UNIVERSITAT POLITÈCNICA DE VALÈNCIA

_ TELECOM ESCUELA TÉCNICA VLC SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER
Implantación de Qradar en un entorno genérico multi-cliente para SOC

Alexis Sánchez Sanz

Tutor: Carlos Enrique Palau Salvador

Tutor en empresa: Bernat Canadell García

Trabajo Fin de Máster presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València, para la obtención del Título de Máster en Ingeniería Telecomunicación

Curso 2018-19

Valencia, 29 de septiembre de 2019

Escuela Técnica Superior de Ingeniería de Telecomunicación
Universitat Politècnica de València
Edificio 4D. Camino de Vera, s/n, 46022 Valencia
Tel. +34 96 387 71 90, ext. 77190
www.etsit.upv.es

VLC/ CAMPUS VALENCIA, INTERNATIONAL CAMPUS OF EXCELLENCE

EMAS Gestión ambiental verificada REG.NO. ES-CV-000030

Resumen

El TFM está basado en un proyecto realizado en la empresa EY, en el apartado de Ciberseguridad. El proyecto consta de la implementación de un SOC (Security Operation Center) en un entorno multi-cliente con Qradar y su posterior puesta en producción e integración con distintas fuentes, así como el desarrollo del día a día y distintos indicadores útiles a nivel interno y para el cliente.

Este proyecto fue llevado a cabo por un equipo, así que las partes que personalmente dirigí o hice serán ampliamente explicadas, mientras que las que me limité a observar serán explicadas de soslayo.

Resum

El TFM està basat en un projecte realitzat en l'empresa EY, en l'apartat de Ciber-seguretat. El projecte consta de la implementació d'un SOC (Security Operation Center) en un entorn multi-client amb Qradar i la seua posterior posada en producció i integració amb distintes fonts, així com el desenvolupament del dia a dia i distints indicadors útils a nivell intern i per al client.

Este projecte va ser dut a terme per un equip, així que les parts que personalment vaig dirigir o vaig fer seran àmpliament explicades, mentre que les que em vaig limitar a observar seran explicades de gaidó.

Abstract

This MFW is based on a project done by EY, in cyber-security field. The project consists of the implementation and deployment of a multi-client SOC (Security Operation Center) with Qradar and subsequent operation and linking to different sources, as well as the daily operation and various indicators, useful for client and for internal use.

This project was carried out by a team, so the tasks that I personally did or closely supervised will be largely explained, while the ones that just observed will be very little explained.

Agradecimientos:

A mis padres, a mi hermano y mi pareja por soportarme (en general).

A EY por aceptar que realizara el TFM en su empresa.

A mis compañeros de EY que han aportado datos, diagramas e información referenciadas al TFM, y en particular a Cris.

Nota: Para evitar que el TFM sea secreto y mantener los datos de la empresa a salvo, se usará un nombre genérico para las empresas a las cuales se les realizó este proyecto, llamándolos "cliente". Puede que algunos nombres y configuraciones se vean modificados con el fin de mantener la confidencialidad de los datos y el proyecto.

Índice

Capítulo 1. Introducción

1.1 Motivación

Hoy en día la seguridad es una pieza fundamental en cualquier empresa, las noticias sobre brechas de seguridad que nos llegan son cada día mas alarmantes, y los delincuentes informáticos poseen máquinas y herramientas cada vez más capaces. Se pueden encontrar múltiples herramientas de código abiertas que pueden ser usadas para realizar acciones maliciosas, y la facilidad de uso de las mismas aumenta con el tiempo.

Debido a estos factores, la motivación de este proyecto es aumentar la seguridad en el cliente, evitando los posibles desenlaces de acciones maliciosas, tales como un filtrado de información o una extorsión.

1.2 Objetivos

El objetivo final de este TFM es implantar un SIEM en cada uno de los clientes para que lo use el SOC en un entorno multi-cliente. Esto significa que cada cliente dispondrá de su instalación personalizada en local, mientras que el equipo de SOC inspeccionará a la vez todas las instalaciones, y focalizará sus esfuerzos en el cliente más conveniente en cada instante.

Como objetivos secundarios, se persigue normalizar en la medida de lo posible todo lo referente a la operativa diaria, piezas clave tales como las ofensas, las reglas, los eventos…

Un objetivo paralelo es informar al cliente de cómo se desarrolla el proyecto a medida que se avanza en el tiempo.

1.3 Organización de la memoria

Se planteará el desarrollo del proyecto de implantación de un SIEM, y cada una de sus fases, pasando posteriormente a su configuración. Se reflejará la estructura de un SOC y su funcionamiento, y se explicará como se transmite este trabajo al cliente. Por último se mostrará la planificación temporal y se concluirá.
Como anexos se incluyen definiciones de siglas, palabras técnicas y de jerga de seguridad, bibliografía y un índice con documentos adjuntos.

Capítulo 2. Desarrollo

2.1 Contexto

Un SOC (Security Operation Center)[1][2] es un equipo que monitoriza y analiza comportamientos extraños y amenazas, con el objetivo de detectarlas, y dado el caso de detección correcta, tiene la potestad de pedir o realizar acciones contraofensivas.

Un SIEM (Security Information and Event Management)[3] es un sistema centralizado que se encarga del almacenamiento y la interpretación de los datos relevantes en un entorno de seguridad informática/cibernética. Estos datos son transmitidos por medio de mensajes llamados 'logs'. El SIEM es el "el cerebro" capaz de correlar distintos eventos y extraer así información útil para el SOC. Un SIEM también es útil en cuanto a la realización de investigaciones, ya sea de forma pasiva (reglas y ofensas que veremos posteriormente) o activa, mediante investigación forense.

El equipo de SOC recibió el proyecto de implantar y configurar un SIEMee para varios clientes los cuales de ahora en adelante llamaremos de forma genérica "cliente", al referirnos a cualquiera de ellos de forma única.

Este proyecto se realizó con un equipo de ocho personas y la ayuda puntual de otra al principio del proyecto, cuando el SOC no era el encargado de su realización. En otras palabras, el proyecto nos fue traspasado con el SIEM y hasta la arquitectura del mismo decidida. Antes de ser traspasado el proyecto, se realizó la elección del SIEM a implantar. Para la realización del proyecto, contaba con siete empleados, de los cuales yo era el coordinador. De los integrantes del equipo, el resto no estaba dedicado en exclusividad a la realización del proyecto, y el equipo estaba dividido en 4 empleados de turno de mañana y tres de tarde.

2.2 Elección de SIEM

Previamente a la implantación de cualquier SIEM, hay que elegir qué tipo de SIEM se va a implementar. Existen SIEMs de pago y SIEMs gratuitos.

La ventaja de las herramientas gratuitas es el mínimo desembolso económico que el cliente debe realizar al principio, sin embargo, las herramientas de pago suelen ser mantenidas y dar un mejor soporte.

La persona encargada de la elección de SIEM no formaba parte del proyecto, y realizó la elección antes de que el equipo de SOC fuese el encargado del proyecto; propuso los siguientes SIEMs: Qradar (IBM), Alienvault USM, Splunk, y ArcSight ESM.

Después de exponer las ventajas e inconvenientes de cada uno de estos SIEM, que no serán detalladas en profundidad en este TFM, se decidió que el SIEM que usaríamos sería el Qradar de IBM (de ahora en adelante, Qradar).

Un apartado que se tuvo en cuenta fue la posición de los distintos SIEM en el "cuadrado mágico de Gartner" de 2018 [Tabla 1]

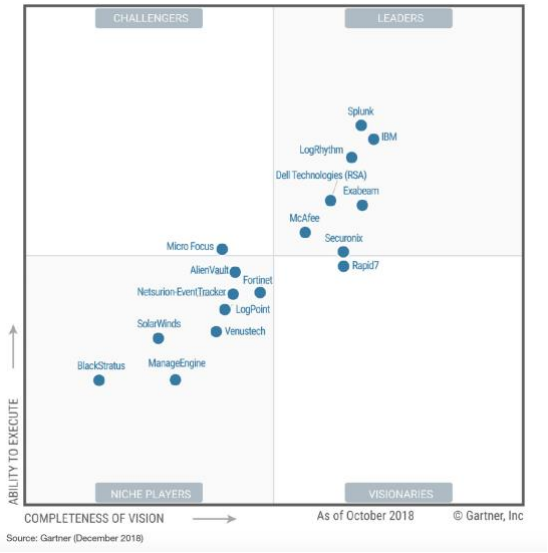


Tabla 1. Cuadrado mágico de Gartner 2018 [4].

2.3 Qradar

Una vez decidida la marca que se adquiriría, se debe decidir cual de las arquitecturas es la adecuada para el entorno de trabajo del cliente. Primero explicaremos las funciones que realiza el SIEM Qradar, para posteriormente ahondar en las distintas arquitecturas disponibles.

2.3.1 Funciones[5]

- Collector and parser: esta función es la que se encarga de recoger los logs y "parsearlos", esto es, analizar sintácticamente, ordenar y clasificar la información de los logs que llegan al Qradar como texto o JSON de las distintas fuentes y darles una estructura para almacenarlas e indexarlas de forma eficiente.
- Correlator: Es la "inteligencia" de Qradar. Se encarga de correlar los eventos y en función de las alertas/casos de uso/procedimientos marcados crea ofensas u otros eventos
- Console and apps: es la parte en la cual se pueden visualizar los logs, realizar búsquedas y tratar con los datos recibidos. Existen aplicaciones que se incorporan en esta capa y hacen que estas tareas se faciliten.

2.3.2 Arquitecturas[6]

En el caso particular de Qradar, las arquitecturas de la solución se pueden dividir en tres tipos agrupados a su vez en dos clases:

- **Qradar on premise:** Este Qradar se instala físicamente en las instalaciones del cliente o en un CPD que pertenezca al mismo. Pueden ser uno o varios dipositivos que interatuen entre sí como un ente único. Existen principalmente dos tipos:
    - All-in-one: como su propio nombre indica, es un dispositivo en el que todas las partes que componen un Qradar se integran en un único dipositivo, o como se suele llamar en la jerga de seguridad, un único "appliance".

  - Various appliances: Cada una de las funciones que componen el Qradar se lleva a cabo en un dispositivo diferente, con lo que tenemos varios dispositivos y una sola consola de control. A ojos del usuario es indistinguible de la primera opción, y se puede pasar de la primera a esta opción mas avanzada en caso de que el proyecto así lo necesitara.
- **Qradar on cloud:** En este tipo de instalación, la empresa no posee ningún hardware, de modo que los logs se envían a internet y no tiene que comprar o alquilar ningún dispositivo.

2.4 Elección de tipo Qradar

Al igual que ocurría con la elección de la marca, esta parte del proyecto la realizó un compañero de empresa, y no ahondaremos en cómo se llegó a la conclusión final, ya que no es parte del trabajo realizado para el presente TFM. Es suficiente con conocer que para el proyecto encomendado al equipo SOC, esta circunstancia era impuesta y no fue un trabajo realizado por el equipo. En cualquier caso, este dato no afecta al desarrollo del presente TFM.

La elección final fue Qradar on premise "All-in-one"

2.5 Implementación en cliente

2.5.1 Información previa

Para realizar la implementación en el cliente hace falta disponer de una información previa que se pidió con posterioridad a la elección del tipo de Qradar.

La información pedida fue la siguiente:

- Mapa de red
- Localización de las fuentes solicitadas, tanto física como digital (IPs)
- Localización de la consola central de logs, en caso de existir.
- Persona responsable de la gestión por cada una de las fuentes solicitadas.
- Volumetrías de carga de las fuentes
- Por cada fuente, el fabricante y modelo, así como el número de dispositivos, y si poseen más de un fabricante o modelo distinto.

Estas son las fuentes de las cuales se pedía que se comprobara en el cliente su disponibilidad:

- Flows
- Proxy
- DNS
- DHCP
- Sandbox red
- Sandbox email
- Antivirus
- EDR/Anti-APT
- FW/UTM
- Servidor de Correo
- AD
- Anti-Spam

- WAF
- IPS
- Jump servers
- DLP
- VPN Concentrador de usuarios
- VPN Punto a Punto
- NAC
- Servidores DMZ Publicación (Cantidad de servidores Midleware que publican y sus servicios)
- SCCM
- CMDB
- Change Auditor
- Sistema monitorización
- Aplicaciones afectadas por la GDPR (Optativo)
- Balanceadores
- Escáner vulnerabilidades (Optativo)
- Electrónica de red para identificar las NATs
- Salidas libres a internet (IFA, WIFA)
- Aplicaciones web y servidores

Como norma general aunque no siempre exacta, estas fuentes son las ampliamente soportadas por cualquier SIEM, y son el núcleo de cualquier sistema de seguridad. Cada una de ellas puede ser extremadamente compleja, con varias sondas, un sistema de alta disponibilidad, repartidas en varias localizaciones… o un simple dispositivo único, de cualquier manera, todas ellas deben transmitir un log en texto, que normalmente sigue un formato CEF, LEEF o JSON (explicados posteriormente).

2.5.2 Conexión física

Para implementar Qradar en el cliente, la primera operación debe ser instalarlo físicamente, y se aplicó la instalación "Appliance installation" mencionada en la guía de instalación de Qradar[7], apartado dos.

Siendo un "All-in-one", se mandó directamente a la dirección del CPD del cliente. Una vez se reservaron los recursos tales como las conexiones a la red del cliente mediante ethernet/fibra y el hueco en el RAC del CPD correspondiente, procedimos a instalarlo físicamente, configurando los parámetros básicos como la IP de ingesta y la IP de gestión, también el DNS, contraseñas y usuario root.

Las IPs de gestión e ingesta son distintas, y el funcionamiento es diferente: la de gestión se usa para controlar la consola, mientras que la de ingesta es la que se utiliza para recibir los distintos logs y poder analizarlos sintácticamente.

Con esta instalación tenemos acceso mediante SHH a la consola con el usuario 'root' y acceso mediante web con el usuario 'admin', con ambas contraseñas definidas por nosotros.

2.5.3 Como configurar usuarios

Una vez que se puede acceder mediante web a Qradar, estamos preparados para configurar los usuarios. Estos usuarios pueden ser tanto locales en el sistema, (opción

menos habitual) como la opción que se suele emplear en las empresas que disponen de Active Directory, que es la de usar usuarios de dominio.

En cualquier caso, la manera de configurar usuarios y roles es acceder al menú de inicio, y posteriormente hacer clic sobre "Admin" [Fig.1]. Así accedemos al menú de administración [Fig. 2], donde podemos ver que tenemos el apartado "Users" que es usado para crear usuarios y añadirlos a grupos, y el apartado "User Roles" que se usa para darles permisos a los grupos.

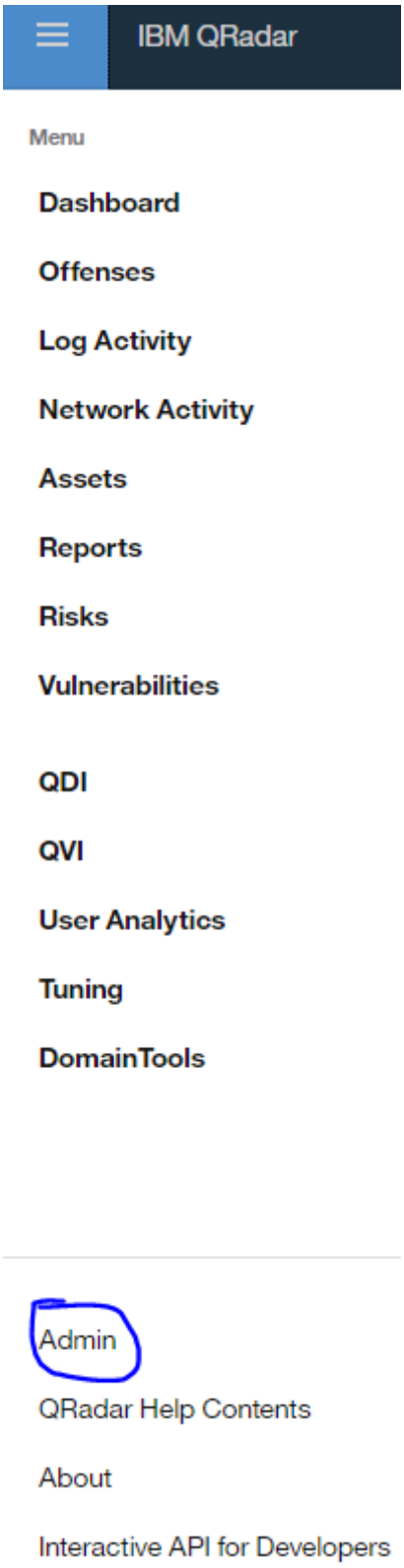


Figura 1. Acceso Admin.

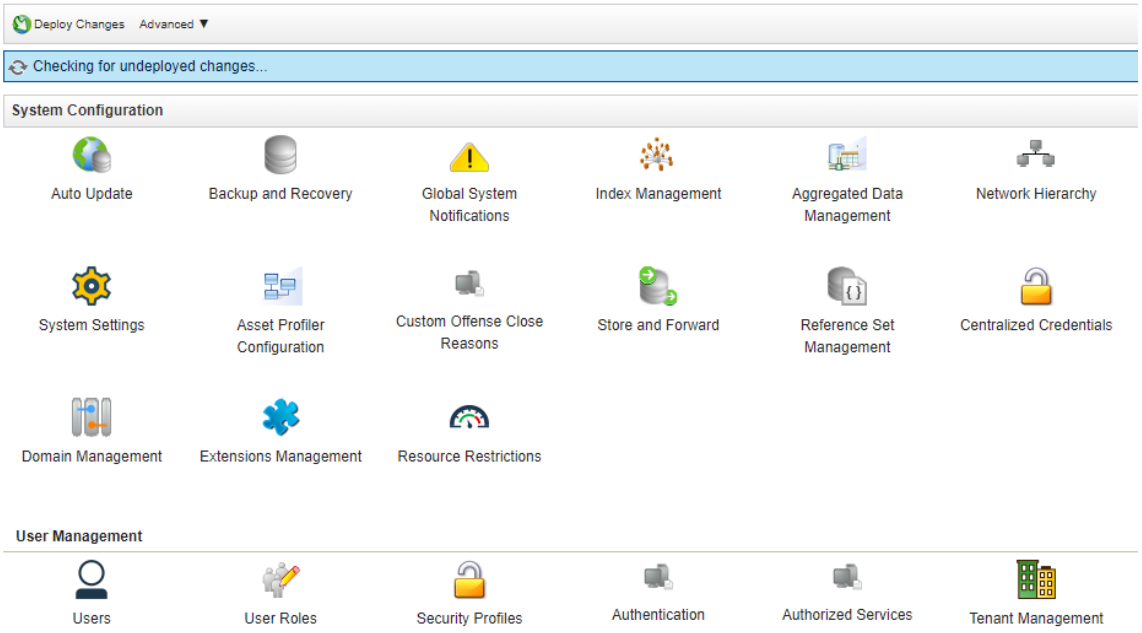


Figura 2. Menú Admin.

Los usuarios configurados deben ser al menos tres: un administrador del sistema, un usuario para el cliente, y un usuario de nivel L1 que es el que se encargará de tratar las ofensas (explicadas posteriormente), las investigaciones y el día a día del SOC con la herramienta.

2.5.4 Implementar aplicaciones básicas

Estas aplicaciones que se mencionan como básicas, son las que deberían formar parte de Qradar por defecto, y de hecho, es posible que en futuras versiones se integren desde el inicio por parte de IBM.

De cualquiera de las maneras, son aplicaciones que considero deben formar parte de cualquier instalación de Qradar. Esta apreciación se debe a las múltiples instalaciones realizadas, a las que en etapas posteriores he tenido que añadir estas aplicaciones.

Una aplicación es "Qradar assistant"[8] [Fig. 3]. Esta aplicación realiza una doble función esencial para el desarrollo diario de la operativa del SOC y para el manejo de la herramienta por parte de los administradores:

Figura 3. Logo Qradar assistant.

- Dentro de la pestaña "Home", existe guía técnica de uso de Qradar, consejos, aplicaciones recomendadas, y una academia en la cual los videos son la

herramienta de aprendizaje.

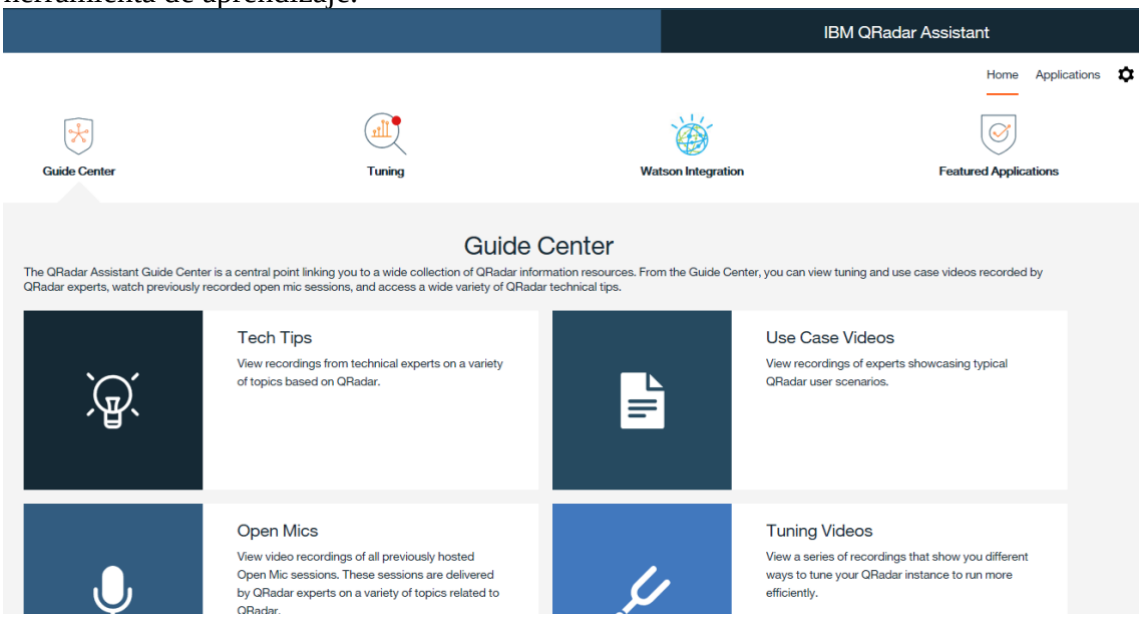


Figura 4. Pestaña "Home" Qradar assistant.

- Si se realiza un click sobre la pestaña "Applications" aparece una tienda de aplicaciones, al igual que podría ser la conocida Play Store o la Apple Store para smartphones. Esta parte supone una ventaja considerable a la hora de añadir aplicaciones y extensiones a Qradar, ya que en caso de no disponer de esta aplicación instalada, el proceso de instalación y búsqueda de aplicaciones es distinto, teniendo que acceder a la página de X-force, descargando el paquete correspondiente a la aplicación, subiendo el paquete al Qradar, y accediendo a la zona de admin para cargar la app.
  a partir de este punto, se supondrá que en cualquier cliente se ha instalado la aplicación de "tienda".

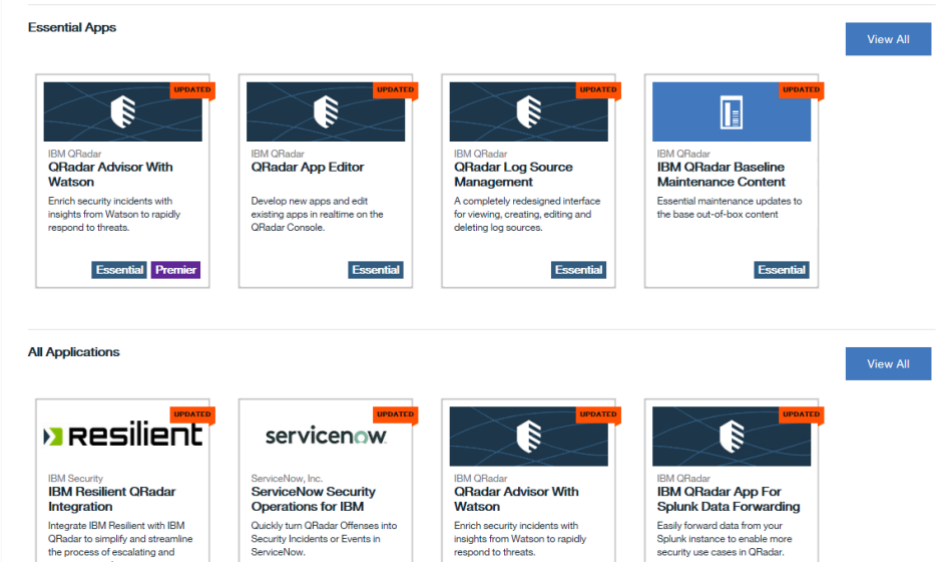


Figura 5. Pestaña "Applications" Qradar assistant.

Otra aplicación sería "Tuning"[9], que como su propio nombre indica, se usa para realizar modificaciones con el objetivo de mejorar la eficiencia de la propia herramienta.

2.5.5 Taxonomía (genérica)

Una taxonomía es una forma de clasificar, que empleada en el mundo de la seguridad y específicamente en referencia a las fuentes de logs que dan información al SIEM, sirve para clasificar la información de dichos logs mediante varios parámetros, con el objetivo de estandarizar e integrar todas las fuentes de todos los clientes del SOC en una serie de campos comunes para todas ellas, previendo el hecho de añadir otras fuentes, y facilitando las búsquedas a los integrantes del SOC de nivel 1, independientemente del cliente en el cual busquen la información y la fuente que proporcione dicha información.

Para realizar una taxonomía genérica en el presente TFM, no es suficiente con basarse en crear una que sea válida para un único cliente, sino que debe ser intercambiable entre clientes, fuentes de logs, SIEMs, y herramientas de defensa/ataque, tales como un EDR (Endpoint Detection and Response) o un FW (Firewall), así como herramientas de análisis y visualización como un ELK (Elasticsearch+Logstash+Kibana).

Para realizar esta taxonomía, es necesario examinar el conjunto de logs que disponemos en la compañía, evaluar los distintos campos que poseen e incluirlos todos en la misma. Así mismo, es necesario definir nombres comunes para los distintos campos, para evitar confusiones a la hora de llamar a un campo; un ejemplo podría ser "IP de origen", que en inglés, algunos logs lo llaman "IPsource" otros "srcIP", otros "origin_IP"…

Para evitar estas confusiones, se tomaron dos medidas, la primera de ellas fue basarse en una taxonomía genérica, y la segunda, una serie de reglas tales como:

- Basar la taxonomía en campos principales y subcampos
- Elegir un formato de separación por puntos entre campo y subcampo,
- Idioma: inglés
- Contracciones siempre que fuera posible y no ambiguas
- Minúsculas
- Separación de palabras mediante barra baja "_"

Siguiendo el ejemplo comentado hasta ahora, el nombre dado a la IP de origen de un dispositivo sería: "source.ip".

La taxonomía se basó en el esquema de Elasticsearch, conocido como ECS (Elastic Common Schema), que puede ser encontrado en su página oficial de github[10].

Para realizar esta taxonomía se creó un Excel en el cual la primera página contenía todos los campos de primer nivel, y un enlace a cada una de las hojas de estos campos, en las cuales se explicaban los posibles subcampos para cada uno de los campos. Podemos ver un ejemplo gráfico en la [Fig. 6]

Modelo de datos Genérico

Objetivo: Crear un modelo de datos unificado y transversal, con el fin de facilitar las búsquedas y las integraciones.

Base	Agent	Cloud	Container	Destination	Device	ECS	Firewall	Web
Error	Event	File	Geo	Host	Log	Network	SMTP	Threat
Organization	Operating System	Process	Service	Source	URL	User	Query	VPN

Figura 6. Taxonomía, campos de nivel 1

Para tener un ejemplo mas preciso de qué son los subcampos y de cual es su papel, además de cómo se componen con los campos principales, veremos la pestaña del campo "source", con todos sus posibles subcampos en la taxonomía creada [Fig. 7]:

Field	Description	Level	Type	Example
source.ip	IP address of the source. Can be one or multiple IPv4 or IPv6 addresses.	core	ip	
source.ipv6	Only IPv6	custom	ipv6	
source.port	Port of the source.	core	long	
source.mac	MAC address of the source.	core	keyword	
source.domain	Source domain.	core	keyword	
source.nat_ip	If source NAT performed, the post-NAT source IP address	custom	ip	
source.nat_port	If source NAT performed, the post-NAT source port	custom	long	3426
source.user_name	user that started the event	custom	string	
source.location	Location of the destination host	custom	string	
source.fqnd	A fully qualified domain name (FQDN) is the complete domain name for a specific computer, or host, on the internet. The FQDN consists of two parts: the hostname and the domain name	custom	string	
source.hostname		custom	string	

Figura 7. Campos nivel 2 (subcampos) del campo "source"

En este ejemplo podemos ver que en el segundo nivel es donde se explica para que sirve cada subcampo.

A parte de la explicación y del nombre que se le va a dar al campo/subcampo, podemos apreciar que tenemos más información en esta tabla. El nivel o "level" en inglés se refiere a los niveles definidos en la entrada al blog de ELK[11], en el apartado "What is the Elastic Common Schema?"; estos pueden ser tres:

- **Core:** Núcleo en inglés, son los campos más comunes entre distintas fuentes, y es altamente probable que cualquier sistema los posea. Los campos con este nivel deben aparecer en cualquier implementación del ECS. La definición e interpretación queda completamente definida en este tipo de campos.
- **Extended:** Extendido en inglés, son los campos que, aunque comunes, solo lo son en fuentes de un tipo determinado, como por ejemplo los antivirus. La definición e interpretación puede variar en función del caso de uso.
- **Custom:** Personalizados en inglés, estos son los campos que son exclusivos o muy poco comunes de cada sistema, y no tienen un equivalente en ECS, con lo que la especificación es o nula, y se deben definir desde cero por parte del implementador del sistema.

Otra columna que se puede ver en la [Fig. 7] es la de "type", que se refiere a qué tipo de dato se espera recibir y guardar en el campo correspondiente. Esto es útil en lo que se refiere a afinar y optimizar las búsquedas, así como ordenar resultados y encontrar fallos en análisis sintácticos incorrectos. También es útil para filtrar la información y comprobar que nos llega la que esperamos y no una distinta.

Por último se dispone de la columna "example", que es en la que se ponen ejemplos de los datos que pueden entrar en el campo indicado. Se ha eliminado la información ya que es confidencial, de la misma manera que no se adjunta el Excel completo por el mismo motivo.

2.6 Integración de fuentes

Para integrar las distintas fuentes mencionadas en Qradar, existe un procedimiento estándar que se encuentra en la propia web de IBM [12], aunque existen fuentes personalizadas a las cuales hay que dedicarle un esfuerzo mayor.

A continuación se mostrarán dos ejemplos, uno con una fuente de Microsoft que es frecuente en un entorno de producción como es el Active Directory, y otra con una fuente imaginaria basada en una real del cliente para el cual se realizó el proyecto. Para ambas se explicarán las distintas maneras existentes de integrarlas, las particularidades de cada una de ellas y cómo, a partir de la versión 7.3.2, desde IBM se sugiere una manera alternativa de integrarla de forma más sencilla. Así mismo, se comparará la versión anterior 7.2.X con la nueva 7.3.2, ya que en el apartado de integración de fuentes se han realizado mejoras sustanciales entre estas versiones.

2.6.1 Tipo de fuente preexistente en Qradar: Windows active directory

Como ejemplo se mostrarán los logs de seguridad de directorio activo en un servidor Windows 2016, aunque con el método expuesto se pueden conseguir integrar gran parte de las fuentes de logs de Windows existentes en el mercado.

Antes de realizar acciones en el Qradar para la integración de la fuente, es importante configurar un agente de Wincollect. Este agente es un pequeño programa que permite a Qradar recolectar información del sistema Windows (WINdows COLLECTor) y transferirla por un protocolo propietario al SIEM. Las instrucciones de instalación de este agente se les deben transmitir al departamento de arquitectura de red o departamento IT. Las instrucciones y los archivos necesarios para ello se encuentra disponibles en el propio repositorio de información de IBM[13].

Uno de los parámetros será necesario que empleen los arquitectos cuando instalen el agente, es llamado "Authentication Token". Es una combinación de caracteres alfanuméricos de la forma de "af111ff6-4f30-11eb-11fb-1fc117711111", y que debe generar a priori en el Qradar. A continuación se explica como generarlo:

1- Se accede al menú de administrador de Qradar

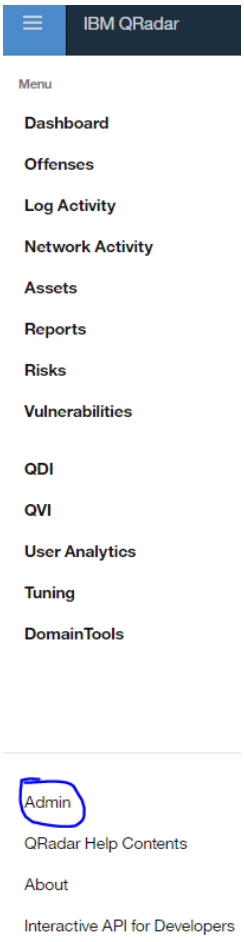


Figura 8. Acceso Admin.

2- Se accede a "Authorized services" haciendo clic sobre la imagen sombreada en azul de la [Fig. 9]

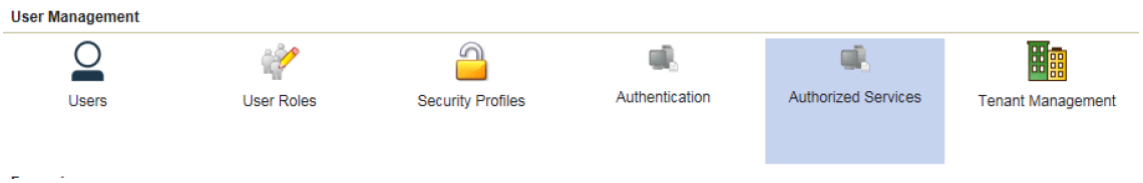
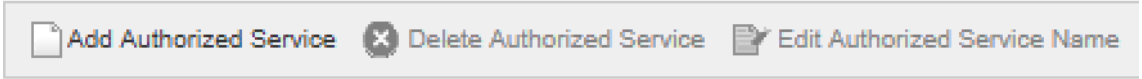


Figura 9. Acceso Admin.

3- Aparecerá una ventana con los token ya creados, y hay que pulsar sobre "Add authorized Service"



4- Aparece un formulario para rellenar en el cual hay que completar los datos como requiera cada aplicación. En este caso, se enscogerá “admin” en ambas opciones, y no se dispondrá de fecha de caducidad. También se le dará un nombre al servicio (en este ejemplo se le llama ‘test_alexis_wincollects’)

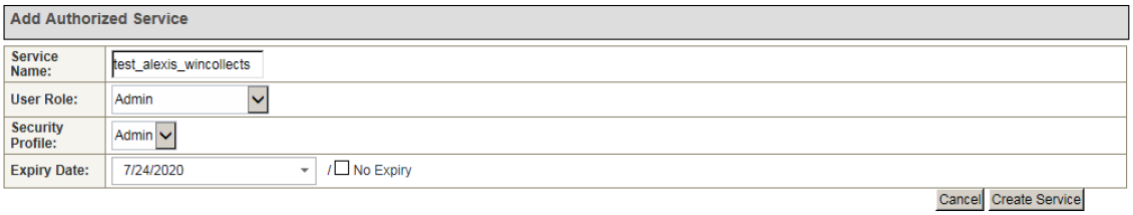


Figura 10. Formulario para crear un Authorization token.

El wincollect se instalará en este momento en el servidor Windows 2016 que es usado como ejemplo, pero falta sincronizarlo con Qradar.
Cuando se instala un wincollect, éste intenta conectarse con la dirección IP que se le ha proporcionado en los parámetros que se le han dado al instalarse. En caso de que esta acción falle (en Qradar es llamada ‘autodiscover’), queda la opción de realizar una detección manual, realizada como se muestra a continuación:

1- Acceder al panel de administrador [Fig. 8]
2- Dentro del apartado “data sources”, acceder a “wincollects”, siendo esta la imagen sombreada en azul en la [Fig. 11]

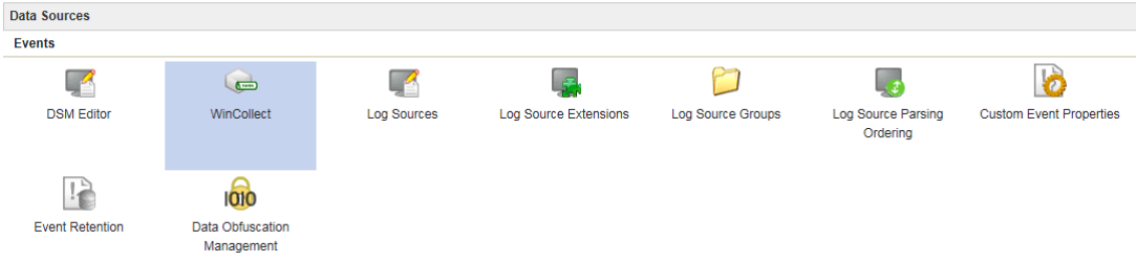


Figura 11. Acceso a Wincollect

3- Aparecerá una ventana con todos los agentes disponibles y con la apariencia de la [Fig. 12]. Se debe hacer clic sobre el botón “Add”

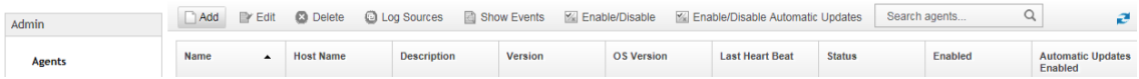


Figura 12. Añadir Wincollect

4- Por último, Se rellena el formulario que aparece en la [Fig. 13] y se verifica su correcto funcionamiento.

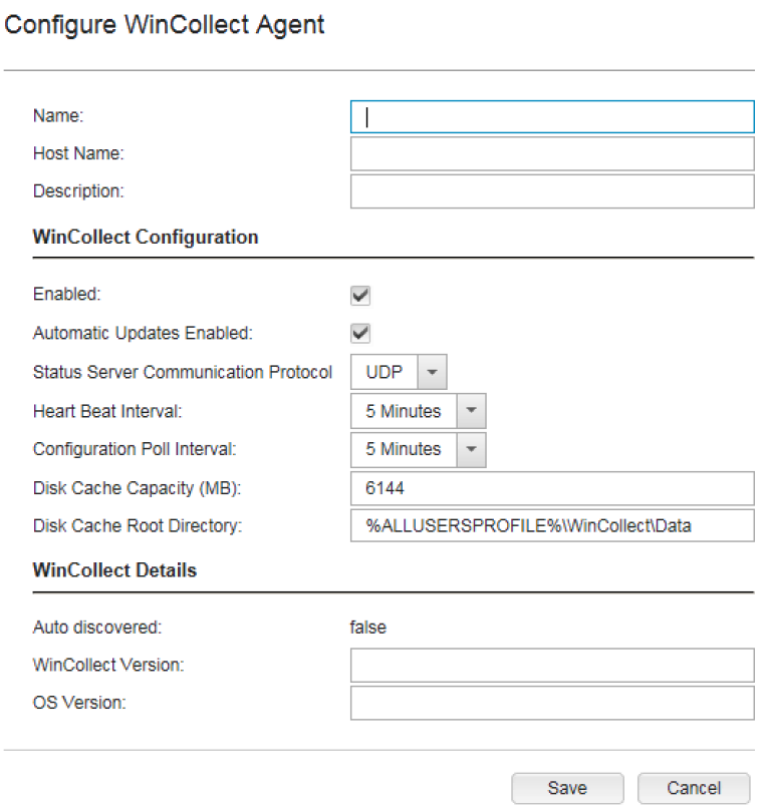


Figura 13. Formulario para añadir Wincollect

El wincollect se encuentra ahora instalado y sincronizado con el Qradar, y está disponible para ser usado por una o varias fuentes concretas. Para ello será necesario acudir al panel de administrador, y pulsar sobre "Log Sources", como se indica en la imagen sombreada en azul de la [Fig. 14]

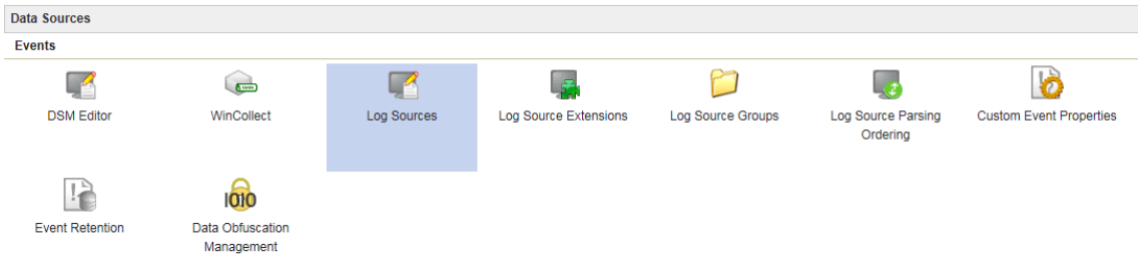


Figura 14. Acceso a "Log Sources"

Aparece otra ventana en la cual es posible añadir una fuente pulsando sobre el botón "Add" que se muestra en la [Fig. 15], y posteriormente surge un formulario como el de la [Fig. 16], en el cual se elige el tipo de fuente que se prentende crear. En él elegimos como tipo de fuente "Microsoft Windows Security event log" escogemos nombre, descripción, identificador (este parámetro es importante ya que las búsquedas se realizarán con él), y el resto de parámetros que pide el formulario, de los cuales no se adjunta captura.

Figura 15. Añadir "Log Sources"

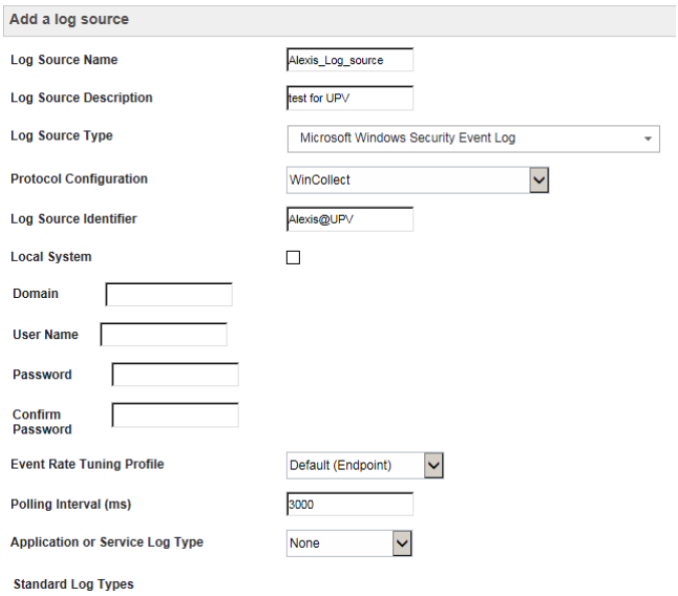


Figura 16. Formulario para añadir nueva "Log Source"

A pesar de que los campos del formulario son múltiples y cada uno de ellos es importante, es conveniente fijarse en un dos de ellos, indicados en la [Fig. 17]:

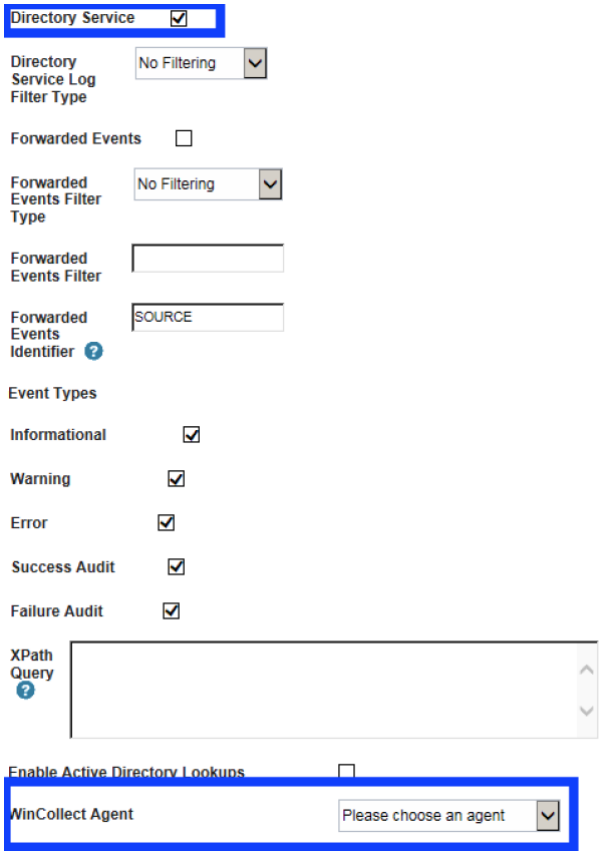


Figura 17. Formulario para añadir nueva "Log Source"

El campo "Directory service" debe estar marcado para transmitir logs de directorio, y en el apartado de "WinCollect Agent" es dónde se elige el agente de wincollect creado anteriormente.

La creación de fuentes con wincollect con la versión previa es la explicada hasta ahora. Se compararán las ventajas de la nueva versión de Qradar a continuación.

Cuando se accede a "log sources" como se muestra en la [Fig. 14], aparece un aviso que se puede apreciar en la [Fig. 18]

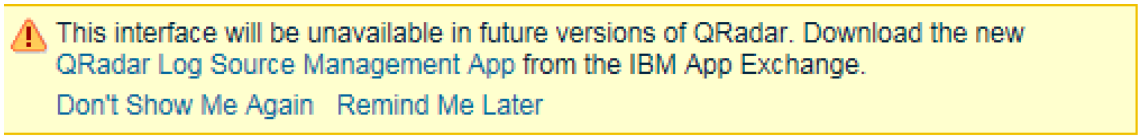


Figura 18. Aviso nueva interfaz para "Log Sources"

Si se siguen las instrucciones dadas por este aviso, y se descarga la aplicación "Qradar Log source Management App", la interfaz y el modo de integrar la fuente se simplifican, y puede ser comprobado en la [Fig. 19].

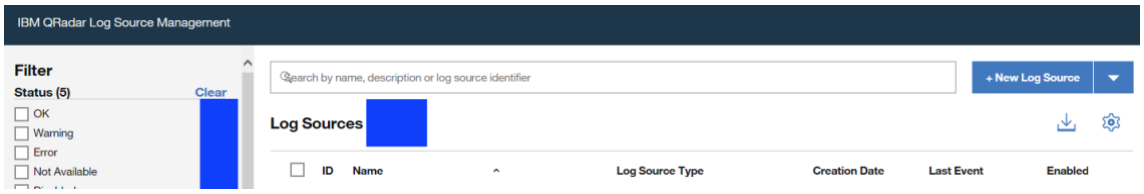


Figura 19. Nueva interfaz "Log Sources"

Esta nueva interfaz, añade filtros mejorados, un rediseño visual y un "wizard", un instalador paso a paso que va pidiendo de forma ordenada los datos y explica qué es cada uno de ellos y dónde obtenerlos. Ver [Fig. 20]

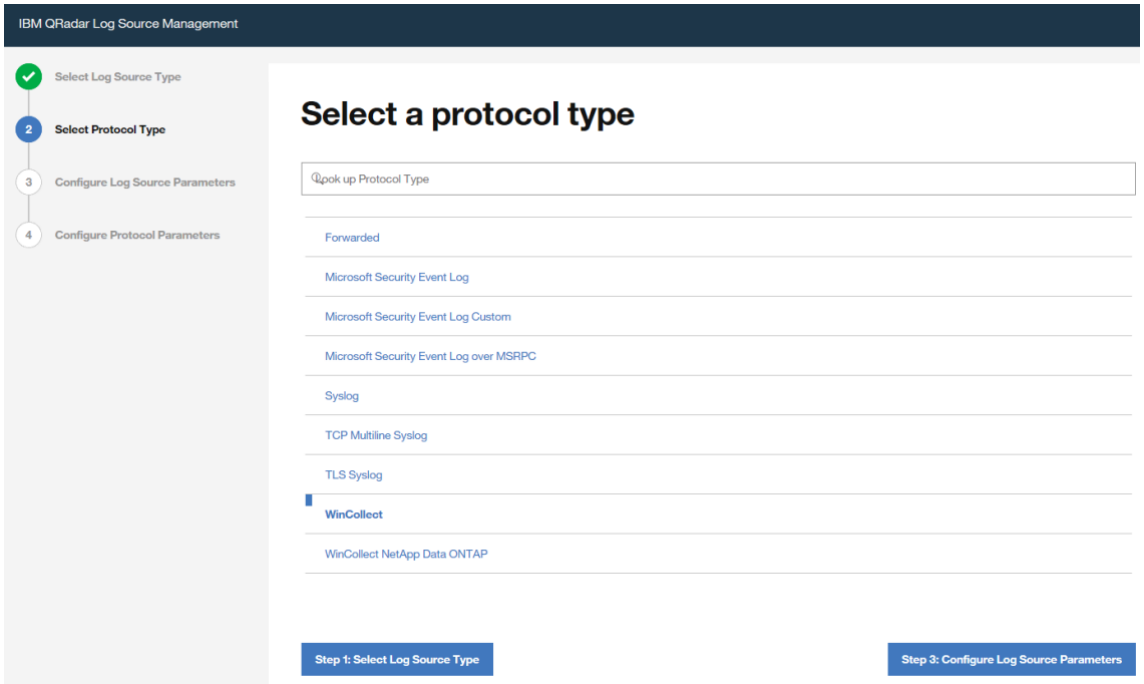


Figura 20. Nuevo formulario para añadir "Log Source"

Cualquiera de estos métodos es válido para obtener la integración de fuentes de logs de seguridad de Windows, como podría ser el directorio activo, no obstante, se pueden optimizar los datos obtenidos de este tipo de fuentes realizando las siguientes acciones: Instalar la aplicación correspondiente a la fuente que hemos instalado, que funciona como una extensión para el análisis sintáctico y la creación de relaciones entre logs y su contenido con eventos (mapear en jerga de seguridad). En este caso, la app existe, está disponible y actualizada; en el caso en el cual alguno de estos tres parámetros no se cumpliese, se recomienda no descargar la aplicación y proceder como se explicará en el apartado siguiente.

En este ejemplo, la aplicación se denomina "IBM QRadar Custom Properties for Microsoft Windows"[14], y está disponible para instalarla desde el asistente de Qradar comentado en apartados anteriores de esta memoria.

Esta app añade 46 "Custom properties", que son equivalentes a índices de almacenado de información (o columnas en el cas de tablas SQL), así como relaciones log/evento y análisis sintáctico de los logs.

2.6.2 Tipo de fuente NO existente en Qradar

Si el tipo de fuente no existe en Qradar, es necesario crearlo y añadir a mano todo el análisis sintáctico de logs y las relaciones logs/eventos que se sustentan sobre en análisis.

En el apartado anterior de la memoria debíamos elegir un tipo de fuente concreto, se debe a que Qradar ya disponía del tipo de fuente que se integraba, y por tanto incluía el análisis sintáctico que hay que realizar con los logs y las relaciones necesarias para transformarlos en eventos. Cuando las fuentes de logs no son de ningún tipo predefinido por Qradar, no se sigue el proceso explicado anteriormente, sino que se le añaden pasos y se modifican otros: hay que definir qué análisis sintácticos son necesarios para que Qradar extraiga del texto que le llegará de la fuente la información que será de utilidad, y también crear de forma manual las relaciones entre logs y sus contenidos y eventos.

A continuación, se explicará cómo crear un nuevo tipo de fuente, cómo realizar los distintos pasos, y se compararán las versiones antigua y moderna de Qradar.

Sin embargo, se necesita el conocimiento de las expresiones regulares para realizar la tarea propuesta. Se realizará un inciso para explicar estas expresiones y posteriormente se proseguirá con la explicación.

Las expresiones regulares son expresiones usadas para búsqueda de texto, y se usan en múltiples lenguajes de programación, así como en el sistema Linux, con el comando "grep". Debido a que la información sobre expresiones regulares, también conocidas como 'RegEx' podría ser objeto de otro proyecto completo, se ha elaborado un resumen el cual se explica a continuación con las expresiones básicas y su significado, de tal manera que pueda ser comprendido y no sea un ente extraño para el lector.

En el momento del desarrollo del proyecto en el cual se debían usar las RegEx, no se disponía de este conocimiento previo, con lo que se obtuvo a través del autoaprendizaje[15][16], y posteriormente se realizó una formación al equipo para que pudiera realizar el resto de las tareas relacionadas con análisis sintáctico y relaciones

log/evento. El contenido mostrado a continuación se encuentra en inglés ya que forma parte de la formación que se realizó a los miembros del equipo.

RegEx [Regular Expressions] (search pattern):

- Quantifiers
  - ? :Zero or once
  - * :Zero or more
  - + :One or more
  - {x} :exact x times
  - {x,} :x times or more
  - {0,y} :less than y times or y
  - {x,y} :min x times and max y times
- Greedy/Lazy behaviour:
  - ".+"
  - ".+?"
- Metacharacters
  - . :ANY character except new-line
  - [^abc] :All characters BUT "a", "b" and "c"
  - [tuv] :some of "t", "u", or "v"
  - [1-9] :All characters between "1" and "9"
- Operators
  - | : OR operator
- Anchors
  - ^ :Start of string or line
  - $:End of the string or line
- Special modifiers (Must be used with "\")
  - \d :All digits
  - \D :All NON-digits
  - \s :All whitespaces
  - \S :All NON-whitespaces
  - \w :All alphanumerics "_" included
  - \W :All NON-aphanumeric "_" excluded
- Capturing group
  - (V) :Create capturing group (where V is the capturing group. EJ: (ab?))

- (?<t>V) :Name capturing group (V is the original capturing group, t is the name given to the group)

- (?:V) :ignores capture group V

Ciertos temas avanzados no serán expuestos en el presente TFM, tales como los 'lookarounds' o las llamadas a los grupos de captura dentro de la propia RegEx.

Retomando la creación de fuentes no predefinidas, seguiremos los siguientes pasos:

2.6.2.1 Creación del tipo de fuente

Al comienzo, se debe crear un tipo de fuente, el cual se elegirá posteriormente cuando se vincule la susodicha fuente con el Qradar. Para ello se accede al panel de administración como se muestra en la [Fig. 8], y luego se realiza un clic sobre la pestaña remarcada en azul de la [Fig. 21], accediendo a la herramienta de Qradar llamada DSM editor. Esta es la herramienta principal que se usará tanto para el análisis sintáctico como para la relación log/evento. Dentro de DSM editor, se aprecia un selector como el de la [Fig. 22], y pulsando sobre "Create New" aparece un único campo que rellenar con el nombre del tipo de fuente personalizado elegido; para este ejemplo se ha usado el nombre 'Test_Alexis_tipo_de_fuente'.

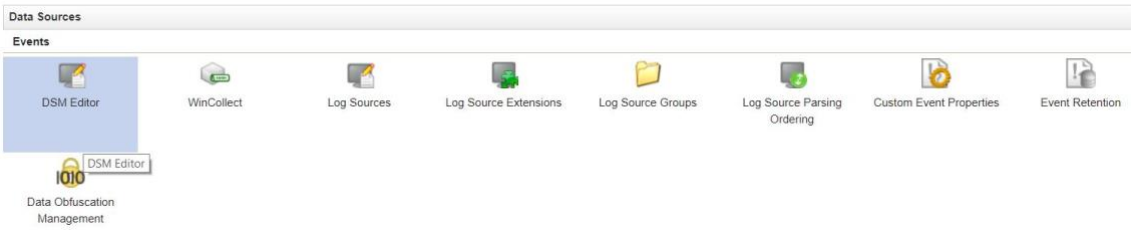


Figura 21. Acceso "DSM editor"

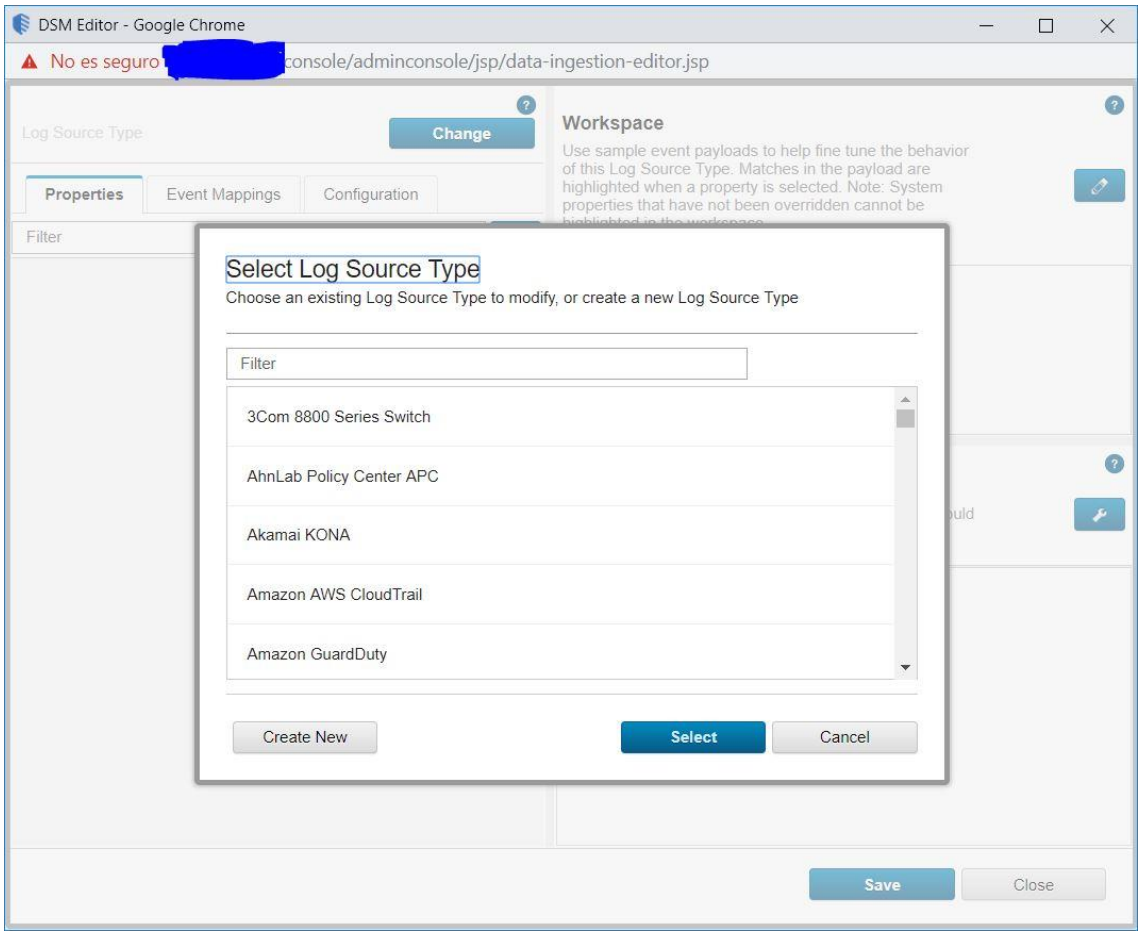


Figura 22. Selector "DSM editor"

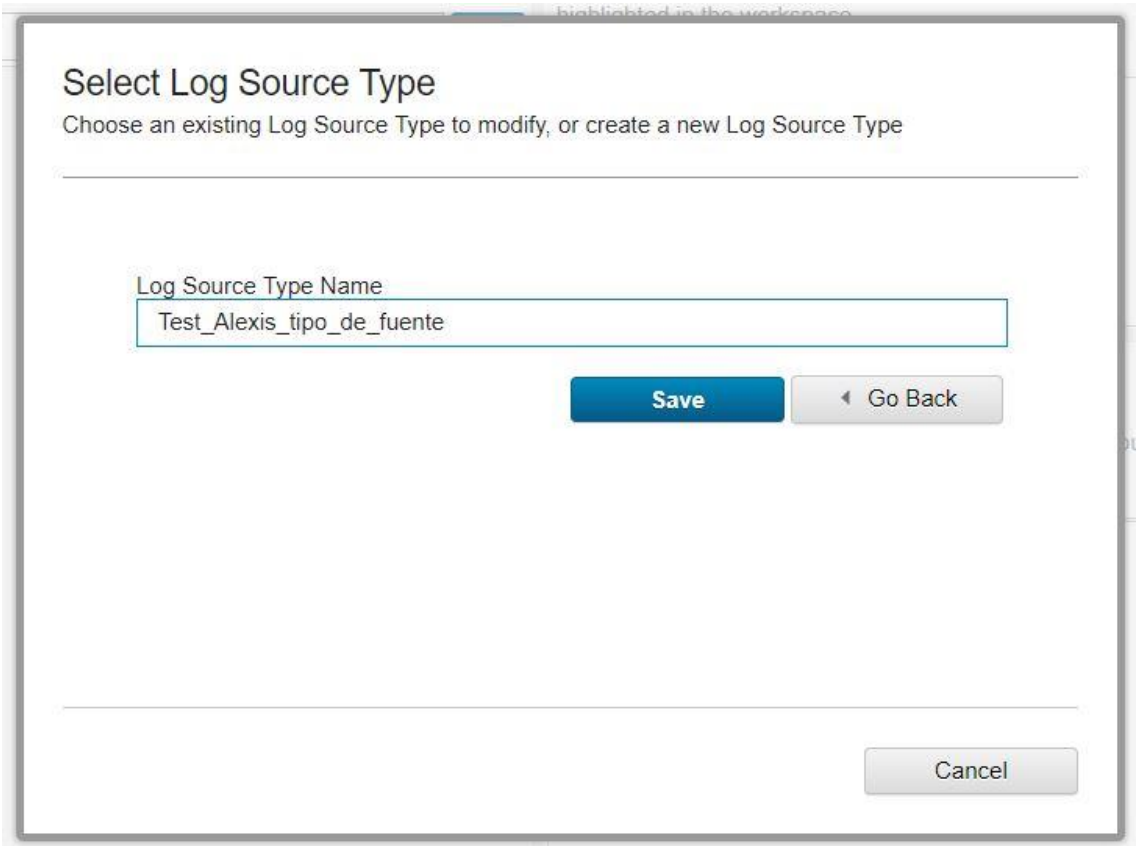


Figura 23. Selector "DSM editor"

2.6.2.2 Análisis sintáctico de los distintos campos. Automático

Aunque se ha definido un nombre para el tipo de fuente específico creado, no se dispone de un tipo de fuente definido como tal, ya que no se han introducido los parámetros necesarios para extraer información de los logs de esta fuente. Para este ejemplo se han utilizado los logs libres existentes en la dirección: https://ossec-docs.readthedocs.io/en/latest/log_samples/ftp/microsoft.html, Sample 3. Estos logs son de Microsoft FTPD, y se van a usar para ejemplificar el comportamiento y el desarrollo de un tipo de fuente propia, ya que normalmente las fuentes propias son instalaciones poco habituales y particulares de cada cliente.

Para analizar sintácticamente los distintos campos, es necesario copiar varios logs de prueba en el DSM editor, para comprobar que los cambios surten efecto. en la pestaña "Configuration" [Fig. 24], se activa la segunda opción consiguiendo que a medida que a Qradar le vayan llegando eventos de este tipo de fuente los detecte de forma automática.

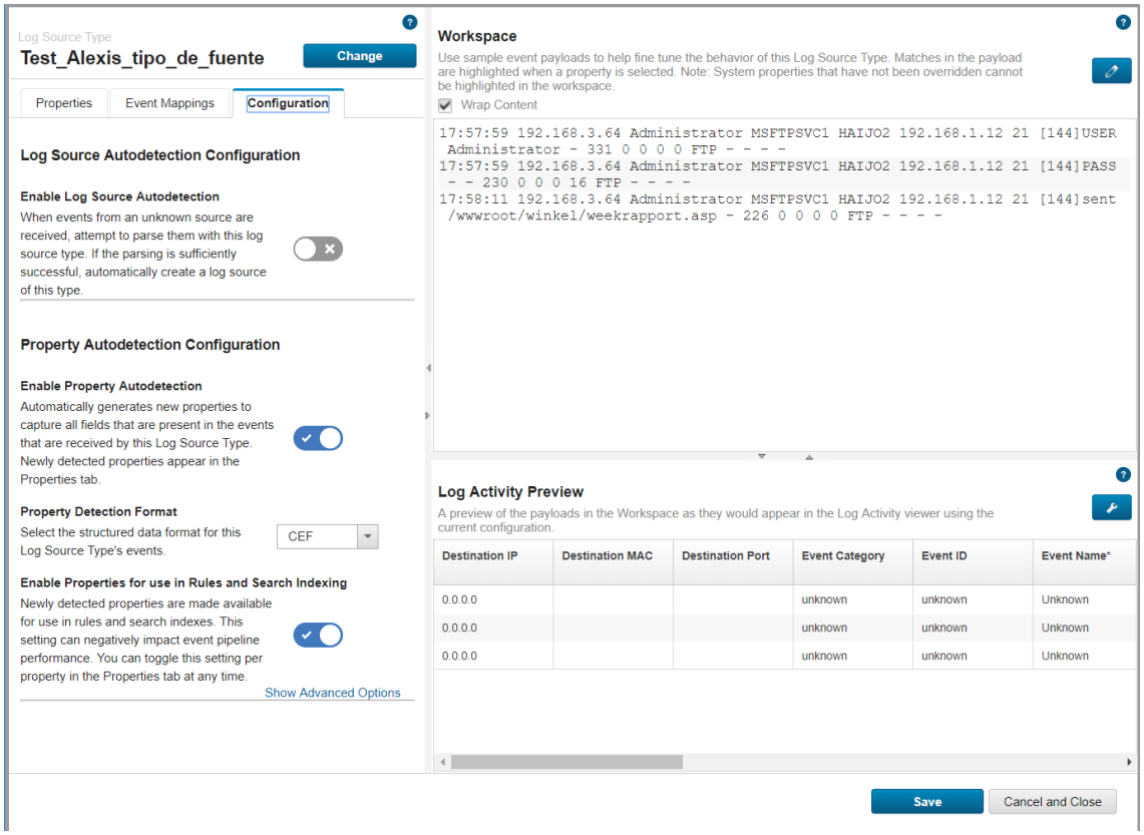


Figura 24. Auto-detección de propiedades/campos "DSM editor"

2.6.2.3 Análisis sintáctico de los distintos campos. Manual

Será necesario sobrescribir el comportamiento por defecto de la fuente si Qradar no es capaz de detectar los parámetros básicos, si la detección no es satisfactoria o si se desea incluir algún campo nuevo o que se recibirá en un futuro.

En la versión anterior de Qradar, solo existían las posibilidades de "JSON" y "Regex", que son respectivamente:

- El log es un array/objeto con una sintaxis JSON, y solo tendríamos que darle la ruta,
- Buscar el texto mediante una expresión regular, como las comentadas anteriormente.

Generalmente se usaba la opción "Regex"; un ejemplo puede visualizarse en la [Fig. 25]. En este ejemplo se puede apreciar una regex muy sencilla y poco precisa cuya funcionalidad es no hacer "match" hasta pasar los 9 primeros caracteres y asegurarse de que después se encuentra una IP versión 4. En realidad se asegura de que son tres números seguidos de un punto cuatro veces; Las RegEx que no se usan con intenciones formativas suelen ser más complejas.

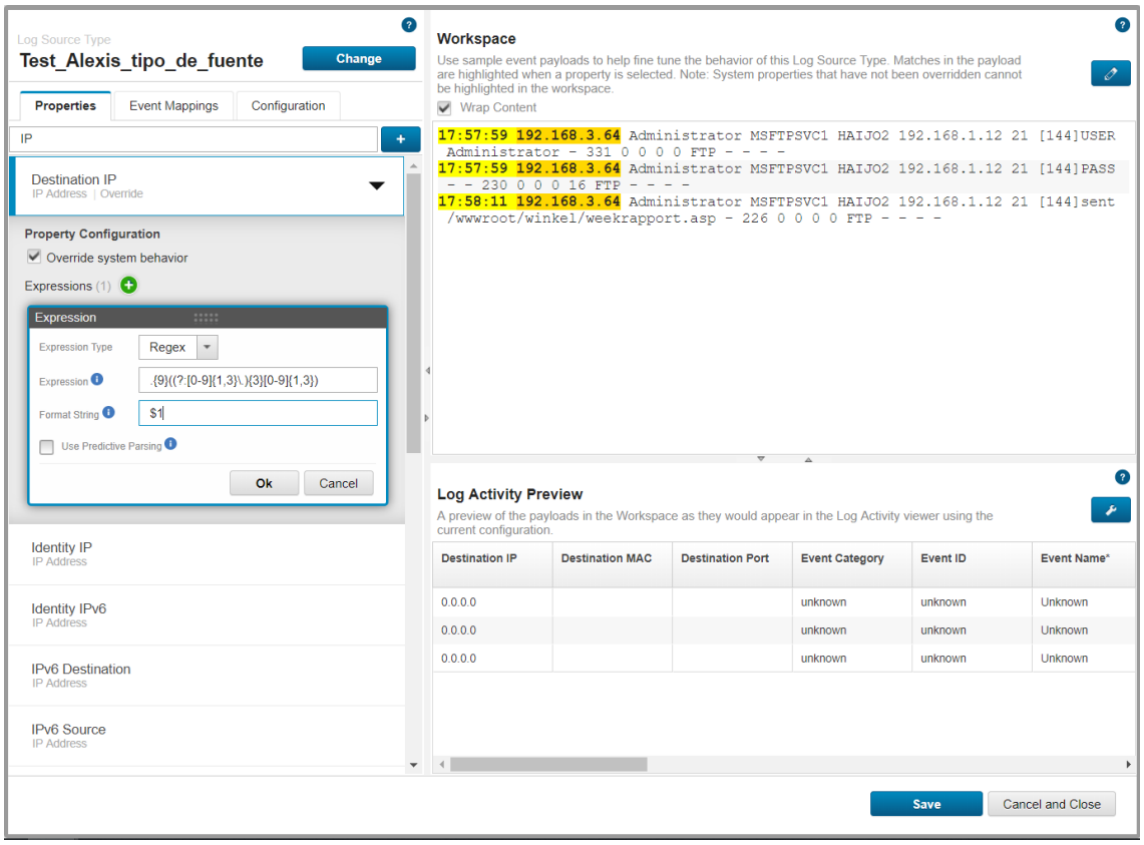


Figura 25. Análisis sintáctico con RegEx "DSM editor"

Se tiene también el campo de "format string", que sirve precisamente para darle un formato correcto a la cadena de texto que se desea almacenar, por ejemplo, si la IP se encuentra separada por comas en vez de puntos, se pueden sustituir dichas comas por el carácter que convenga con este campo. No siempre existe esta opción de darle formato al texto, sólo se puede realizar en algunos campos por defecto.

En los campos en los cuales se espera una fecha (como "Log Source Time"), se añade un campo más, llamado "Date Format", visible en la [Fig. 26], que sirve para comunicar al SIEM qué parte de la fecha y la hora es cada parte del texto capturado. Este campo existe ya que las diferentes herramientas que son fuentes de log para Qradar u otro SIEM, tienen distintos formatos de fecha y hora dependiendo de la región, el idioma, la herramienta, el desarrollador… con lo que se hace necesario definir qué son las horas, los minutos, los segundos, el día, el mes y el año.

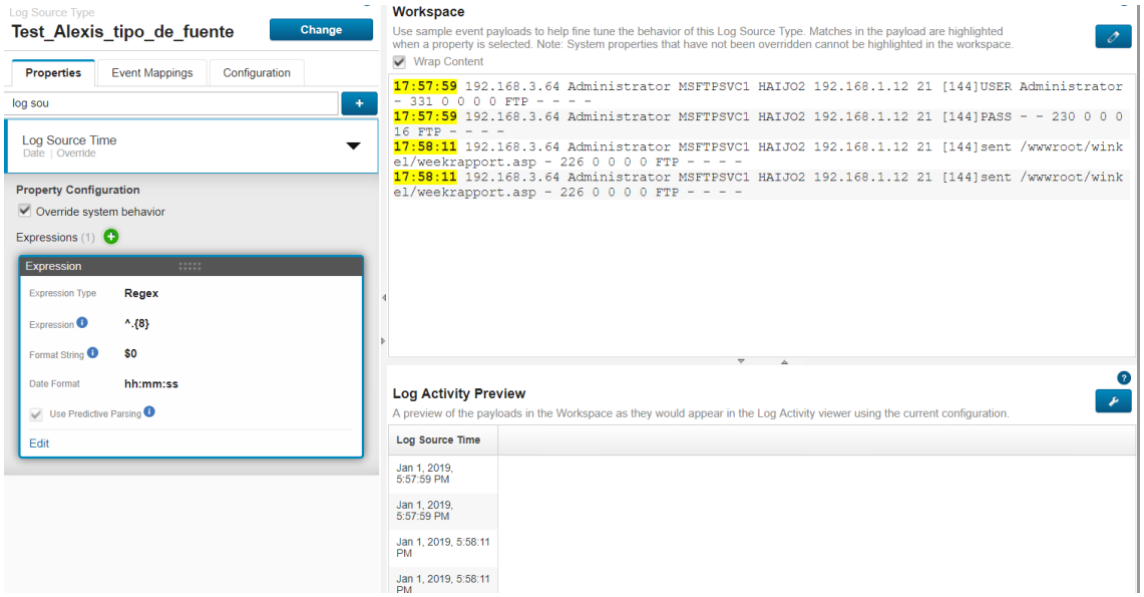


Figura 26. Análisis sintáctico de campos específicos "DSM editor"

Comparando con la nueva versión de Qradar, esta incluye dos tipos de análisis sintáctico clave que no poseía la anterior, y son los tipos "LEEF" y "CEF", ambos son formatos para la transmisión de logs en texto, y ambas opciones tienen el mismo nombre que los formatos que son capaces de reconocer. Este hecho presenta una ventaja notable frente a las regex de cara al usuario, ya que no debe conocer este lenguaje y ahorrará tiempo en pensar y escribir una RegEx que encaje sin ser necesario. Estas nuevas opciones de detección relegan a un uso circunstancial a la opción de RegEx, y aportan un gran valor a la solución global.

Se puede ver en la [Fig. 27], que una explicación completa emerge cuando acercamos el ratón al panel de información, hay que escribir el nombre del campo que se quiere obtener rodeado por símbolos de dólar ($), o en caso de que el campo no pertenezca a la parte fija de CEF, la palabra clave que lo define y existe previa al símbolo igual (=). En este caso particular, bastada introducir el texto 'act' para que Qradar identifique el campo al cual se hace referencia y lo extraiga de manera satisfactoria.

En la versión antigua es necesaria una Regex personalizada. Se creó por tanto una RegEx genérica con la misma intención con la cual ahora Qradar nos da estas dos opciones, simplificar el proceso de análisis sintáctico. La expresión es la siguiente: 'act=(?:(.*?)(?:(?: \b\w+=)|$))', donde "act" se sustituye por la palabra clave que ahora puede ponerse directamente en DSM editor con la opción CEF o LEEF. Esta expresión buscaba la siguiente palabra seguida de un símbolo '=', o el final del log, con lo que era capaz de actuar de forma correcta aunque el valor buscado se encontrara en distintas posiciones, siendo intercambiable entre palabras clave sustituyendo la misma al principio de la expresión, sin depender de ningún otro parámetro.

Con la incorporación en Qradar de las opciones CEF y LEEF, se simplifica este proceso.

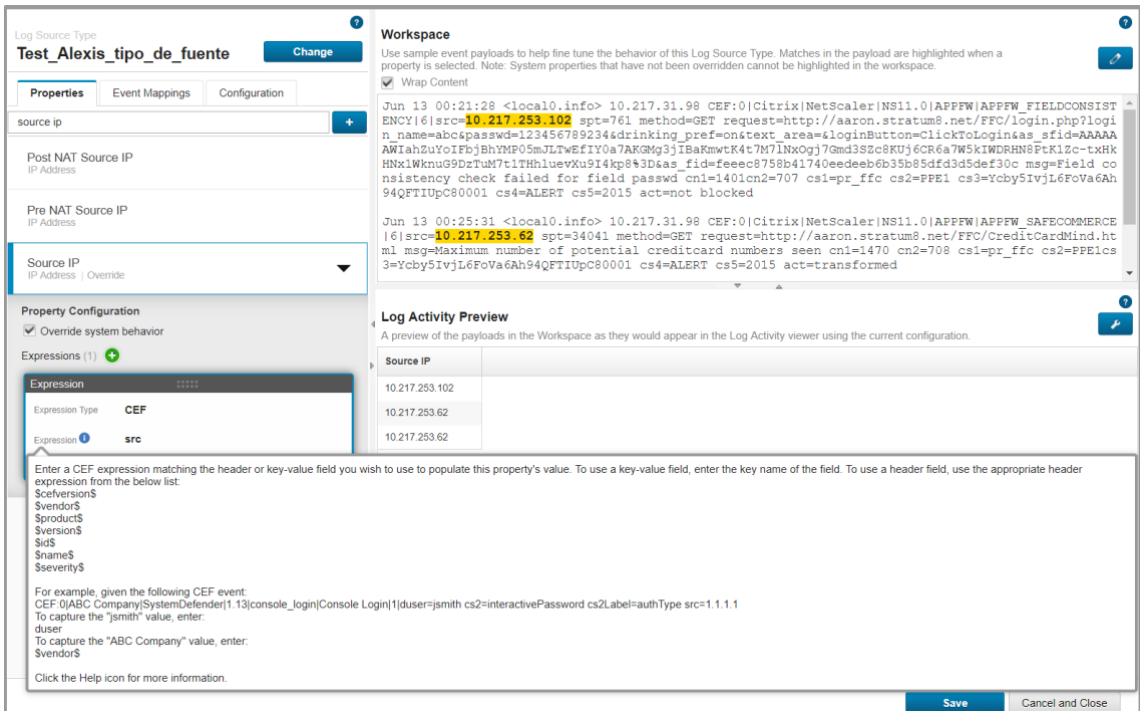


Figura 27. Opción CEF, "DSM editor"

2.6.2.4 Relaciones logs/eventos

Crear relaciones log/evento es establecer una correspondencia entre un log proveniente de una fuente concreta y con una parte de información fija, de tal manera que por cada log se genere un evento con ciertas propiedades definidas, como un nombre, una severidad, una categoría, una subcategoría… Los eventos deben ser representativos de la situación ocurrida que el log transmite. Estos eventos serán usados posteriormente en búsquedas, creación de reglas etc.

Un ejemplo es un log de un firewall, en el cual se especifican las direcciones IP de origen y destino, también la acción que el firewall ha llevado a cabo, cuál era la petición original, y el porqué de la acción llevada a cabo. Estos campos pueden variar en función de qué haya ocurrido aunque la información importante y prioritaria de estos logs es conocer si se ha permitido o denegado la conexión, con lo que aparte de analizar sintácticamente toda la información contenida en el log para que quede almacenada, es conveniente crear relaciones log/evento (mapear en jerga de seguridad) estos logs de firewall a dos eventos distintos:

- **Firewall Allow:** Este evento significa que la conexión se ha permitido, y dentro del mismo encontraremos todos los datos comentados anteriormente. Para distinguirlo de cualquier otro evento, se usa al menos la información del campo en el cual se encuentra la acción realizada.
- **Firewall Deny:** La conexión a través del firewall se ha denegado, encontraremos más información dentro del evento o en el log original. Para distinguirlo usaremos el campo en el cual se encuentra la acción realizada.

En Qradar se realizan las relaciones log/evento mediante dos campos claves que deben aparecer siempre en cualquier log. En caso de no disponer de ellos o de ser estos

"unknown", no se podrá realizar una asociación correcta, debido a este dato, estos campos deben ser analizados y almacenados antes de comenzar a realizar asociaciones log/evento.

Estos campos son:

- Event ID
- Event Category

Como su propio nombre indica, el campo "Event ID" es el identificador del evento, que puede ser una cadena de texto alfanumérica. El campo "Event Category" es usado para definir la categoría del evento. Siguiendo con el ejemplo anterior del firewall, el campo "Event ID" podría ser el mismo valor del campo action (véase 'Allow'), mientras que el campo "Event Category" podría ser la categoría del evento de firewall, si existiese. Cualquier valor es válido para estos dos campos, siempre y cuando existan y además identifiquen de forma unívoca el tipo de log que se asociará al evento correspondiente.

Para llevar a cabo el proceso necesario para relacionar logs con eventos, se puede realizar de dos formas distintas, desde "DSM editor" o desde un evento que ha aparecido como 'unknown' cuando se buscan eventos por cualquier motivo. En ambos casos, se deben tener definidos los campos comentados con anterioridad, "Event ID" y "Event Category". A continuación se mostrará cómo realizar la asociación log/evento, y para ello se usarán eventos en formato CEF y con el protocolo Syslog[17] (también adjuntos como .txt a este TFM).

1- Se copian los eventos en el entorno de pruebas de DSM editor
2- Se analizan sintácticamente "Event ID" y "Event Category", como se muestra en la [Fig. 28]. Se aprecia en la figura en la parte derecha superior tenemos los logs en crudo, a la izquierda cómo parsearlos y a la derecha abajo el resultado.
3- Se accede a la pestaña "Event Mappings", se pulsa sobre el botón azul con un "+", y se rellena el formulario con los datos que nos proporcionaba el análisis sintáctico del log en cuestión [Fig. 29]
4- Se pulsa sobre "Choose QID…", aparece una lista de eventos [Fig. 30] y se filtran hasta elegir el que se desea asociar con este log, o se crea uno nuevo y se asocia el mismo.
5- Mediante el botón "Create" se termina el proceso, y se comprobará cómo los campos con asterisco que antes eran 'unknown' toman los valores definidos por el evento [Fig. 31]

El proceso definido se repetirá tantas veces como relaciones se deseen crear.

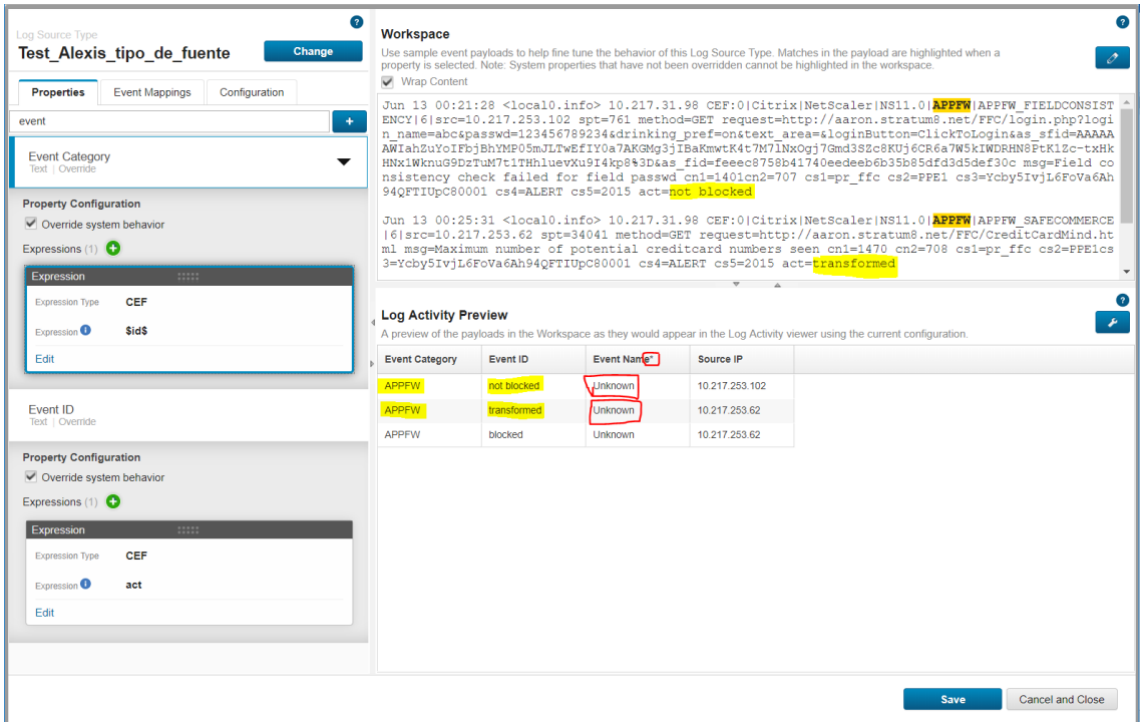


Figura 28. Análisis sintáctico con el objetivo de crear relación log/evento

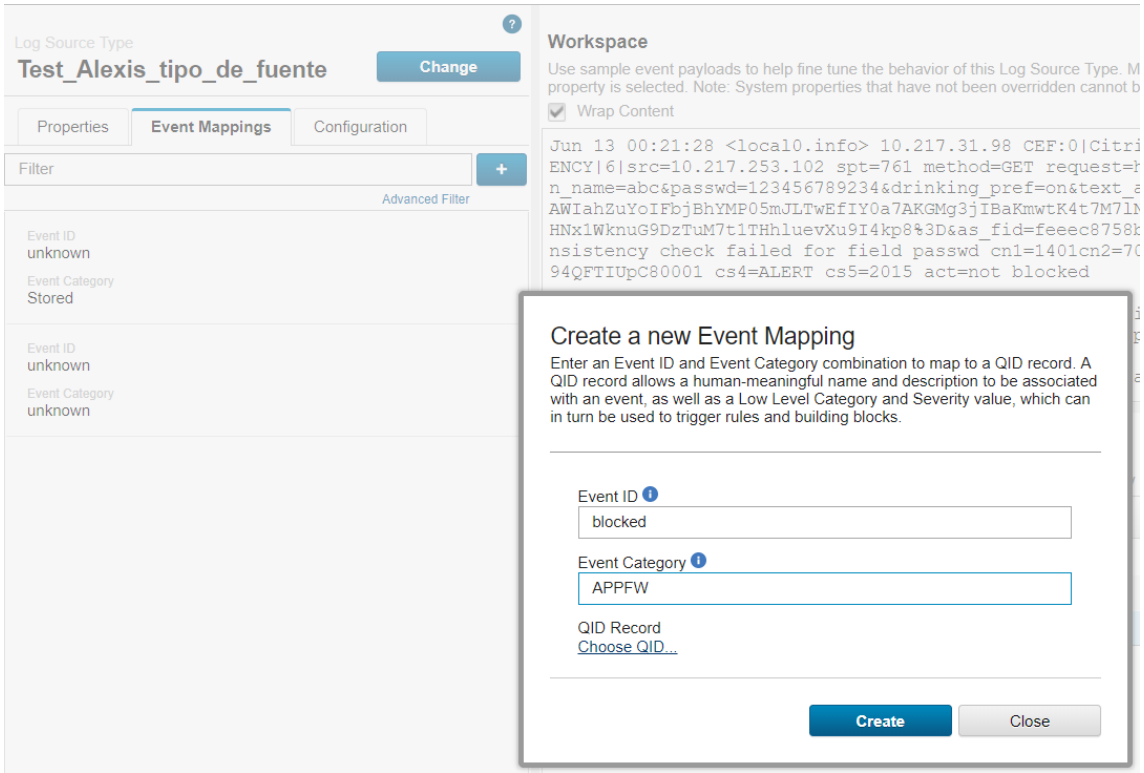


Figura 29. Proceso de creación de relación log/evento

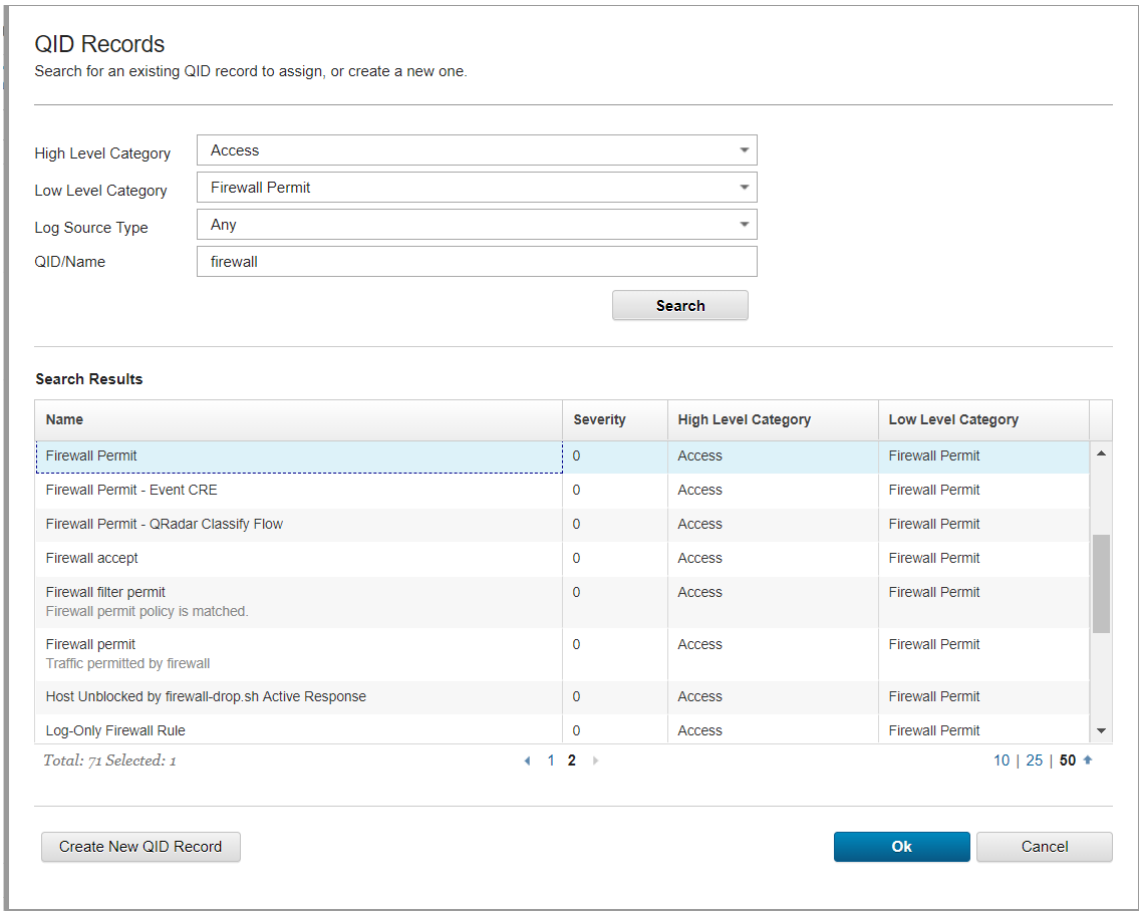


Figura 30. Selección de QID

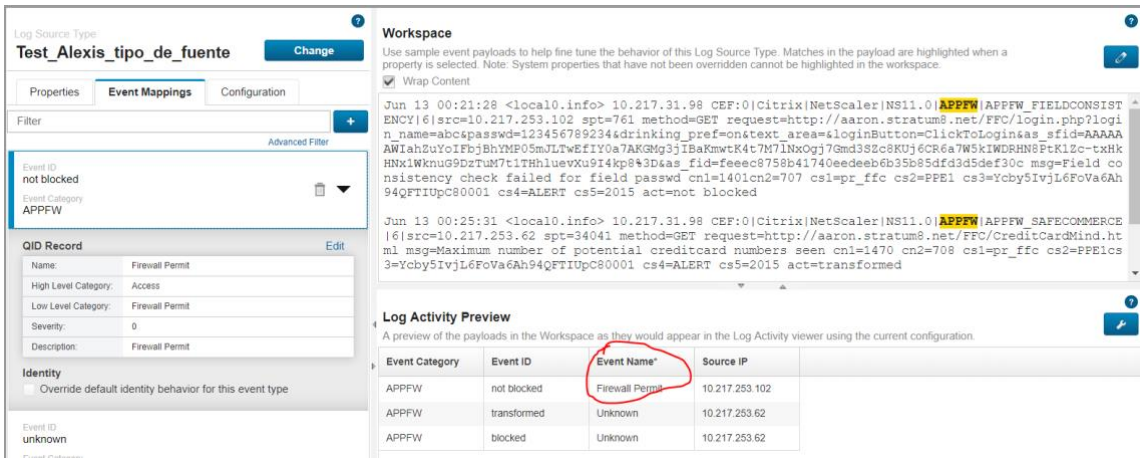


Figura 31. Comprobación de la correcta creación de la relación log/evento

Capítulo 3. Configuración y optimización

Se dispone de una instalación de Qradar con los parámetros básicos y las fuentes de logs, y se tiene que configurar Qradar, las ofensas que generará y cómo responderá a los distintos eventos. Se llamará ofensa a la alerta interna de Qradar que se genera respondiendo a una regla, ya sea creada a mano, predeterminada por el sistema o incluida en una aplicación que se haya descargado.

Las ofensas generadas estarán basadas en casos de uso, que son situaciones definidas que se quieren detectar; un ejemplo de caso de uso es el de "brute force", cuyo escenario es: un usuario intenta acceder a un recurso o sistema de forma repetida sin conseguirlo, de forma repetida, en un periodo breve de tiempo, con lo que podría ocurrir que este usuario fuese un atacante que intenta acceder al recurso o sistema probando distintas contraseñas.

Estos casos de uso pueden ser inventados o basados en algún marco de trabajo, en el caso de este desarrollo se realizó una mezcla de ambos. El marco de trabajo usado fue MITRE[18].

Los casos de uso personalizados y creados para cada cliente no se expondrán en este TFM, con la intención de preservar la privacidad de los datos de los clientes de este proyecto.

En este capítulo, se tratará cómo realizar todos los pasos desde la idea hasta tener un caso de uso funcionando de forma correcta.
Esta tarea se realizó para cada cliente con 59 casos de uso genéricos y 10-20 casos específicos para cada cliente, teniendo 5 clientes el multi-SOC.

3.1 Casos de uso basados en MITRE

El marco de trabajo de MITRE ATT&ACK, presenta una matriz dividida por columnas (adjunta a este TFM como Mitre_matrix.xlsx), en la cual cada columna es una táctica (el porqué, el objetivo técnico) o técnica (el cómo, las acciones a realizar) de ataque que un posible 'actor malicioso' referido como 'atacante'; éste podría usar las técnicas y tácticas para acceder a lugares/sistemas/información/… de forma no deseada por el cliente.

Como ejemplo se expondrá el ataque por táctica de fuerza bruta que se introducía anteriormente, que en el marco de MITRE está definido en inglés como "brute force". Siguiendo la explicación de esta técnica, se repasará la composición de cada una de las páginas de MITRE de las distintas técnicas, ya que el formato es estándar.

La lectura e identificación de esta técnica es la primera tarea a realizar; seleccionando en la matriz de MITRE la misma, nos lleva a la página explicativa[19] que define la táctica y las posibles técnicas que un atacante podría usar englobadas dentro de este método. Primero explica en qué se basa, siendo 'Adversaries' un sinónimo de 'atacante' en este caso. Un ataque por fuerza bruta se usa para obtener la contraseña de un usuario cuando no es conocida.

El siguiente párrafo relaciona esta con otras técnicas, explicando si se usa antes o después, a la vez, o cómo puede influir en un ataque múltiple sencillo.

El tercer párrafo trata sobre la situación en la cual se suele lanzar esta táctica, y en el caso que nos atañe, es cuando el atacante desconoce las contraseñas o tiene una lista que cree que contiene una contraseña correcta. Los riesgos para el atacante también se contemplan,

si realiza esta técnica, generará muchos logs de autenticación fallida, con lo que es posible que sea detectado.

En el caso de existir tácticas o técnicas parecidas, que se pueden confundir o se basan en conceptos parecidos, aparecen en el siguiente párrafo, en el caso contemplado, esta táctica es la de "password spraying", basada en probar pocas contraseñas (las mismas normalmente) en varios usuarios, con tal de dejar menor rastro, como se comentaba anteriormente.

Posteriormente, se encuentra la parte descriptiva de la técnica, donde se muestran los métodos, y cómo un atacante podría realizarla.

El segundo apartado trata sobre mitigaciones, con una tabla en la que se comenta la mitigación y se da un enlace a una explicación más detallada, y una descripción en la siguiente columna, que explica cómo implementar la mitigación en el caso concreto de la táctica con la se trata.

Por último, existen dos apartados, el primero es de ejemplos, que como su propio nombre indica, aporta una serie de ejemplos de distintas herramientas/actores que han implementado dicha táctica o que se ha observado que la usan. El siguiente es el de detección, en el cual no se aportan datos concretos, sino el valor de comunicar cuales son los servicios, logs, o acciones a monitorizar si se quiere tener monitorizada esta táctica; en el caso de "brute force", estos eventos son los de autenticación, de los controladores de dominio específicamente y de todos los sistemas que los tengan en general. El último apartado, es el de referencias.

3.1.1 Ciclo de vida del caso de uso

El ciclo de vida de un caso de uso es el proceso que sigue desde la intención de crearlo hasta que está funcionando en producción, y las fases por las que pasa se explicarán en los siguientes subapartados. Un resumen visual se puede apreciar en la [Fig. 32]

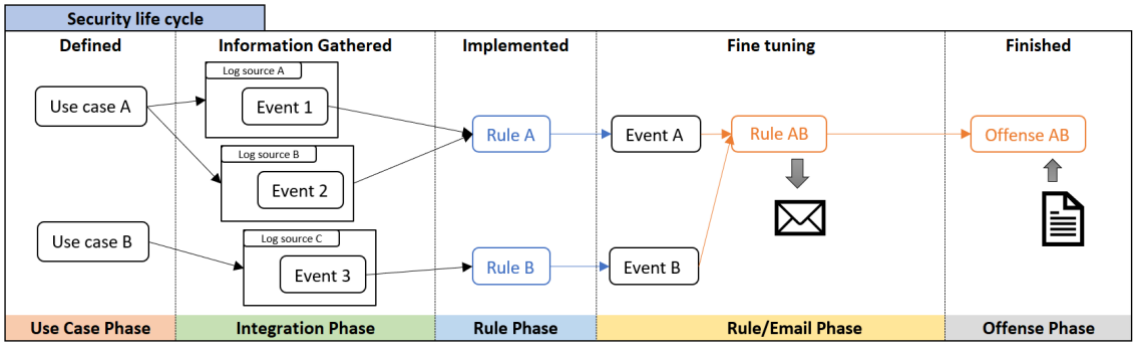


Figura 32. Ciclo de vida de caso de uso

3.1.2 Creación y definición

Un caso de uso es la combinación de la detección técnica específica y la descripción a alto nivel de un comportamiento no deseado, así como las guías o directrices a seguir en caso de detectarlas. Para la parte de detección, se usan alertas e incidentes de seguridad. En cuanto a la parte de actuación, se definen los protocolos y las acciones a realizar, así como las herramientas. Un caso de uso puede dar como resultado una alerta o un incidente (ofensa), y puede tomar como input logs y alertas u ofensas de otros casos de uso.

Con los datos de los cuales se dispone en MITRE, podemos crear un caso de uso genérico para el cliente, sin tener en cuenta si aplicará o no, simplemente como un ejercicio de casos que deberían ser implementados en cualquier sistema que quiera considerarse mínimamente vigilado. Uno de los casos en esta categoría, es el basado en la táctica "brute force", que es la táctica con la que se ilustra la creación del caso de uso completo.

Lo primero es entender cómo funciona la técnica y cómo puede ser detectada en un sistema maduro. Cuando se comprende, se crean una serie de reglas genéricas, sin especificar número concretos, que se determinarán en función del cliente a aplicar. Estas reglas incluyen la prioridad de la alerta, y también diferencian entre alerta e incidente de seguridad. Antes de continuar, es necesario comentar la diferencia entre alerta e incidente (ofensa) de seguridad [Fig. 33], ya que es importante a la hora de la operativa diaria, que será explicada posteriormente en este mismo TFM.

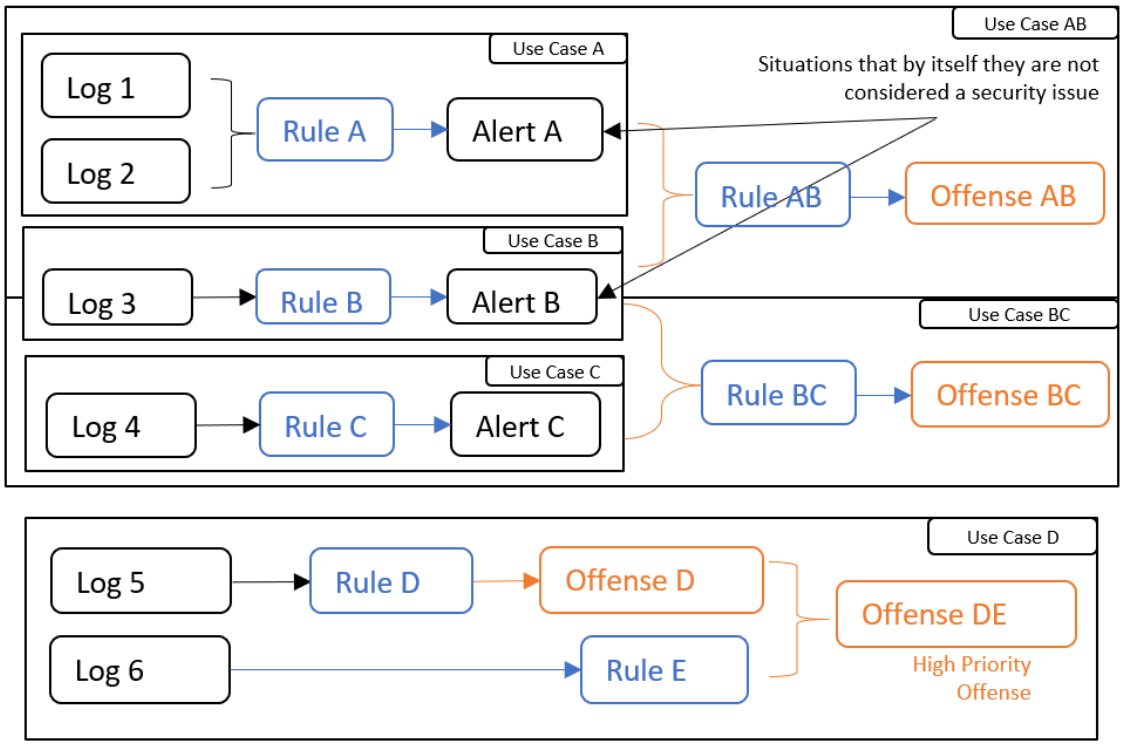


Figura 33. Relaciones logs/reglas/alertas/ofensas

- **Alerta:** una alerta de seguridad es un aviso que salta cuando un comportamiento poco habitual o sospechoso ocurre, detectado por uno o más logs, pero no se considera un riesgo o una amenaza para la seguridad de la red monitorizada. Estas alertas serán investigadas de forma proactiva cuando los incidentes sean resueltos. Una alerta podría transformarse en un incidente de seguridad.
- **Incidente (ofensa):** es un aviso que debe ser investigado dependiendo de la prioridad del mismo. Puede provenir de un conjunto de alertas, así como de una repetición de la misma, o una sucesión específica de alertas, dependiendo de cómo esté planteado el caso de uso.

Para la táctica de "brute force", un log podría ser un "login failure", o un log de identificación fallida. Un único log no significa nada, podría ser un usuario que ha fallado o podría deberse a cualquier otro motivo.

Una posible alerta que influiría en este caso de uso sería "x eventos de login failure detectados en y segundos". Los números dependerán de cada cliente, y se tendrá que estudiar la red antes de fijarlos. Para realizar esta tarea, se estudian las conexiones en un entorno de trabajo real de la red, dejando que Qradar recoja los logs con los cuales se trabajará, pero sin habilitar ninguna regla. Dependiendo de los ciclos de trabajo del cliente, este tiempo puede variar entre uno y tres meses en casos normales, mientras que puede alcanzar el medio año bajo condiciones especiales. Pongamos por caso que el cliente tiene un script personalizado que consulta un recurso compartido cifrado y protegido, y tiene una política de caducidad de la contraseña de los usuarios de un mes. Si el script se ejecuta cada 10 minutos, lo intenta 10 veces una vez cada 0.1 segundo y si no puede se espera otros 10 minutos, la contraseña de acceso que usa el script depende de un proceso que se ejecuta cada semana, entonces tendremos hasta 6 días de logs fallidos por este script, cada 10 minutos 10 fallos concentrados en 1 segundo y seguiría siendo el comportamiento habitual de la compañía.

Suponiendo que el caso expuesto anteriormente es el único comportamiento "poco habitual" que afecta a este cliente, estos números podrían fijarse por ejemplo en x=15, y=1, de tal manera que una alerta saltara si ocurriesen 15 logins fallidos en 1 segundo del mismo usuario.

Teniendo esto, una combinación de z alertas de las explicadas podría dar lugar a un incidente de seguridad de prioridad p, como por ejemplo 30 logins fallidos en 1 seg, es decir, z=dos alertas de las explicadas y prioridad p=3 (menor número es mayor prioridad, siendo 0 la máxima urgencia). Este comportamiento no sería el único que monitorizaría este incidente de seguridad, sino que podría comprobar una cantidad de logins erróneos mayor en un tiempo mayor, haciendo uso de otra alerta. Así mismo, la alerta creada puede servir para hacer saltar otros incidentes de seguridad, como por ejemplo el caso de uso del "password spraying", que es probar una pocas o incluso una contraseña común en varios usuarios. En este caso, el incidente sería con 4 alertas de las mencionadas, pero de usuario distintos en un minuto.

3.1.3 Actuación del caso de uso (documentos guía)

Cuando se tiene la parte del caso de uso en la que se dice cómo detectarlo, falta saber cómo responder ante el mismo. De esto se encargan los documentos guía.

Estos documentos pueden ser un manual por pasos que explica exactamente qué hacer, cómo el flujo de comunicaciones con un diagrama de visio, quién tendrá la responsabilidad de realizar las acciones correspondientes en un momento determinado con una matriz RACI en la que se explica cómo y a quién pedir que se realicen determinadas acciones, y qué herramientas usar y cómo, con enlaces y tutoriales, dependiendo de la madurez del sistema en el cual se implante, una simple guía de las posibles acciones a realizar y las herramientas a usar, sin tanto detalle y dejando más libertad al analista de SOC L1.

Para la creación de estos documentos, se realiza una versión preliminar que el cliente examina y corrige o sugiere si así lo creyese conveniente, y cuando el flujo de trabajo encaja con la compañía cliente, se pone en marcha y se prueba.

3.1.4 Recolección de la información

En este TFM se da por supuesto que los casos de uso que no se puedan implementar con las fuentes disponibles, no se implementan, aunque queden definidos, y esto no tiene porqué ser así. Si el cliente quiere un determinado caso de uso, y por las fuentes disponibles no se puede implantar, se realiza otro proyecto de implantación de la fuente necesaria o de extracción personalizada de información para poder incluirla en Qradar y posteriormente implantar el caso de uso.

3.1.5 Implementación

En lo referido a "caso de uso", haré referencia a la parte de detección, y obviaremos la parte de respuesta.

Se han expuesto los pasos necesarios para integrar las fuentes de las cuales el cliente disponía, se ha comprobado un ejemplo teórico de un caso de uso, pero falta introducirlo en el Qradar para que empiecse a correlar eventos.

Qradar dispone de una pestaña llamada "offenses" (ofensa en castellano), Equivalente a "incidente de seguridad". En esta pestaña, podemos ver las ofensas que tenemos abiertas, a quién están asignadas y un pequeño resumen para cada una de ellas. En la [Fig. 34], se aprecia que dentro de esta pestaña existe un enlace o botón en el cual está escrito 'rules'. Accedemos a "Rules" para comenzar a crear nuestro caso de uso.

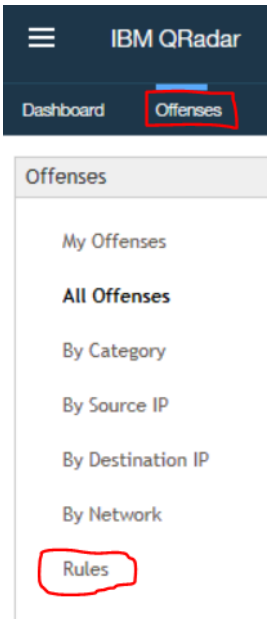


Figura 34. Acceso a configuración de reglas

Vemos las reglas ya implementadas, y tenemos un desplegable que nos deja elegir entre reglas (rules) y BB (Building Block). Vamos a detallar qué es cada uno de estos conceptos y se explicara cómo crear una regla paso a paso.

3.1.5.1 Building Blocks (BB)

Los building blocks con como piezas de construcción (LEGOs) o variables, ya que son "trozos de reglas" que se pueden usar en otras reglas. Las ventajas de usar estos BB son las mismas que las de usar variables en los lenguajes de programación, y se crean

prácticamente igual que las propias reglas en sí, sólo que en vez de realizar acciones, sirven para implementar otras reglas.

Podemos tener un caso especial en el que los servidores y ciertas máquinas especiales hacen peticiones continuas de login y suelen fallar algunas veces, con lo que podríamos detectarlos como "brute force" cuando no lo son. Una manera de evitar estos falsos positivos es crear un BB que contenga las máquinas especiales, y por otro lado crear un "reference set" que se autogestione con los servidores encontrados.

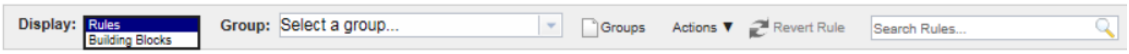


Figura 35. Selector de reglas o building blocks

Para crear un building block, hay que hacer pulsar sobre 'Actions' y en el menú emergente sobre 'New evento rule', aunque parezca al principio contraintuitivo. En este punto nos aparece un asistente de creación [Fig. 36], que es exactamente el mismo que el de las reglas. Elegimos 'Events', y presionamos siguiente.

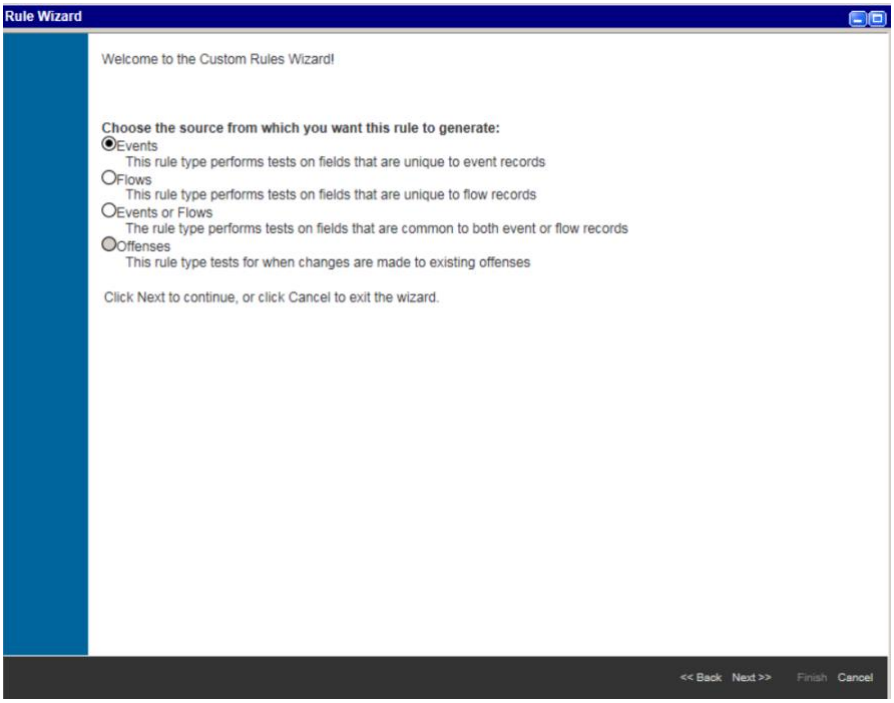


Figura 36. Acceso a configuración de reglas

En el siguiente paso debemos elegir una serie de filtros y condiciones editables (la parte en negrita del filtro es editable) con las cuales crearemos nuestro BB, y cuando lo tengamos completado, para hacer que sea un BB y no una regla, de debe pulsar sobre el botón "Export as Building Block" visible en la [Fig. 37], con lo que aparecerán otra ventana distinta una entrada de formulario en el cual insertar el nombre a nuestro BB [Fig. 38].

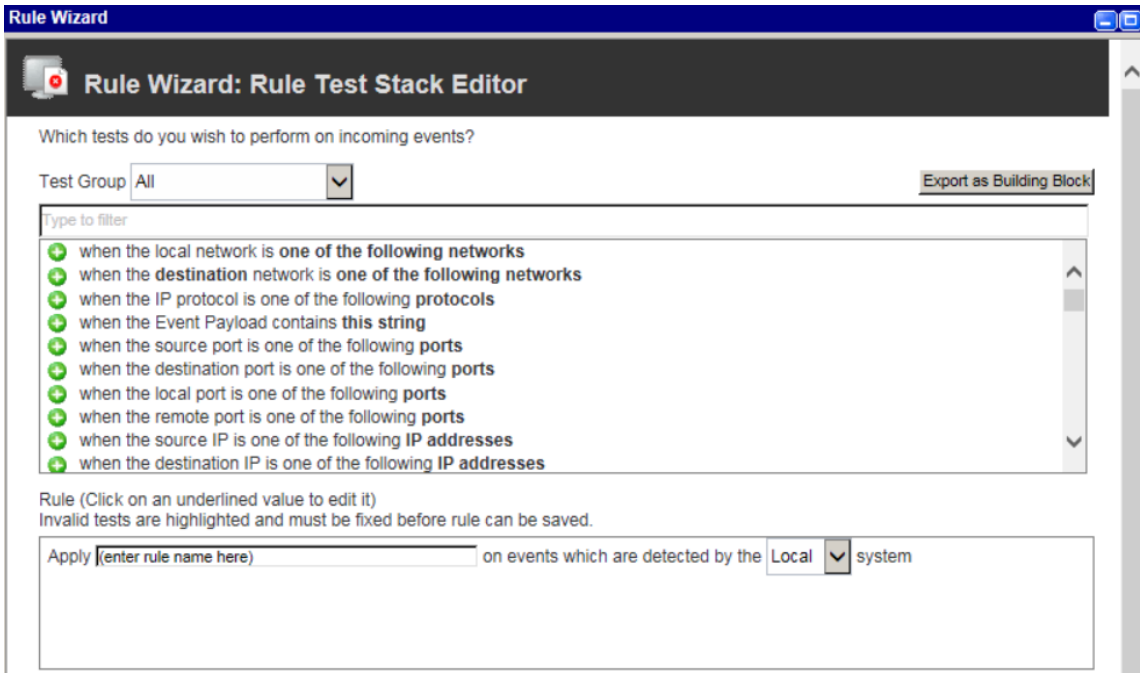


Figura 37. Creación de BB

Qradar, aunque clasifica los BB en un apartado distinto de las reglas, los nombra de una manera especial, ya que muchas aplicaciones los usan, y crean sus BB que quedan visibles para el usuario.

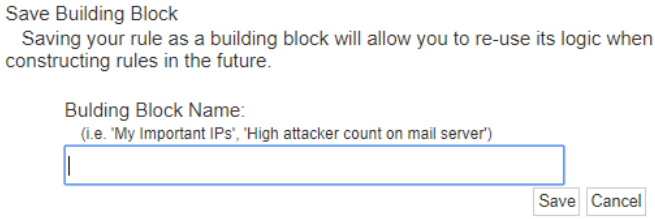
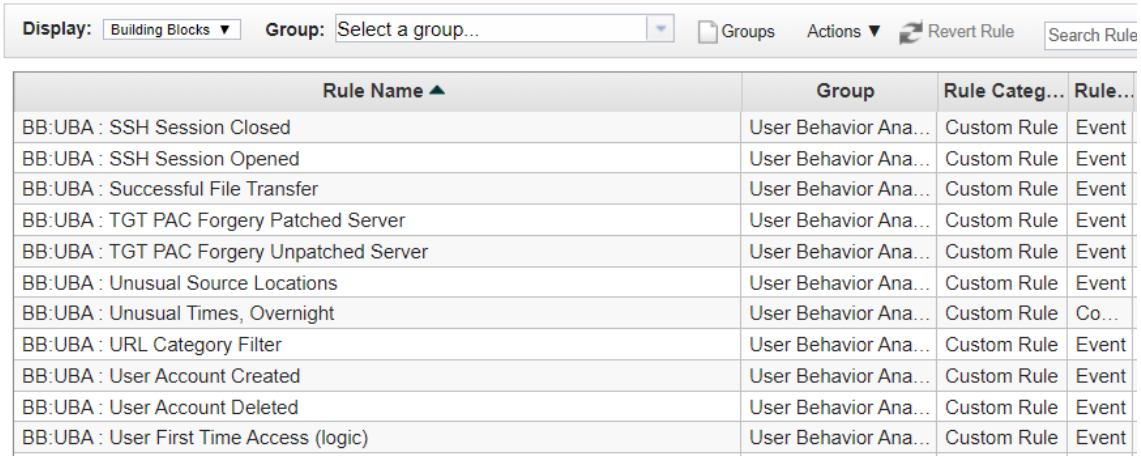


Figura 38. Guardado de BB

Esta forma de nombrarlos especial es añadir ‘BB:’ al principio del nombre, y en caso de que sea un BB especificado por el usuario, escribirlo de forma explícita. Para ejemplificarlo, supongamos que se ha creado un BB que se quiere llamar ‘Alexis_test’. Este BB en realidad debería tomar el nombre de ‘BB: User defined: Alexis_test’.

Se pueden ver los bloques que crean las aplicaciones en la ventana donde se almacena el bloque que acabamos de crear, en la [Fig. 39] se muestran algunos de los BB de la aplicación UBA (User Behavioural Analytics).



Rule Name	Group	Rule Categ...	Rule...
BB:UBA : SSH Session Closed	User Behavior Ana...	Custom Rule	Event
BB:UBA : SSH Session Opened	User Behavior Ana...	Custom Rule	Event
BB:UBA : Successful File Transfer	User Behavior Ana...	Custom Rule	Event
BB:UBA : TGT PAC Forgery Patched Server	User Behavior Ana...	Custom Rule	Event
BB:UBA : TGT PAC Forgery Unpatched Server	User Behavior Ana...	Custom Rule	Event
BB:UBA : Unusual Source Locations	User Behavior Ana...	Custom Rule	Event
BB:UBA : Unusual Times, Overnight	User Behavior Ana...	Custom Rule	Co...
BB:UBA : URL Category Filter	User Behavior Ana...	Custom Rule	Event
BB:UBA : User Account Created	User Behavior Ana...	Custom Rule	Event
BB:UBA : User Account Deleted	User Behavior Ana...	Custom Rule	Event
BB:UBA : User First Time Access (logic)	User Behavior Ana...	Custom Rule	Event

Figura 39. Guardado de BB

3.1.5.2 Reglas

Las reglas son una serie de filtros y condiciones que en caso de cumplirse desatan ciertas acciones en el propio SIEM.

Son una de las partes principales de la herramienta en sí, gracias a ellas se pueden detectar distintas amenazas y monitorizar los casos de uso como se mostraba anteriormente en la [Fig. 33], siendo una parte relevante de la 'inteligencia' de este SIEM. Muchas aplicaciones generan reglas propias, como por ejemplo el UBA comentado en la sección anterior.

Las reglas son el método principal, aunque no el único con el cual pueden saltar ofensas en Qradar, y pueden ser de varios tipos[20]:

- Basadas en eventos (logs cuando se parsean y mapean)
- Basadas en 'flows' (Flujo de datos en la red)
- Basadas en eventos y 'flows'
- Basadas en ofensas
- Detección de Anomalías
- Detección comportamental
- Detección de límites

Las últimas tres son especiales y se realizan a partir de las búsquedas. Dichas búsquedas deben cumplir ciertos requisitos para poder derivar en una de estas reglas [Fig. 40].

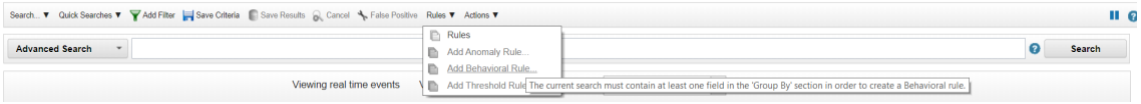


Figura 40. Opciones para reglas especiales

La explicación del funcionamiento y la creación de cada una de ellas es materia suficiente para otro TFM, con lo cual se centrarán los esfuerzos únicamente en las reglas basadas en eventos. Con estas reglas también se pueden realizar los casos de uso que como input tienen eventos y ofensas, ya que cuando se genera una ofensa, también se generará un evento informando de esta situación, con lo que se podrán controlar la apertura de ofensas mediante la creación de estos eventos internos de Qradar.

Siguiendo con el ejemplo de “brute force”, se creaó la regla especificada anteriormente usando los filtros adecuados, de tal manera que se tuviesen en cuenta los logs de Windows AD, los de Kerberos, los de Linux referidos a login, y tamben se creó un BB llamado ‘BB: User definded: failed login events’ para usarlo posteriormente en la regla, de tal manera que si se detectan la cantidad de estos logs especificada y en un tiempo menor del especificado, salte esta regla.

Se pasa a la siguiente pestaña pulsando sobre el botón “Next”, y aparece un menú para seleccionar qué se desea que ocurra cuando la regla salta. Este menú puede llegar a ser confuso, pero en esencia dispone de cuatro apartados diferenciados que serán explicados con detalle a continuación:

Rule action: Es la acción que se lleva a cabo la primera vez que salta la regla. Se puede escoger que los eventos detectados formen parte de la ofensa, de tal manera que se marquen con una etiqueta especial. Si se selecciona esta opción, la ofensa debe indexarse por algún campo. Esto significa que la parte importante de la ofensa es el campo por el cual se indexe. En nuestro caso, sin lugar a dudas este campo debe ser el nombre de usuario. [Fig. 41].

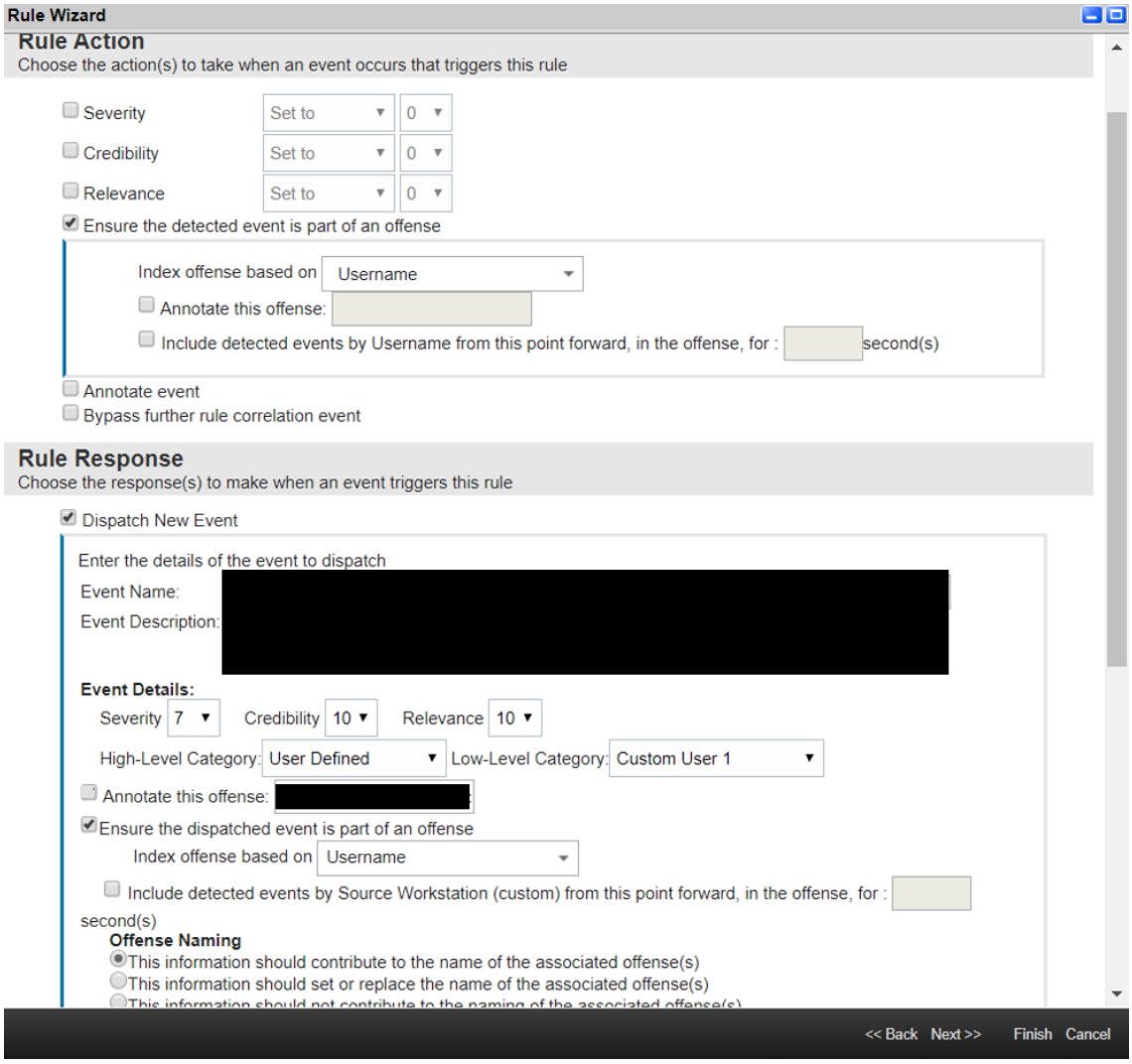


Figura 41. Respuesta y acciones de una regla

Una alternativa es anotar la ofensa, aunque no suele realizarse.
Existe un checkbox es que permite incluir los eventos que tengan el mismo campo que el indexado durante los segundos que se indiquen. En el caso de "brute force", este campo es muy interesante marcarlo, ya que las siguientes acciones del usuario quedan registradas, con lo cual se podría conocer si después de la detección de este comportamiento ha cesado la actividad maliciosa, o el atacante ha intentado realizar otra acción no detectada en otra ofensa, o ha realizado un login y ha entrado…

- **Rule response:** Con este apartado se ajusta cómo responderá la ofensa cada vez que sea detectada, no siendo este comportamiento coincidentecon la primera vez que se cumplen las condiciones impuestas por la regla. En el caso de detectar 500 eventos, y la regla detectara de 100 en 100, se obtendría una ofensa con cinco eventos de detección.
  El campo que crea un evento cuya información recoge la regla que ha detectado un positivo es "Dispatch New Event". Dentro de este checkbox, se contemplan varias opciones, muchas parecidas a la del apartado anterior, y una especial, que se refiere al nombre de la ofensa. Si se marca alguna de ellas, el nombre de la ofensa que hemos puesto al principio se vería modificado de distinta manera, aportando nueva información o incluso reemplazando el nombre original. Esto suele usarse si la ofensa es genérica, ya que aporta mas información; en el caso de que sea específica, lo habitual es que no se modifique el nombre de la alerta.
  Existen también una multitud de acciones que el SIEM puede realizar de forma autónoma [Fig. 42]:
  - **Mandar un email:** se puede enviar un email con plantillas editables
  - **Mandar a un Syslog:** enviar esta información a otro server syslog
  - Reenviar
  - **Notificar:** aparecerá una notificación web/push si la web de Qradar se encuentra abierta en un navegador
  - **Añadir/eliminar de "reference set":** Los "reference set" son listas que se usan para comprobar si un valor está en ellas o no. Pueden ser caracteres alfanuméricos (sesibles a mayúsculas o no), direcciones IPs… Añadir o eliminar sirve para hacer dicha colección dinámica; un ejemplo sencillo serían los usuarios que se dan de baja en un servicio, y tardan en replicarse en el resto. Este BB se usa para evitar que esta situación genere falsos positivos de la regla de "brute force".
  - **Añadir/eliminar de "reference data":** Como su nombre indica, son una colección de datos de referencia, que se diferencian de los "reference set" ya que no son una lista, sino que pueden ser mapas, mapas de listas, mapas de mapas o tablas. La complejidad de las reglas realizadas con estos elementos es mayor y se escapa de los conocimientos que se pretenden transmitir en el presente TFM, con lo que no serán explicadas con más detalle.
  - **Lanzar un escaneo:** Qradar dispone de un escáner de vulnerabilidades (licencia aparte) que puede ser usado cuando aparece una ofensa, con uno de los perfiles llamado escáner bajo demanda. Con este escáner,

cuando aparezca la ofensa se escaneará el equipo afectado. Suele usarse para casos de uso en los cuales se intente explotar alguna vulnerabilidad, para comprobar si, en el momento del ataque el equipo era vulnerable.

- **Realizar acciones personalizadas:** Se puede recurrir a una acción personalizada, que son scripts programados para funcionar en el Red Hat que sobre el que corre Qradar. Con esta opción se obtendría la máxima personalización de respuesta.

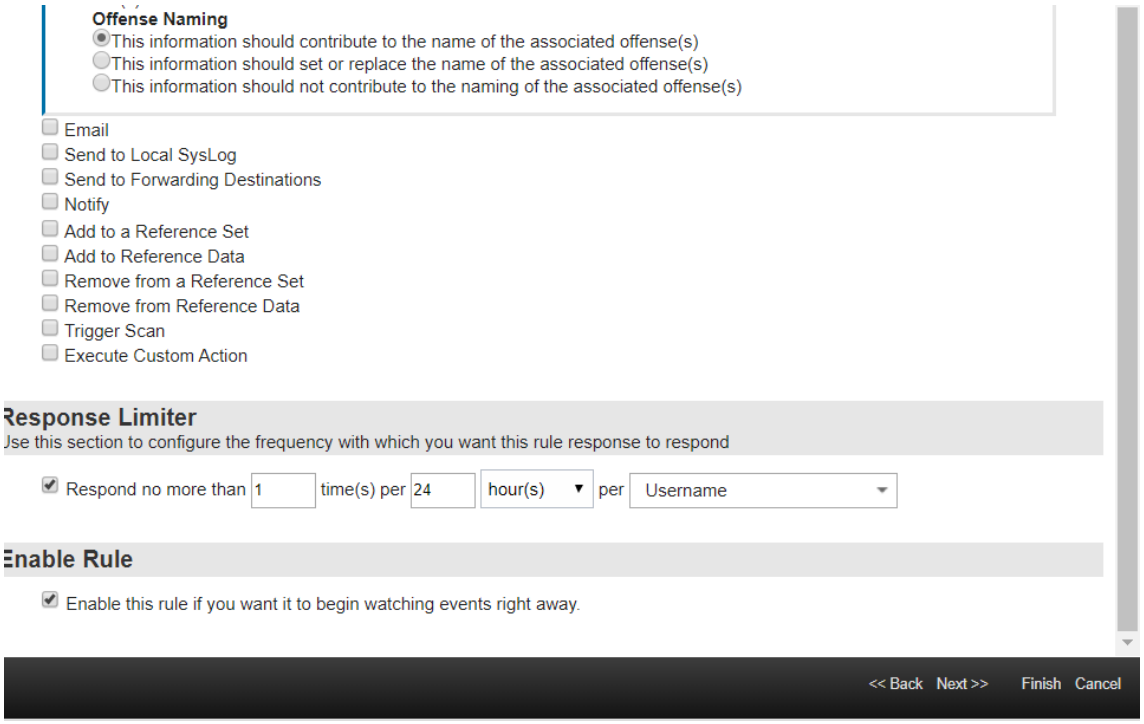


Figura 42. Opciones de respuesta

- **Response Limiter:** Este apartado es el que se encarga de limitar la cantidad de veces que la regla hace que aparezca una nueva ofensa cuando detecta un positivo. Esto es útil en el caso en el cual un atacante intenta una táctica con una técnica muy agresiva, o por algún motivo se descontrola la red y empiezan a aparecer ofensas de forma rápida. Este apartado limita las veces que una regla puede ejecutar acciones de respuesta del apartado anterior, limitado por tiempo y un índice que se elija.
  En la [Fig. 42], se limita a una vez al día por cada usuario, ya que en cuanto se cumplan los requisitos impuestos por esta regla, se comenzará con la investigación a investigarla, y no es necesario que aparezca en múltiples ocasiones.
- **Enable rule:** Por último, tenemos la opción de habilitar o no la regla, que habilitaremos para hacer que la regla empiece a funcionar, o lo deshabilitaremos para guardar los ajustes exclusivamente.

Cuando se escogen los ajustes, se pulsa sobre “finish” para acabar o “Next” para ver un resumen, y posteriormente “finish” para acabar.

3.1.6 Afinación del caso de uso

La regla está diseñada e implementada en Qradar, pero no es el final de la implementación completa, como se puede ver en la [Fig 32], resta una parte importantes del proceso: Afinar el caso de uso y la regla para el cliente (‘tuning’ en adelante). Qradar dispone de una app que debería incorporarse en la propia instalación base (según mi punto de vista), llamada Tuning, y que facilita enormemente las tareas que se comentarán en este apartado.

El tuning se basa en adaptar la regla genérica a la red la cual será monitorizada, ya que puede ocurrir que una regla genérica no funcione como se esperaba en el entorno del cliente por diversos motivos.

Se desactiva la respuesta de la regla por ofensa si estaba activada, y se activa la respuesta por email. Esta acción se realiza por dos motivos principales: Si la regla se descontrola (que es habitual en redes complejas y con comportamientos anormales), no se genera ruido en el SIEM ni se molesta al SOC de nivel 1 (los encargados de las investigar las ofensas); por otra parte, se diferencian los casos de uso en estado de ‘tuning’ y los terminados, informando al cliente de cuales se encuentran en cada fase, ya que éste tiene la potestad de observar Qradar cuando crea conveniente (y las ofensas suelen replicarse en una herramienta de ticketing del cliente, como veremos más adelante).

Al llegan por email al buzón del equipo, se realizan reglas y carpetas específicas para cada una de las reglas, y se prueban de cinco en cinco cada semana, habilitando una por día, y controlando la cantidad de alertas que se crean. Estas alertas son investigadas y se buscan comportamientos habituales de la red no contemplados previamente en la creación de la regla, con la intención de incluirlos en la misma y evitar futuros falsos positivos.

Siguiendo con el ejemplo de “brute force”, después de habilitar la regla, a la semana saltaba entre 5 y 7 veces, con lo que se investigaron estos casos para verificar qué es lo que ocurría. Se encontraron varias casuísticas que se resolvieron y se excepcionaron en la regla, ya que formaban parte del comportamiento habitual de la red, y a continuación se explicarán dos de ellas, evitando desglosar información clasificada sobre la empresa.

- **Usuarios con la contraseña caducada:** se encontró en un cliente una casuística muy marcada: ocurría que la política de renovación de la contraseña se encontraba limitada a pocos días, y los usuarios cambiaban de contraseña de forma muy habitual. Al ser un cliente con muchos usuarios, los cambios de contraseña eran frecuentes en términos globales. En sí, este hecho no supone ningún problema, pero un programa de uso empresarial hacia intentaba acceder con el nombre y contraseña del usuario al iniciar el sistema y se quedaba con la contraseña del inicio, de tal modo que el SOC observaba ‘logins’ (usuarios verificando su identidad) fallidos de este usuario continuamente cuando cambiaba la contraseña, mientras que el usuario sólo percibía que “de vez en cuando no se actualizan los gráficos del programa”. Se comprobó que el programa realizaba consultas cada cierto tiempo y si no las conseguía de forma exitosa, reintentaba hasta un total de 20 veces, con la contraseña del último inicio de sesión, y no la nueva que se había cambiado.
  Para excluir este comportamiento se creó un “reference set” llamado ‘Users with changed passwords’, de tal manera que caducara cada 6 horas cualquier nombre

de usuario que se introdujese en él, y una regla que introducía a los nombres de usuario cuando se detectaba un cambio de contraseña. En la regla original se excluyeron los usuarios que pertenecían a este "reference set".

- **Migraciones:** Las migraciones de la suite de office antiguas (las que en el nombre tienen un año, como 'office 2016') a office365, dependiendo de cómo se realicen pueden producir inconsistencias de forma temporal en los usuarios de directorio activo de Windows.
  Estas inconsistencias se deben a los nuevos nombres que usa oficce365 para que sus usuarios sean capaces de usar este office online y de manera global, añadiendo una cadena de texto con '@office365' o similar al usuario, dando como resultado que algunos usuarios que tiene las versiones antiguas y la de office365 aparezcan como duplicados.
  Estas inconsistencias pueden derivar en que la regla detecte este comportamiento sin ser un comportamiento malicioso.
  Para evitarlo se usó la misma técnica de reference set explicada en la anterior casuística.

Cuando se supera esta fase, la regla produce una ofensa y se desactiva el aviso por email.

Ahora se dispone de una ofensa adaptada al cliente.

3.1.6.1 Plan de pruebas

A pesar de que esta fase no se encuentra enmarcada dentro de 'tuning' en el ciclo de vida, suele realizarse a la vez o justo después de la misma, en cuyo caso es considerada parte de la misma.

En esta fase se prueba que la regla está bien definida y se pone a prueba . Las ofensas que se hayan creado en la fase anterior de tuning se han podido ajustar, pero hay que comprobar que los ajustes no han afectado a su eficacia, y las que no hubiesen saltado por ser situaciones complejas o extremas, se debe comprobar que saltarían en caso de ocurrir, debido a su naturaleza crítica.

Para ello se realizaron otros proyectos a parte: proyectos de 'Resilience', que son aquellos en los que se realizan las acciones que se esperan detectar de forma específica y proyectos de 'pentest' (PENetration TEST, test de intrusión en castellano), que son aquellos en los que un hacker ético sin intenciones maliciosas comprueba qué podría ser vulnerado, fingiendo ser un actor malicioso. Los 'pentest', pueden ser realizados por la misma empresa que realiza el proyecto de SOC, aunque comúnmente se contrata a otra empresa, para comprobar de forma independiente que la primera cumple con lo establecido. En estos casos, en la jerga de seguridad, el que comprueba la vulnerabilidad de la red es llamado 'red team', y a los que monitorizan y defienden se les llama 'blue team'; cualquier referencia a estos términos indica normalmente que es una prueba, y si no se tiene constancia de ella es necesario contactar con el responsable de seguridad de cliente de forma inmediata.

En el plan de pruebas se incluyen ambos proyectos, y dependiendo del cliente aceptaban o declinaban esta parte del proyecto.

Cuando el caso de uso llega a esta fase, se mantiene en un 'fine-tuning' (adaptación a pequeños cambios) constante en función de los cambios en la red y las mejoras añadidas a Qradar.

3.1.7 Seguimiento del ciclo de vida

Se ha explicado el proceso realizado por el equipo para la realización del proyecto, sin embargo, no se ha reflejado la forma en la que se reportaban los distintos estados de los casos de seguridad.

Para cada cliente se creó un Excel con los distintos casos, tanto los genéricos como los personalizados, y mediante este Excel compartido a cada uno de los clientes, se realizaba el seguimiento de la madurez de los casos de uso. La madurez se entiende como la etapa del ciclo de vida en la cual se encuentra el caso.

En el Excel comentado existen varias pestañas, una con el inventario de fuentes, otra con el inventario de casos de uso, y otra con las reglas necesarias para esos casos de uso. Por último, existen los dashboards (cuadros de mando) a modo de resumen, tal y como puede verse en la [Fig. 43] y la [Fig. 44]

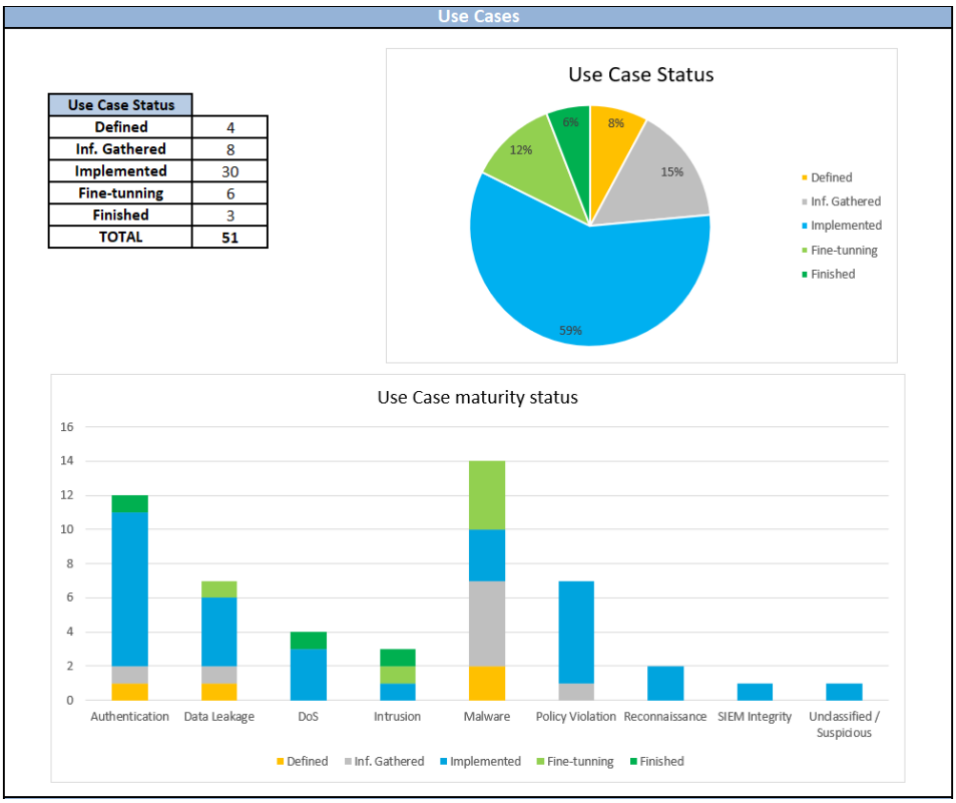


Figura 43. Dashboard casos de uso 1

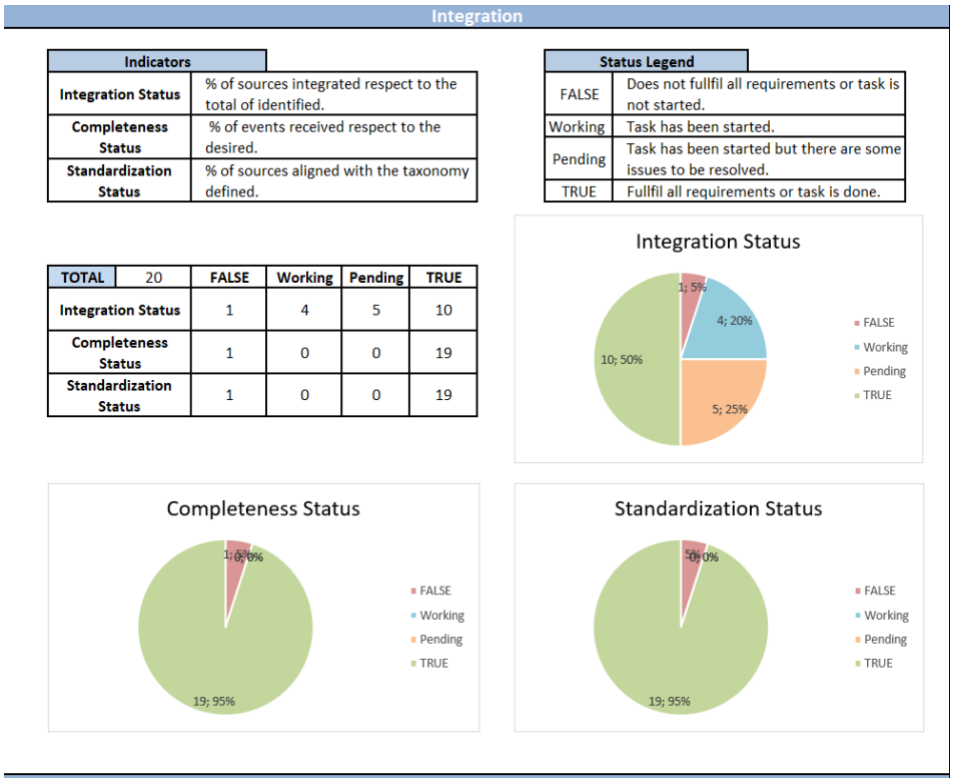


Figura 44. Dashboard casos de uso 2

3.2 Implementación con herramienta de ticketing

Una integración con la herramienta de ticketing del cliente se realiza como parte del proyecto, debido a que es la plataforma donde el cliente realiza peticiones de forma genérica, y donde suelen pedir que el SOC las realice. Por esto mismo, este apartado forma parte del proyecto general de SOC, y no es un proyecto a parte, aunque no es estrictamente necesario para que el SOC empiece a operar.

Existen varias maneras, dependiendo de la integración a la que quiera llegar el cliente:

- **Email:** Integración mínima; se basa en abrir un ticket por cada ofensa generada, y continuar la investigación en el propio ticket, abriendo un ticket relacionado (también llamado ticket hijo) en caso de necesitar alguna acción por parte del cliente. Este tipo de integración tiene la ventaja de ser sencilla y asequible económicamente, y los inconvenientes de ser genérica y lenta en lo referido a la comunicación entre equipos. Si se necesita un bloqueo, hay que pedirlo mediante un ticket, y éste sigue todas las vías preestablecidas que puede seguir cualquier otra petición, significando esto que se realizará con alta probabilidad con los tiempos que se manejen en la herramienta de forma habitual, siendo estos mayores de los deseables en acciones críticas de seguridad.
- **Herramienta de seguridad:** Esta integración es posible si en vez de una herramienta genérica de ticketing, el cliente posee una herramienta especializada en seguridad, tal como The hive[21] (código abierto) o Resilient (IBM)[22].
  En caso de tener este tipo de herramientas de orquestación, se puede integrar con ellas, y ellas a su vez con los distintos sistemas de seguridad, de forma que la investigación queda en estas herramientas y además se pueden realizar acciones

de forma directa a través de las mismas mediante API. Esta opción es la recomendada para integración, aunque es mas costosa en cuanto a esfuerzos y de forma económica.

3.3 Extras:

3.3.1 Integración y segmentación de redes

Qradar es un almacenador/correlador de datos, y usa todos los datos proporcionados para mejorar los avisos sobre posibles amenazas, avisos denominados ofensas. Uno de los datos resulta clave en cualquier compañía con una red IT considerable, es cómo está dividida la red, para qué se usa cada segmento, si hay una red de servidores, una de invitados, un segmento de red secreta… Puede ocurrir que, dependiendo de la red, esta sea mas sensible a la carga o la información que corre por la misma sea de un nivel de confidencialidad mayor.

Múltiples casuísticas como las expuestas anteriormente pueden acaecer en cualquier empresa, y muchas otras que no se contemplan. Todas ellas influyen en el nivel y tipo de ofensa que debe crearse, haciendo incluso que no se cree ninguna si no fuese necesario. Para ejemplificarlo, se continuará con el ejemplo de los apartados anteriores, el caso de uso de "brute force".

La prioridad de la alerta podría ir relacionada con la red en la cual esté ocurriendo: si ocurre en una red de invitados, no supone ningún problema, ya que por defecto no está unida a la red interna de ningún cliente, sin embargo, si ocurre en una parte de red secreta, a parte del posible brute force realizado se detecta que un atacante ha llegado a este segmento de red. Por ello un añadido importante es la segmentación de redes, que comprando un módulo de Qradar y si se configura un usuario con permiso de lectura en la electrónica de red, se puede realizar de forma automática; en caso de no estar disponible este módulo o el usuario necesario, se tiene que realizar y mantener la segmentación a mano.

Para realizar esta tarea de forma manual, existe una aplicación de Qradar[23] que permite incluir redes desde un .csv, permitiendo realizar introducción de redes en masa; otra opción es introducirla manualmente. Para llevar esta tarea a cabo primero se pide el mapa de red, y se accede a la pestaña "Admin", como se ha realizado anteriormente, se pulsa sobre "Network Hierarchy" [Fig. 45]. Al hacerlo aparecen las redes que estén configuradas, en caso de tener alguna, y con el botón "Add", es posible añadir una red de forma manual. Pulsando este botón aparece el formulario reflejado en la [Fig. 46]

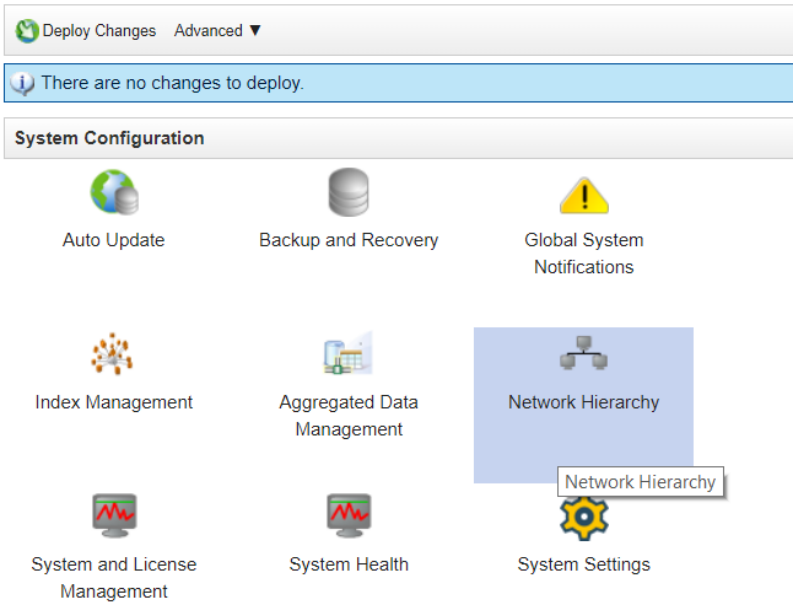


Figura 45. Acceso a Network Hierarchy

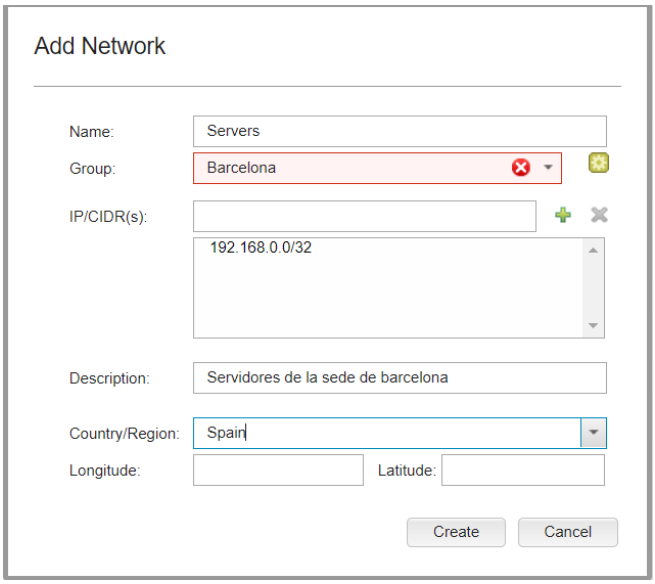


Figura 46. Añadir nueva red

Esta información queda reflejada en varios apartados de Qradar, como en el resumen de la ofensa y puede ser usada en la creación de reglas. Es un complemento a tener en cuenta, sobre todo si la red es compleja y está segmentada.

3.3.2 APP: User behavior Analytics (UBA)

Es otra de las aplicaciones que desde mi punto de vista debería incluirse por defecto en Qradar. En la web oficial[24] pueden verse las ventajas que aporta, siendo estas muy considerables, sobre todo teniendo en cuenta que la propia aplicación es gratuita (a excepción del módulo Watson). Permite obtener información de los usuarios a un nivel mucho mas detallado que los propios logs, ya que bien configurada aporta datos extraídos del Active Directory de Microsoft mediante el protocolo LDAP.

También es capaz de habilitar un módulo de machine learning (dependiente de los recursos de la máquina en la cual se ejecute) que aprende de la red y avisa de comportamientos extraños. Se pueden buscar acciones de los usuarios independientemente de la fuente que aporte los logs, y la interfaz es intuitiva y altamente navegable.

Dispone de una gran cantidad de casos de uso y reglas habilitables de forma sencilla y que permiten ahorrar gran cantidad de trabajo., en concreto 230 reglas.

Otra capacidad importante es añadir propiedades nuevas al análisis sintáctico de ciertas fuentes.

Capítulo 4. Funcionamiento del SOC y métricas

4.1 Estructura y tareas del SOC

A parte de los proyectos de mejora, los proyectos de integración de fuentes y el propio proyecto explicado en este TFM, existe una operativa diaria en el SOC, y es el análisis e investigación de las ofensas producidas.

El SOC se estructura en tres niveles principales, en el que cada uno de ellos recibe una función principal:

- **L1:** El nivel uno del SOC son los primeros que ven e investigan las ofensas, siguen los procedimientos establecidos en los casos de usos, y realizan acciones que se detallan en ellos. Si la ofensa se repite, no está contemplada la acción a realizar, o no saben cómo seguir la investigación, avisan al nivel 2 de SOC. Se encargan de ofensas con prioridad 3 y 2, y algunas con prioridad 1 en el caso de que estén altamente procedimentadas (menor número equivale a mayor prioridad, siendo prioridad 0 la máxima posible).
- **L2:** El nivel dos de SOC tiene experiencia suficiente como para resolver con éxito un caso no documentado, con capacidades de decisión básicas y medias, pudiendo pedir acciones que no se encuentren procedimentadas tras una investigación y una explicación de las mismas. Se encargan de las ofensas con prioridad 1 y de comunicar las de prioridad 0, así como de empezar la investigación de forma urgente. Como tarea extra, también se encargan de afinar las reglas en producción, en caso de que sea demasiada carga para el SOC L1. Si llegara el caso en el que el nivel dos se queda sin recursos o tiene que realizar alguna acción que pudiera afectar al cliente de manera negativa, se encargaría el nivel tres.
- **L3:** El nivel tres del SOC son los expertos en seguridad. Ellos son los que se encargan de las ofensas de prioridad 0, es decir, aparecen en caso de necesidad apremiante: ataques reales de expertos, situaciones de infección de rápida expansión (como un ransomware)… y pueden tomar decisiones de seguridad que afecten al cliente. A pesar de que el cliente puede realizar o no las acciones que plantee un componente de SOC L3, se recomienda encarecidamente que las lleve a cabo.

4.2 Métricas nativas de Qradar

Las métricas que buscamos en este capítulo no están relacionadas con los casos de uso, sino con el trabajo diario del SOC.

4.2.1 Dashboards

Qradar posee una serie de cuadros de mando por defecto, que se van incrementando a medida que se instalan aplicaciones, algunos de estos cuadros de mando pueden ser útiles, si se personalizan. No se usa esta herramienta para reportar datos.

4.2.2 APP: Pulse

Es una aplicación específica de Qradar que mejora los cuadros de mando anteriores y permite editarlos de forma personalizada. Incluye varios tipos de gráficos, por ejemplo un mapa 3D.

En la página web oficial en X-force disponen de ejemplos. Esta aplicación se valoró, pero no permitía una personalización total ni el acceso desde otro punto que no fuese Qradar, con lo que se descartó, ya que los informes no sólo llegarían a clientes con un perfil técnico.

4.3 Implementación con Qlik

Se ofrece un apartado de informes mensuales, con la intención de que el cliente tenga los datos verídicos respecto a las ofensas que se producen, quién las resuelve, cuanto tiempo tardan en ser respondidas, la efectividad del servicio…

Para ello se realizan una serie de cuadros de mando distintos, elegiendo el programa Qlik Sense, ya que es un programa altamente interactivo, accesible y editable, y los clientes disponían del mismo y estaban acostumbrados a él.

Aprovechando la API de Qradar (que es una de las mejor documentadas del mercado) y Qlik, se realizaron cuadros de mando [Fig. 46] con información mensual sobre las ofensas. Estos contenían la siguiente información:

Figura 47. Diferentes dashboards

- Cuadro de mandos con gráficos sobre ofensas [Fig. 47]
  - Ofensas totales
  - Tiempo medio de resolución
  - Media de ofensas al día (con y sin fin de semana)
  - Gráficos sobre cantidad de abiertas y cerradas
  - Gráfica con el motivo de cierre
  - Cantidad de ofensas abiertas al día
  - Ofensas por grupo
- Ofensas resueltas por usuario
  - Resueltas al día de media
  - Resueltas en total
  - Resuletas en porcentaje
- Ofensas por usuario y tipo
  - Tabla pivote con usuarios y tipos, con información sobre la cantidad
  - Gráfico de árbol de tipo e integrante del SOC con medidas porcentuales
  - Gráfico de barras con cantidades absolutas
- Tiempos
  - Tiempos medios de resolución

  - Tiempos medios de reacción
  - Tiempo por ofensa
  - Tiempo por tipo de ofensa
- Datos en crudo de las ofensas

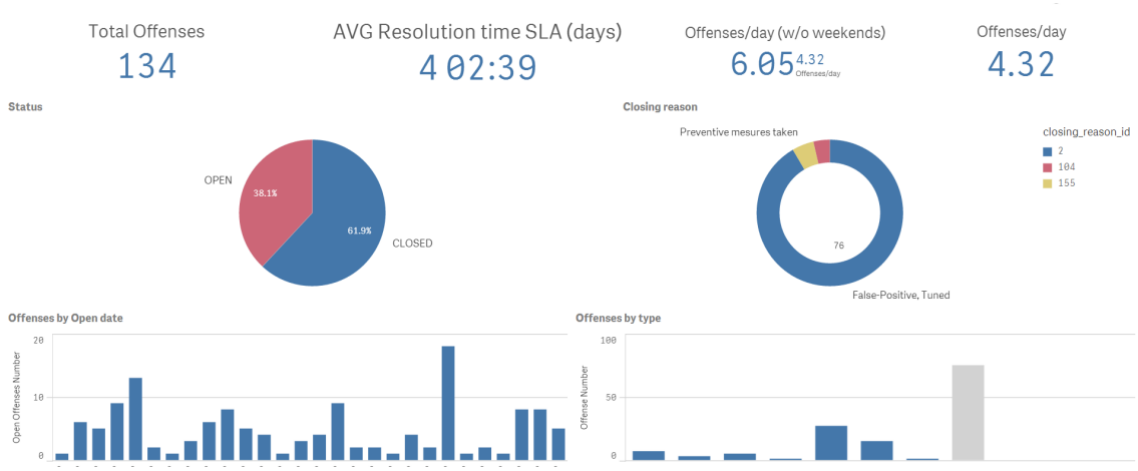


Figura 48. Cuadro de mandos 'Offenses Graphics'

En la [Fig. 48] se aprecia uno de los cuadros de mando que se realizaron. De derecha a izquierda y de arriba abajo, encontramos un indicador numérico sobre las ofensas totales que se han creado en un mes, el tiempo medio de resolución de dichas ofensas, la media de ofensas al día (contando los fines de semana, y sin contarlos), un gráfico de tarta sobre cantidad de ofensas abiertas y cerradas, una gráfica de donut con el motivo de cierre de cada ofensa, un gráfico de barras temporal con el eje x los días del mes, y en el eje y la cantidad de ofensas abiertas al día, y por último, las ofensas clasificadas por grupo.

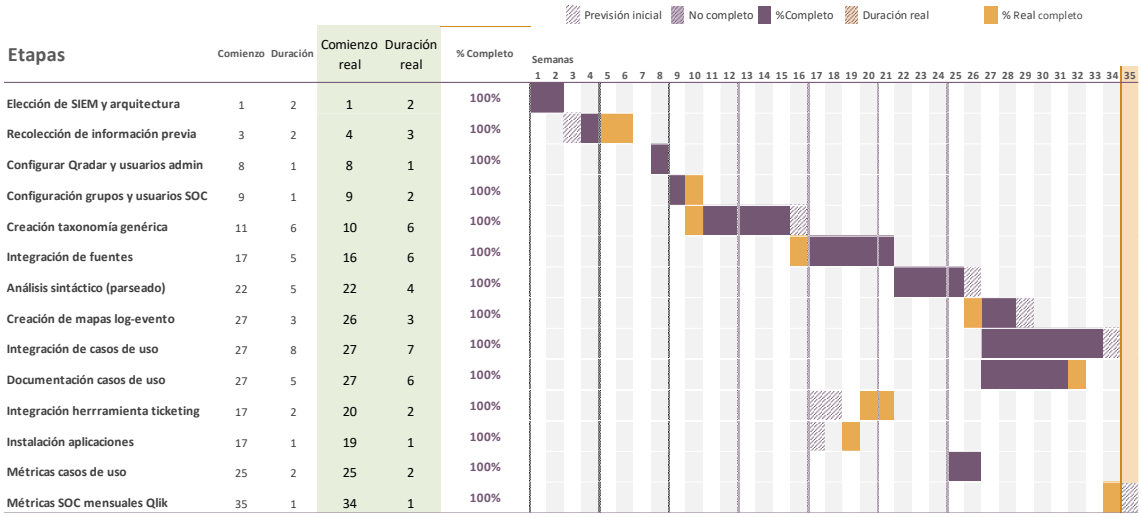
La imagen de la [Fig. 48] está tomada en la fase de 'tuning', y como se puede comprobar, la mayor parte de las ofensas son cerradas por falsos positivos que estaban siendo afinados.

Capítulo 5. Planificación temporal

Para la realización de este proyecto se siguió una planificación temporal por semanas, presentada al cliente al comienzo del mismo. Esta planificación fue sufriendo varios cambios a lo largo del proyecto, hasta que finalmente acabó una semana antes de lo previsto.

5.1 Diagrama de Gantt

El diagrama de Gantt se desarró en Excel, y se adjunta como documento. Este Excel se modificaba en función de los cambios del proyecto, y el resultado una vez terminado fue en que se puede apreciar en la [Fig. 49]



Etapas	Comienzo	Duración	Comienzo real	Duración real	% Completo
Elección de SIEM y arquitectura	1	2	1	2	100%
Recolección de información previa	3	2	4	3	100%
Configurar Qradar y usuarios admin	8	1	8	1	100%
Configuración grupos y usuarios SOC	9	1	9	2	100%
Creación taxonomía genérica	11	6	10	6	100%
Integración de fuentes	17	5	16	6	100%
Análisis sintáctico (parseado)	22	5	22	4	100%
Creación de mapas log-evento	27	3	26	3	100%
Integración de casos de uso	27	8	27	7	100%
Documentación casos de uso	27	5	27	6	100%
Integración herrramienta ticketing	17	2	20	2	100%
Instalación aplicaciones	17	1	19	1	100%
Métricas casos de uso	25	2	25	2	100%
Métricas SOC mensuales Qlik	35	1	34	1	100%

Figura 49. Diagrama de Gantt del proyecto

Las partes amarillas son aquellas que no coincidieron con la planificación inicial, mientras que las púrpura son aquellas que sí lo hicieron.

En esta etapa del diagrama, existe también una versión púrpura a rayas, siendo estos tramos aquellos en los que inicialmente se pensaba desarrollar la tarea pero se desarrolló en otro instante.

Capítulo 6. Conclusiones

Se ha implantado el SIEM Qradar en cada uno de los clientes, personalizando cada una de las instalaciones y en el tiempo esperado. Con esto se ha conseguido aumentar la seguridad de los clientes en cuanto a vigilancia y análisis forense.

Se ha cumplido también el objetivo de transmitir al cliente el desarrollo del proyecto a la hora de su realización y las métricas al volverse un servicio SOC.

Durante la instalación y personalización se han obtenido conocimientos de los cuales no se disponía con anterioridad: sobre cada uno de los entornos, diferentes herramientas de seguridad, RegEx…

Al trabajar con un equipo, se ha mejorado la comunicación, se ha tenido que sincronizar tres integrantes del equipo en turno de tarde y cuatro de mañana, aprendiendo diferentes técnicas de dirección de proyectos, trabajo en equipo y asignación de tareas. Se han evaluado, usado y configurado diferentes recursos, herramientas en la nube, recursos compartidos, comunicación vía chat, las webs de RegEx, Trello…

Se ha paralelizado el proyecto en cada uno de los clientes, y me he comunicado por email y en reuniones con cada uno de los responsables de seguridad de los distintos clientes, adaptando el discurso para cada uno de ellos, y transformando posteriormente a tareas técnicas a repartir entre los integrantes del equipo dichas comunicaciones.

Como resumen, se han cumplido los objetivos establecidos en el plazo previsto, haciendo consciente al cliente y al equipo, y se han obtenido conocimientos técnicos de seguridad y generalistas. Finalmente, se han desarrollado las capacidades empáticas y de trabajo en equipo, así como el liderazgo.

Capítulo 7. Anexos: definiciones, bibliografía y documentos

7.1 Definiciones de acrónimos, palabras técnicas y jerga de seguridad:

- **SIEM:** Security Information and Event Management. Es un sistema que centraliza el almacenamiento y la interpretación de los datos relevantes en un entorno de seguridad informática/cibernética. Es el "el cerebro" que es capaz de correlar distintos eventos.
- **MW:** Malicious Software (también conocido como "malware"). Es un software malicioso, es decir, una pieza de código no tangible que está diseñada para realizar acciones no deseadas por el usuario, generalmente en beneficio de una persona externa conocida vulgarmente como "hacker".
- **On premise:** Significa que el dispositivo está disponible físicamente, por ejemplo, en un CPD de la compañía.
- **CPD:** Centro de Procesamiento de Datos (Data Center en inglés). Lugar en el que se localizan recursos de una empresa o compañía necesarios para el procesado de datos de la misma.
- **EDR:** Endpoint Detection and Response. Son las herramientas que se focalizan en detectar (e investigar) posibles comportamientos sospechosos en el propio equipo de usuario final, ya sea un PC/portátil o un servidor. Pueden detectar comportamientos extraños en memoria, disco, programas, emisiones a internet…
- **APT:** Advanced Persistent Threath. Es una amenaza avanzada persistente, es decir, un MW que anida en la máquina afectada y puede realizar acciones en ella. Suelen estar controladas de forma remota.
- **ELK:** Elasticsearch+Logstash+Kibana. Esta combinación de software libre da como resultado el producto ELK, que combina la capacidad de Logstash (una pila dinámica enriquecida con plugins, para ingestar datos), la velocidad de búsqueda de Elasticsearch (motor para almacenar, buscar y analizar grandes volúmenes de datos en tiempo real) y la interfaz de Kibana (interfaz gráfica web encargada de la visualización mediante distintos diagramas y gráficos)
- **'Parsear':** Análisis sintáctico. Extraer información de un texto (en nuestro caso un log) y categorizarla para almacenarla y guardarla de forma ordenada.
- **'Mapear':** Crear una relación entre un log con una información concreta y una parte no variable, de tal manera que por cada log se genere un evento con ciertas propiedades definidas, como un nombre, una severidad, una categoría, una subcategoría… Estos eventos se usarán posteriormente por ejemplo para las reglas.
- **CEF:** Common Event Format. Es un formato de logs estándar con el que muchas herramientas pueden transmitir y recibir información, se basa en una cabecera fija y una parte variable. En la cabecera se transmite información sobre la fuente, y el resto varía en función de la fuente
- **LEEF:** Log Event Extended Format. Es un formato personalizado para Qradar, tal y como se enuncia en la siguiente web: https://www.ibm.com/support/knowledgecenter/en/SS42VS_DSM/com.ibm.dsm.doc/c_LEEF_Format_Guide_intro.html?cp=SS42VS_7.3.2.

7.2 Anexos

Como anexos incluidos en este TFM se tienen los siguientes documentos:

- **cef-log-components.pdf:** Usado en la sección 1.5.2. El documento es un ejemplo libre de logs CEF. Este loge es de Microsoft FTPD, y se ha usado para ejemplificar el comportamiento y el desarrollo de un tipo de fuente propietaria.
- **Mitre_matrix.xlsx:** es la matriz de los casos de uso en la cual se basan los genéricos de este proyecto.
- **example.log:** Contiene los logs de ejemplo usados para crear la relación log/evento.
- **Gantt_TFM:** Excel original usado para la creación del diagrama de Gantt de la planificación temporal.

7.3 Bibliografía

[1] Renaud Bidou, "Security Operation Center Concepts & Implementation" [Online], https://pdfs.semanticscholar.org/1ffa/f58ab9379b1d3ef11d18091fc08df777481b.pdf

[2] MITRE, "Ten Strategies of a World-Class Cybersecurity Operations Center", 2014, [Online] https://www.mitre.org/sites/default/files/publications/pr-13-1028-mitre-10-strategies-cyber-ops-center.pdf

[3] ForcePoint, "What is a SIEM" [Online] https://www.forcepoint.com/cyber-edu/siem

[4] Gartner, "SIEM magic cuadrant", [Online] https://www.gartner.com/en/documents/3894573

[5] IBM, "QRadar architecture overview", [Online], https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.3.2/com.ibm.qradar.doc/c_qradar_deployment_guide_arch.html

[6] IBM, "Qradar components", [Online], https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.2.8/com.ibm.qradar.doc/shc_qradar_comps.html

[7] IBM, "Instation guide", [Online], https://www.ibm.com/support/knowledgecenter/SS42VS_7.3.1/com.ibm.qradar.doc/b_siem_inst.pdf?view=kc

[8] IBM, "Qradar assistant app for Qradar", [Online], https://exchange.xforce.ibmcloud.com/hub/extension/ed8aee4440f98f9c8bedaff4c5c644de

[9] IBM, "Tuning app for Qradar", [Online], https://exchange.xforce.ibmcloud.com/hub/extension/bf01ee398bde8e5866fe51d0e1ee684a

[10] GitHub, "Elastic Common Schema", [Online], https://github.com/elastic/ecs

[11] Elastic Blog, "Introducing the Elastic Common Schema", [Online], https://www.elastic.co/blog/introducing-the-elastic-common-schema

[12] IBM, "Log sources user guide", [Online], ftp://ftp.software.ibm.com/software/security/products/qradar/documents/71MR1/LogMgr/LogSources-71MR1.pdf

[13] IBM, "Installing the WinCollect agent on a Windows host", [Online] https://www.ibm.com/support/knowledgecenter/en/SSKMKU/com.ibm.wincollect.doc/t_ug_wincollect_install_wincollect_agent.html

[14] IBM, "IBM QRadar Custom Properties for Microsoft Windows", [Online], https://exchange.xforce.ibmcloud.com/hub/extension/IBMQRadar:MicrosoftWindowsCustomProperties

[15] RexEgg, "Fundamentals", [Online], https://www.rexegg.com

[16] Regexr, "Learn build & test RegEx", [Online], https://www.rexegg.com

[17] Citrix, "Web App Firewall logs", [Online], https://docs.citrix.com/en-us/citrix-adc/12-1/application-firewall/logs.html

[18] Mitre, "MITRE ATT&CK™", [Online], https://attack.mitre.org

[19] Mitre, "Brute Force", [Online], https://attack.mitre.org/techniques/T1110/

[20] IBM, “Rule types”, [Online], https://www.ibm.com/support/knowledgecenter/en/SS42VS_7.2.6/com.ibm.qradar.doc/c_qradar_rul_typ.html

[21] Thehive project, “Thehive project”, [Online], https://thehive-project.org

[22] IBM, “IBM Resilient Security Orchestration, Automation and Response (SOAR)”, [Online], https://www.ibm.com/es-es/marketplace/resilient-incident-response-platform

[23] IBM Security App Exchange, “Network Hierarchy Management for QRadar”, [Online], https://exchange.xforce.ibmcloud.com/hub/extension/a35d463d776708c88a7fed70204d9953

[24] IBM Security App Exchange, “Network Hierarchy Management for QRadar”, [Online], https://exchange.xforce.ibmcloud.com/hub/extension/IBMQRadar:UserBehaviorAnalytics