPN.
- Configuración del tipo de conexión a elegir (TCP o UDP).
- Puerto por defecto de conexión.
- Selección de DNS.
- Instalación desatendida de actualizaciones.
- Tipo de cifrado elegido.

Tras ello el proceso estaría listo y el servicio iniciado listo para admitir conexiones.

3.9.2. Creación del cliente

Una vez se haya realizado la instalación del servidor, se crea el usuario con el siguiente comando:

```
pivpn –add
```

Tras ello, se procede a la exportación del perfil creado y se incluirá en el sistema que se pretenda utilizar, en nuestro caso, Windows.

Ilustración 22. Confirmación creación del perfil OpenVPN. OpenVPN.

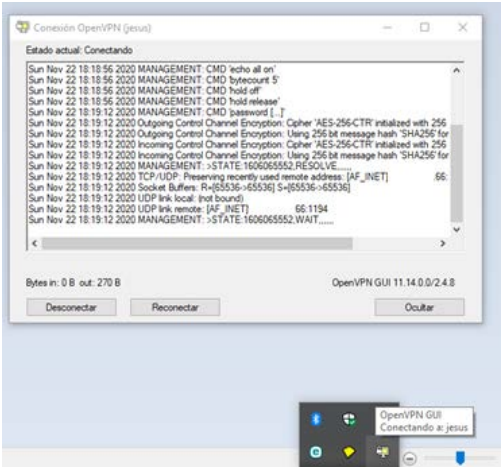


Ilustración 23. Conexión mediante VPN. OpenVPN.

3.10. Pi-hole

Pi-hole es una aplicación que permite el bloqueo de publicar y rastreadores por internet que va a permitir, entre otros, mejorar la experiencia en la navegación en nuestros dispositivos gracias a su servicio de DNS.

Adicionalmente a estas características se suma la posibilidad de controlar todo el tráfico de red actuando como una versión light de un firewall.

La información proporcionada por este sistema es valiosa y candidata a ser integrada dentro del panel de ciberseguridad que se pretende conformar con la ayuda de Kibana.

3.10.1. Instalación

Con un procedimiento similar al de PiVPN se realiza el proceso de instalación de este servicio a través de un script de instalación.

```
curl -sSL https://install.pi-hole.net | bash
```

El proceso resumido es el siguiente (se amplía información de la instalación completa en Anexo III):

- Selección de la interfaz de escucha (red inalámbrica o cableada).
- Importación de listas de bloqueo predefinidas.
- Configuración TCP/IP.
- Habilitar interfaz gráfica.
- Almacenamiento de *log*s.
- Configuración de los niveles de privacidad.
- Configuración de clave de administrador.

Tras ello, el servicio de seguridad de red estará listo. Será necesario configurar en los enrutadores este dispositivo como servidor DNS. El resultado tras unas horas de uso es el siguiente:

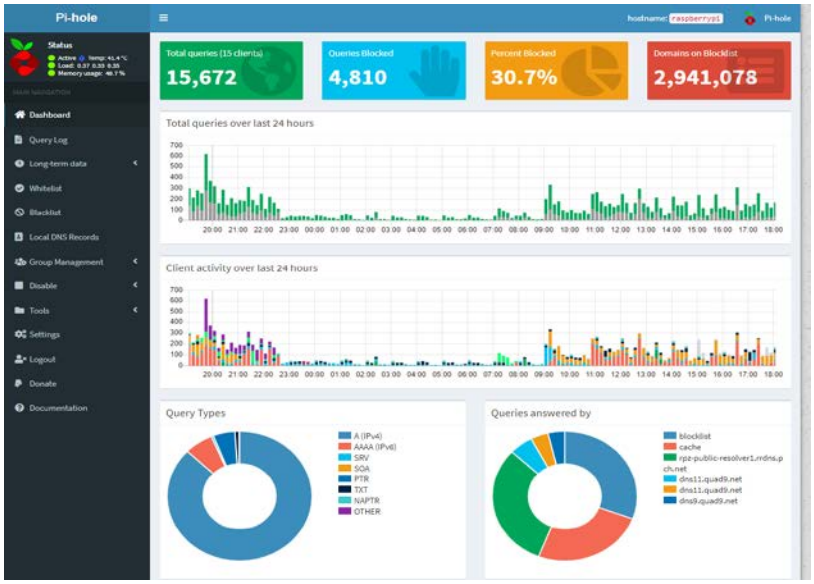
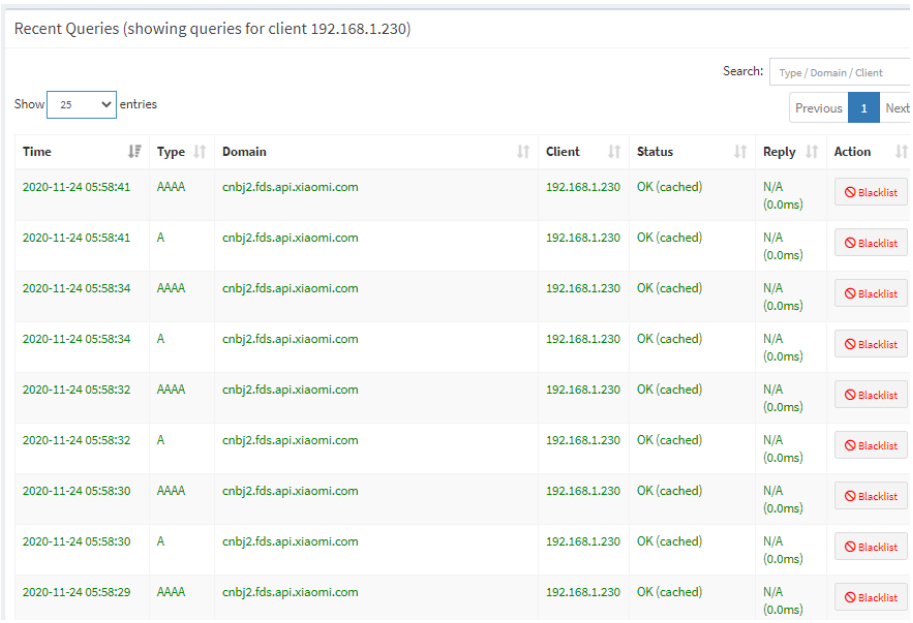


Ilustración 24. Panel de administración Pi-hole. Pi-hole.

3.10.2. *Logs*

Uno de los objetivos que se plantearon era el control de la red. Es probable que con esta aplicación se pueda hacer parcialmente, pues, como se puede comprobar a continuación, es posible hacer la revisión visual de todas las conexiones que un equipo realiza de forma visual y sin necesidad de tener a mano una consola y bloquear, si así se desea, las peticiones a golpe de clic (Action – Blacklist).

Cada vez es más frecuente la conexión a redes de los llamados dispositivos inteligentes. El caso que se muestra a continuación es un equipo de la marca Xiaomi, en el que se puede apreciar el número de peticiones constantes que realiza a servidores pese a estar en un supuesto reposo.


Recent Queries (showing queries for client 192.168.1.230)

Search: Type / Domain / Client

Show 25 entries

Previous 1 Next

Time	Type	Domain	Client	Status	Reply	Action
2020-11-24 05:58:41	AAAA	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:41	A	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:34	AAAA	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:34	A	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:32	AAAA	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:32	A	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:30	AAAA	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:30	A	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist
2020-11-24 05:58:29	AAAA	cnbj2.fds.api.xiaomi.com	192.168.1.230	OK (cached)	N/A (0.0ms)	Blacklist

Ilustración 25. Listado de peticiones de un dispositivo inteligente. Pi-hole

Estos *log*s, almacenados en /var/*log*/Pi-hole.*log* serán tratados posteriormente la pila ELK que nos permitirá integrar esta fuente interesante de datos.

4. Casos de uso

4.1. Detección de archivo malicioso alojado en cloud privada a través de VirusTotal

Uno de los puntos críticos de una infraestructura se puede localizar en sistemas que almacenan contenidos provenientes de fuentes desconocidas como puede ser un sistema compartido de almacenamiento o una nube privada, como es nuestro caso, a través de Nextcloud.

Con la ayuda de VirusTotal[4] y, concretamente, a través de su API para incluir nuestra sonda en el sistema para que analice los directorios en los que los diversos usuarios almacenan sus archivos.

1) El proceso se inicia con una alerta a través del File Integrity Monitoring que detecta un nuevo archivo en el sistema.
2) Esta alerta se comunica al servidor que reenvía la petición a VirusTotal para analizar el archivo que se ha subido en el sistema.
3) VirusTotal, tras analizar el archivo devuelve una respuesta.
4) En caso de que haya detectado alguna incidencia el servidor Wazuh enviará una alerta o iniciará un procedimiento de respuesta activa para actuar ante una posible amenaza.

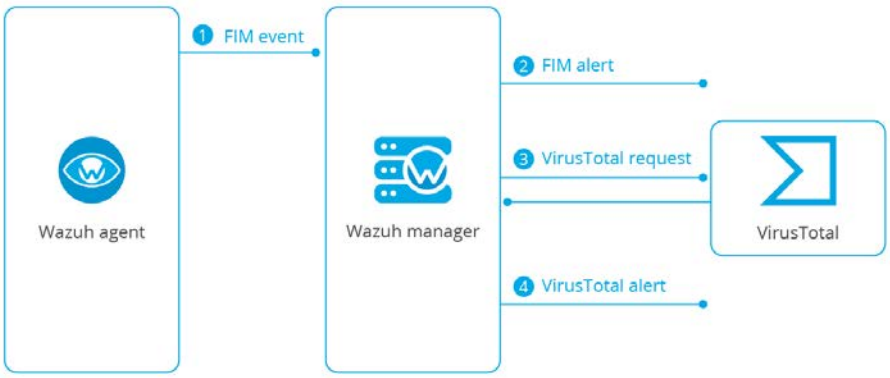


Ilustración 26. Proceso escaneo VirusTotal. Wazuh.

En primer lugar, se genera nuestro token desde el portal web de VirusTotal. Para ello, tras realizar el registro y acceder al apartado API Key encontraremos la clave necesaria

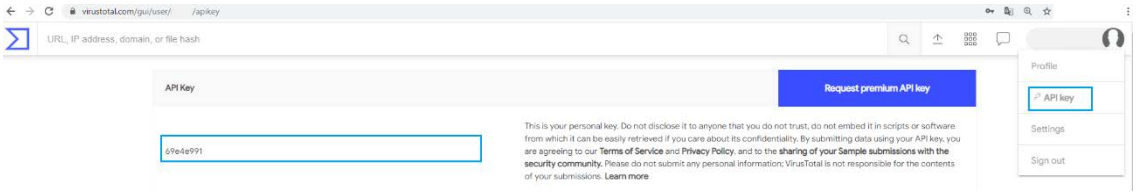


Ilustración 27. Generación de API key en VirusTotal. VirusTotal.

[4] Plataforma de referencia que permite el análisis de archivos, sitios web, hashes gracias a los motores de búsqueda disponibles.

Para que el FIM funcione correctamente con esta funcionalidad se incluye el siguiente en el archivo ossec.conf del Manager de Wazuh.

```
<integration>
 <name>VirusTotal</name>
 <api_key>VIRUSTOTAL_API_KEY</api_key> <!-- Replace with your VirusTotalAPI key -->
 <group>syscheck</group>
 <alert_format>json</alert_format>
</integration>
```

Una vez incluida la API, configurado los directorios a escanear en tiempo real, y reiniciado el sistema, éste quedará en funcionamiento.

Para comprobar el correcto funcionamiento y evitar infectar nuestros sistemas alojamos en el archivo EICAR Antivirus Test File. Se trata un archivo de prueba desarrollado por el Instituto Europeo para la Investigación de los Antivirus Informáticos (EICAR) que se emplea para probar la efectividad de los sistemas de protección implementados.

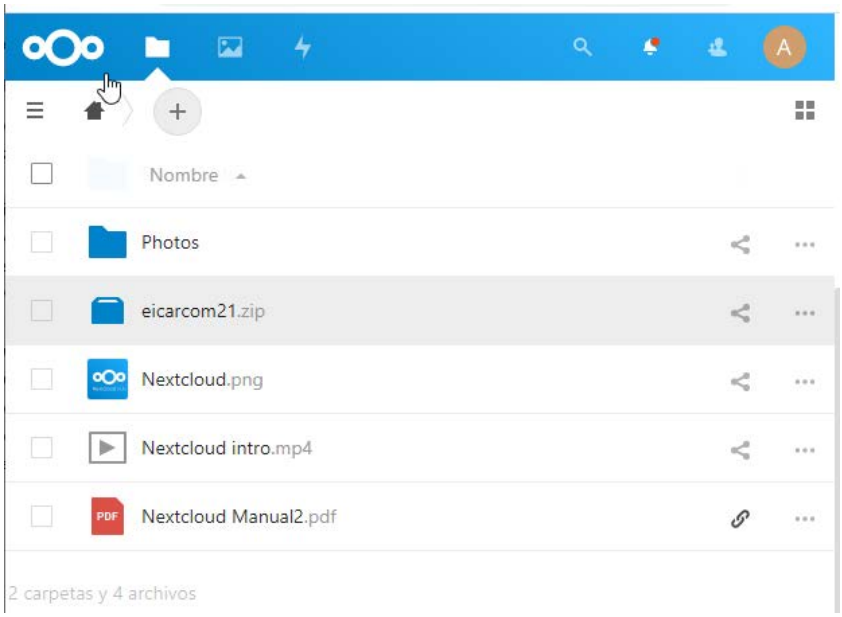


Ilustración 28. Alojamiento archivo EICAR en Nextcloud. Nextcloud.

Una vez alojado el archivo dentro de nuestra nube privada y tras ser analizado por la API de VirusTotal, se procede a comprobar el funcionamiento del FIM (File Integrity Monitoring) de Wazuh, el cual permite la detección de amenazas dentro de nuestro sistema (data.VirusTotal.positives: 53) como se puede apreciar en la siguiente ilustración.

Ilustración 29. Análisis de archivos alojado en Nextcloud. Kibana / Wazuh.

Este evento podría dar lugar a otras acciones tales como la notificación vía email o la eliminación de la amenaza como parte del plan de respuesta ante incidentes mediante la ejecución de nuevas tareas desembocadas por la aparición de dicho evento.

4.2. Prevención y detección ante un ataque de Ransomware

Como se comentaba al inicio, el número de ataques a sistemas ha aumentado exponencialmente en los últimos años. Uno de los más empleados es el Ransomware, el cual, mediante un ataque de Phishing, consigue acceder a nuestro sistema encriptando nuestros archivos y pidiendo una cantidad de dinero a modo de rescate.

En este caso se hará una simulación de este tipo de ataque en una estación de trabajo corriendo Windows 10. Para ello, se cuenta con un script que creará directorios y subdirectorios, así como archivos que alojará en los mismos y que posteriormente serán encriptados. El código del mismo, desarrollado por Wazuh, puede ser consultado en el Anexo V.

En primer lugar, se configura el agente ossec, indicando la ruta a monitorizar “C:\users\Jesús\Desktop\ransomware”.

Ilustración 30. Configuración FIM en agente Wazuh. Wazuh.

Se ejecuta el script que creará automáticamente las carpetas.

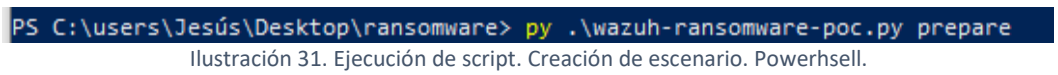


Ilustración 31. Ejecución de script. Creación de escenario. Powerhsell.

Se revisa que el FIM detecte los nuevos eventos. En este caso la creación de nuevos ficheros en un determinado directorio.

Ilustración 32. Comprobación del funcionamiento del FIM en directorio objetivo. Kibana.

Una vez creadas las carpetas se lanza la ejecución del “ataque” encriptando el contenido alojado en el directorio que se le ha pasado al script.

```
PS C:\users\Jesús\Desktop\ransomware> py .\wazuh-ransomware-poc.py attack
```

Ilustración 33. Simulación ataque Ransomware. Encriptación de archivos. Powershell

En la siguiente ilustración se puede comprobar como los ficheros han sido encriptados correctamente.

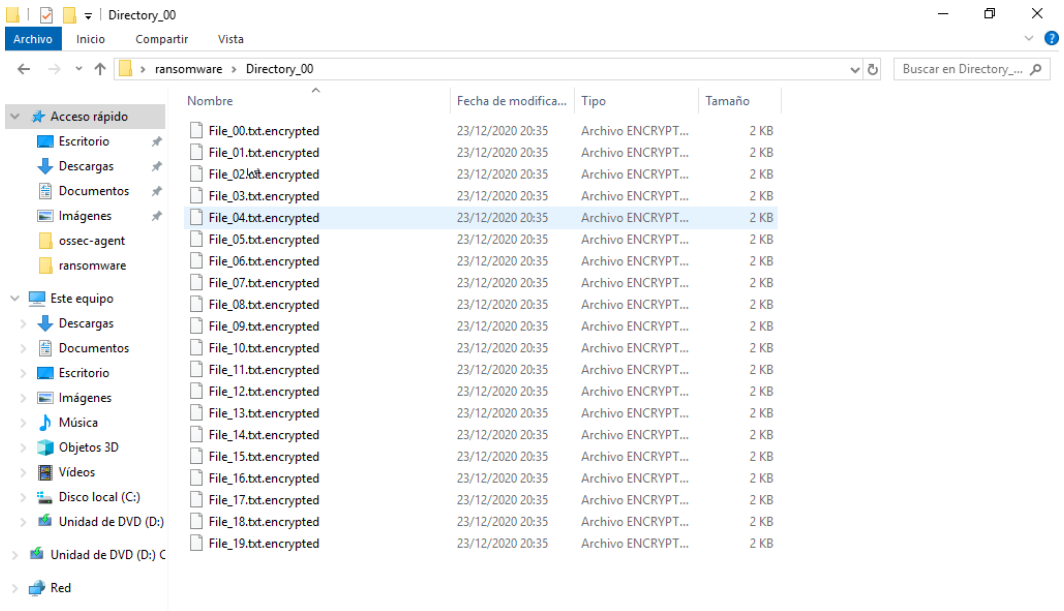


Ilustración 34. Simulación ataque Ransomware. Comprobación archivos encriptados. Windows 10.

Automáticamente, el FIM, como es posible comprobar en la siguiente ilustración, se hace eco de los cambios que han ocurrido en los directorios. Se puede observar el procedimiento seguido por este tipo de ataque en el que primero encripta los archivos y posteriormente elimina los archivos originales.

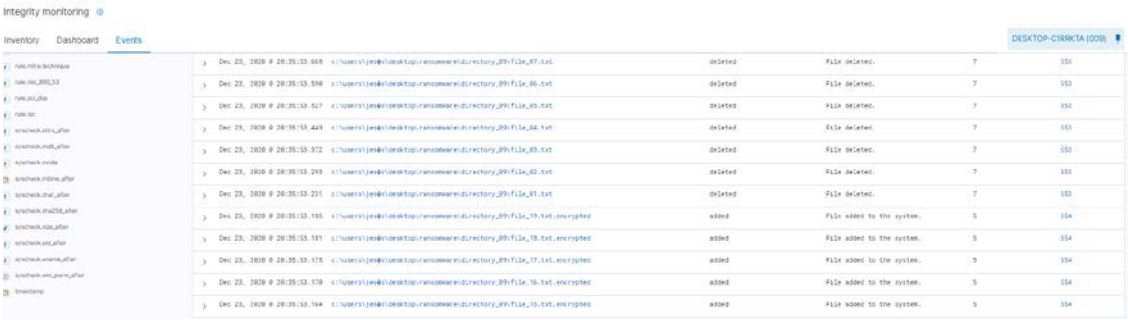


Ilustración 35. Simulación ataque Ransomware. Revisión de eventos en FIM. Kibana / Wazuh.

Como conclusión, nuestro sistema sería válido para la detección de ataques de tipo Ransomware. Faltará por definir el proceso para el tratamiento y la mitigación de este tipo de ataques mediante la definición de un plan de respuesta que permita, por ejemplo, aislar estos equipos de la red, o apagar el mismo para evitar que se propague tanto por el sistema como por la red. Además, generar alertas para proceder a su rápida investigación.

4.3. Detección de ataque de fuerza bruta por SSH

Uno de los puntos vulnerables y que suelen ser testeados y atacados son las conexiones a través de SSH.

Con el fin de detectar posibles intrusiones se ha definido una alerta que permite controlar la actividad que se realiza sobre este tipo de conexión a través de los eventos recabados por el sistema en el fichero **auth.*log***.

En nuestro caso, se simula un ataque realizando intentos de autenticación contra el sistema, con un usuario admin hacia la máquina virtual en la que se ha desplegado Nextcloud.

Ilustración 36.Intento de validación SSH usuario admin. Putty.

Pasados unos segundos, se puede ver en nuestro panel de alertas cómo ha recogido el evento informando del intento de conexión fallido hacia nuestra máquina.

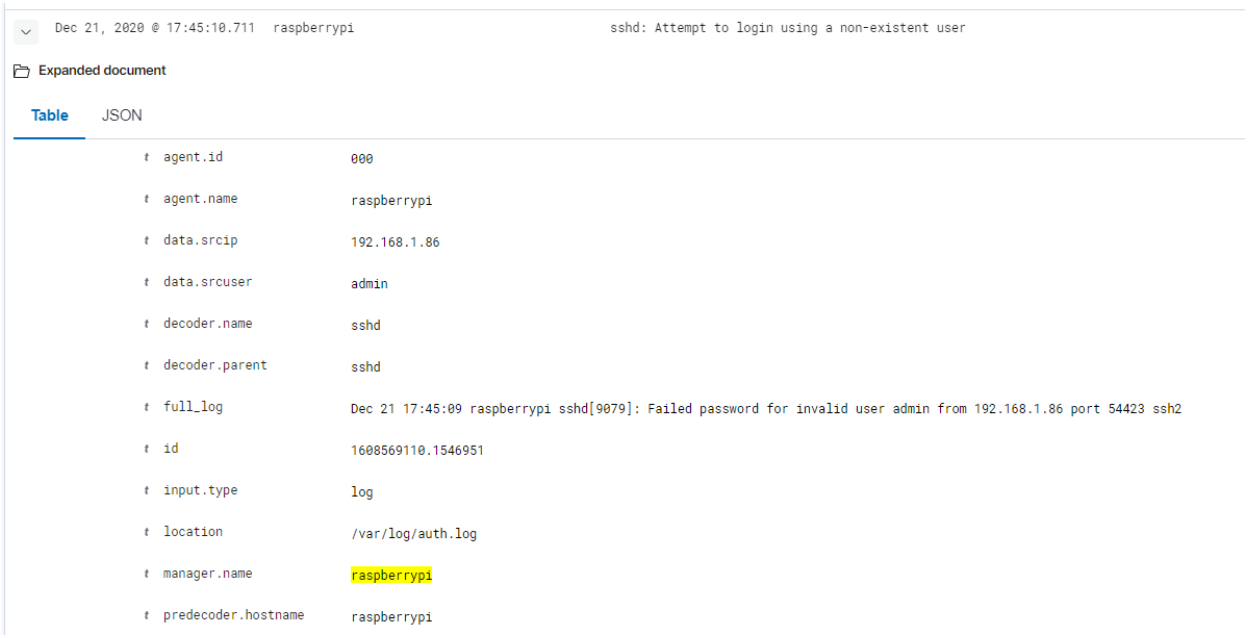


Ilustración 37. Simulación ataque fuerza bruta SSH. Wazuh.

Este tipo de ataques es bastante común como paso posterior a un escaneo de puertos en el que se detecta que sobre el puerto 22 -en este caso- corre el servicio de SSH.

Aunque existen mecanismos que, a un determinado intento, bloquean los intentos de conexión, la notificación de este tipo de eventos nos servirá para determinar posibles puntos débiles en nuestra infraestructura.

4.4. Detección de conexiones a la VPN desplegada

Otro de los puntos que es interesante controlar consiste en toda aquella conexión que se realiza a través de la VPN. Para ello se incluye dentro del Manager de Wazuh un nuevo *log* a monitorizar y se crea una regla para que nos notifique de la ocurrencia de un evento.

Se añaden al archivo ossec.conf del Manager las siguientes líneas. Indicamos que el *log* de Ppenvpn será monitorizado, de este modo será posible revisar, de forma pormenorizada, los eventos que surgen.

```
<localfile>
 <location>/var/log/openvpn.log</location>
 <logformat>syslog</logformat>
</localfile>
```

Se incluye también una alerta que permita la notificación al mail, de este modo avisará de la realización de una conexión por esta vía.

```
<email_alerts>
 <email_to>destinatario@correo.com</email_to>
 <rule_id>81801</rule_id>
 <do_not_delay />
</email_alerts>
```

Tras reiniciar el servicio y realizar de nuevo la conexión, podemos apreciar que los eventos ya son detectados por el SIEM.

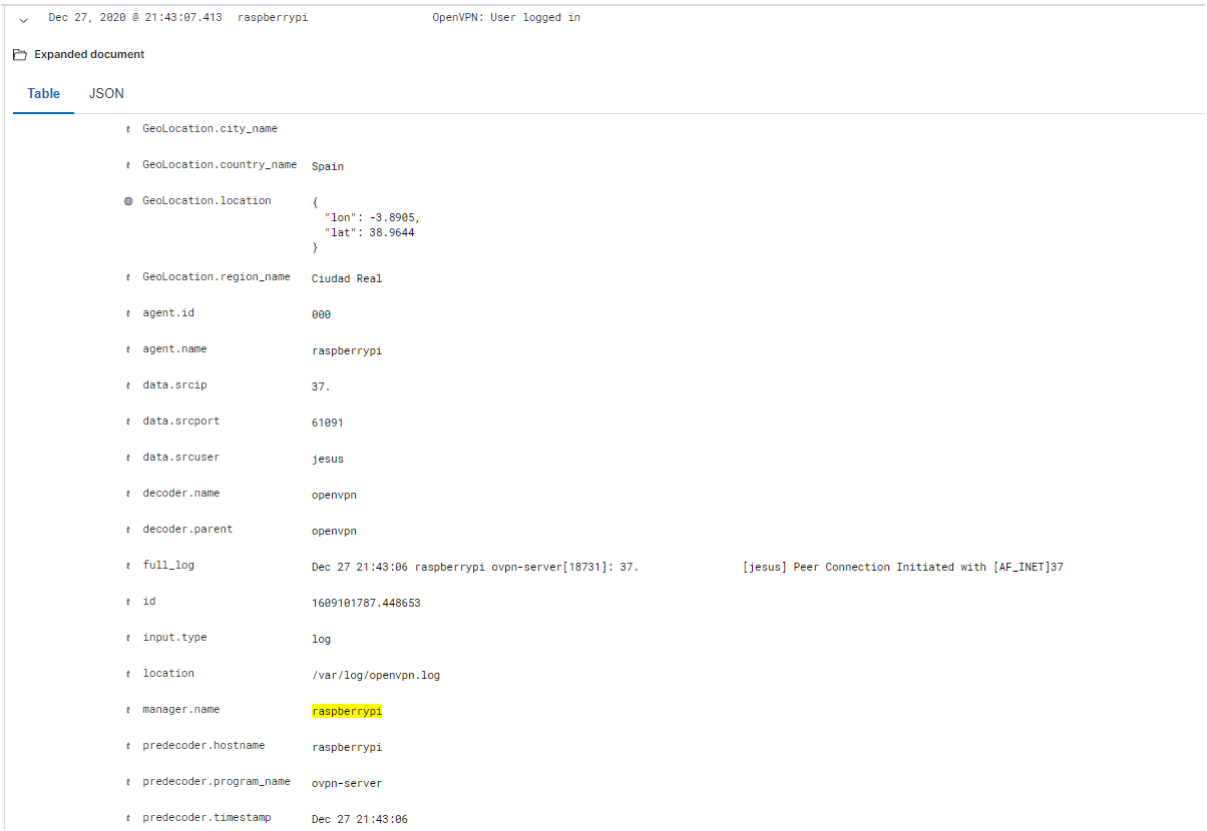


Ilustración 38. Detección de conexión por VPN. Wazuh.

Se recibe la comunicación con información del evento.

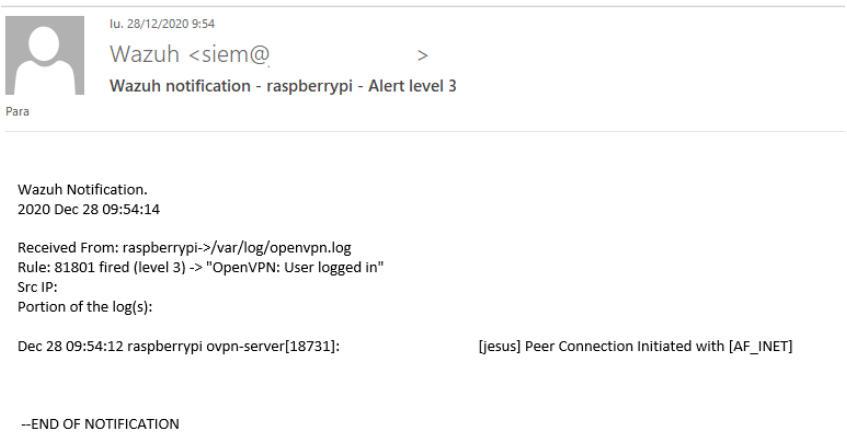


Ilustración 39. Notificación de alerta usuario conectado por VPN. Outlook 2016.

Durante las pruebas realizadas se detecta un intento de conexión desde una IP no identificada lo cual nos podría indicar que estamos ante un ataque.

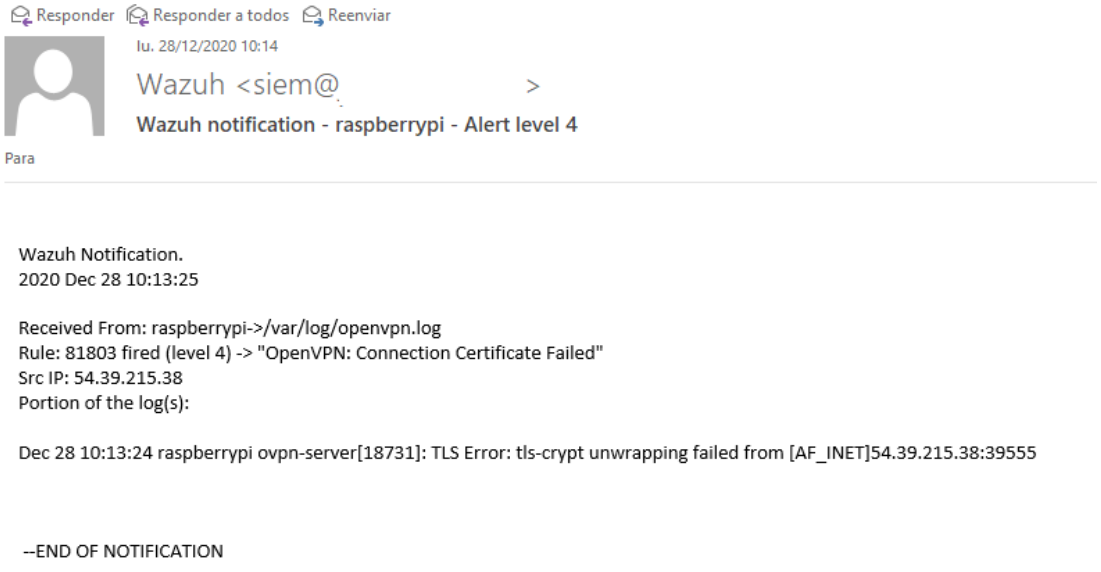


Ilustración 40. Notificación de intento de conexión no identificada. Outlook 2016.

4.5. Detección de vulnerabilidades

Con el despliegue de los agentes en nuestros equipos, además de recopilar los eventos Wazuh, es posible, a través de un módulo implementado por defecto, la posibilidad de realizar escaneos periódicos y automáticos para detectar posibles vulnerabilidades en nuestros sistemas.

Para ello es necesario habilitar la opción dentro del archivo ossec.conf del servidor.

```
<vulnerability-detector>
 <enabled>yes</enabled>
 <interval>5m</interval>
 <ignore_time>6h</ignore_time>
 <run_on_start>yes</run_on_start>
```

Una vez se ha realizado el primer escaneo podremos acceder a la información desde nuestro dashboard para revisar aspectos como la clasificación por criticidad de las alertas, el CVE (Common Vulnerabilities and Exposures) asociado entre otros.

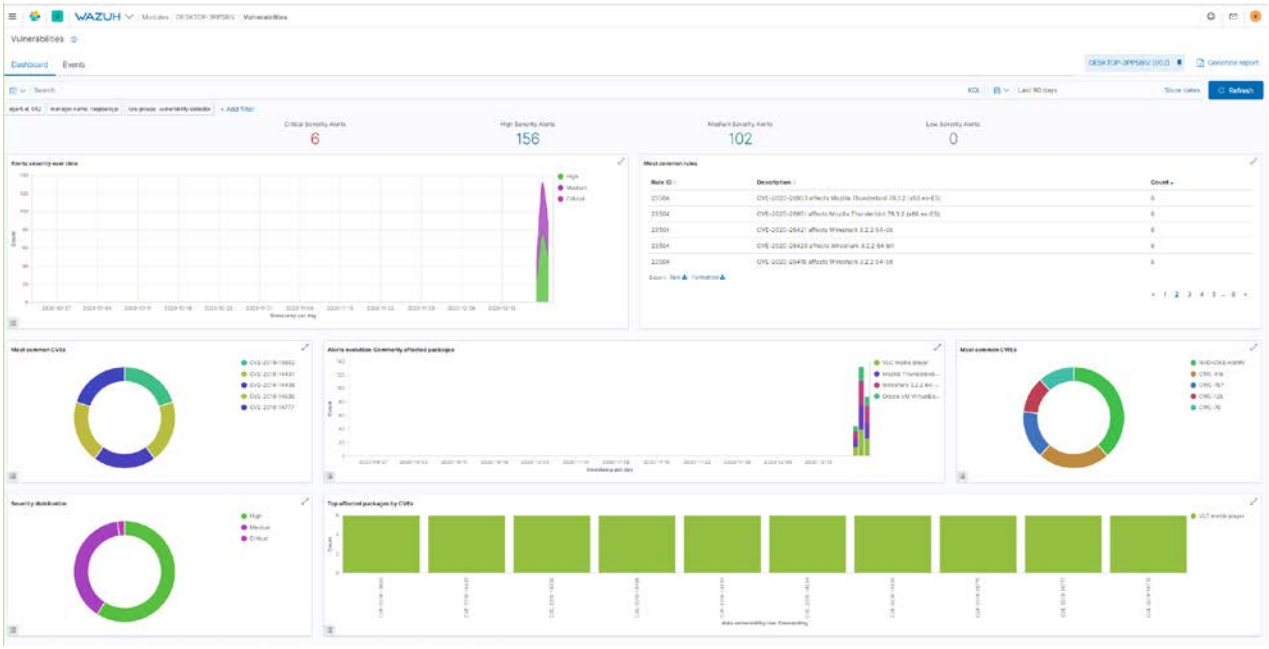


Ilustración 41. Identificación vulnerabilidades Workstation. Kibana / Wazuh.

El CVE determinará la criticidad de las alertas que se basan en las métricas que se definen a través del MITRE, organismo internacionalmente reconocido que permite, a través de su metodología, clasificar las vulnerabilidades en función a diversos parámetros.

4.6. Notificación de alertas vía correo electrónico

Uno de los puntos más importantes del SIEM es la generación de alertas mediante las cuales el analista podrá tratar el evento que ha generado la misma.

En este caso la alerta llegará vía email. Dada la cantidad de eventos que recoge nuestro sistema será imprescindible filtrar los mismos para tratar aquellas que realmente sean interesantes bien por la fuente de la cual provienen o el nivel de criticidad de dichas alertas.

En primer lugar, se debe configurar el servicio de correo, el cual se realiza a través de la herramienta Postfix. En el Anexo II se detalla la operación realizada para dejar operativo nuestro mail.

Por defecto Wazuh tiene deshabilitadas las alertas vía correo electrónico por lo que es necesario modificar la configuración del fichero ossec.conf para habilitar los envíos.

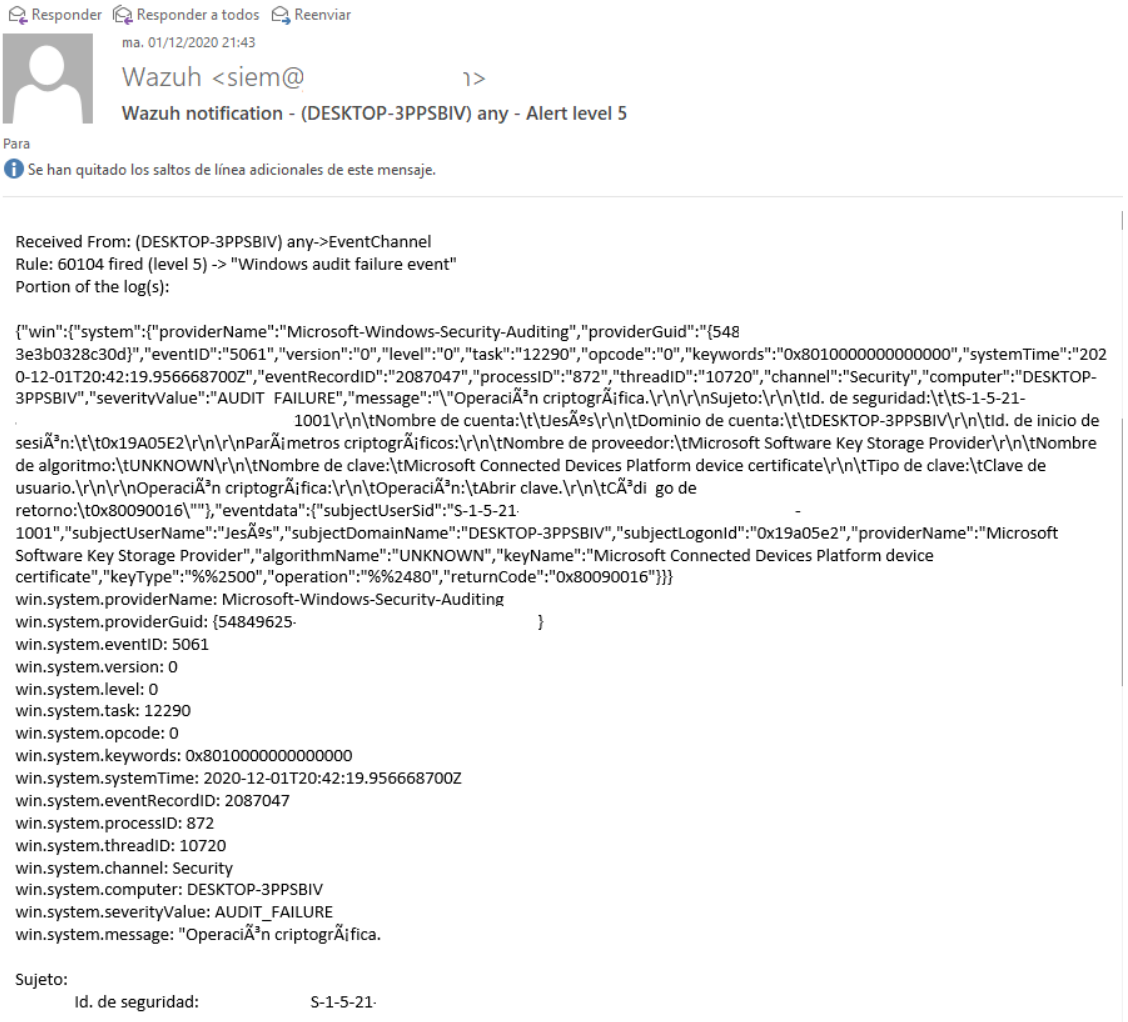
```
<ossec_config>
 <global>
  <jsonout_output>yes</jsonout_output>
  <alerts_log>yes</alerts_log>
  <logall>no</logall>
  <logall_json>no</logall_json>
  <email_notification>yes</email_notification>
  <smtp_server>localhost</smtp_server>
  <email_from>remitente@correo.com</email_from>
```

```
 <email_to>destinatario@correo.com</email_to>
 <email_maxperhour>70</email_maxperhour>
 <email_log_source>alerts.log</email_log_source>
</global>
```

Tras ello, teniendo como objetivo el tratamiento de las alertas se definen como umbral para el envío del correo aquellas alertas que tengan un nivel 6 o superior.

```
<alerts>
 <log_alert_level>6</log_alert_level>
 <email_alert_level>6</email_alert_level>
</alerts>
```

Forzamos un evento especificando una contraseña incorrecta a la hora de iniciar sesión con un equipo Windows. Al momento podemos comprobar como recibimos el correo electrónico avisando del evento ocurrido.


Responder Responder a todos Reenviar

ma. 01/12/2020 21:43

Wazuh <siem@ >

Wazuh notification - (DESKTOP-3PPSBIV) any - Alert level 5

Para

Se han quitado los saltos de línea adicionales de este mensaje.

Received From: (DESKTOP-3PPSBIV) any->EventChannel
Rule: 60104 fired (level 5) -> "Windows audit failure event"
Portion of the log(s):

{"win":{"system":{"providerName":"Microsoft-Windows-Security-Auditing","providerGuid":"{548
3e3b0328c30d}","eventID":"5061","version":"0","level":"0","task":"12290","opcode":"0","keywords":"0x8010000000000000","systemTime":"202
0-12-01T20:42:19.956668700Z","eventRecordID":"2087047","processID":"872","threadID":"10720","channel":"Security","computer":"DESKTOP-
3PPSBIV","severityValue":"AUDIT_FAILURE","message":"\"OperaciÃ³n criptogrÃ¡fica.\r\n\r\nSujeto:\r\n\tId. de seguridad:\t\tS-1-5-21-
1001\r\n\tNombre de cuenta:\t\tJesÃºs\r\n\tDominio de cuenta:\t\tDESKTOP-3PPSBIV\r\n\tId. de inicio de
sesiÃ³n:\t\t0x19A05E2\r\n\r\nParÃ¡metros criptogrÃ¡ficos:\r\n\tNombre de proveedor:\tMicrosoft Software Key Storage Provider\r\n\tNombre
de algoritmo:\tUNKNOWN\r\n\tNombre de clave:\tMicrosoft Connected Devices Platform device certificate\r\n\tTipo de clave:\tClave de
usuario.\r\n\r\nOperaciÃ³n criptogrÃ¡fica:\r\n\tOperaciÃ³n:\tAbrir clave.\r\n\tCÃ³di go de
retorno:\t0x80090016\""},"eventdata":{"subjectUserSid":"S-1-5-21- -
1001","subjectUserName":"JesÃºs","subjectDomainName":"DESKTOP-3PPSBIV","subjectLogonId":"0x19a05e2","providerName":"Microsoft
Software Key Storage Provider","algorithmName":"UNKNOWN","keyName":"Microsoft Connected Devices Platform device
certificate","keyType":"%%2500","operation":"%%2480","returnCode":"0x80090016"}}}
win.system.providerName: Microsoft-Windows-Security-Auditing
win.system.providerGuid: {54849625- }
win.system.eventID: 5061
win.system.version: 0
win.system.level: 0
win.system.task: 12290
win.system.opcode: 0
win.system.keywords: 0x8010000000000000
win.system.systemTime: 2020-12-01T20:42:19.956668700Z
win.system.eventRecordID: 2087047
win.system.processID: 872
win.system.threadID: 10720
win.system.channel: Security
win.system.computer: DESKTOP-3PPSBIV
win.system.severityValue: AUDIT_FAILURE
win.system.message: "OperaciÃ³n criptogrÃ¡fica.

Sujeto:
Id. de seguridad: S-1-5-21-

Ilustración 42. Notificación de alerta nivel 5. Error en la autenticación de Windows. Outlook 2016.

Este punto es importante para limitar el número de envíos en un determinado envío, pues de lo contrario se podría saturar la red con nuestros mensajes. Si a esto se le suma el uso de un servidor de correo contratado con un tercero, es posible que, como vemos a continuación, se proceda al bloqueo de la cuenta o el puerto de envío, al ser considerada como una actividad inusual (envío de correo malicioso). Como se aprecia, este tipo de aplicaciones emplean patrones de comportamiento para desarrollar determinadas acciones.

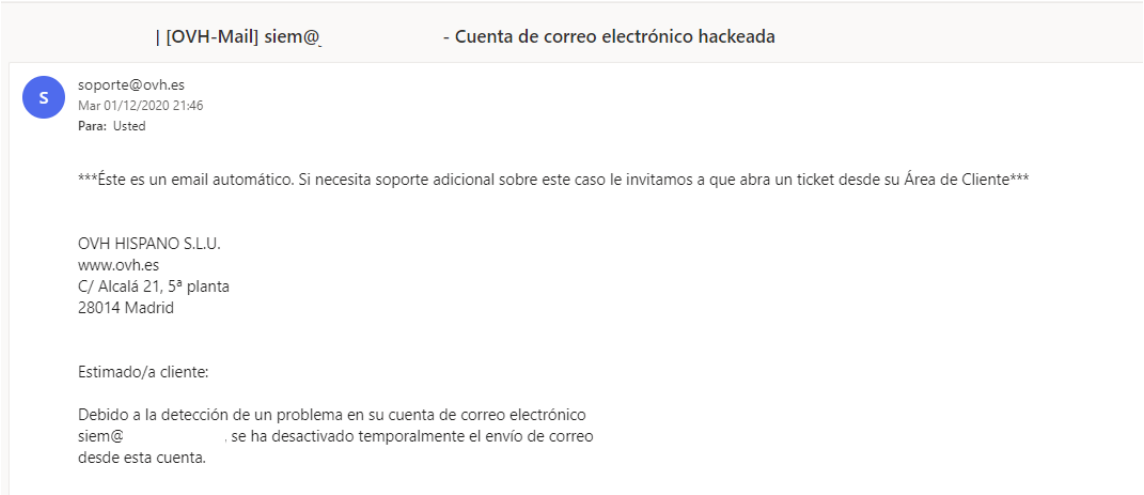


Ilustración 43. Notificación de bloqueo de cuenta. Outlook 2016.

La reiteración de los eventos permitirá al analista depurar la tipolog
ía y la criticidad de alertas con el fin de no evaluar eventos que no conllevan riesgos para el sistema, así como no saturar a los recursos disponibles.

Tras la realización de las pruebas, se decide subir el nivel de alertas hasta 9 con el fin de tratar las más críticas.

4.7. Notificación de alertas a través de un bot de Telegram

Una alternativa al correo cada vez más empleada para diversos usos son las notificaciones mediante la aplicación de mensajería Telegram.

Gracias a su facilidad de uso y a las múltiples posibilidades han hecho que sea una opción muy recomendable para notificar las diversas alertas generadas por nuestro SIEM.

En primer lugar, se crea el bot en Telegram a través del bot **BotFather**, un asistente que sirve de guía en el proceso de configuración de nuestro bot de notificaciones.

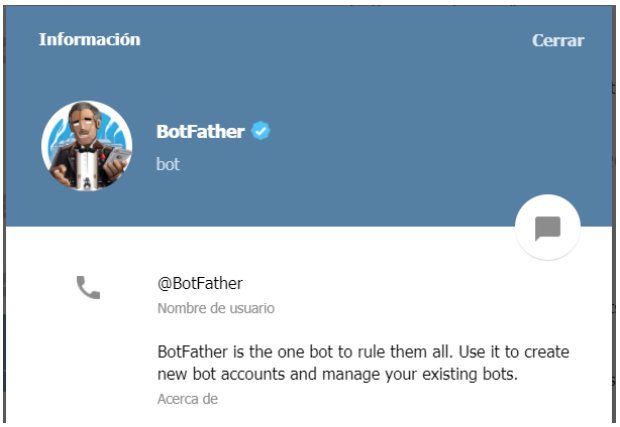


Ilustración 44. Cuenta BotFather. Telegram.

Tras su creación, se modifica el archivo ossec.conf de Wazuh para la ejecución del disparador que envíe la alerta al bot.

Se indica el comando que ejecutará el script y que permitirá enviar la alerta generada a telegram.

```
<command>
  <name>send-event</name>
  <executable>ossec-telegram.sh</executable>
  <expect></expect>
</command>
```

Como parte de la gestión de respuesta activa, se le indica que este evento únicamente se envíe cuando el nivel de criticidad de las alertas sea igual o superior a 7. Con ello, se evitará la recepción de notificaciones de bajo nivel de impacto.

```
<active-response>
  <command>send-event</command>
  <location>local</location>
  <level>7</level>
</active-response>
```

Se crea el script ossec-telegram.sh el cual recolectará las alertas y enviará la notificación a nuestro chat de Telegram.

```
PATH=$PATH:/bin:/sbin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin
# Telegram settings
TOKEN="TELEGRAM-TOKEN"
CHAT_ID="ID_CHAT"

ACTION=$1
USER=$2
IP=$3
ALERTID=$4
```

```
RULEID=$5

LOCAL=`dirname $0`;
cd $LOCAL
cd ../
PWD=`pwd`

# Logging the call
echo "`date` $0 $1 $2 $3 $4 $5 $6 $7 $8" >> ${PWD}/../logs/active-responses.log

# Getting alert time
ALERTTIME=`echo "$ALERTID" | cut -d "." -f 1`

# Getting end of alert
ALERTLAST=`echo "$ALERTID" | cut -d "." -f 2`

# Getting full alert

ALERT=`grep -A 10 "$ALERTTIME" ${PWD}/../logs/alerts/alerts.log | grep -v
"\.$ALERTLAST: " -A 10 | grep -v "Src IP: " | grep -v "User: " |grep "Rule: " -A 4 | cut -c -
139 | sed 's/\"//g'`

curl -s \
-X POST \
https://api.telegram.org/bot$TOKEN/sendMessage \
-d text="$ALERT" \
-d chat_id=$CHAT_ID

ALERT=`grep -A 10 "$ALERTTIME" ${PWD}/../logs/alerts/alerts.log | grep -v
"\.$ALERTLAST: " -A 10 | grep -v "Src IP: " | grep -v "User: " |grep "Rule: " -A 4 | cut -c -
139 | sed 's/\"//g'`
curl -s \
-X POST \
https://api.telegram.org/bot$TOKEN/sendMessage \
-d text="$ALERT" \
-d chat_id=$CHAT_ID
```

Tras darle permisos de ejecución y reiniciar el gestor de Wazuh comienzan a llegar las alertas con anomalías.

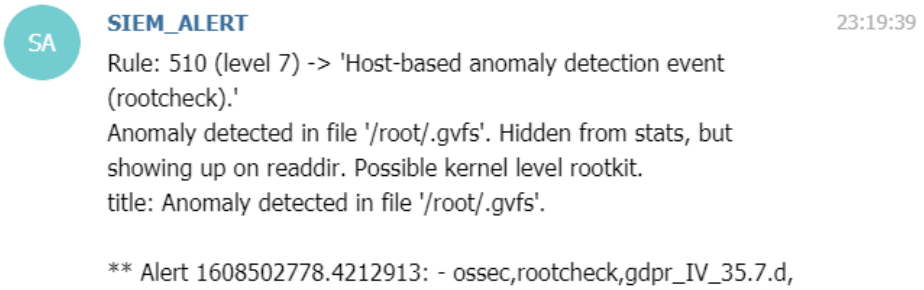


Ilustración 45. Notificación alerta Bot SIEM. Telegram.

4.8. Generación de informes

Como se comentó al inicio de este proyecto, una de las premisas de las que se partían consistía en que el sistema permitiese mostrar información relevante a los diversos grupos de interés tales como analistas o CIOs / CISOs, y que estos obtuvieran información sesgada para la toma de decisiones.

Kibana y el plugin de Wazuh permiten la generación de varios tipos de reporte, así como estadísticas de uso del propio sistema los cuales se pueden exportar para su tratamiento posterior en los formatos estándar.

- Inventario de hardware y software a bajo nivel de un determinado agente.
- Auditoría Compliance de un sistema o conjunto de sistemas.
- Monitorización de políticas.
- Reportes para la verificación del cumplimiento de normativa PCI, GDPR o NIST.
- Informes de Monitorización de la Integridad.
- Informes de eventos de seguridad.
- Informes de Vulnerabilidades de seguridad encontrados en los agentes.


WAZUH

info@wazuh.com
https://wazuh.com

Inventory data report

ID	Name	IP	Version	Manager	OS	Registration date	Last keep alive
005	nextcloud2020	192.168.1.5	Wazuh v4.0.3	raspberrypi	Debian GNU/Linux 10	2020-12-20T11:34:40Z	2020-12-20T17:33:09Z

Applications, network configuration, open ports and processes running on your monitored systems.

Hardware information

- 2 cores
- Intel(R) Celeron(R) CPU G1610T @ 2.30GHz
- 4.00GB RAM

OS information

- Linux
- #1 SMP PVE 4.15.18-58 (Fri, 12 Jun 2020 13:53:01 +0200)
- x86_64
- 4.15.18-30-pve
- Debian GNU/Linux 10 (buster)

Ilustración 46. Inventory data report agente. Wazuh.

4.9. Cumplimiento normativo: GDPR / RGPD

Como se ha comentado en secciones anteriores, uno de los puntos interesantes de este tipo de aplicaciones es la cobertura que permite, desde el punto de vista de IT (Information Technology), ayudar a otras áreas de negocio a cumplir con determinados controles requeridos por las diversas normativas relacionadas con la seguridad de la información o algunas más concretas como la de protección de datos.

Wazuh dispone de un módulo exclusivo para abordar todas aquellas cuestiones que atañan al compliance normativo.

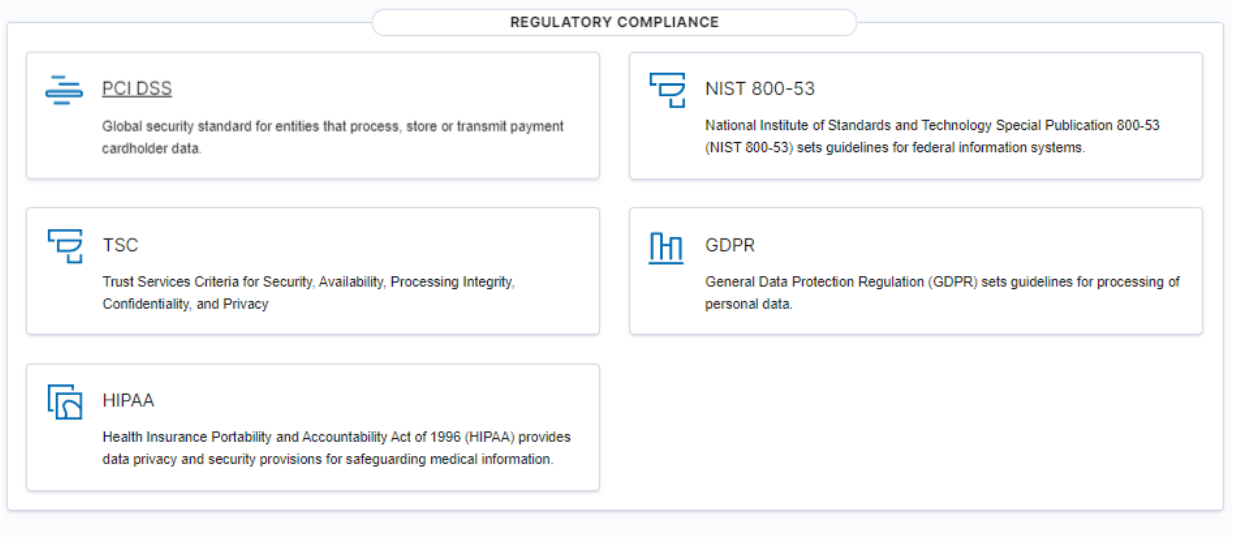


Ilustración 47. Módulo compliance. Kibana / Wazuh.

A lo largo de este trabajo, se han ido recopilando un alto número de eventos. Estos eventos afectan a la normativa de protección de datos nacional (LOPDGDD) y europea (RGPD) que nos es de aplicación.

En la siguiente ilustración se puede conocer el número de eventos que han afectado a un determinado artículo.

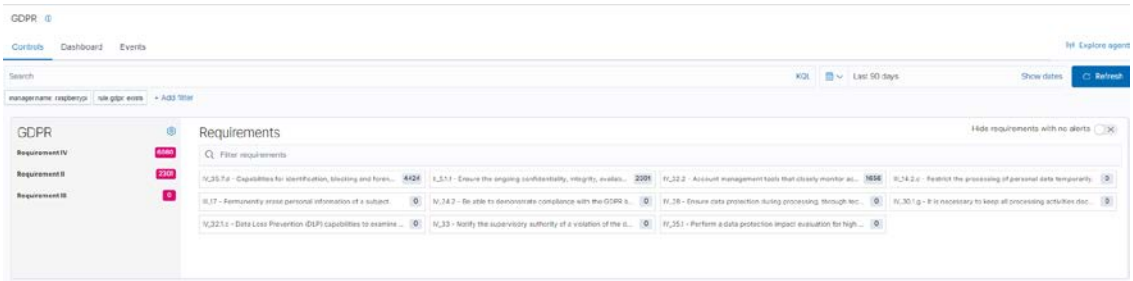


Ilustración 48. Controles RGPD. Fuente: Kibana / Wazuh.

Tras acceder a uno de ellos se comprueban, de forma pormenorizada, aquellos eventos que afectan directamente al artículo seleccionado, así como una descripción de ayuda que permite identificar los requerimientos realizados en el mismo.

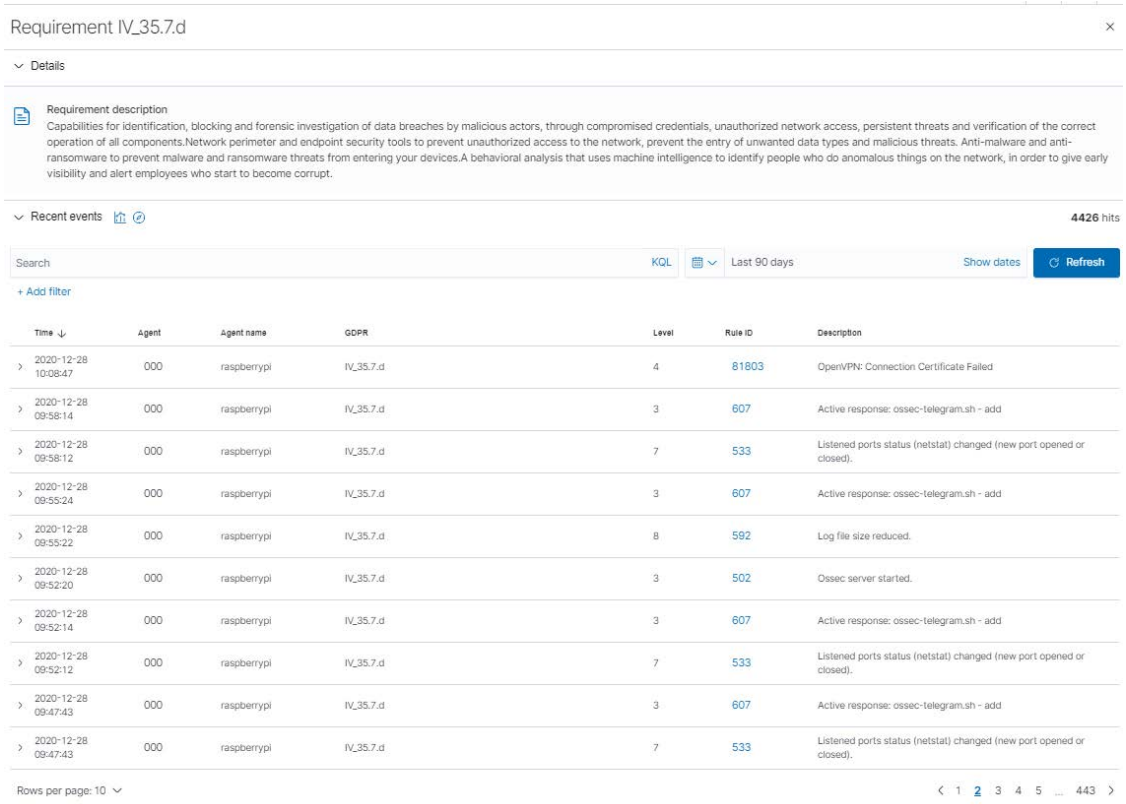


Ilustración 49. Eventos asociados al requerimiento IV_35.7. Kibana / Wazuh

Se muestra, a continuación, un dashboard detallado en el cual se muestra información acerca de los agentes que más alertas tienen, conteo del número de veces que un evento ha sido asociado a un determinado requerimiento (artículo de la normativa).

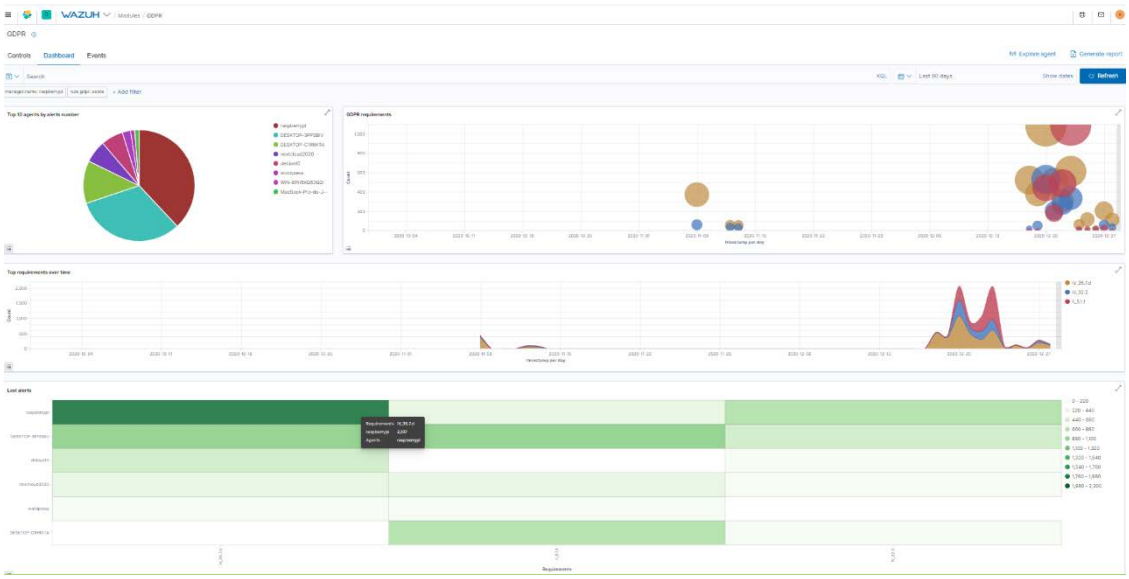


Ilustración 50. Dashboard GDPR. Kibana / Wazuh.

Por último, este apartado muestra los eventos producidos en los que se puede apreciar a qué artículo está haciendo referencia.

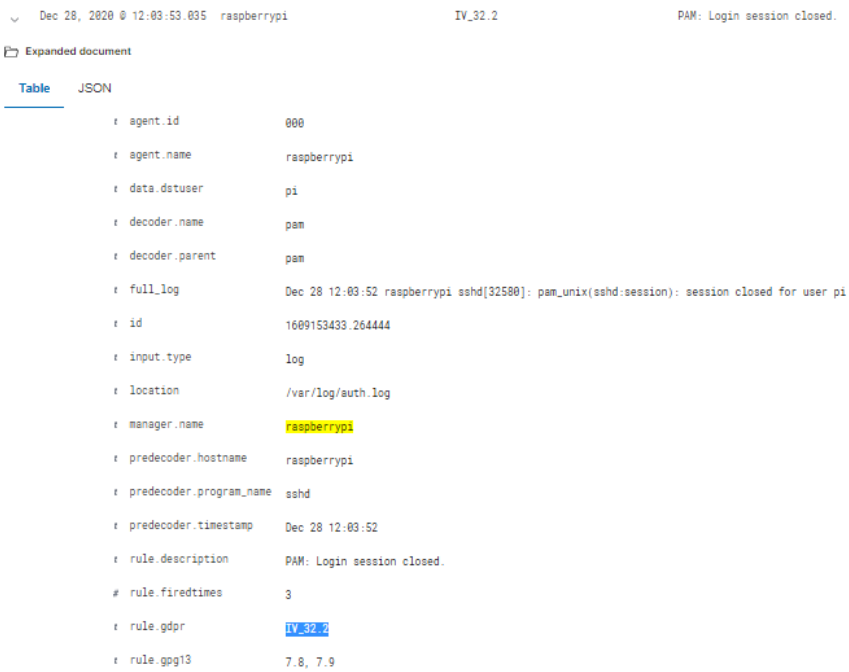


Ilustración 51. Evento de seguridad asociado a art IV 32.3. Kibana / Wazuh.

En Anexo VII se puede comprobar un ejemplo de informe válido para aportar como evidencia en una posible auditoría en relación a esta materia.

5. Conclusiones

La tecnología de E*dge Computing* se plantea como un complemento perfecto para la computación en la nube, gracias a las ventajas que proporciona el hecho de que un sistema se encuentre dentro de la misma red que sensores y dispositivos. Entre estas ventajas como se ha podido comprobar se encuentra la privacidad y la seguridad que proporciona el establecimiento de una conexión rápida y fiable sobre la que desplegar servicios como puede ser una nube privada a la que acceder a través de un túnel VPN desde el exterior si fuera necesario.

En este trabajo se han definido los componentes principales que integran un SOC desarrollando uno de los elementos principales, el SIEM a través de la herramienta de código abierto Wazuh.

Kibana, gracias a su potente motor gráfico permite la definición de cuadros de mando integral con el que poder conocer, en tiempo real, el estado de seguridad de nuestra red y de los componentes conectados a ella. Como se ha podido comprobar en algunos puntos de la prueba de concepto, son múltiples las posibilidades que nos brinda la aplicación tanto para tratar los datos de Wazuh como de otras fuentes de recursos. Debido a la extensión del trabajo, se deja definido en puntos a desarrollar a futuro.

Las integraciones de diversos elementos de seguridad presentan una problemática en cuanto al tratamiento unificado de la información que éstos generan. Con ellos se ha conseguido en la prueba de concepto y, en especial gracias a Elasticsearch, parsear los *log*s y homogeneizar los mismos para que éstos hayan podido ser tratados de igual forma. Ello permite, además de presentar los datos, ahorrar bastante tiempo en el tratamiento de los mismos.

En lo referente a la escalabilidad, es posible descentralizar los distintos módulos de la pila ELK, rotar *log*s con una mayor frecuencia y derivar los mismos a un sistema de almacenamiento adaptado a las necesidades de cada momento consiguiendo así, una mejora en la seguridad de estos registros.

La problemática con la que se encuentran los administradores de sistemas relativos al tratamiento de *log*s de diversas fuentes se puede afirmar que en un alto grado ha sido solventada gracias a una gestión integral y homogeneizada de los registros generados por los diversos sistemas.

¿Dónde radicará el principal inconveniente? Como en este tipo de proyectos, en la planificación e investigación para conseguir que los *log*s de los diversos sistemas sean legibles.

A través de la herramienta Pi-hole se ha comprobado cómo, además de filtrar el contenido y más concretamente la publicidad, ha permitido visualizar y bloquear el

tráfico de la red de todos los dispositivos que tienen como servidor de resolución de nombres de dominio al equipo sobre el que gira este proyecto.

Este sistema ha permitido mostrar cómo, a golpe de clic, las peticiones que realizan dispositivos. Cobra especial relevancia en la instalación de éste en redes con dispositivos IoT los cuales son cada vez más frecuentes tanto en redes empresariales como personales.

El empleo de OpenVPN como solución para establecer una VPN es una opción muy interesante que nos ha permitido cumplir con varios puntos entre los que se encuentran:

- Posibilitar de incluir bajo el mismo paraguas de protección a dispositivos ubicadas en distintas localizaciones.
- Solución muy a tener en cuenta para implementar en organizaciones y hogares con el fin de establecer un canal seguro de comunicación con fines, por ejemplo, como el teletrabajo.
- Acceder a los dispositivos que se han estado empleando a lo largo del TFM desde diversas localidades.
- Integrar como fuente de datos los *log*s generados para establecer mecanismos de control a la red interna.

Gracias a los sistemas de Elastic se han podido recopilar, tratar y presentar de forma gráfica los datos recabados por el SIEM de tal forma que los mismos ayudan a los analistas, administradores de sistemas y responsables tales como CIO's CISO's comprobar el estado actual de seguridad de un determinado entorno, el nivel de cumplimiento al cual se llega o la ubicación de la que provienen las diversas amenazas.

A lo largo de los últimos años personalmente hemos podido comprobar, desde el punto de vista de auditor de sistemas de seguridad de la información, la ausencia de medidas de seguridad implementadas en los sistemas, junto con la libertad de interpretación en el caso de la RGPD o LOPDGDD en cuanto a medidas técnicas hacen que este tipo de soluciones sean perfectas para dar cumplimiento a artículos, dominios y controles de las diversas normativas relacionadas con la seguridad de la información.

Las pruebas llevadas a cabo han servido para comprobar la potencia y la versatilidad que un SIEM y más concretamente Wazuh nos ofrece solventando la problemática planteada. Es por tanto opción que, gracias a su capacidad de integración con la suite Elastic y a ser una solución Open Source que se puede desplegar sobre un hardware de bajo coste.

Para la prueba de concepto se emplea un hardware de bajo coste el cual puede llegar a plantear, a la larga, limitaciones en cuanto a la capacidad de almacenamiento a medida que avanza el tiempo o se incrementa el número de agentes. Para ello, queda definido un punto de mejora en la que se valora la posibilidad de ampliar la capacidad o de optar por un modelo en el que los componentes estén descentralizados.

Tras evaluar los costes, es posible afirmar que nos encontramos ante una posibilidad muy a tener en cuenta por su bajo coste –no llega a superar los 100€ en total (placa Raspberry Pi de 8 GB de RAM, memoria, adaptador de corriente y micro HDMI) en su modelo base. Este minicomputador ha rendido correctamente durante todas las pruebas realizadas. Faltará por depurar una versión del sistema operativo y mejorar la compatibilidad de las aplicaciones con el tipo de arquitectura sobre el que éste está diseñado para añadir nuevas funcionalidades al sistema.

Tras comprobar los requisitos tanto a niveles de software como de hardware y verificar las distintas arquitecturas existentes podemos decir que estamos ante un sistema escalables y replicables. Se trata de dos factores determinantes en la elección de un determinado sistema para conocer si es viable o no su implantación.

Se espera que la prueba de concepto desarrollada sea válida para abordar la problemática de red actual y tenga aplicabilidad tanto para fines educativos, personales como empresariales.

Para terminar, y a modo de reflexión global, con el crecimiento del IoT se abren nuevas posibilidades. Debido a la forma en la que está pensada esta tecnología lleva intrínsecamente aparejado el incremento de unos riesgos que afectan cada vez más a la privacidad de los usuarios que ven cómo ésta es vulnerada cada vez con más intensidad y frecuencia.

Gran parte de la culpa reside en la ausencia de la cultura de la privacidad y la seguridad de los datos la cual deberíamos trabajar en conjunto antes de que sea demasiado tarde.

Esta cuestión daría para otro interesante trabajo.

6. Líneas de trabajo futuro

Debido a la limitación en cuanto a tiempo y extensión del ensayo no se han podido desarrollar tantas opciones como nos hubiera gustado. Pese a la entrega del mismo, definimos una serie de líneas de actuación para perfeccionar e incrementar el número de funcionalidades que el sistema desarrollado dispone hasta la fecha.

Administración del sistema

Gestión de copias de seguridad del sistema. Queda pendiente por definir un procedimiento que permita la creación de imágenes del sistema (snapshots) así como copias periódicas de las partes más críticas del sistema. Elasticsearch ya contempla la opción por lo que se deberá valorar esta funcionalidad u optar por otras alternativas que Linux nos proporciona.

Configuración de alta disponibilidad (HA). Con el fin de cumplir con dos de los principios básicos de la seguridad de la información (disponibilidad e integridad) se valorará la opción de crear un clúster con otro sistema idéntico para mejorar la fiabilidad del sistema implementado.

Supervisión del estado del sistema. A fecha de finalización del presente proyecto nos encontramos con un servicio instalado (RpiMonitor) que permite la monitorización de los principales recursos del sistema. Queda pendiente la revisión para optimizar los recursos y detectar posibles anomalías en el rendimiento de los componentes.

Inclusión de disco SSD para la gestión del sistema y la mejora en el rendimiento. El sistema de almacenamiento sobre el que se ha realizado la instalación consiste en una tarjeta microsd de 64 gb. Lo deseable, para mejorar el rendimiento y la fiabilidad de los sistemas pasa por la migración del contenido alojado en el actual soporte a un disco duro sólido (Solid State Disk, SSD).

Externalización y protección de los *log*s. Uno de los principales –sino el principal- componentes sobre el que gira el SIEM son los *log*s. Con el fin de evitar la pérdida de información, así como la integridad de la misma se valorará la opción de tratar los registros de las diferentes fuentes en un repositorio externo al SIEM que cuente con las medidas suficientes para evitar la modificación del contenido de los mismos. Ello servirá, para dar cumplimiento a aspectos de compliance, más concretamente para aquellos posibles casos de auditoría / forense.

Funcionalidades de Wazuh

Integración de nuevas fuentes de monitorización. Queda pendiente por incluir nuevas fuentes de monitorización, principalmente aquellas para las cuales no se dispone de una aplicación como es el caso de Android, equipos de red o dispositivos IoT.

Integración del SIEM en una plataforma que permita el tratamiento de los incidentes. Para *log*rar llevar un control más eficaz de los casos generado valoraremos la posibilidad

de integrar las alertas con un sistema como puede ser el TheHive que permita aprovechar, en pro de automatizar los procesos, la apertura de casos a través de la información proporcionada directamente del SIEM o bien a través de mail o Telegram como hemos visto a lo largo de la prueba de concepto.

Desarrollo de cuadro de mandos integral de ciberseguridad. El plugin de Wazuh desarrollado por Kibana permite establecer un dashboard predeterminado con las funcionalidades más relevantes del SIEM. Quedará por definir nuevos paneles personalizados a través de Kibana en los que se pueda mostrar tanto información de Wazuh como de diversas fuentes (normativo, compliance, reportes del malware, etc…) que permitan construir un marco de control de ciberseguridad integral.

Desarrollo de conocimiento a través de Machine Learning. El entorno de la ciberseguridad es totalmente cambiante por lo que el estudio, la adaptación y la implementación de nuevas funcionalidades a través de esta tecno*logí*a permitirá mejorar la precisión de nuestro sistema.

Creación de nuevas alertas personalizadas. La incorporación de nuevas fuentes, así como las necesidades cambiantes en cuanto a compliance, por ejemplo, hará que sea necesario el tratamiento y la creación de nuevas notificaciones de alerta. Se estudiará la creación de plantillas tipo para determinados grupos de eventos.

Mejora en la respuesta activa ante incidentes. Desarrollar casos de uso y scripts de respuesta que permitan incrementar el nivel de madurez de seguridad de nuestros sistemas.

7. Bibliografía

Referencias bibliográficas

A. Sforzin, F. G. Mármol, M. Conti and J. Bohli, "RPiDS: Raspberry Pi IDS — A Fruitful Intrusion Detection System for IoT," 2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld), Toulouse, 2016, pp. 440-448, doi: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0080.

K. Detken, T. Rix, C. Kleiner, B. Hellmann and L. Renners, "SIEM approach for a higher level of IT security in enterprise networks," 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Techno*log*y and Applications (IDAACS), Warsaw, 2015, pp. 322-327, doi: 10.1109/IDAACS.2015.7340752.

M. Satyanarayanan, "The Emergence of Edge Computing," in Computer, vol. 50, no. 1, pp. 30-39, Jan. 2017, doi: 10.1109/MC.2017.9.

R.J. Millán, "Conectrónica nº 204, GM2 Publicaciones técnicas, 2017". 5G La red móvil que habilitará loT.

R.Montesino,, Fenz, S. and Baluja, W. (2012), "SIEM-based framework for security controls automation", Information Management & Computer Security, Vol. 20 No. 4, pp. 248-263

S. Bhatt, P. K. Manadhata and L. Zomlot, "The Operational Role of Security Information and Event Management Systems," in IEEE Security & Privacy, vol. 12, no. 5, pp. 35-41, Sept.-Oct. 2014, doi: 10.1109/MSP.2014.103.

Premsankar, G., Di Francesco, M., & Taleb, T. (2018). Edge computing for the Internet of Things: A case study. IEEE Internet of Things Journal, 5(2), 1275-1284.

Van Os, R.M. (2016). SOC-CMM: Designing and Evaluating a Tool for Measurement of Capability Maturity in Security Operations Centers (Dissertation). Recuperado desde: http://urn.kb.se/resolve?urn=urn:nbn:se:ltu:diva-59591

Referencia normativa

Reglamento General de Protección de Datos (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016.

Ley Orgánica de Protección de Datos y Garantía de los Derechos Digitales, (LO 3/2018 de 8 de diciembre).

ISO 27001:2013. Normativa de Seguridad de la Información.

Webgrafía

- **Artículos**

Qué es el Edge Computing, explicado de manera sencilla. Carlos Rebato (2020, 27/11/2020, 20:30. Recuperado desde: https://empresas.b*log*thinkbig.com/edge-computing-que-es/

Mejorar la analítica de Seguridad con el Elastic Stack, Wazuh e IDS. Elastic, 10/10/2020, 15:25. Recuperado desde: https://www.elastic.co/es/b*log*/improve-security-analytics-with-the-elastic-stack-wazuh-and-ids

Threat Landscape 2020 - List of top 15 threats. ENISA, 19/11/2010, 19:30. Recuperado desde: https://www.enisa.europa.eu/publications/enisa-threat-landscape-2020-list-of-top-15-threats

- **Manuales**

Install Kibana with Debian Package. Elastic. 17/11/2020 10:20. Recuperado desde: https://www.elastic.co/guide/en/kibana/7.9/deb.html#deb-repo

The Open Source Security Platform. Wazuh,17/09/2020 20:30. Recuperado desde: https://www.wazuh.com/ | https://documentation.wazuh.com/

VPN Tunnel. OpenVPN, 18/09/2020, 19:45. Recuperado desde: https://www.pivpn.io/

Pi-hole. 19/09/2020, 10:20. Recuperado desde: https://www.pivpn.io/

8. Anexos

I. Securización de la Suite Elastic

Con el fin de securizar el sistema se llevará a cabo la creación de certificados y se modificará la configuración para que el contenido que se envía vaya cifrado. Adicionalmente, se configurará a través de la utilidad X-Pack la interfaz para que sea necesario el inicio de sesión antes de acceder al sistema.

En primer lugar, se crean y se comprimen los certificados.

```
# /usr/share/elasticsearch/bin/elasticsearch-certutil cert ca --pem --in instances.yml --out certs.zip
```

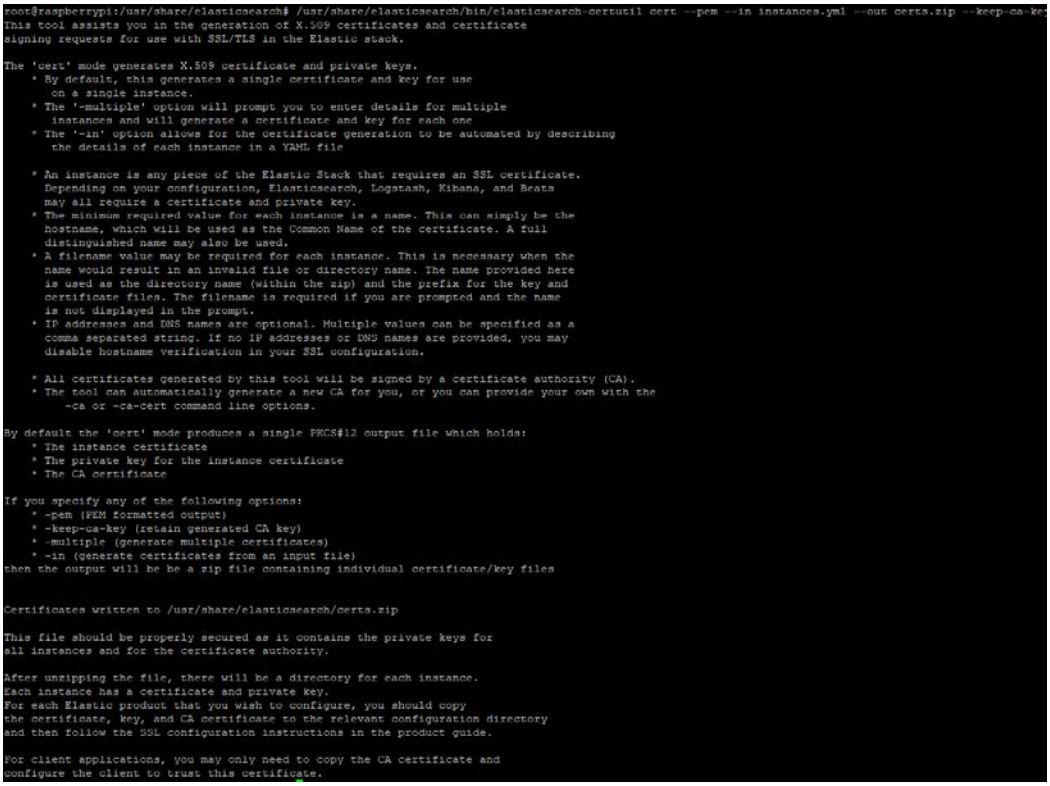


Ilustración 52. Generación de certificados. Raspberry Pi OS.

Se descomprimen los archivos y se alojan en la raíz /usr/share/elasticsearch.

```
unzip /usr/share/elasticsearch/certs.zip -d /usr/share/elasticsearch/
```

```
root@raspberrypi:/usr/share/elasticsearch# unzip /usr/share/elasticsearch/certs.zip -d /usr/share/elasticsearch/
Archive:  /usr/share/elasticsearch/certs.zip
   creating: /usr/share/elasticsearch/ca/
  inflating: /usr/share/elasticsearch/ca/ca.crt
  inflating: /usr/share/elasticsearch/ca/ca.key
   creating: /usr/share/elasticsearch/wazuh-manager/
  inflating: /usr/share/elasticsearch/wazuh-manager/wazuh-manager.crt
  inflating: /usr/share/elasticsearch/wazuh-manager/wazuh-manager.key
   creating: /usr/share/elasticsearch/elasticsearch/
  inflating: /usr/share/elasticsearch/elasticsearch/elasticsearch.crt
  inflating: /usr/share/elasticsearch/elasticsearch/elasticsearch.key
   creating: /usr/share/elasticsearch/kibana/
  inflating: /usr/share/elasticsearch/kibana/kibana.crt
  inflating: /usr/share/elasticsearch/kibana/kibana.key
```

Ilustración 53. Alojamiento de certificados. Raspberry Pi OS.

Securización de Elasticsearch

Se crea la carpeta de certificados dentro de la ruta de configuración de Elasticsearch para posteriormente alojar los certificados. Se asignan los permisos correspondientes a la carpeta que alojará los ficheros.

```
mkdir /etc/elasticsearch/certs/ca -p
# cp /usr/share/elasticsearch/ca/ca.crt /etc/elasticsearch/certs/ca
# cp /usr/share/elasticsearch/elasticsearch/elasticsearch.crt /etc/elasticsearch/certs
# cp /usr/share/elasticsearch/elasticsearch/elasticsearch.key /etc/elasticsearch/certs
# chown -R elasticsearch: /etc/elasticsearch/certs
# chmod -R 770 /etc/elasticsearch/certs
```

Una vez se han creado y almacenado los certificados, se procede a la configuración de las opciones de X-Pack. La configuración a incluir en el fichero elasticsearch.yml es la siguiente:

```
xpack.security.transport.ssl.enabled: true
xpack.security.transport.ssl.verification_mode: certificate
xpack.security.transport.ssl.key: /etc/elasticsearch/certs/elasticsearch.key
xpack.security.transport.ssl.certificate: /etc/elasticsearch/certs/elasticsearch.crt
xpack.security.transport.ssl.certificate_authorities:
[ "/etc/elasticsearch/certs/ca/ca.crt" ]
xpack.security.http.ssl.enabled: true
xpack.security.http.ssl.verification_mode: certificate
xpack.security.http.ssl.key: /etc/elasticsearch/certs/elasticsearch.key
xpack.security.http.ssl.certificate: /etc/elasticsearch/certs/elasticsearch.crt
xpack.security.http.ssl.certificate_authorities: [ "/etc/elasticsearch/certs/ca/ca.crt" ]
xpack.security.enabled: true
```

Reiniciamos el servicio

```
systemctl restart elasticsearch
```

Securización de Filebeat

Se copian los certificados a la carpeta de configuración de Filebeat

```
mkdir /etc/filebeat/certs/ca -p
# cp /usr/share/elasticsearch/ca/ca.crt /etc/filebeat/certs/ca
# cp /usr/share/elasticsearch/elasticsearch/elasticsearch.crt /etc/filebeat/certs
# cp /usr/share/elasticsearch/elasticsearch/elasticsearch.key /etc/filebeat/certs
# chown -R elasticsearch: /etc/ filebeat /certs
# chmod -R 770 /etc/ filebeat/certs
```

Para ello, tras haber copiado los certificados en la ruta correspondiente cambiamos la configuración del archivo filebeat.yml para que muestre la siguiente información:

```
output.elasticsearch.hosts: ['192.168.1.2:9200']
output.elasticsearch.protocol: https
output.elasticsearch.ssl.certificate: "/etc/filebeat/certs/wazuh-manager.crt"
output.elasticsearch.ssl.key: "/etc/filebeat/certs/wazuh-manager.key"
output.elasticsearch.ssl.certificate_authorities: ["/etc/filebeat/certs/ca/ca.crt"]
output.elasticsearch.username: "elastic"
output.elasticsearch.password: "password"
```

Se reinicia el servicio:

```
systemctl restart filebeat
```

Securización de Kibana

Se indica la ruta y protocolo de autenticación en Filebeat.

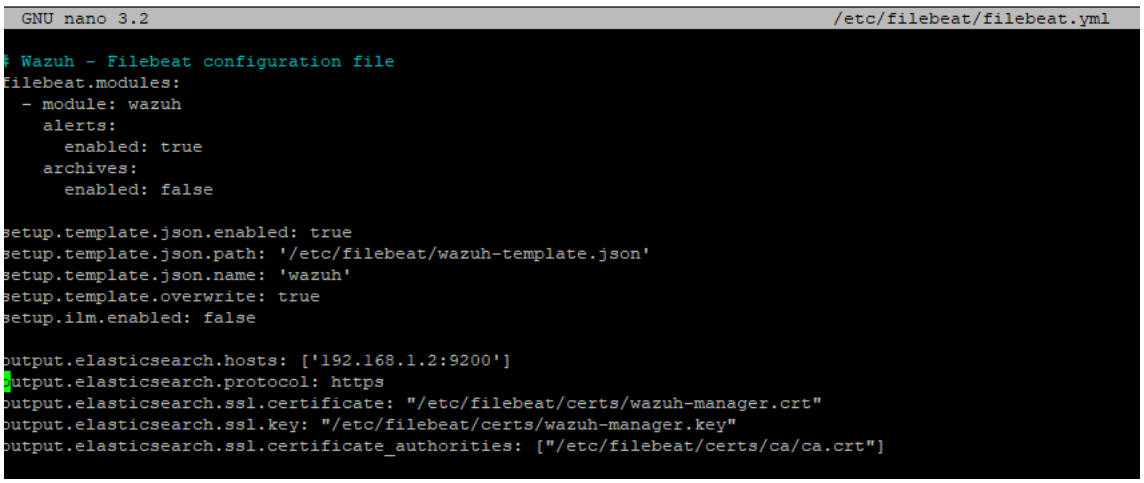


Ilustración 54. Configuración de certificados Filebeat. Raspberry Pi OS.

Se añade la configuración en Kibana, se crean directorios para el almacenaje de los certificados y se asignan permisos a la carpeta /etc/kibana/certs

```
mkdir /etc/kibana/certs/ca -p
# cp /usr/share/elasticsearch/ca/ca.crt /etc/kibana/certs/ca
# cp /usr/share/elasticsearch/kibana/kibana.crt /etc/kibana/certs
# cp /usr/share/elasticsearch/kibana/kibana.key /etc/kibana/certs
# chown -R kibana: /etc/kibana/certs
# chmod -R 770 /etc/kibana/certs
```

```
root@raspberrypi:/usr/share/elasticsearch# cp ca/ca.crt /etc/filebeat/certs/ca
root@raspberrypi:/usr/share/elasticsearch# cp wazuh-manager/wazuh-manager.crt /etc/filebeat/certs
root@raspberrypi:/usr/share/elasticsearch# cp wazuh-manager/wazuh-manager.key /etc/filebeat/certs
root@raspberrypi:/usr/share/elasticsearch# chmod 770 -R /etc/filebeat/certs
root@raspberrypi:/usr/share/elasticsearch# nano /etc/filebeat/filebeat.yml
root@raspberrypi:/usr/share/elasticsearch# service filebeat restart
root@raspberrypi:/usr/share/elasticsearch# mkdir /etc/kibana/certs/ca -p
root@raspberrypi:/usr/share/elasticsearch# cp ca/ca.crt /etc/kibana/certs/ca
root@raspberrypi:/usr/share/elasticsearch# cp kibana/kibana.crt /etc/kibana/certs
root@raspberrypi:/usr/share/elasticsearch# cp kibana/kibana.key /etc/kibana/certs
root@raspberrypi:/usr/share/elasticsearch# chown -R kibana: /etc/kibana/certs
root@raspberrypi:/usr/share/elasticsearch# chmod -R 770 /etc/kibana/certs
```

Ilustración 55. Configuración de certificados Kibana 1. Raspberry Pi OS.

Reiniciamos el servicio

```
systemctl restart kibana
```

Se configura el archivo kibana.yml descomentando las líneas que se aprecian en la siguiente ilustración e indicando las rutas de los certificados.

```
elasticsearch.hosts: ["https://192.168.1.2:9200"]
elasticsearch.ssl.certificateAuthorities: ["/etc/kibana/certs/ca/ca.crt"]
elasticsearch.ssl.certificate: "/etc/kibana/certs/kibana.crt"
elasticsearch.ssl.key: "/etc/kibana/certs/kibana.key"
server.ssl.enabled: true
server.ssl.certificate: "/etc/kibana/certs/kibana.crt"
server.ssl.key: "/etc/kibana/certs/kibana.key"
```

```
 GNU nano 3.2                        /usr/share/kibana/config/kibana.yml                        Modificado

# The URLs of the Elasticsearch instances to use for all your queries.
elasticsearch.hosts: ["https://192.168.1.2:9200"]

#elasticsearch.hosts: 192.168.1.2:9200

# When this setting's value is true Kibana uses the hostname specified in the server.host
# setting. When the value of this setting is false, Kibana uses the hostname of the host
# that connects to this Kibana instance.
#elasticsearch.preserveHost: true

# Kibana uses an index in Elasticsearch to store saved searches, visualizations and
# dashboards. Kibana creates a new index if the index doesn't already exist.
kibana.index: ".kibana"

# The default application to load.
#kibana.defaultAppId: "home"

# If your Elasticsearch is protected with basic authentication, these settings provide
# the username and password that the Kibana server uses to perform maintenance on the Kibana
# index at startup. Your Kibana users still need to authenticate with Elasticsearch, which
# is proxied through the Kibana server.
elasticsearch.username: "kibana_system"
elasticsearch.password: "pass"

# Enables SSL and paths to the PEM-format SSL certificate and SSL key files, respectively.
# These settings enable SSL for outgoing requests from the Kibana server to the browser.
server.ssl.enabled: true
#server.ssl.certificate: /path/to/your/server.crt
server.ssl.certificate: "/etc/kibana/certs/kibana.crt"
#server.ssl.key: /path/to/your/server.key
server.ssl.key: "/etc/kibana/certs/kibana.key"

# Optional settings that provide the paths to the PEM-format SSL certificate and key files.
# These files are used to verify the identity of Kibana to Elasticsearch and are required when
# xpack.security.http.ssl.client_authentication in Elasticsearch is set to required.
#elasticsearch.ssl.certificate: /path/to/your/client.crt
elasticsearch.ssl.certificate: "/etc/kibana/certs/kibana.crt"

#elasticsearch.ssl.key: /path/to/your/client.key
elasticsearch.ssl.key: "/etc/kibana/certs/kibana.key"


# Optional setting that enables you to specify a path to the PEM file for the certificate
# authority for your Elasticsearch instance.
#elasticsearch.ssl.certificateAuthorities: [ "/path/to/your/CA.pem" ]
elasticsearch.ssl.certificateAuthorities: ["/etc/kibana/certs/ca/ca.crt"]

# To disregard the validity of SSL certificates, change this setting's value to 'none'.
#elasticsearch.ssl.verificationMode: full

# Time in milliseconds to wait for Elasticsearch to respond to pings. Defaults to the value of
# the elasticsearch.requestTimeout setting.
#elasticsearch.pingTimeout: 1500
```

Ilustración 56. Configuración de certificados Kibana 2. Kibana.

Generación de credenciales para la autenticación

Por último, se generan las contraseñas de acceso a las diversas interfaces web a los usuarios de cada aplicación.

```
# /usr/share/elasticsearch/bin/elasticsearch-setup-passwords interactive
```

```
root@raspberrypi:/etc/elasticsearch# /usr/share/elasticsearch/bin/elasticsearch-setup-passwords auto
Initiating the setup of passwords for reserved users elastic,apm_system,kibana,kibana_system,logstash_system,beats_system,remote_monitoring_user.
The passwords will be randomly generated and printed to the console.
Please confirm that you would like to continue [y/N]y


Changed password for user apm_system
PASSWORD apm_system = Wz5E1T

Changed password for user kibana_system
PASSWORD kibana_system = nU7sl

Changed password for user kibana
PASSWORD kibana = nU7sl

Changed password for user logstash_system
PASSWORD logstash_system = MEr

Changed password for user beats_system
PASSWORD beats_system = KTN6

Changed password for user remote_monitoring_user
PASSWORD remote_monitoring_user = I72P

Changed password for user elastic
PASSWORD elastic = cHt

root@raspberrypi:/etc/elasticsearch#
```

Ilustración 57. Generación de contraseñas de acceso Elastic. Elasticsearch.

Tras ello ya podremos acceder al entorno securizado.

II. Instalación de Postfix

A continuación, se detallan los pasos seguidos para la instalación del servicio de postfix en el sistema.

```
root@raspberrypi:/home/pi# apt-get install postfix
```

Ilustración 58. Postfix: Instalación por consola. Raspberry Pi OS.

Se selecciona la configuración genérica del correo. En nuestro caso, al utilizar un proveedor de correo externo (OVH) elegimos sitio de internet.

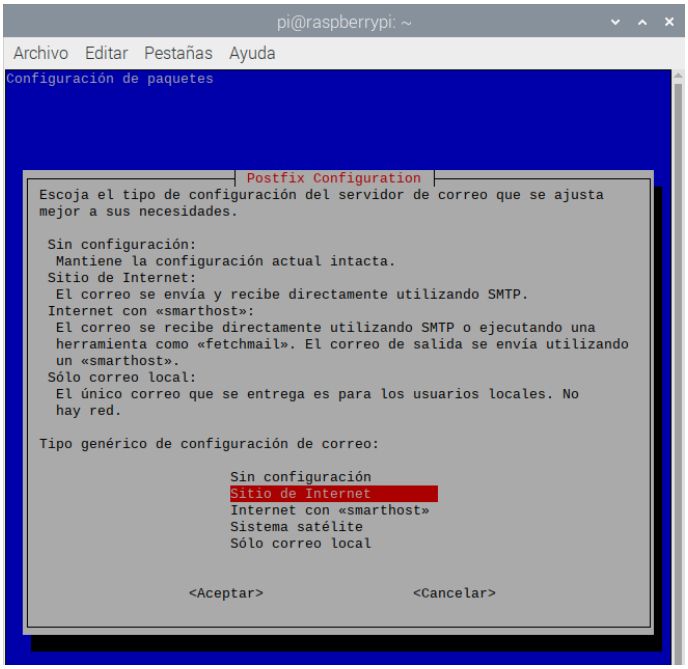


Ilustración 59. Postfix: Selección de tipo de correo Postfix.

Se informa del dominio a emplear.

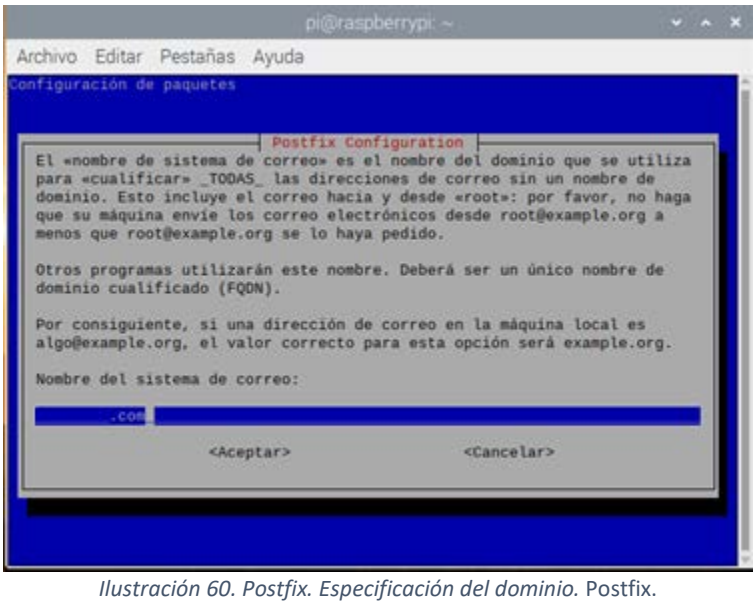


Ilustración 60. Postfix. Especificación del dominio. Postfix.

Se edita el archivo de configuración /etc/postfix/main.cf con las siguientes opciones de configuración.

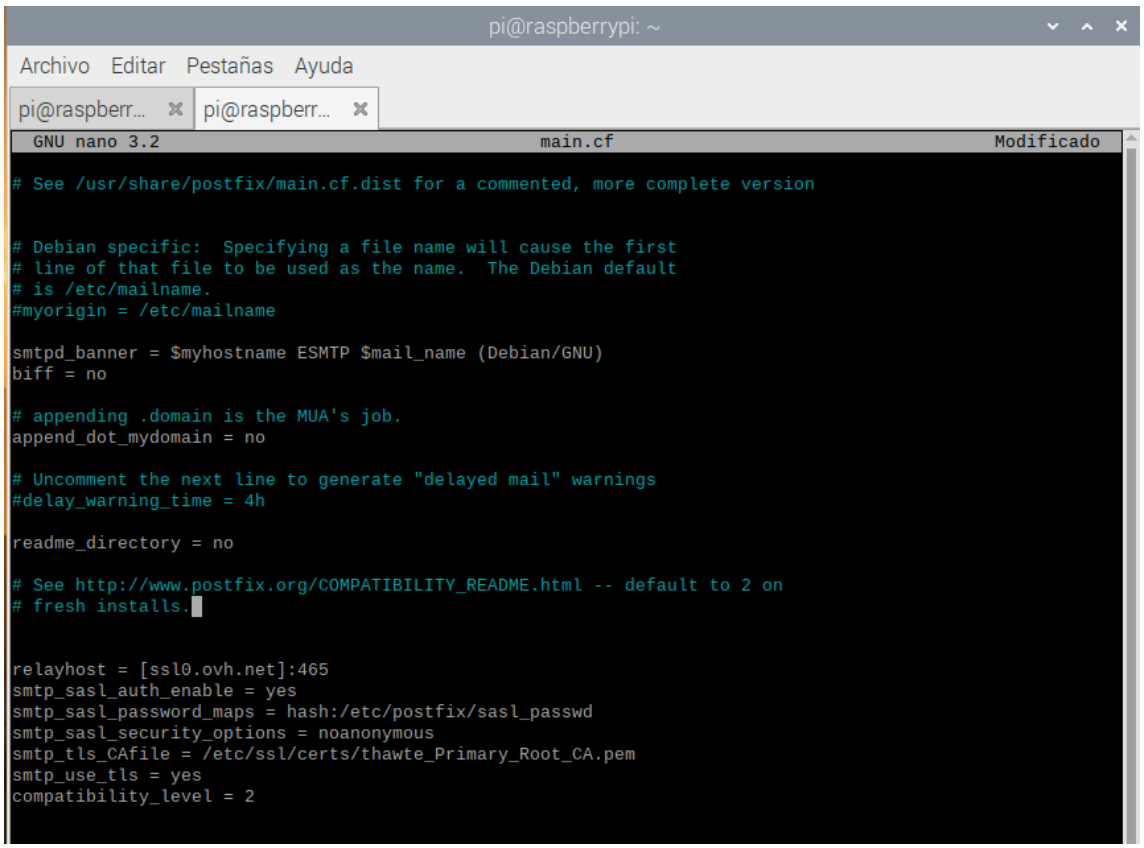


Ilustración 61. Postfix: Configuración de archivo main.cf. Postfix.

Tras ello, se crea el archivo en el que se almacenarán las contraseñas y se le asignan los permisos para acceder al mismo. Se finaliza reiniciando el servicio para que cargue los nuevos parámetros de configuración.

```
root@raspberrypi:/etc/postfix# echo [ssl0.ovh.net]:465 siem@jmorenoc.com:monitorizacion1 > /etc/postfix/sasl_passwd
root@raspberrypi:/etc/postfix# postmap /etc/postfix/sasl_passwd
root@raspberrypi:/etc/postfix# chmod 400 /etc/postfix/sasl_passwd
root@raspberrypi:/etc/postfix# chown root:root /etc/postfix/sasl_passwd /etc/postfix/sasl_passwd.db
root@raspberrypi:/etc/postfix# chmod 0600 /etc/postfix/sasl_passwd /etc/postfix/sasl_passwd.db
root@raspberrypi:/etc/postfix# service postfix restart
```

Ilustración 62. Postfix: Configuración archivo sasl_password y asignación de persmisos. Postfix.

Por último, se realizauna prueba de envío.

```
root@raspberrypi:/home/pi# echo "Test" | mail -s "Test" mail@mail.com
```

Ilustración 63. Postfix. Prueba de envío. Postfix.

Se comprueba que se recibe correctamente.

Ilustración 64. Postfix: Recepción del correo de prueba. Outlook 2016.

III. Instalación de Pi-hole

El proceso de instalación de este servicio se inicia a través de un script de instalación.

```
curl -sSL https://install.pi-hole.net | bash
```

Tras la ejecución por consola del comando anterior inicia el asistente de instalación.

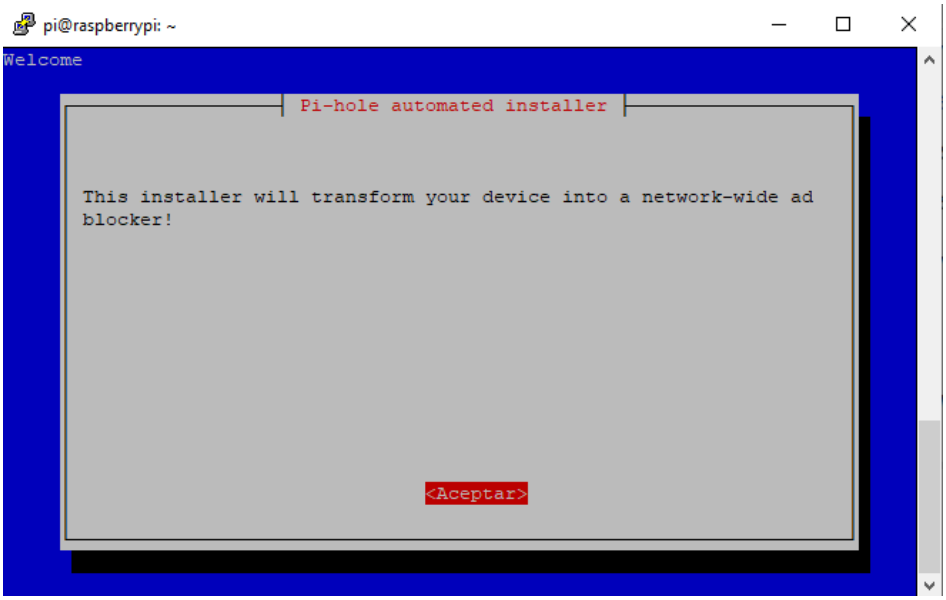


Ilustración 65. Pi-hole: Inicio del asistente. Pi-hole.

Se asigna el adaptador eth0 (red cableada) como interfaz predeterminada para la escucha de las conexiones.

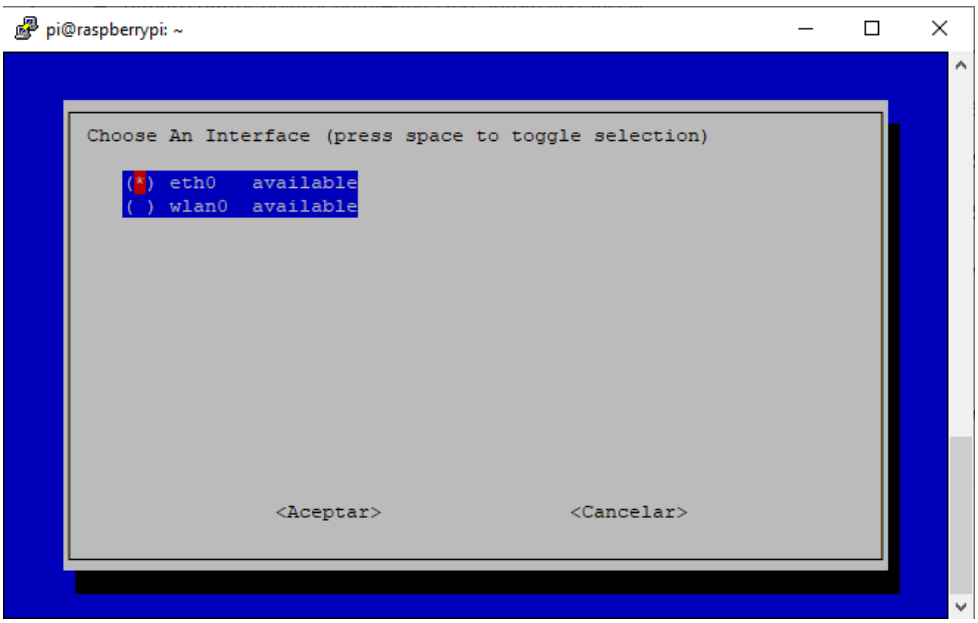


Ilustración 66.Pi-hole:Selección interfaz de red. Pi-hole.

Se incluyen las listas con dominios a bloquear (posteriormente se pueden añadir más).

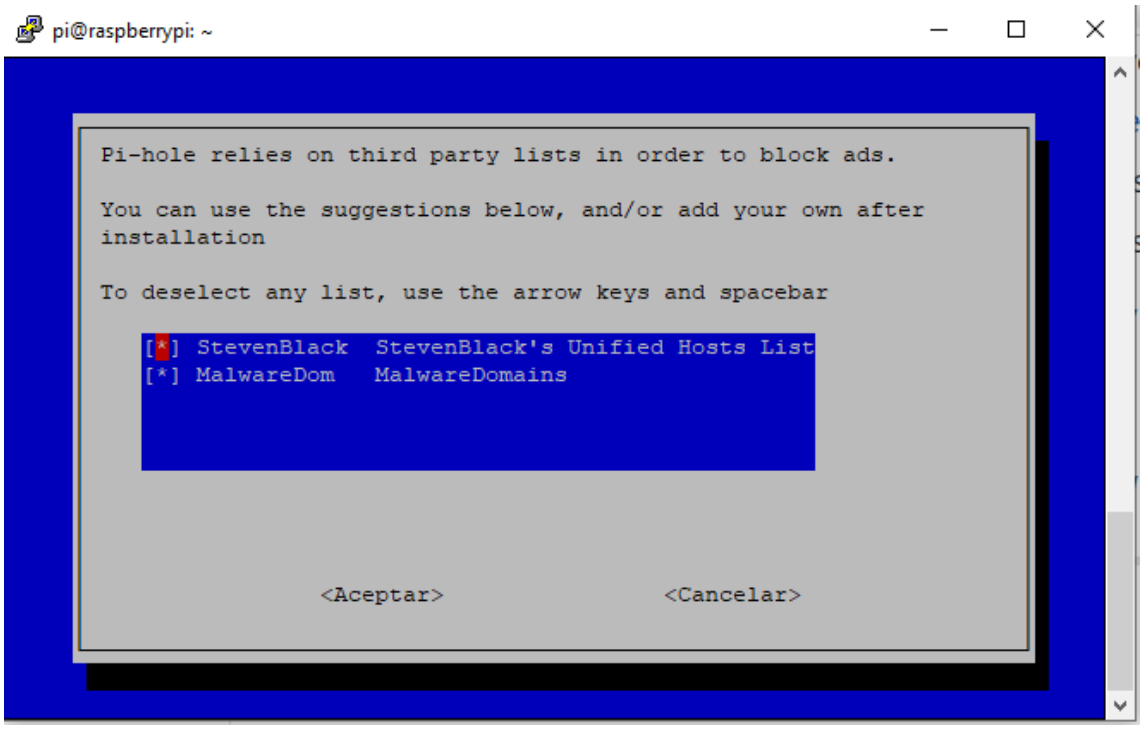


Ilustración 67.Pi-hole:Inclusión de listas de bloqueo. *Pi-hole*

Se seleccionan los protocolos a bloquear, en este caso, ambos.

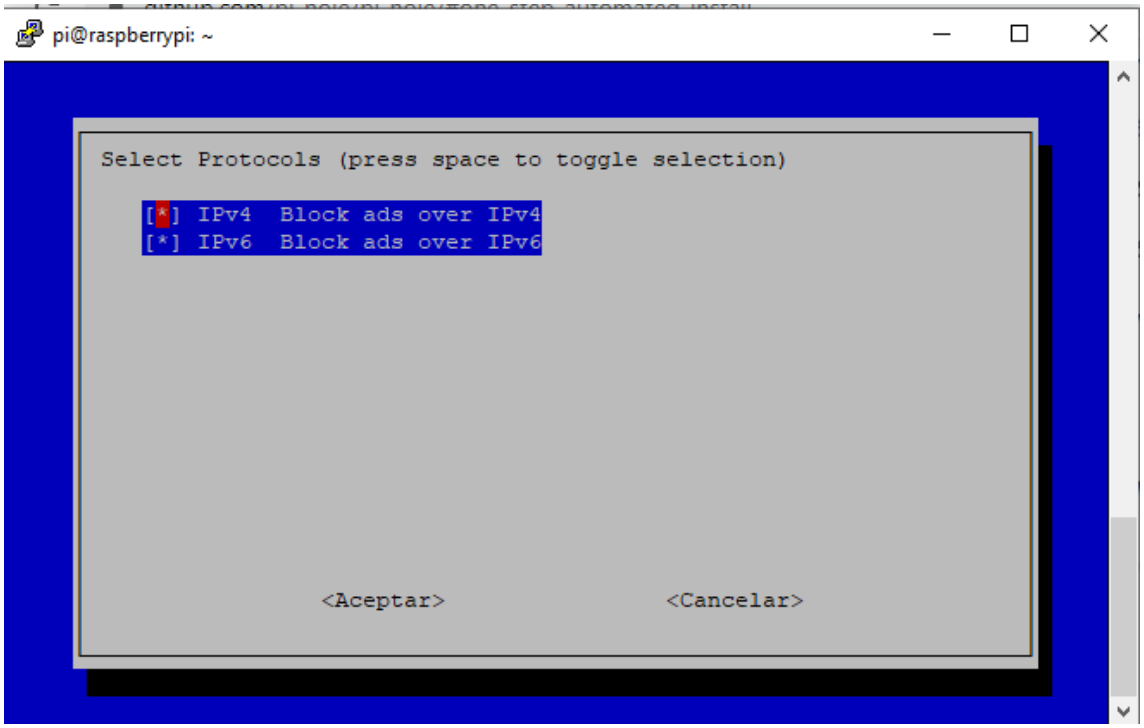


Ilustración 68. Pi-hole:Selección de protocolos. *Pi-hole*.

Muestra la configuración de red que se va a aplicar para que podamos confirmar.

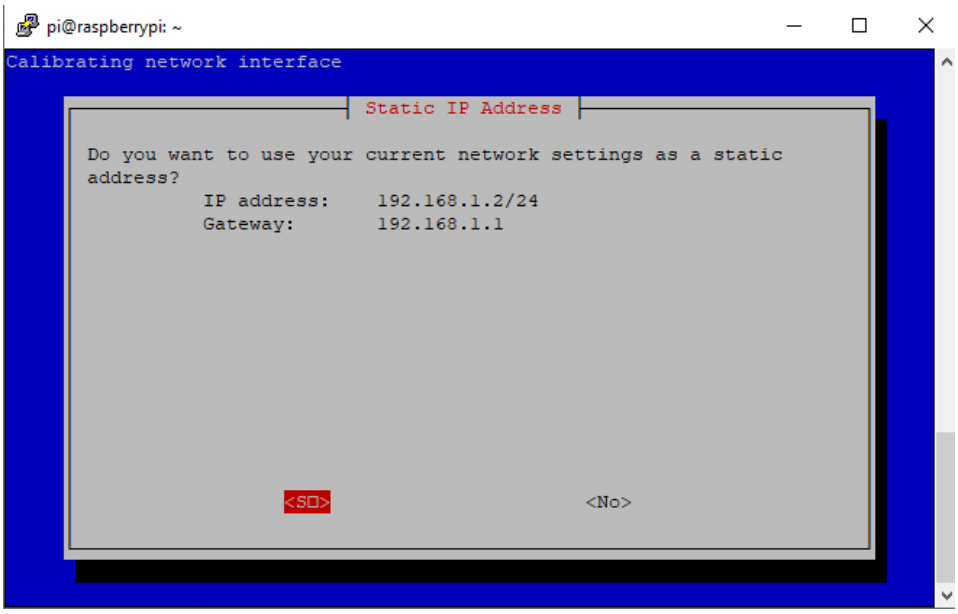


Ilustración 69. Pi-hole: Confirmación de la configuración de red. *Pi-hole*.

Se muestra a continuación un cartel informativo con el fin de que se lleven a cabo las configuraciones pertinentes en el router para que no haya una duplicidad de las direcciones IP en la red.

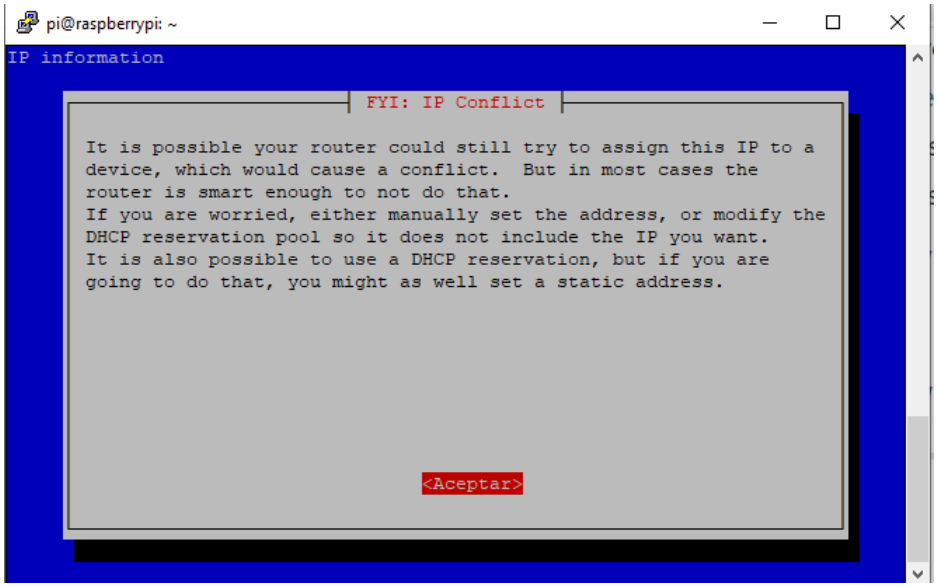


Ilustración 70. Pi-hole: Notificación de conflicto IP. *Pi-hole*

Se instala la interfaz web para que, a través de la dirección del sistema /admin sea posible acceder a la parte gráfica.

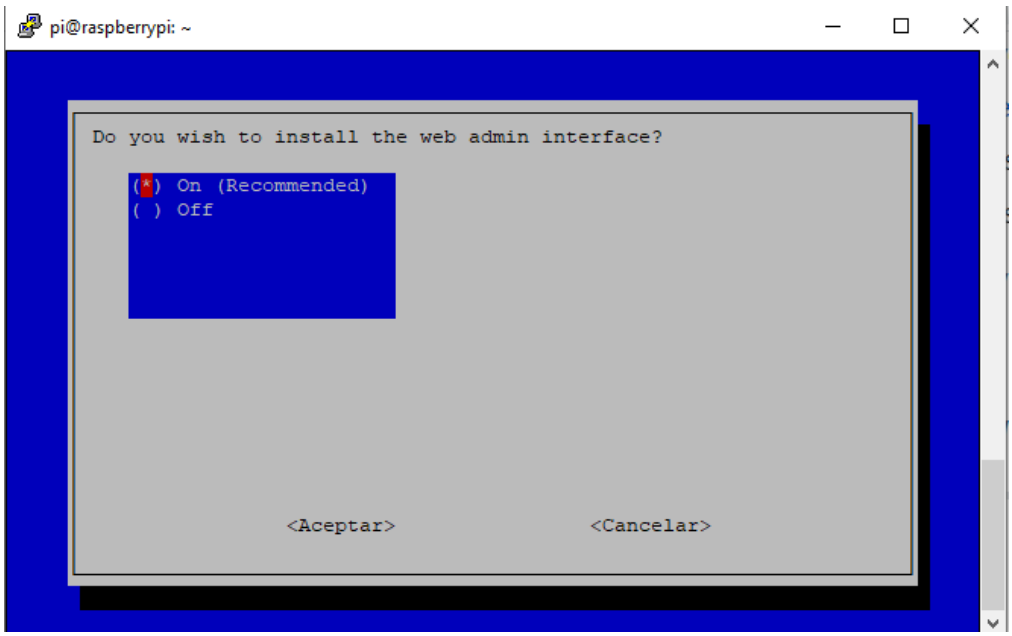


Ilustración 71. Pi-hole:Confirmación de instalación de interfaz web. *Pi-hole*

Se instala el servidor web (lighttpd) para que la interfaz GUI (*Grafical User Interface*) funcione correctamente.

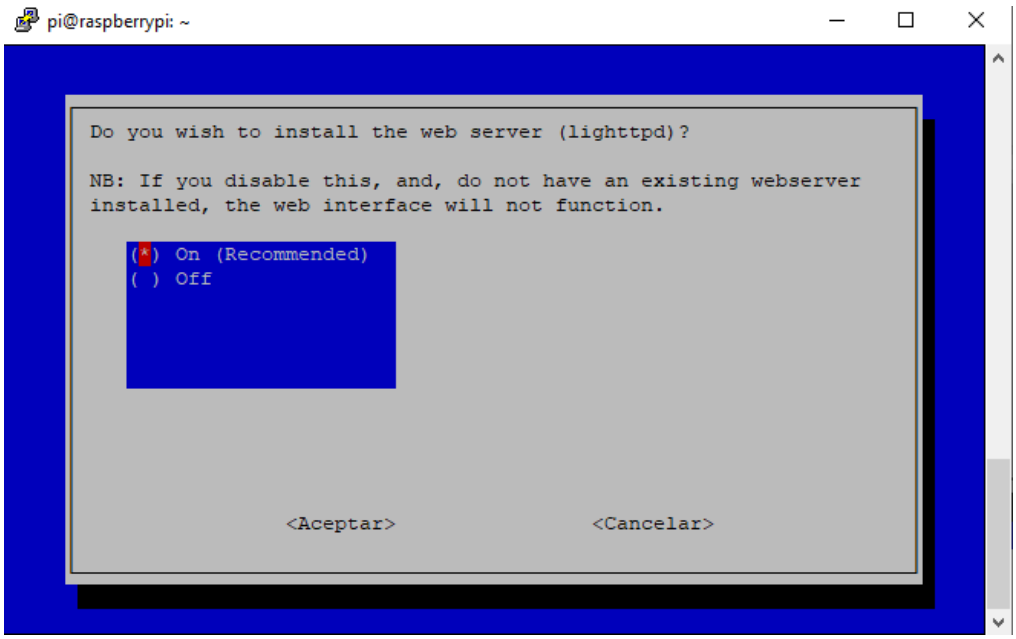


Ilustración 72. Pi-hole: Instalación de servidor web. *Pi-hole*.

Se configura para que almacene en *log*s las consultas para que posteriormente, a través de nuestro SIEM puedan ser procesados.

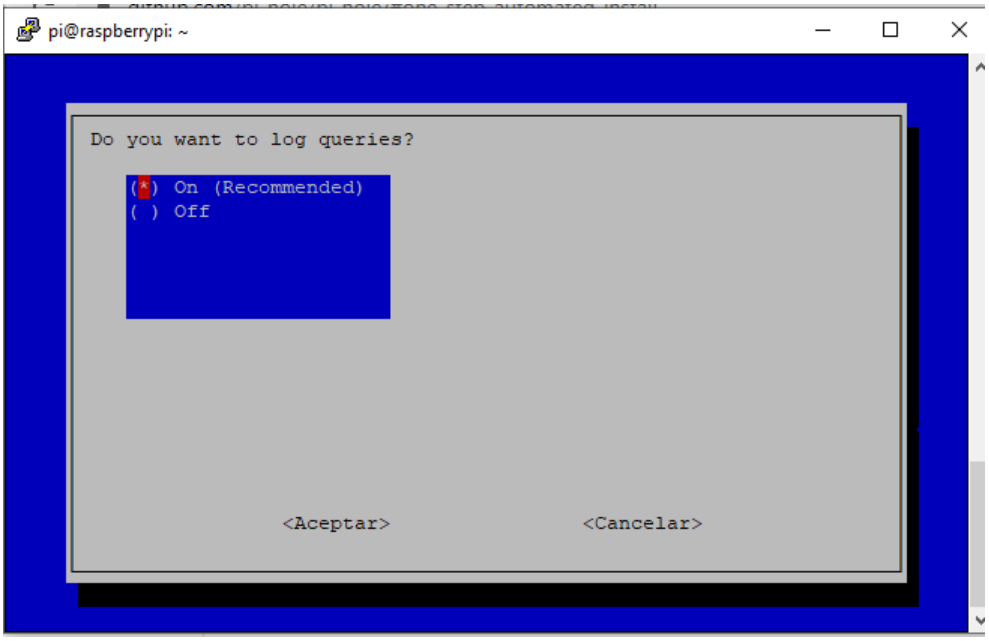


Ilustración 73.Pi-hole: Confirmación almacenado de logs. Pi-hole

Se informa del nivel de privacidad que queremos aplicar en las consultas que nos va a mostrar. Para realizar un tratamiento más profundo se elige la primera opción.

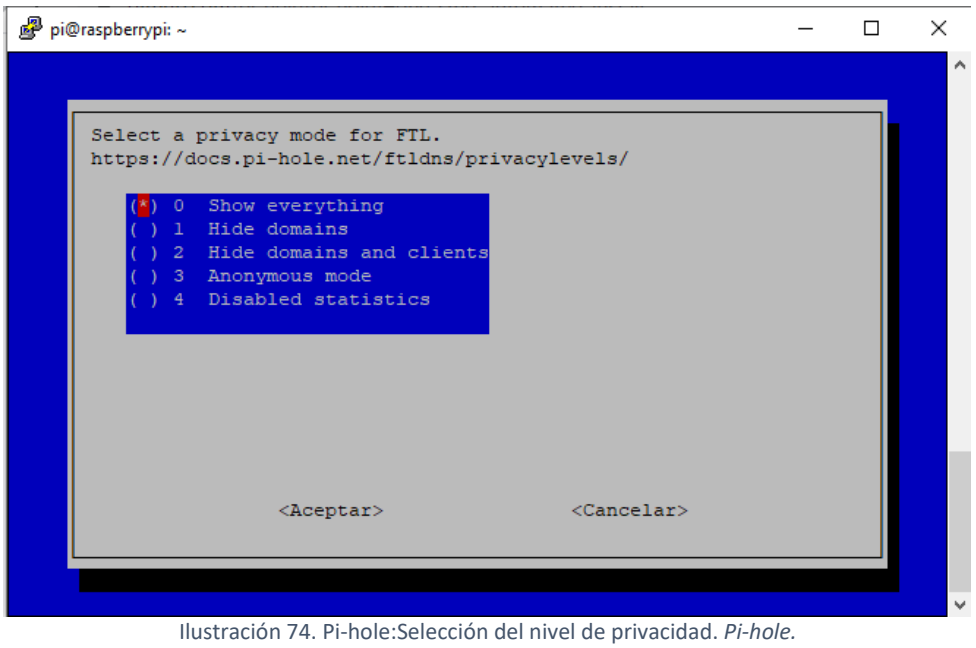


Ilustración 74. Pi-hole:Selección del nivel de privacidad. *Pi-hole.*

Finalmente muestra un resumen de la configuración que se ha realizado.

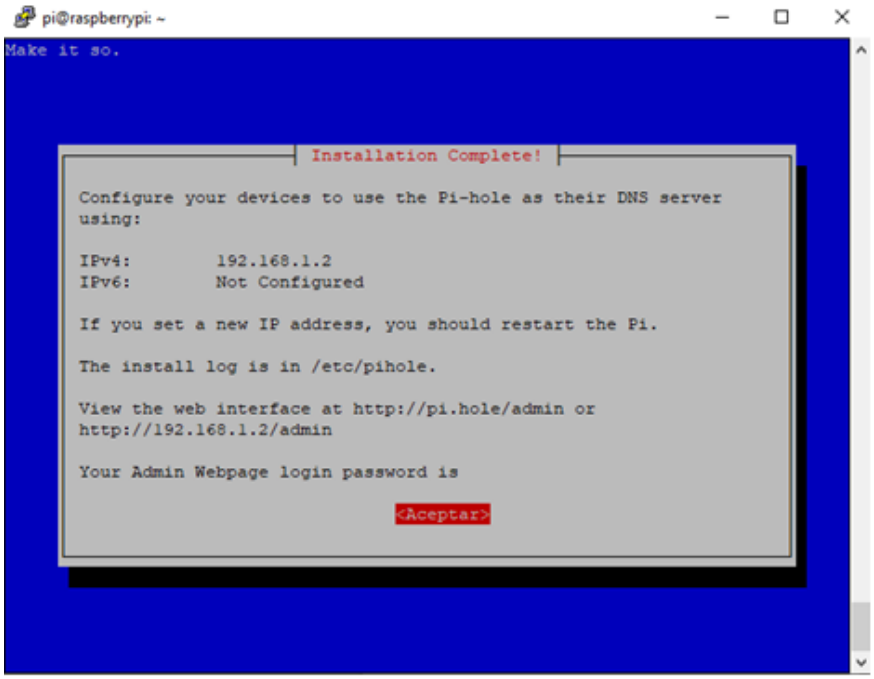


Ilustración 75. Pi-hole: Confirmación de la configuración. *Pi-hole*.

Se muestra la interfaz web a través de la cual podremos gestionar de forma sencilla el servicio.

Ilustración 76. Pi-hole: Panel de administración Pi-hole en funcionamiento. *Pi-hole.*

IV. Instalación de PiVPN - OpenVPN

Se lanza el script y automáticamente comienza la instalación. Se instala iptables y las reglas que tiene predefinidas.

```
curl -L https://install.pivpn.io | bash
```

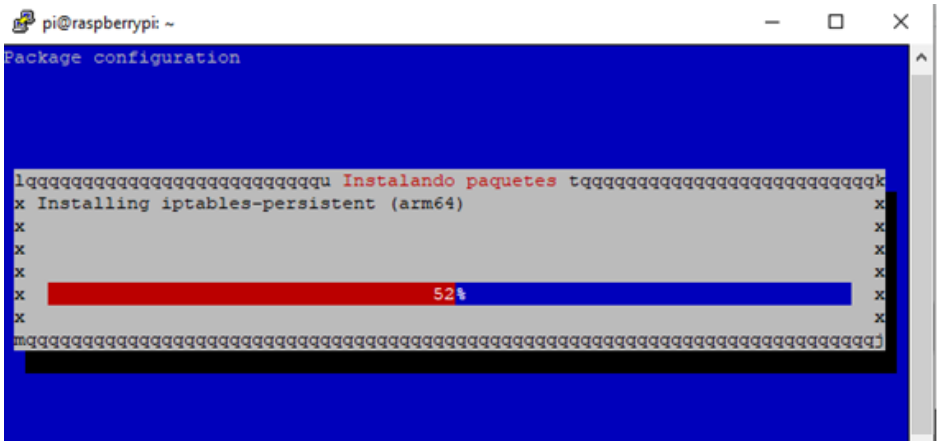


Ilustración 77. PiVPN: Instalación de IPtables. PiVPN.

Informa acerca de la necesidad de que la IP del dispositivo sea estática.

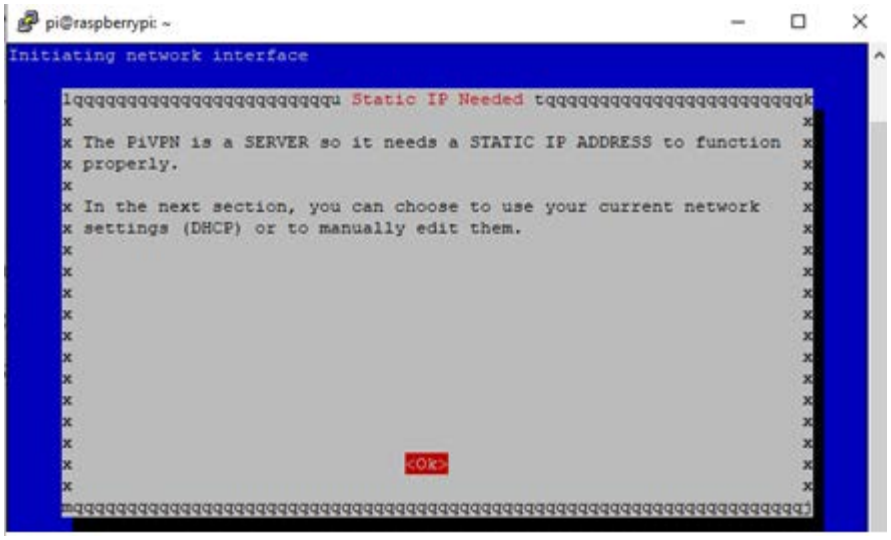


Ilustración 78.PiVPN: Información tipo de direccionamiento IP necesario. PiVPN.

Permite seleccionar el tipo de VPN a instalar, bien Wireguard o bien OpenVPN. Se selecciona la segunda opción.

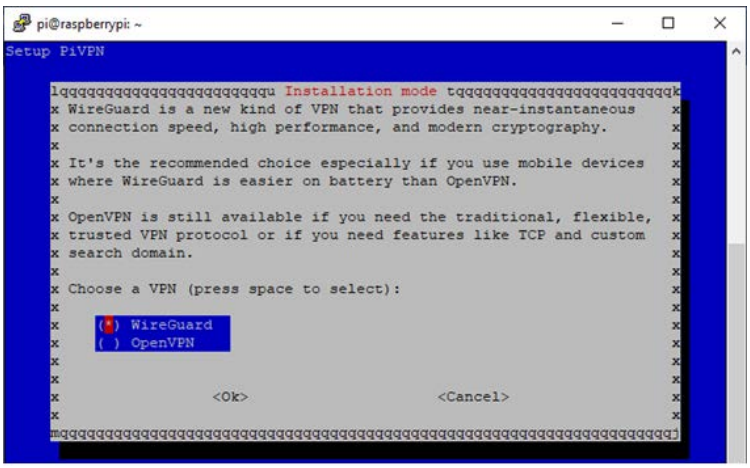


Ilustración 79.PiVPN:Elección de servicio VPN a instalar. PiVPN.

Se elige el tipo de protocolo a emplear (UDP).

Ilustración 80.PiVPN: Selección de protocolo de conexión. PiVPN.

Se selecciona puerto de conexión.

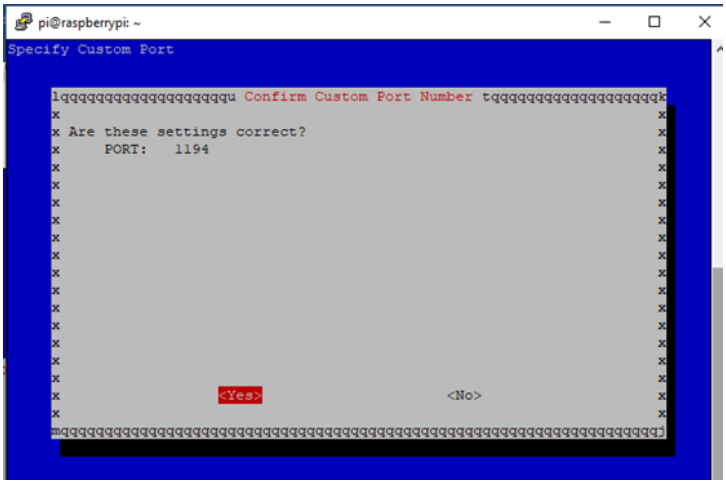


Ilustración 81. PiVPN: Selección de puerto de conexión. PiVPN.

Al detectar la instalación previa de Pi-hole nos da la opción de poder usar el servicio como servidor DNS.

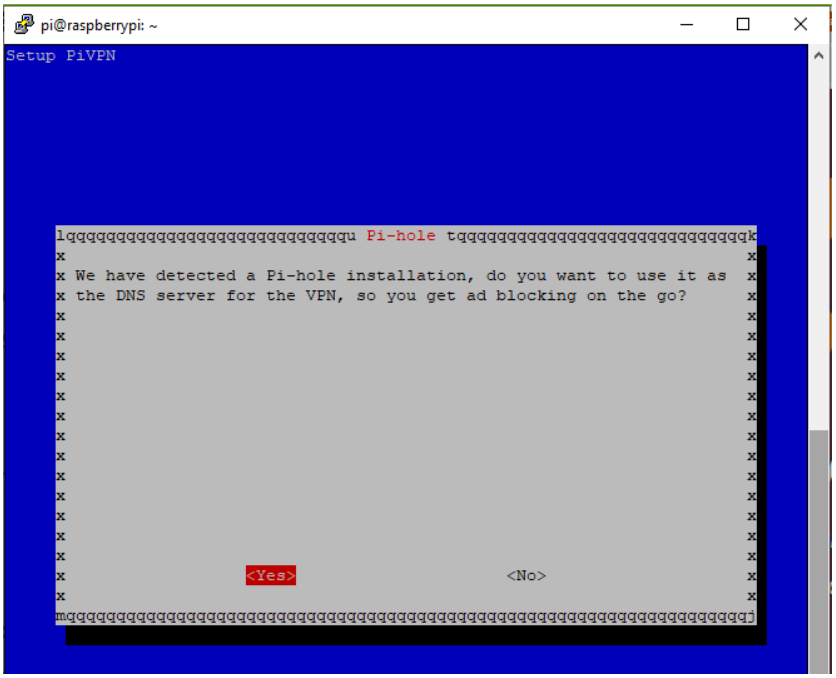


Ilustración 82.PiVPN: Detección de servidor DNS Pi-hole. PiVPN.

Tras ello, se configura la dirección IP pública. En este caso, al ser dinámica, asignamos un servicio de DDNS.

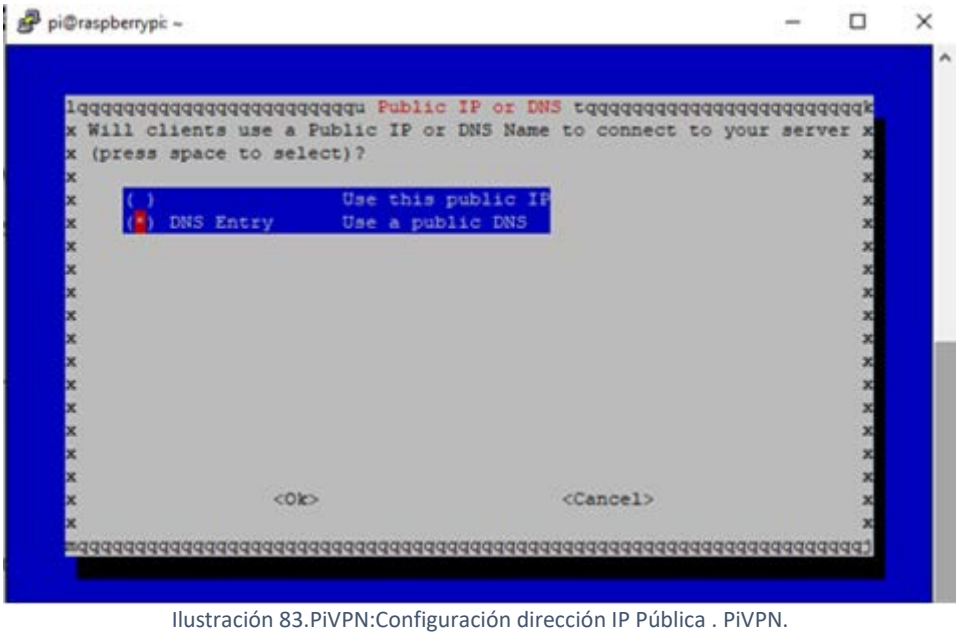


Ilustración 83.PiVPN:Configuración dirección IP Pública . PiVPN.

Se procede a crear una cuenta gratuita para conseguir el token que permita enlazarlo con nuestro router.

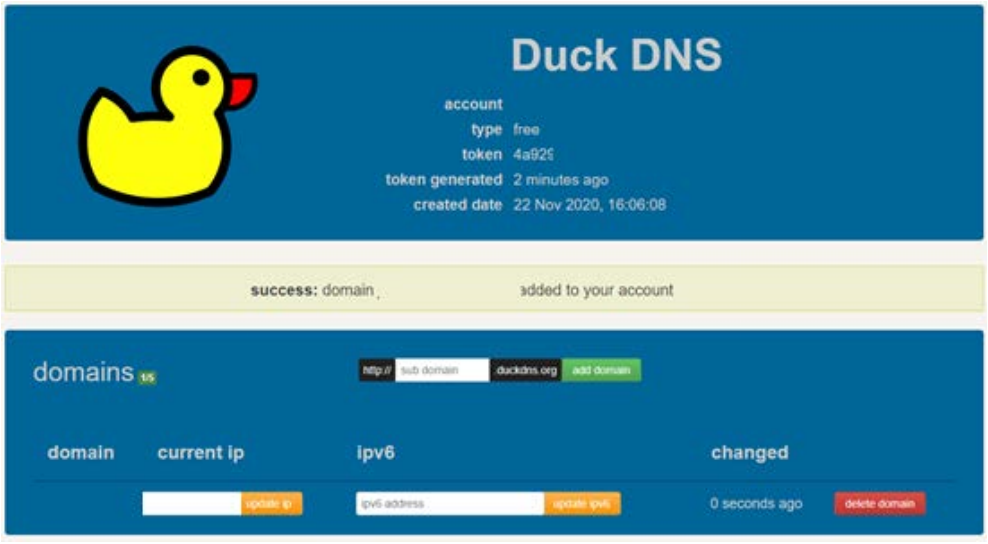


Ilustración 84.PiVPN: Generación Token Duck DNS. Duck DNS.

Se asigna el valor de la DDNS que acabamos de crear con el servicio DuckDNS.

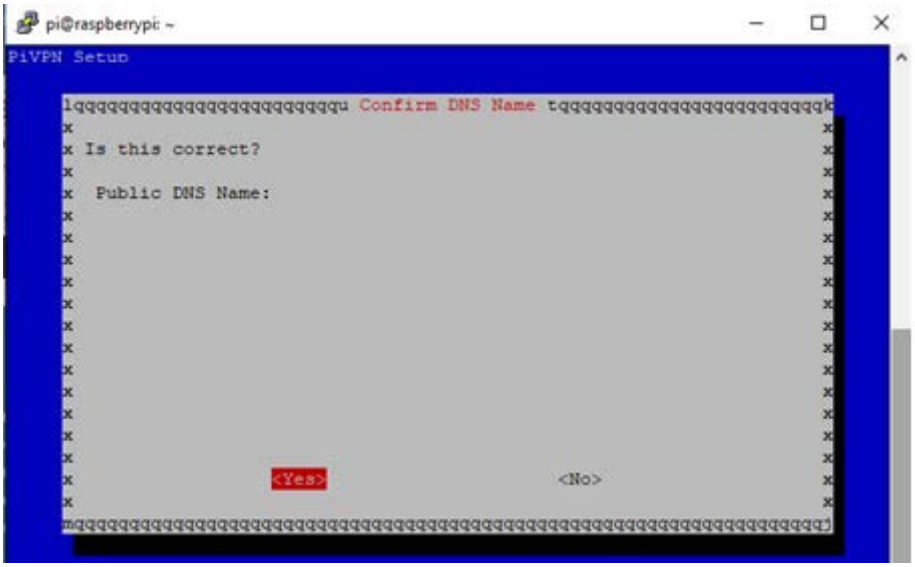


Ilustración 85.PiVPN: Confirmación configuración. PiVPN.

Se habilita la funcionalidad para mejorar la seguridad en el cifrado.

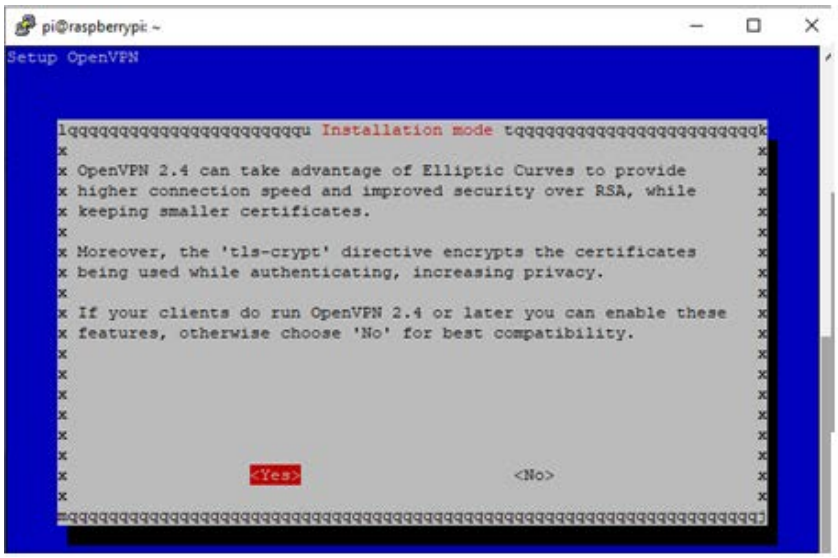


Ilustración 86.PiVPN: Configuración para mejora en sobre cifrado RSA. PiVPN.

Se usa el cifrado recomendado para el certificado.

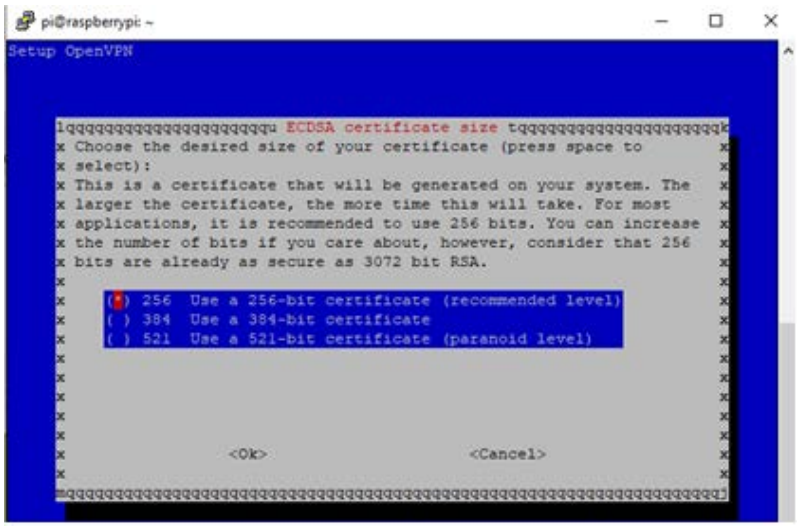


Ilustración 87.PiVPN: Selección del tipo de cifrado pra el certificado a generar. PiVPN.

Se habilitan las actualizaciones de seguridad desatendidas para hardenizar el sistema.

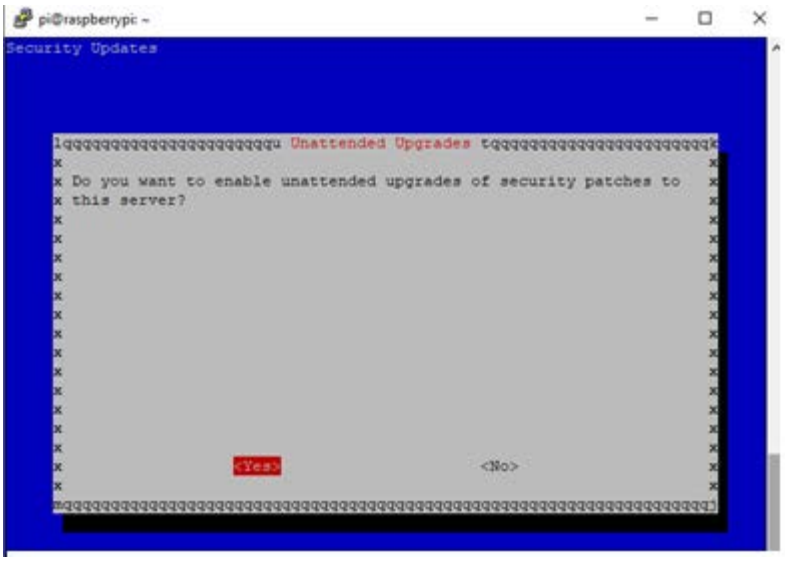


Ilustración 88.PiVPN:Configuración de actualizaciones de seguridad desatendidas. PiVPN.

Al final de la instalación, se nos informa de que el proceso ha finalizado correctamente. Ahora, simplemente, habrá que añadir los perfiles de clientes vpn que tenemos para poder establecer conexión con los distintos dispositivos.

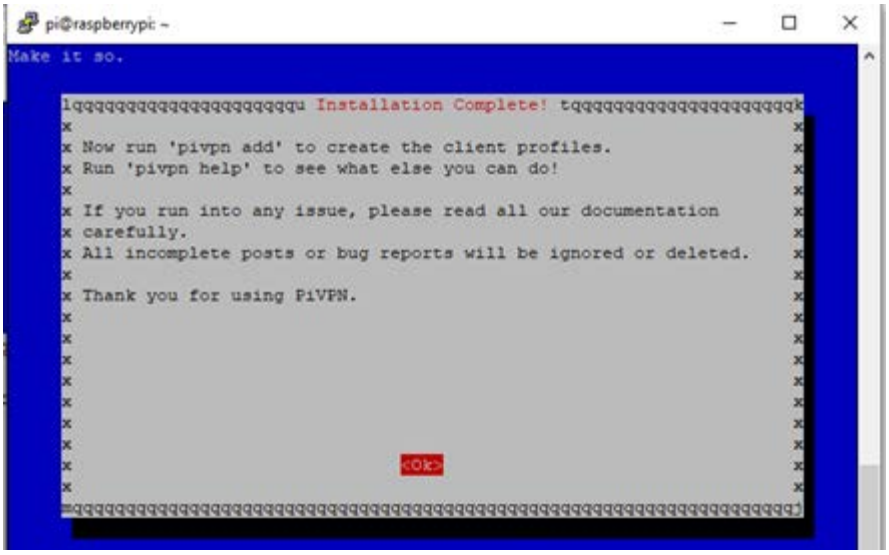


Ilustración 89.PiVPN: Confirmación de instalación. PiVPN.

Confirmada la configuración en el servidor se procederá a crear los perfiles VPN que servirá para que los clientes se conecten.

Ilustración 90.PiVPN: Creación de cliente VPN . PiVPN.

V. Script para simulación de ataque Ransomware

```
#!/usr/bin/env python3

# Copyright (C) 2015-2019, Wazuh Inc.
# Created by Wazuh, Inc. <info@wazuh.com>.
# This program is free software; you can redistribute it and/or modify it under the terms of GPLv2

import os
import random
import string
import base64
import sys
from pathlib import Path
from cryptography.fernet import Fernet


def create_random_files(basedir, n_directories, n_files_per_directory, size_file=1024):

    for root, dirs, files in os.walk(str(basedir)):
        for n_dir in range(n_directories):
            p = Path(root) / 'Directory_{}'.format(str(n_dir).zfill(2))
            p.mkdir(exist_ok=True)

    for root, dirs, files in os.walk(str(basedir)):
        if root is basedir:
            continue

        for n_file in range(n_files_per_directory):
            new_file = '{}/File_{}.txt'.format(root, str(n_file).zfill(2))
            text = ''.join([random.choice(string.ascii_letters) for i in range(size_file)]) #1
            with open(new_file, 'w') as f:
                f.write(text)

    return None


def encrypt_file(filepath, plain_key, output_filepath):

    encoded_key = base64.urlsafe_b64encode(plain_key.encode())

    # Encrypt file using cryptography.fernet library
    with open(filepath, mode='rb') as f_clear:
        fernet_cipher = Fernet(encoded_key)
        encrypted_data = fernet_cipher.encrypt(f_clear.read())
```

```
    # Remove sensitive variables
    del plain_key, encoded_key

    # Write content to file
    with open(output_filepath, mode='wb') as f_encrypt:
        f_encrypt.write(encrypted_data)


def decrypt_file(filepath, plain_key, output_filepath):

    encoded_key = base64.urlsafe_b64encode(plain_key.encode())

    # Decrypt file using cryptography.fernet library
    with open(filepath, mode='rb') as f_clear:
        fernet_cipher = Fernet(encoded_key)
        clear_data = fernet_cipher.decrypt(f_clear.read())

    # Remove sensitive variables
    del plain_key

    # Write content to file
    with open(output_filepath, mode='wb') as f_encrypt:
        f_encrypt.write(clear_data)


def encrypt_files(basedir, key):

    # Read operation
    for root, dirs, files in os.walk(str(basedir)):
        for file in files:
            # Write operation
            src_file = os.path.join(root, file)
            dst_file = '{}.{}'.format(src_file, "encrypted")
            encrypt_file(src_file, key, dst_file)
            # Delete operation
            os.remove(src_file)


def decrypt_files(basedir, key):

    # Read operation
    for root, dirs, files in os.walk(str(basedir)):
        for file in files:
            if file.endswith('.encrypted'):
                # Write operation
                src_file = os.path.join(root, file)
                dst_file = os.path.splitext(src_file)[0]
                decrypt_file(src_file, key, dst_file)
```

```
            # Delete operation
            os.remove(src_file)


if __name__ == '__main__':
    try:
        action = sys.argv[1]
    except:
        print("Error: Bad arguments. Valid arguments: 'prepare', 'attack', 'restore'")
        sys.exit(1)

    basedir = "/home/vagrant/test"
    key = "nso42FGdswR0805tnVqeww0u3Rubwk2a"

    if action == "prepare":
        create_random_files(basedir,        n_directories=10,        n_files_per_directory=20,
size_file=1024)
    elif action == "attack":
        encrypt_files(basedir, key)
    elif action == "restore":
        decrypt_files(basedir, key)
```

VI. Creación de Bot de Telegram

Se detallan a continuación los pasos para la creación de un bot de Telegram.

En primer lugar, se ha de disponer de una cuenta en la aplicación. Tras ello, se buscará el bot a través del buscador.

Ilustración 91. Búsqueda de BotFather. Telegram.

Al acceder al mismo, muestra la información que necesitamos para crear nuestro bot de alertas. Por un lado, cómo crear un bot, distintas integraciones y por otro lado los comandos necesarios para crear, inicializar y customizar nuestro bot.

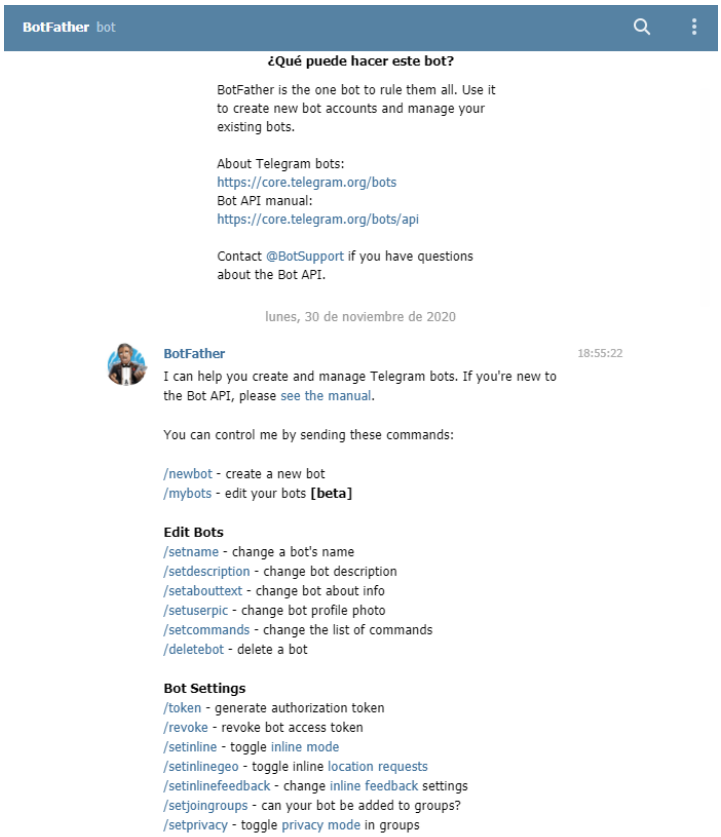


Ilustración 92. Información BotFather. Telegram.

Se inicia la creación del bot con la instrucción /newbot y se proporcionan los parámetros mínimos necesarios para que el proceso finalice con éxito. Tras ello, nos facilita la clave que se empleará en nuestro script para *log*rar la conexión.

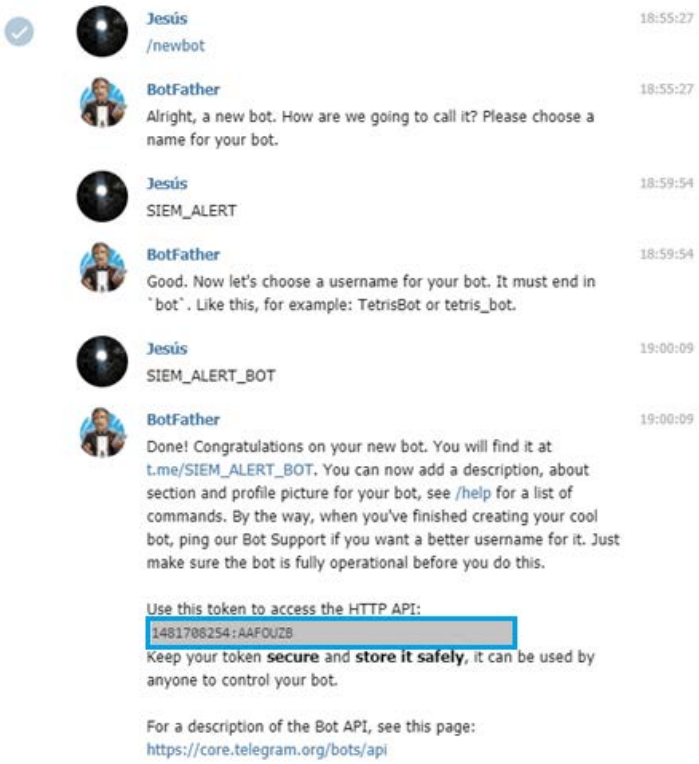


Ilustración 93. Creación de bot y generación de token. Telegram.

Tras obtener el token será necesario ir a la dirección api.telegram.org/bot[IDBOT]/getUpdates para conseguir el ID del chat al cual queremos enviar las notificaciones de alerta.

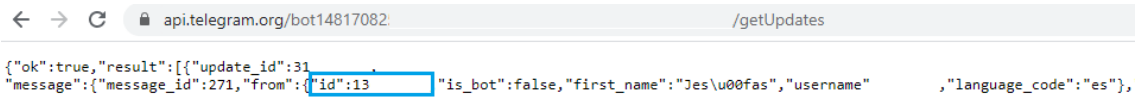


Ilustración 94. ID Chat Bot Telegram. *Telegram.*

VII. Informe ampliado Compliance: RGPD

info@wazuh.com
https://wazuh.com

GDPR report

General Data Protection Regulation (GDPR) sets guidelines for processing of personal data.

2020-09-29T10:42:01 to 2020-12-28T10:42:01
manager.name: raspberrypi AND rule.gdpr: exists

Most common GDPR requirements alerts found

Requirement IV_35.7.d

Capabilities for identification, blocking and forensic investigation of data breaches by malicious actors, through compromised credentials, unauthorized network access, persistent threats and verification of the correct operation of all components. Network perimeter and endpoint security tools to prevent unauthorized access to the network, prevent the entry of unwanted data types and malicious threats. Anti-malware and anti-ransomware to prevent malware and ransomware threats from entering your devices. A behavioral analysis that uses machine intelligence to identify people who do anomalous things on the network, in order to give early visibility and alert employees who start to become corrupt.

Top rules for IV_35.7.d requirement

Rule ID	Description
2904	Dpkg (Debian Package) half configured.
2902	New dpkg (Debian Package) installed.
607	Active response: ossec-telegram.sh - add

Requirement II_5.1.f

Ensure the ongoing confidentiality, integrity, availability and resilience of processing systems and services, verifying its modifications, accesses, locations and guarantee the safety of them. File sharing protection and file sharing technologies that meet the requirements of data protection.

Top rules for II_5.1.f requirement

Rule ID	Description
550	Integrity checksum changed.
554	File added to the system.

Ilustración 95. Informe Compliance 1/6. Wazuh.

info@wazuh.com
https://wazuh.com

Rule ID	Description
553	File deleted.

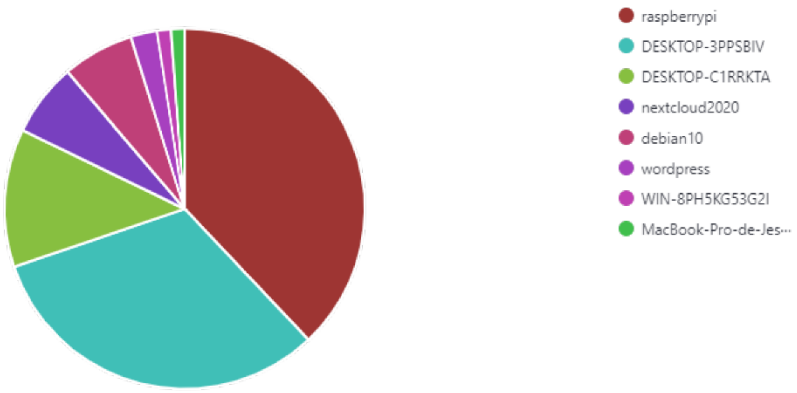
Requirement IV_32.2

Account management tools that closely monitor actions taken by standard administrators and users who use standard or privileged account credentials are required to control access to data.

Top rules for IV_32.2 requirement

Rule ID	Description
60106	Windows Logon Success
5501	PAM: Login session opened.
5502	PAM: Login session closed.

GDPR Agents



GDPR Requirements by agent

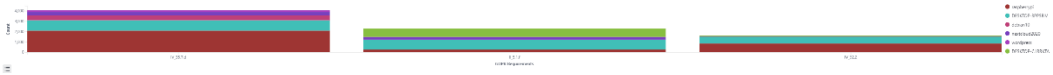
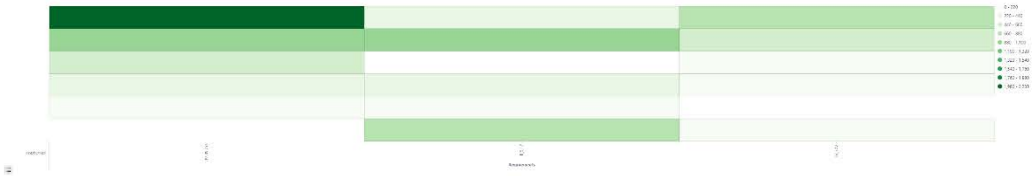


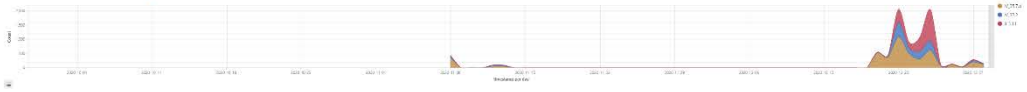
Ilustración 96 Informe Compliance 2/6. Wazuh.

info@wazuh.com
https://wazuh.com

Last alerts



GDPR requirements over time



GDPR requirements

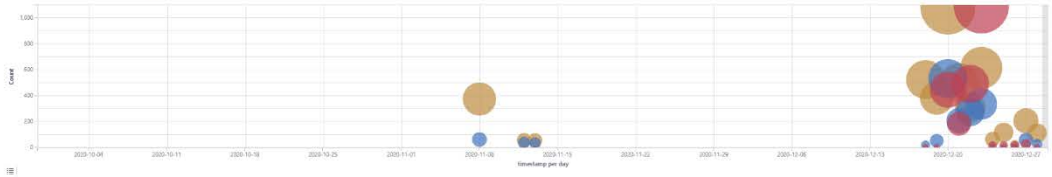


Ilustración 97 Informe Compliance 3/6. Wazuh.

info@wazuh.com
https://wazuh.com

Alerts summary

Agent name	Requirement	Rule description	Count
DESKTOP-3PPSBIV	II_5.1.f	Integrity checksum changed.	783
DESKTOP-3PPSBIV	IV_32.2	Windows Logon Success	500
raspberrypi	IV_35.7.d	Active response: ossec-telegram.sh - add	440
raspberrypi	IV_35.7.d	Dpkg (Debian Package) half configured.	439
DESKTOP-C1RRKTA	II_5.1.f	File added to the system.	402
DESKTOP-C1RRKTA	II_5.1.f	File deleted.	402
raspberrypi	IV_35.7.d	New dpkg (Debian Package) installed.	340
raspberrypi	IV_32.2	PAM: Login session opened.	297
raspberrypi	IV_32.2	PAM: Login session closed.	243
raspberrypi	IV_35.7.d	Listened ports status (netstat) changed (new port opened or closed).	242
raspberrypi	IV_32.2	Successful sudo to ROOT executed.	201
debian10	IV_35.7.d	Dpkg (Debian Package) half configured.	160
raspberrypi	II_5.1.f	Integrity checksum changed.	153
raspberrypi	IV_35.7.d	New dpkg (Debian Package) requested to install.	150
raspberrypi	IV_35.7.d	Systemd: Service exited due to a failure.	132
DESKTOP-3PPSBIV	II_5.1.f	File deleted.	111
debian10	IV_35.7.d	New dpkg (Debian Package) installed.	104
nextcloud2020	II_5.1.f	File added to the system.	79
DESKTOP-3PPSBIV	II_5.1.f	File added to the system.	77
raspberrypi	II_5.1.f	File added to the system.	76
nextcloud2020	II_5.1.f	Integrity checksum changed.	75
DESKTOP-3PPSBIV	IV_32.2	Windows User Logoff	70
DESKTOP-3PPSBIV	IV_32.2	Windows Workstation Logon Success	62
DESKTOP-C1RRKTA	IV_35.7.d	Windows audit failure event	62
DESKTOP-C1RRKTA	IV_32.2	Windows Logon Success	57
nextcloud2020	II_5.1.f	File deleted.	53
raspberrypi	IV_35.7.d	Ossec server started.	50
raspberrypi	IV_35.7.d	Host-based anomaly detection event (rootcheck).	40
raspberrypi	IV_35.7.d	VirusTotal: Error: Check credentials	39
raspberrypi	IV_32.2	VirusTotal: Error: Check credentials	39
DESKTOP-3PPSBIV	IV_35.7.d	Windows System error event	38
debian10	IV_35.7.d	New dpkg (Debian Package) requested to install.	35
DESKTOP-3PPSBIV	IV_35.7.d	Service startup type was changed	34
debian10	IV_35.7.d	Listened ports status (netstat) changed (new port opened or closed).	25
debian10	IV_32.2	PAM: Login session opened.	25
DESKTOP-3PPSBIV	IV_32.2	Logon Failure - Unknown user or bad password	24

Ilustración 98. Informe Compliance 4/6. Wazuh.

info@wazuh.com
https://wazuh.com

Agent name	Requirement	Rule description	Count
raspberrypi	IV_35.7.d	Log file rotated.	23
raspberrypi	II_5.1.f	Log file rotated.	23
DESKTOP-3PPSBIV	IV_35.7.d	Logon Failure - Unknown user or bad password	23
raspberrypi	IV_32.2	sshd: authentication success.	22
nextcloud2020	IV_35.7.d	Log file rotated.	19
nextcloud2020	II_5.1.f	Log file rotated.	19
wordpress	IV_35.7.d	Log file rotated.	17
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-13602 affects VLC media player	15
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-14437 affects VLC media player	15
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-14438 affects VLC media player	15
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-14498 affects VLC media player	15
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-14533 affects VLC media player	15
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-14535 affects VLC media player	15
DESKTOP-3PPSBIV	IV_35.7.d	CVE-2019-14777 affects VLC media player	15
raspberrypi	IV_32.2	Postfix SASL authentication failure.	14
DESKTOP-C1RRKTA	IV_35.7.d	Windows System error event	10
debian10	IV_32.2	PAM: Login session closed.	10
debian10	IV_35.7.d	Ossec agent disconnected.	9
debian10	IV_35.7.d	Ossec agent started.	9
raspberrypi	IV_32.2	PAM: User login failed.	8
raspberrypi	IV_32.2	sshd: Attempt to login using a non-existent user	8
raspberrypi	II_5.1.f	File deleted.	8
nextcloud2020	IV_35.7.d	Ossec agent started.	8
debian10	IV_32.2	Successful sudo to ROOT executed.	7
raspberrypi	IV_32.2	OpenVPN: User logged in	6
nextcloud2020	IV_35.7.d	sshd: Attempt to login using a non-existent user	6
nextcloud2020	IV_32.2	sshd: Attempt to login using a non-existent user	6
raspberrypi	IV_32.2	New group added to the system.	5
nextcloud2020	IV_32.2	PAM: Login session closed.	4
wordpress	IV_35.7.d	Ossec agent started.	4
DESKTOP-C1RRKTA	IV_35.7.d	Active response: restart-ossec.cmd - add	3
nextcloud2020	IV_35.7.d	VirusTotal: Alert - /var/www/nextcloud/data/admin/files/eicar22.zip - 56 engines detected this file	3
nextcloud2020	IV_32.2	PAM: Login session opened.	3
nextcloud2020	IV_32.2	sshd: authentication success.	3
debian10	IV_35.7.d	Systemd: Service exited due to a failure.	3
DESKTOP-C1RRKTA	IV_35.7.d	Service startup type was changed	2
nextcloud2020	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Disable Automounting	2
nextcloud2020	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure /etc/hosts.allow is configured	2
nextcloud2020	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure /etc/hosts.deny is configured	2

Ilustración 99.Informe Compliance 5/6. Wazuh.

info@wazuh.com
https://wazuh.com

Agent name	Requirement	Rule description	Count
nextcloud2020	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure Avahi Server is not enabled	2
nextcloud2020	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure DHCP Server is not enabled	2
nextcloud2020	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure GDM login banner is configured	2
debian10	IV_35.7.d	SCA summary: CIS benchmark for Debian/Linux 9 L1: Score less than 50% (35)	2
wordpress	IV_35.7.d	New dpkg (Debian Package) installed.	2
DESKTOP-C1RRKTA	IV_35.7.d	Benchmark for Windows audit: Ensure 'Accounts: Limit local account use of blank passwords to console logon only' is set to 'Enabled'	1
DESKTOP-C1RRKTA	IV_35.7.d	Benchmark for Windows audit: Ensure 'Always install with elevated privileges' is set to 'Disabled'	1
DESKTOP-C1RRKTA	IV_35.7.d	Benchmark for Windows audit: Ensure 'Always prompt for password upon connection' is set to 'Enabled'	1
DESKTOP-C1RRKTA	IV_35.7.d	Benchmark for Windows audit: Ensure 'Audit: Shut down system immediately if unable to log security audits' is set to 'Disabled'	1
DESKTOP-C1RRKTA	IV_35.7.d	Benchmark for Windows audit: Ensure 'Configure Automatic Updates' is set to 'Enabled'	1
DESKTOP-C1RRKTA	IV_35.7.d	Benchmark for Windows audit: Ensure 'Configure Offer Remote Assistance' is set to 'Disabled'	1
nextcloud2020	IV_32.2	PAM: User login failed.	1
nextcloud2020	IV_32.2	syslog: User missed the password more than one time	1
debian10	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Disable Automounting	1
debian10	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure /etc/hosts.allow is configured	1
debian10	II_5.1.f	File added to the system.	1
debian10	II_5.1.f	File deleted.	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Disable Automounting	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure /etc/hosts.allow is configured	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure /etc/hosts.deny is configured	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure /tmp is configured	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure AIDE is installed	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure Avahi Server is not enabled	1
wordpress	IV_35.7.d	CIS benchmark for Debian/Linux 9 L1: Ensure CUPS is not enabled	1

Ilustración 100. Informe Compliance 6/6. Wazuh.