INTEGRACIÓN DE OSSIM Y UNTANGLE

MARCOFI ANDRETTI TORRES MANRIQUE
DIEGO ALEJANDRO VILLEGAS OLIVEROS

UNIVERSIDAD ICESI
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA TELEMÁTICA
SANTIAGO DE CALI
2010

INTEGRACIÓN DE OSSIM Y UNTANGLE

MARCOFI ANDRETTI TORRES MANRIQUE
DIEGO ALEJANDRO VILLEGAS OLIVEROS

Trabajo de Grado presentado como requisito para optar al título de Ingeniero Telemático

Tutor del proyecto
Juan Manuel Madrid
Director del programa de Ingeniería Telemática

UNIVERSIDAD ICESI
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA TELEMÁTICA
SANTIAGO DE CALI
2010

Nota de aceptación:

_____
_____
_____
_____
_____
_____

_____
Firma del Presidente del Jurado

_____
Firma del Jurado

Santiago de Cali, 15-12-2010

AGRADECIMIENTOS

El presente proyecto de grado es el resultado de la idea que nos dio el director de carrera de Ingeniería Telemática, Juan Manuel Madrid. Gracias a él ampliamos nuestro conocimiento en el campo de la seguridad en las redes, viendo la necesidad de contar con un sistema robusto, sencillo y centralizado para manejar la seguridad de la información. Agradecemos también al profesor Carlos Andrey Montoya quien desde un principio, se preocupó por el avance del proyecto, compartió ideas para lograr la integración, resolvió algunas dudas en la instalación de los dos sistemas operativos en las máquinas virtuales y nos contactó con Rodrigo Bedoya, un experto en el manejo de la consola OSSIM.

El señor Rodrigo Bedoya se ofreció a colaborarnos en todo lo que le fue posible, nos dio acceso a algunas guías que emplea en las capacitaciones que ofrece en la compañía TGR[1], finalmente nos relacionó con Cristian Latorre, para asesorarnos con la integración del agente OSSEC en el servidor OSSIM. Latorre asistió durante un par de sábados al sitio donde se encontraba nuestro escenario de trabajo, con el fin de guiarnos. Estas asesorías fueron de gran ayuda, en una de ella nos mostró como ejemplo el uso de la plataforma OSSIM como herramienta de monitoreo del campus party[2] 2010 en Bogotá.

También agradecemos al profesor de la Universidad Icesi Juan David Osorio, quien nos explicó sobre el proyecto en el cual participó (Plugin Zoneminder). Nos dio una capacitación que permitió entender el manejo de las expresiones regulares que son interpretadas por Python.

Finalmente a nuestros padres que nos apoyaron emocional y económicamente en lo necesario para alcanzar los objetivos propuestos en este proyecto de grado.

[1] TGR. Empresa especializada en Servicios Informáticos en el mercado tecnológico; lidera el mercado Open Source y el ámbito de Seguridad Informática en la región del Valle del Cauca.

[2] El mayor evento de tecnología, creatividad, ocio y cultura digital en red del mundo.

CONTENIDO

pág.

LISTA DE TABLAS

pág.

LISTA DE ILUSTRACIONES

pág.

LISTA DE ANEXOS

GLOSARIO

ARCHIVO DE LOG: archivo en donde se registra los sucesos ocurridos en un sistema o una aplicación, almacenando parámetros relevantes que permiten identificar cuándo, dónde y por qué ocurrió un evento.

ATAQUE INFORMÁTICO: evento que atenta contra el correcto funcionamiento de un sistema, puede ser exitoso o no.

FIREWALL: es un sistema (o conjunto de ellos) ubicado entre dos redes para garantizar la seguridad. Mediante una serie de reglas permite o bloquea la comunicación entre dispositivos de la red.

FIRMAS: parámetros que son característicos o identifican contenidos de ataques de red dentro de los paquetes.

GATEWAY: dispositivo que permite conectar redes con protocolos y arquitecturas de diferente tipo. Su principal propósito consiste en retransmitir los paquetes que recibe, haciendo posible la traducción de direcciones (NAT) y cambio de formatos de los mensajes entre diferentes redes.

MALWARE: es un tipo de software que tiene como objetivo infiltrarse o dañar una computadora.

OPEN SOURCE: es una filosofía manejada por la comunidad de desarrolladores que consideran altos los beneficios de compartir el código de las aplicaciones. Un programa o aplicación debe cumplir con los siguientes criterios para ser considerado *Open Source*: distribución libre, código fuente de fácil acceso, trabajos desarrollados sobre el mismo código, el reconocimiento del autor o autores y mantener la filosofía *Open Source* sobre las aplicaciones derivadas.

PHISHING: modalidad de estafa cuyo objetivo es adquirir información confidencial del usuario, como contraseñas, números de tarjetas de creditos ó datos financieros y bancarios. En ocasiones funciona mediante la duplicación de sitios web que confunden al visitante con el sitio original y por eso depositan su confianza para ingresar los datos.

PLUGIN: es un módulo de software que añade características específicas para una aplicación de software más grande, permitiendo personalizar sus funcionalidades.

PROTOCOLO SSL: el protocolo SLL (Secure Sockets Layer) permite establecer conexiones seguras a través de internet, de forma sencilla y transparente. Su funcionamiento consiste en interponer una fase de codificación de los mensajes antes de enviarlos por la red y una fase de decodificación al recibirlos, estableciendo una comunicación segura.

RED *PEER TO PEER (P2P)*: red de computadores en la que todos los nodos se comportan iguales entre sí, es decir que actúan simultáneamente como clientes y servidores respecto a los demás nodos de la red. Este tipo de redes permiten el intercambio directo de información en cualquier formato, entre los computadores interconectados.

SEGURIDAD INFORMÁTICA: área de la informática que se enfoca en la protección de la infraestructura computacional. Comprende software, bases de datos, metadatos, archivos y todo lo que la organización valore (activo) y signifique un riesgo si ésta llega a manos de terceras personas.

SERVIDOR WEB: programa diseñado para la transferencia de hipertexto, páginas web o páginas HTML, implementa el protocolo HTTP (*HyperText Transfer Protocol*) que pertenece a la capa de aplicación del modelo OSI.

SIEM: es la combinación de SIM (S*ecurity Information Management*) y SEM (S*ecurity Event Management*). Una Plataforma SIEM es una herramienta informática utilizada principalmente en redes de datos empresariales para centralizar el almacenamiento y la interpretación de registros o eventos generados por las diferentes herramientas de hardware o software.

SISTEMA INFORMÁTICO: es el resultado de la interrelación entre componentes de Hardware, software y el recurso humano (Administrador de red).

SOFTWARE LIBRE: programas que se pueden utilizar, modificar y distribuir gratuitamente.

SPYWARE: Software que recopila información de un computador y después transmite a una entidad externa sin el conocimiento ó el consentimiento del propietario.

URL (Uniform Resource Locator): se refiere a la dirección única que identifica a una página web en internet.

INTRODUCCIÓN

En los últimos años se han desarrollado múltiples servidores de aplicaciones basados en directivas de Software Libre, que brindan servicios para la administración, control de los usuarios y servicios de una red. Sus bajos (y en muchas ocasiones nulos) costos de licenciamiento e implementación han permitido que dichos sistemas puedan ser adoptados por organizaciones que pretenden obtener tanto una reducción en sus costos de infraestructura, como una mayor solidez y estabilidad en sus plataformas. Existen numerosos ejemplos de programas de Código Abierto y Software Libre que se han constituido como verdaderos estándares en las funciones que llevan a cabo. El servidor web Apache[3] y el firewall *IPTables*[4] son ejemplos de ellos.

Un sistema informático puede ser más seguro en la medida en que las variables que afectan sus métricas de seguridad y riesgo puedan ser monitorizadas y controladas. Con éste propósito, las compañías adquieren diversas herramientas de seguridad para ser implementadas en su infraestructura informática. La dificultad de lograr una efectiva reducción del riesgo asociado a variables de seguridad informática no está en la implementación de las herramientas de seguridad, sino en el conocimiento de las ocurrencias reportadas por dichas herramientas en un marco de tiempo útil, y en la relación de estas ocurrencias. Una vez resuelto este problema, las herramientas de seguridad suelen ser un magnifico aliado en la detección y prevención de violaciones a las políticas de seguridad de los sistemas.

Con el objetivo de solucionar las dificultades de monitorización en el proceso tecnológico de gestión de la seguridad[5] informática se han creado las consolas de administración de seguridad, cuya arquitectura está diseñada para la recolección, la integración, la normalización y la correlación de eventos provenientes de diferentes fuentes, facilitando en este modo la labor del administrador de red al proveer información centralizada y en tiempo real. En esta forma se reduce el tiempo de respuesta y se mejora el proceso de manejo de incidentes cuando se detecta un ataque informático o vulnerabilidad en el sistema.

Al día de hoy, una de las herramientas más populares en este campo es la consola de seguridad OSSIM (Open Source Security Information Management). La

[3] www.apache.org
[4] www.iptables.org
[5] Los programas manejados en la consola OSSIM, los SIEM, solo resuelven problemas asociados al riesgo tecnológico.

integración de OSSIM con otras aplicaciones es efectuada a través de plugins, cuya función es recolectar y normalizar la información definida para detección, para luego enviarla al servidor de correlación de la consola de seguridad, que lleva a cabo la labor de interpretarla, desplegando así, los datos más relevantes en el análisis de un posible ataque informático.

1. PLANTEAMIENTO DEL PROBLEMA

Es de común consenso que los indicadores de seguridad de un sistema informático se incrementan favorablemente en la medida en que se implementan controles y herramientas específicas para solucionar problemas relacionados a la seguridad del sistema, la red y los usuarios. Sin embargo, la implementación de una gran cantidad de soluciones de seguridad en una red con frecuencia conlleva un aumento sustancial en la complejidad de administración de los dispositivos instalados para tal fin, observándose al final resultados adversos en los indicadores de gestión de la seguridad.

Así pues, un administrador de seguridad que sea el responsable de diferentes aplicaciones de seguridad enfrenta un problema complejo; identificar potenciales situaciones de peligro para los sistemas a su cargo a partir de diversos dispositivos con formatos de reporte y archivo completamente heterogéneos. De este modo se presentan situaciones en las cuales la atención de un incidente de seguridad requiere de un manejo avanzado de dos, tres o incluso más interfaces de administración pertenecientes a distintos dispositivos. El problema se agrava cuando debe hacerse una correlación de los eventos presentados por parte del encargado de la gestión del incidente, dado que la capacidad de un ser humano no es óptima para este tipo de tareas.

La dispersión y descentralización de los sistemas de información ha llevado al desarrollo de aplicaciones de tipo SIEM[6] (*security information and event manager*), que permiten centralizar el almacenamiento y la interpretación de registros o eventos generados por otras herramientas. Si bien existen casos de software comercial con funciones de SIEM (*ArcSight*, por ejemplo), hay una motivación especial para este tipo de arquitecturas en el mundo del Software Libre. La versatilidad y extensibilidad de las arquitecturas abiertas ha permitido que numerosos desarrolladores puedan escribir piezas de código con el objetivo de integrar sus sistemas específicos a una plataforma de monitorización estándar. Dichas piezas de código se conocen como plugins, y son los encargados de normalizar los protocolos de contenido y transmisión de información concerniente a seguridad desde cualquier dispositivo que se desee integrar.

[6] Tecnología SIEM es una herramienta informática utilizada principalmente en redes de datos empresariales para centralizar el almacenamiento y la interpretación de los registros, o eventos, generados por diferentes herramientas hardware o software que se ejecutan en la red.

2. OBJETIVOS

Objetivo general:

Realizar el desarrollo necesario para integrar eventos de monitorización generados por aplicaciones de UNTANGLE a la arquitectura tipo SIEM provista por la consola de seguridad de OSSIM, para lograr de esta manera la consecución de un sistema de gestión y control centralizado y eficiente.

Objetivos específicos:

1. Identificar los métodos que permiten generar archivos de log (información de actividad) en Untangle.

2. Identificar los métodos de comunicación que permiten a OSSIM obtener la información generada por las diferentes herramientas de seguridad integradas en su arquitectura.

3. Diseñar el plugin o la serie de plugins que permitan relacionar las salidas de Untangle con las entradas de OSSIM.

4. Desarrollar una interfaz que permita visualizar las entradas de Untangle en OSSIM o complementar la existente.

5. Crear un manual de implantación de las dos herramientas integradas.

3. JUSTIFICACIÓN

Una de las principales dificultades que plantean las redes de hoy en día, es obtener un sistema de información altamente seguro. Se logra mejorar los indicadores de seguridad en un sistema en la medida que se implementan múltiples controles, herramientas y políticas de gestión y monitorización en la red, que generen registros con suficiente trazabilidad acerca de incidentes de seguridad o usos indebidos del sistema.

A medida que se decide implementar herramientas que controlen la red, la diversidad de estas puede acarrear varios problemas. Por un lado, los datos que registra cada una de ellas en general carecen de un estándar de formato que permita unificarlos y centralizarlos, y en la mayoría de los casos se requieren conocimientos especializados en la interpretación de los datos arrojados por cada herramienta. En muchas ocasiones se presenta una heterogeneidad en las interfaces de manejo y control de las herramientas implementadas, lo cual dificulta la gestión, interpretación y seguimiento de incidentes y ocurrencias relevantes a la seguridad por parte del administrador del sistema informático, al tener que reunir información de varias fuentes y tiempos en forma manual.

Por lo anterior, surge la necesidad de normalizar los datos recolectados por las diferentes herramientas y centralizarlos, de esta manera se mejoran los indicadores de seguridad del sistema sin adicionar complejidad en la gestión y detección de anomalías generadas por malos usos o ataques informáticos. La integración de Untangle y Ossim resulta ser un desarrollo necesario para complementar diversas funciones, al permitir la vinculación entre sistemas que permitan obtener una información estandarizada y centralizada para un manejo adecuado.

4. ANTECEDENTES

La información, junto con las personas y los equipos, es uno de los activos más importantes para una organización. En estándares de seguridad informática, como la norma ISO/IEC 27002, se hace referencia a que este tipo de seguridad debe estar basada en generar copias de respaldo periódicamente, tener instalado potentes antivirus para el sistema de información, y también en implementar una serie de controles, políticas y reglas que ayuden a mejorar la protección de la información[7].

Un razonamiento simple, indica que un sistema de información nunca será 100% seguro, a menos que la superficie de ataque efectiva sobre el mismo sea nula. La existencia del sistema, significa que las métricas de riesgo asociadas al mismo no son nulas, por lo cual es recomendable la implementación de un sistema de detección robusto que garantice la detección de la mayor cantidad posible de amenazas, y que sea capaz de notificar eventos de seguridad en tiempo real, con el fin de tomar decisiones oportunamente en caso de un incidente.

Existen algunas herramientas que pueden ayudar a los administradores de la red a mejorar su desempeño en la labor de control y gestión de la seguridad, dando a conocer comportamientos intrusivos, peligrosos y/o anómalos en los sistemas administrados. Estas utilidades son programas que se instalan en los dispositivos físicos, entre los cuales se encuentran agentes de detección, antivirus y *firewall*. Los agentes pueden ser considerados programas, encargados de monitorizar los archivos de registro del sistema operativo y generar alertas cuando se presentan sucesos anormales; los antivirus de proteger las máquinas de posibles programas, considerados como malignos, previniendo su ejecución y propagación; y finalmente están los Firewall que controlan el acceso a ciertos recursos de la red.

Se encuentran también programas especializados en la monitorización y gestión de los recursos de red, como la consola de seguridad OSSIM y la distribucción Untangle. OSSIM es una de las herramientas *Open Source* más usadas para la recolección y consolidación de información de aplicaciones provenientes de los dispositivos de conectividad, equipos y servidores en las empresas. Untangle es un potente *gateway* capaz de desempeñar diferentes roles de seguridad dependiendo de la aplicación que se habilite, en él se pueden realizar configuraciones que permiten obtener un sistema con un mayor grado de seguridad y control sobre la red.

[7] Seguridad Informatica Norma ISO 27001. Clave internet: http://www.rinconinformatico.net/seguridad-informatica-norma-iso-27001.

5. MARCO TEÓRICO

Los principales objetivos que debe manejar la seguridad informática es proteger la confidencialidad, integridad y disponibilidad de la información. La confidencialidad consiste en asegurar que sólo individuos autorizados puedan acceder a ciertos recursos. La integridad garantiza que la información sólo es modificada por quien posee la autorización, garantizando incluso el manejo en la red. Finalmente la disponibilidad es poder utilizar los datos en el momento deseado, siempre y cuando esté permitido dentro de las políticas de uso.

5.1 OSSIM

La palabra OSSIM es un acrónimo para *Open Source Security Information Management*, en español podría traducirse como: Herramienta de código abierto para la gestión de seguridad de la información. OSSIM no es tan sólo una herramienta, sino una combinación de herramientas, todas de código libre, que construyen una infraestructura de monitorización de la seguridad.

El principal objetivo de OSSIM es establecer una estructura completamente centralizada que permita visualizar y analizar los eventos relevantes que ocurren en una infraestructura de IT[8].

La capacidad de la consola OSSIM para obtener información completa y selecta, de los miles de eventos que reportan otras herramientas, le permite ser una herramienta muy útil. A los administradores de red, les permite elegir el procedimiento que regirá la seguridad en el sistema de información, pudiendo así, detectar amenazas rápidamente y disponer de un nivel adecuado de protección para la información y equipos que permiten la comunicación dentro de la red.

Al Considerar que en muchas ocasiones se presenta grandes cantidades de alertas, de las cuales no todas son confiables, en los últimos años se han generado *plugins* y mejoras, que permiten aumentar su rendimiento y compatibilidad con otras aplicaciones. Lo cual ha permitido la monitorización de equipos físicos de gran importancia para la seguridad de los sistemas de información[9].

La arquitectura de OSSIM esta basada en el modelo cliente-servidor. El servidor OSSIM es un demonio que se ejecuta para el procesamiento de múltiples tareas,

[8] Sistemas de computo y redes.

[9] OSSIM – Descripción general. Clave internet: http://www.alienvault.com/community.php?section=WhatisES

como la recolección de los datos de agentes u otros servidores, la priorización y la correlación de eventos, el almacenamiento de eventos en la base de datos y el envió de eventos o alarmas a otros servidores.

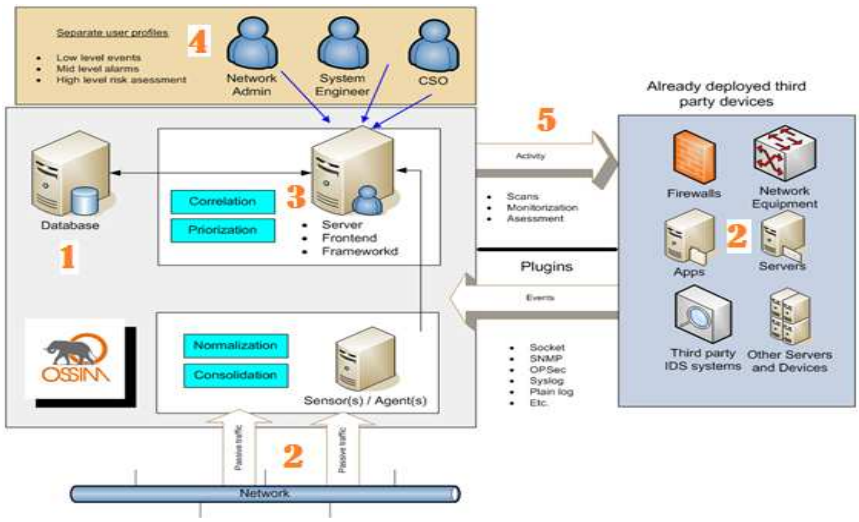
Los agentes de OSSIM son responsables de recolectar todos los datos importantes en el funcionamiento de los diferentes dispositivos de la red, para estandarizarlos y enviarlos al servidor OSSIM. Cuando esta información llega al servidor OSSIM se convierte en un evento. Los puertos manejados por los agentes son el 40001 para la conexión con el servidor y el 3306 cuando se emplea el uso de monitores.

El proceso de recolección de un agente generalmente implica filtrado y extracción de datos desde archivos de *log*. En esta etapa se determina que datos de un sistema pueden ser descartados, dependiendo de la funcionalidad del dispositivo.
La normalización de un evento permite dar un formato particular de la información a OSSIM. La normalización asegura la evaluación y la correlación consistente por parte del servidor OSSIM

5.1.1 Niveles

OSSIM posee una arquitectura de monitorización abierta que integra varios productos *Open Source* y se estructura en cinco niveles. En la Ilustración 1 se puede observar la relación existente entre los diferentes niveles.

Ilustración 1 Niveles de OSSIM



Fuente: CASAL, Julio. OSSIM – Descripción general del sistema. Clave internet: http://www.alienvault.com/docs/OSSIM-desc-es.pdf

En el primero, se encuentra la base de datos donde son almacenados los eventos que llegan al servidor. Se debe diferenciar las tres bases de datos que maneja la consola de seguridad de seguridad, las cuales son: EDB (*event database*), KDB (*Knowledge or framework database*) y UDB (*user or profile database*). EDB es la base de datos de todos los eventos que han llegado por medio de los detectores[10], KDB almacena las configuraciones para las políticas de seguridad y UDB registra datos específicos de los usuarios.

En un segundo nivel se ubican las herramientas de pre-procesamiento, las cuales recolectan toda la información de los eventos que suceden en el sistema informático y los lleva hasta el servidor OSSIM. Como ejemplo de estas herramientas se tienen los *firewalls*, IDS (Detectores de intrusos), detectores de anomalías y otros monitores.

El tercer nivel de OSSIM, es el servidor que realiza el post-procesamiento, el cual posee las herramientas que realizan actividades de procesamiento y análisis de datos. Utilizando la correlación de eventos para dar nivel de prioridad y valorar el riesgo que pueda implicar un suceso para el sistema. Lo anterior se realiza con el objetivo de aumentar la fiabilidad (Grado de certeza de un ataque) y sensibilidad de la detección a posibles ataques (¿Qué eventos se pueden descartar?).

En el cuarto nivel de OSSIM, están las herramientas de monitorización, las cuales permiten visualizar la información que ha sido procesada y clasificada en los niveles de seguridad: alto, medio y bajo. En un nivel alto se utiliza la herramienta de cuadro de mandos, la cual muestra los eventos que resultan ser muy críticos e identifican un perfil de ataque determinado. En un nivel medio se emplean los monitores de riesgo y comportamiento, estos muestran comportamientos que puedan implicar un riesgo informático. Finalmente en un nivel bajo se encuentran la consola forense y monitores de red, los cuales muestran todos los eventos de bajo nivel que son reportados al servidor OSSIM.

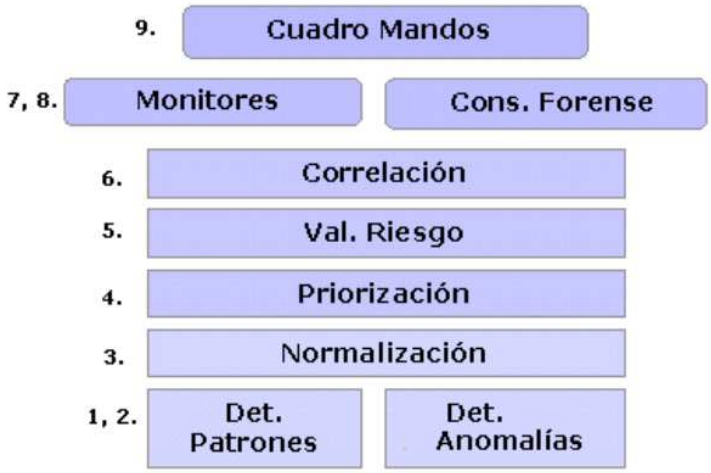
El último nivel es el *framework*, una herramienta que permite administrar y configurar todos los módulos que constituyen la consola de seguridad. En éste se definen reglas de correlación, políticas de seguridad, inventario de activos, topología y se enlaza cada una de las herramientas pertenecientes a los otros niveles.

[10] Los Detectores son programas que escuchan en la red, vía socket o archivos de log, en busca de patrones predefinidos y producen eventos de seguridad cuando encuentra coincidencias con estos.

5.1.2 Funcionamiento

El funcionamiento entre los niveles de OSSIM ocurre en el orden que se muestra en la Ilustración 2 y el cual, se explica a continuación.

Ilustración 2 Funcionamiento OSSIM



Fuente: CASAL, Julio. OSSIM – Descripción general del sistema. Clave internet: http://www.alienvault.com/docs/OSSIM-desc-es.pdf

1. Los detectores de patrones funcionan a través de reglas que se definen por omisión o que pueden ser ajustadas a una necesidad. Se encargan de analizar posibles problemas de seguridad en la red y en caso de detectar uno, alertan al sistema de seguridad.

2. La detección de anomalías consiste en alertar sobre cualquier comportamiento que no sea normal en la red. Un ejemplo de esto, es cuando se hace una copia de archivos que están en la red interna por un usuario que no ha sido autorizado.

3. La normalización unifica todos los eventos provenientes de los detectores en una sola consola y con un mismo formato.

4. La priorización está relacionada con la importancia que se le debe dar a un evento con respecto a un escenario que se puede presentar. Está configurada dentro del *framework*.

5. Después de priorizar los eventos, se debe valorar el riesgo del suceso, el cual está relacionado de forma directa con el activo asociado al riesgo, el tipo de amenaza que involucra y la probabilidad de ocurrencia.

6. La Correlación es una función mediante la cual se relacionan diferentes eventos que pueden estar involucrados en el mismo ataque. Esta función tiene dos formas de realizar esta correlación: mediante una secuencia definida de eventos o mediante un algoritmo heurístico. En una secuencia de eventos existen patrones ya definidos mediante ocurrencias de eventos únicos repetidos durante un intervalo de tiempo. Dichas secuencias de eventos son llamadas directivas de correlación lógicas. En el caso de los algoritmos heurísticos, se intenta detectar un comportamiento de riesgo a través de un proceso de correlación compuesto, con un primer paso de correlación lógica general y un posterior procesamiento en correlación cruzada, en donde se realiza una búsqueda de coincidencias entre los comportamientos observados y patrones de riesgo asociados con vulnerabilidades.

7. Los monitores de riesgos muestran los valores del nivel de riesgo que se le ha dado a un proceso después de efectuada la correlación. Se puede obtener también información sobre el uso de las sesiones de ciertos usuarios.

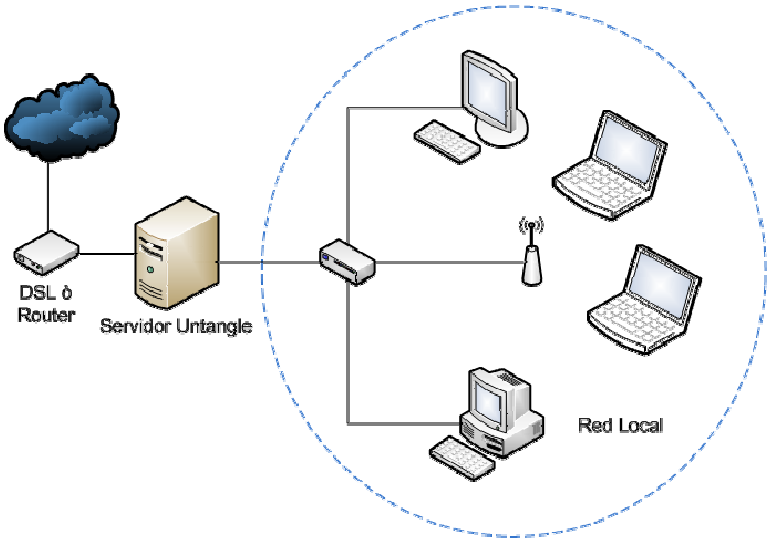
8. La consola forense permite analizar todos los eventos que han sido recolectados y guardados por el sistema, y reúne un poco más de información estadística sobre todos los riesgos que se han presentado.

9. El cuadro de mandos tiene los indicadores que permiten medir el estado de seguridad en la red, definiendo unos límites para un nivel normal de funcionamiento. Estos límites se conocen como umbrales.

5.2 UNTANGLE

Untangle es una plataforma para *gateway* (Ilustración 3) de código abierto, usada principalmente en las pequeñas empresas, colegios y organizaciones sin ánimo de lucro. Esta plataforma permite configurar gran cantidad de equipos para brindar la conexión a Internet u otra red, realizando operaciones de gestión de red, como la traducción de direcciones IP (NAT, *Network Address Translation*).

Ilustración 3 Implementación básica Untangle



Su ubicación de borde en la red, permite ofrecer varios servicios para la seguridad de la red interna y también le permite realizar análisis de todos los paquetes que circulan a través de él, con el fin de obtener reportes sobre la actividad de los usuarios. Untangle ofrece una forma simple para proteger, controlar y monitorear una red de computadores. Tiene la tecnología necesaria para proteger contra virus, *spyware* y ataques. También protege la productividad controlando la navegación web, generando un informe detallado de la actividad de la red en una sola interfaz gráfica.

El servidor de Untangle se puede conectar de dos formas, como router/firewall ó como complemento de éstos en modo bridge. Si se cuenta con un router o firewall, el servidor Untangle debe estar antes del switch principal de la red (Ilustración 4). En este momento el servidor Untangle cumple la función de bridge; no es necesario cambiar la ruta de salida (Gateway) de los equipos que pertenecen a la red local. Si no se tiene un router ó se quiere reemplazar uno existente, se debe conectar el servidor Untangle directamente entre la conexión a internet y el switch principal, así el nuevo servidor dará los servicios de enrutamiento y de protección a la red (Ilustración 5).

Ilustración 4 Conectividad Servidor Untangle en Modo Bridge

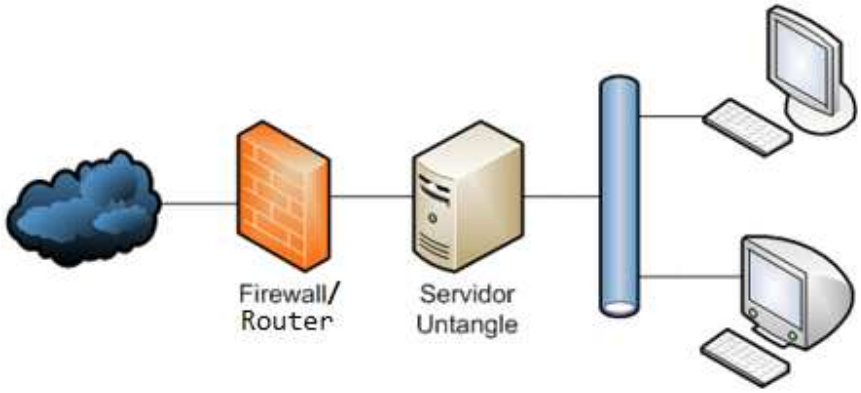
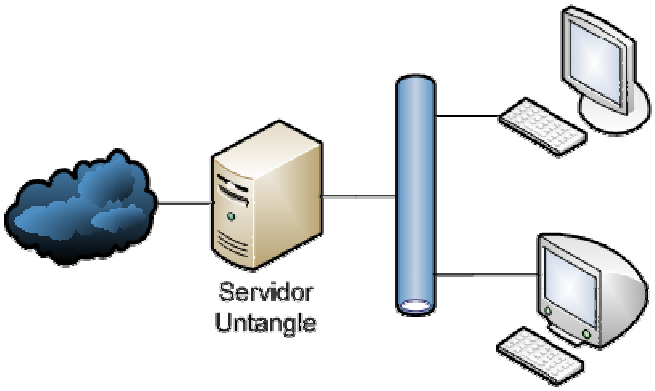


Ilustración 5 Conectividad Servidor Untangle en Modo Router



5.2.1 Aplicaciones de Untangle.

Como se mencionó anteriormente, la ubicación de Untangle en el borde de la red y su función como Gateway le permite utilizar aplicaciones que realicen el análisis de los paquetes y de las solicitudes de los clientes internos. En Untangle se encuentran definidas dos tipos de aplicaciones: aplicaciones de filtrado y aplicaciones de servicio.

Las aplicaciones de filtrado son aquellas que permiten crear políticas para diferentes grupos de usuarios. Generalmente son aplicaciones que bloquean el uso indebido de los recursos de ancho de banda a ciertos usuarios. Las aplicaciones de servicio solo se pueden configurar de manera global, por tanto si se adiciona o remueve alguno de estos servicios el cambio será efectivo para

todos los usuarios. Generalmente son aplicaciones que protegen la integridad de la red.

5.2.1.1 Aplicaciones de Filtrado Open Source

- *Web Filter*. Examina el tráfico del protocolo HTTP con el fin de bloquear o dejar registros de alguna actividad específica. Algunos sitios web pueden ser bloqueados o registrados de acuerdo a las categorías (pornografía, juegos de azar, redes sociales, etc.), de acuerdo con la URL (MySpace, Youtube, Facebook, ESPN, etc.), tipo MIME y al tipo de extensión del archivo (.exe, .mp3, .avi, etc.).[11]

- *Virus Blocker*. Examina todo el correo electrónico del servidor Untangle, evitando que los virus lleguen o salgan de los equipos de la red a través de este medio.[12]

- *Spam Blocker*. Permite clasificar los correos que se reciben en la bandeja de entrada, como no deseados, dependiendo del remitente del mensaje. Cuando un correo electrónico está en los correos no deseados se elimina después de 30 días.[13]

- *Ad Blocker*. Permite bloquear los sitios web con publicidad más conocidos a nivel global. Este bloqueo se realiza en base a una lista que el servidor Untangle descarga cuando se instala la aplicación y se actualiza periodicamente desde la misma aplicación.[14]

- *Phish Blocker*. Esta aplicación protege de la suplantación (*PHISHING*) de correos electrónicos o páginas web. Phish Blocker inspecciona los correos fraudulentos que buscan obtener información confidencial como contraseñas y detalles de tarjetas de crédito, haciéndose pasar por una persona de confianza o una entidad bancaria.[15]

- *Spyware Blocker*. Bloquea los programas espía para evitar perdida de información de los usuarios de la red. Utiliza las firmas de virus para detectar e identificar virus específicos.

---

[11] Untangle Server User's Guide. Web filter. Clave internet: http://wiki.untangle.com/index.php/Web_Filter

[12] Untangle Server User's Guide. Virus blocker. Clave internet: http://wiki.untangle.com/index.php/Virus_Blocker

[13] Untangle Server User's Guide. Spam Blocker. Clave internet: http://wiki.untangle.com/index.php/Spam_Blocker

[14] Untangle Server User's Guide. Ad Blocker. Clave internet: http://wiki.untangle.com/index.php/Ad_Blocker

[15] Untangle Server User's Guide. Phish Blocker. Clave internet: http://wiki.untangle.com/index.php/Phish_Blocker

Proporciona una lista negra de URL's (*Uniform Resource Locator*) para bloquear la descarga de software espía (*malware*) desde esos sitios web.[16]

- *Firewall*. Esta aplicación presta la funcionalidad tradicional de los firewall, bloqueando o dejando registro del tráfico establecido en la lista de reglas. Cada vez que se establece una nueva sesión de conexión, se evalúa la lista de reglas en su respectivo orden, si alguna de las reglas coincide con la configuración realizada, entonces se aplica la acción correspondiente: bloquear o registrar.

Se puede construir una lista de reglas que satisfaga las necesidades de la configuración de la red local para el controlar el tráfico por tipo de protocolo, interfaz de origen o destino, dirección IP de origen y destino o puerto origen y destino.[17]

- *QoS.* Esta aplicación garantiza que determinadas aplicaciones tengan acceso prioritario al ancho de banda, permitiendo reservar un ancho de banda. El administrador podrá decidir qué ancho de banda se reserva dependiendo del tamaño o tipo de conexiones y la sensibilidad de las aplicaciones a correr de manera concurrente. Se pueden clasificar los servicios en colas de alta, media y baja prioridad.

QoS puede mejorar el tráfico de algunos protocolos en la red, sobre todo cuando el ancho de banda está saturado por el uso intensivo de otras aplicaciones, como ejemplo, el video Streaming. Sin embargo, la calidad de servicio puede ser causante de un mal rendimiento de la red, si no se configura de manera adecuada.[18]

- *Intrusion Prevention.* Intercepta todo el tráfico y detecta actividad maliciosa en la red o en los equipos. Para detectar la actividad maliciosa, Intrusion Prevention utiliza detección por firmas, un método que se basa en comparar el contenido de los paquetes con patrones de ataque contenidos en una base de datos, que ha de ser actualizada periódicamente.[19]

- *Control Protocol.* Esta aplicación permite al administrador tener un mayor control de la red y controlar aplicaciones que usan de manera intensa el tráfico de la red hacia determinados puertos, como por ejemplo E-mule, Bit-torrent, Ares o aplicaciones de juegos en línea. El filtrado se basa en el protocolo de las

[16] Untangle Server User's Guide. Spyware Blocker. Clave internet: http://wiki.untangle.com/index.php/Spyware_Blocker
[17] Untangle Server User's Guide. Firewall. Clave internet: http://wiki.untangle.com/index.php/Firewall
[18] Untangle Server User's Guide. QoS. Clave internet: http://wiki.untangle.com/index.php/QoS
[19] Untangle Server User's Guide. Intrusion Prevention. Clave internet: http://wiki.untangle.com/index.php/Intrusion_Prevention

aplicaciones que se desea no permitir. Esto ofrece las siguientes ventajas: conservar el ancho de banda, mejorar la productividad mediante el control de aplicaciones que escapan a las reglas del firewall y diseñar firmas personalizadas para bloquear cualquier protocolo.[20]

5.2.1.2 Aplicaciones de Servicio Open Source.

- *Attack Blocker*. Este servicio realiza el seguimiento de tráfico de todos los equipos, monitoreando el número de conexiones y el volumen de datos manejados. Si un equipo determinado es más activo que otro, aumentará su reputación. Entre mayor sea la reputación del equipo, indicará que está consumiendo una mayor cantidad de recursos y se podrá tomar acciones como administrador.[21]

- *OpenVPN*. Permite configurar acceso remoto a la red interna de Untangle, desde cualquier parte del mundo, a través de Internet. Se basa en el protocolo SSL, para proporcionar un nivel alto de seguridad y protección del tráfico de la información. Funciona sobre una gran cantidad de Sistemas Operativos (Windows 2000/XP, Linux, Mac OS, entre otros).[22]

- *Reports*. Esta aplicación proporciona información de comportamientos de los clientes y su nivel de incidencia sobre la red de Untangle. Proporciona los datos necesarios para investigar incidentes relacionados con la seguridad, y tomar decisiones que permitan hacer cumplir las políticas de la red. También permite analizar los patrones y flujos de tráfico de la red.[23]

5.3 SISTEMAS DE DETECCIÓN DE INTRUSOS (IDS)

Para explicar la definición de un Sistema de Detección de Intrusos, primero se comienza con el concepto de “detector”, que es un programa capaz de procesar información en tiempo real a bajo nivel, para generar alertas.

Un intruso es una entidad (persona o programa), que está haciendo un uso indebido de la red para aprovechar sus recursos; por ejemplo, la obtención de

[20] Detalles Untangle Red Segura. Control de protocolos. Clave internet: http://openti.net/index.php?option=com_content&task=view&id=39&Itemid=45#bspy
[21] Untangle Server User's Guide. Attack Blocker. Clave internet: http://wiki.untangle.com/index.php/Attack_Blocker
[22] Untangle Server User's Guide. Open VPN. Clave internet: http://wiki.untangle.com/index.php/OpenVPN
[23] Untangle Server User's Guide. Reports. Clave internet: http://wiki.untangle.com/index.php/Reports

información confidencial, el aprovechamiento de algún recurso de la red, entre otras acciones que pueden poner en riesgo una compañía.

El sistema de detección de intrusos o IDS, ayuda a detectar posibles intrusiones a un sistema o el uso inadecuado de algún recurso, haciendo más fácil la gestión del administrador de red. Esto se realiza mediante análisis de los registros críticos que se generan en un equipo, o a través del análisis de paquetes que ingresan a la red. De acuerdo con la ubicación del IDS, se puede tener dos clasificaciones: NIDS e HIDS. [24]

En un HIDS (*Host Intrusion Detection System*), el sistema detector se basa en la información de un equipo. El NIDS (*Network Intrusion Detection System*) trabaja con los datos que circulan a través de la red, buscando posibles accesos no autorizados o comportamientos anormales.

5.4 OSSEC

OSSEC es una herramienta open source para la detección de intrusos en equipos (HIDS). Desarrolla análisis de registros, archivos, y archivos ejecutables del sistema, de esta manera detecta y alerta sobre anomalías introducidas al equipo, que podrían significar un grave riesgo para la integridad del sistema.

La eficiencia, versatilidad y ventajas que ofrece esta herramienta le han permitido lograr gran aceptación dentro de la comunidad de herramientas de seguridad y un amplio crecimiento a lo largo de los últimos años. OSSEC está disponible para varios sistemas operativos, como Windows, Linux, AIX, entre otros. Su arquitectura cliente servidor, le permite un manejo y control centralizado a lo largo de toda la red, y permite integración con aplicaciones SEM.

OSSEC posee una base de reglas de detección que ha sido desarrollada por usuarios de todo el mundo, pero también permite que el administrador del sistema defina sus propias reglas. Las reglas de detección se deben actualizar frecuentemente debido a que día a día se desarrollan nuevas formas de explotar vulnerabilidades de los sistemas, y la no actualización u obsolescencia de las reglas podrían dar cabida a una intrusión.

[24] Intrusion Detection System Logs as Evidence and legal aspects. Forensic focus. [Documento en línea]. Clave internet: http://www.forensicfocus.com/intrusion-detection-system-logs (consultado septiembre 02 de 2009)

5.4.1 Arquitectura.

OSSEC está compuesto por múltiples elementos (Servidor y Agentes). OSSEC Server, monitorea y recibe información de múltiples agentes instalados en diferentes equipos de un sistema informático, utilizando una arquitectura cliente servidor.

5.4.1.1 OSSEC Server.

El servidor OSSEC es el elemento principal para el funcionamiento de OSSEC, este contiene una base de datos con la información de integridad de los archivos que se deben monitorizar, los agentes le reportan constantemente registros, eventos o modificaciones en los archivos que monitorizan para constatar cambios, comparando la información reportada con la contenida en su base de datos de integridad.

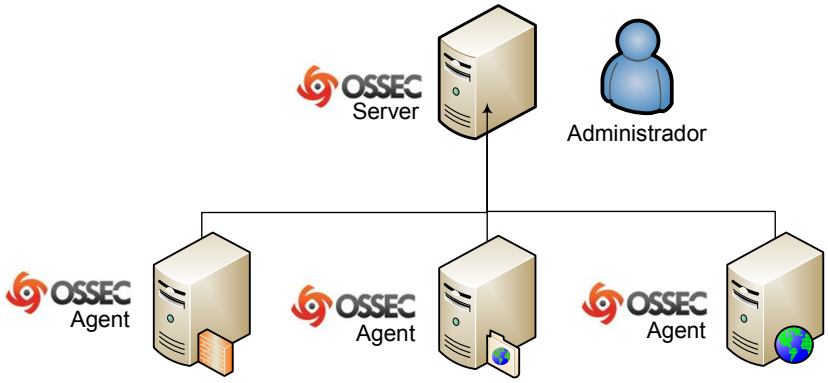
A través del OSSEC server son manejadas las principales opciones de configuración de los agentes, los decodificadores que identificaran cada registro de log recibido y las reglas que permiten identificar un ataque determinado, de esta manera (centralizada) se hace fácil la administración de gran cantidad de agentes instalados en todo el sistema informático.

5.4.1.2 OSSEC Agent.

El agente es una aplicación que se instala en el equipo a monitorizar. Recolecta información en tiempo real y la envía al OSSEC Server para que éste realice su análisis y comparación con la base de datos contenida. El procesamiento y la memoria utilizada por estos agentes son muy reducidos, por lo que no afectan el rendimiento del equipo que están revisando.

A continuación (Ilustración 6) se muestra la arquitectura cliente servidor manejada por OSSEC, en la cual los agentes recolectan la información y le notifican al OSSEC Server, encargado del procesamiento y la detección de eventos que puedan poner al sistema informático en riesgo y a su vez notificar al administrador.

Ilustración 6 Arquitectura Cliente Servidor OSSEC



Adaptación de: http://www.ossec.net/main/ossec-architecture/

5.5 FUNCIONES ALMACENADAS EN BASES DE DATOS

La aplicaciónes de Untangle utilizán bases de datos Postgres para almacenar los registros de eventos de las aplicaciones. Las bases de datos permiten crear funciones almacenadas de acuerdo con las necesidades presentes en el manejo de datos. Estas funciones contienen fragmentos de código que pueden estar en lenguaje SQL[25], C[26], Shell[27], entre otros[28]. Cada sentencia permite realizar cálculos, manejo de cadenas y consultas sobre la base de datos de manera secuencial dentro de la función.

Es importante anotar que las funciones en Postgres son similares a las de otros lenguajes de programación pues estas tienen parámetros de entrada, valores de retorno e internamente se pueden declarar variables. Se debe tener en cuenta que todas las variables declaradas en una función sólo tendrán validez durante la ejecución de la función.

Las funciones almacenadas que se ejecutan, son creadas dentro de un sistema de base de datos, para evitar la redundancia de los datos, eliminar inconsistencias, Mantener la integridad de los datos, realizar validaciones necesarias cuando se realicen modificaciones en la base de datos, entre otras. En el caso del proyecto, la función almacenada se emplea para agregar un evento registrado de las

[25] Lenguaje estándar para acceder y manipular bases de datos.
[26] Es un lenguaje de programación diseñado para la implementación de software y el desarrollo de aplicaciones portables.
[27] Lenguaje que interpreta y ejecuta comandos en un sistema operativo ingresados por el usuario.
[28] ANDRADE, Roberto. Programación de funciones en PL/pgSQL para PostgreSQL, 2002. Clave internet: http://sdi.bcn.cl/desarrollo/doctos/PL_pgSQL.pdf

aplicaciones de Untangle sobre la base de datos, en un archivo de texto plano. En la Tabla 1 se encuentra un ejemplo de una función en una base de datos.

Tabla 1 Ejemplo de Función en Base de Datos

Partes de una Función	Función suma

CREATE FUNCTION nombrefuncion (parametro, parametro)
RETURNS tiporetorno
AS $$
DECLARE
variable;
variable;
variable;
BEGIN
sentencia; -- esto es un comentario
sentencia; /* esto es un bloque de comentario */
RETURN res;
END;
$$
Languaje 'plpgsql';
	CREATE FUNCTION suma (int4, int4)
	RETURNS int4
	AS $$
	DECLARE
	a int4;
	b int4;
	res int4;
	BEGIN
	a :=$1;
	b := $2;
	res := a + b;
	RETURN res;
	END;
	$$
	Languaje 'plpgsql';

Fuente: Funciones en PostgreSQL- http://www.scribd.com/doc/102830/Funciones-en-PostgreSQL

5.6 TRIGGERS EN BASE DE DATOS

Un *trigger* (en español, disparador), especifica una función que debe ejecutarse cada vez que cierto tipo de operación se realice sobre una tabla en la base de datos. En el proyecto serán utilizados con el fin de copiar los eventos que se registran en la base de datos a un archivo de texto. La ejecución de un *trigger* puede definirse antes o después de cualquier modificación (UPDATE), eliminación (DELETE) o inserción (INSERT) de un registro. Una vez se active el *trigger*, la función asociada es llamada para ejecutarse de manera inmediata.[29]

En *postgres* las funciones del *trigger* pueden invocar a otras funciones escritas en otros lenguajes tales como PL/PERL, PL/JAVA, PL/TCL o para el caso particular

[29] Capitulo 20. Disparadores (triggers). Clave internet: http://dev.mysql.com/doc/refman/5.0/es/triggers.html

de este proyecto PL/SH. A continuación se encuentra un ejemplo para invocar una función desde un *trigger*.

```
CREATE OR REPLACE FUNCTION actualizar() RETURNS TRIGGER
AS $ejemplo$
  BEGIN
      NEW.nombre := NEW.nombres || ' ' || NEW.apellidos ;
      RETURN NEW;
  END;
$ejemplo$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER ejemplo
BEFORE INSERT OR UPDATE ON tablaregistros
   FOR EACH ROW EXECUTE PROCEDURE
actualizar();
```

En el trigger utilizado llamado ejemplo, una vez se actualiza o se inserta contenido en una fila sobre los campos nombres y apellidos en la tabla llamada tablaregistros, el trigger ejecuta la función y asigna al campo nombre los campos nombres, apellidos de manera concatenada, como se muestra en la siguiente tabla:

Id	**Nombre**	**Nombres**	**Apellidos**
1	Marcofi Andretti Torres	Marcofi Andretti	Torres
2	Diego Villegas	Diego	Villegas

6. MARCO METODOLÓGICO

El desarrollo de este proyecto se orienta hacia la elaboración de una guía paso a paso de la instalación y configuración de herramientas que permitan monitorizar en la consola de seguridad OSSIM, los eventos que ocurren en las aplicaciones del gateway Untangle.

La elaboración del proyecto consta de cinco etapas que se desarrollaron de forma consecutiva para alcanzar un mayor grado de claridad con el objetivo de poder replicar la información. En la etapa inicial, se cubre el proceso de instalación de ambos sistemas operativos en máquinas virtuales. Cuando estas funcionaron, se realizó el proceso de investigación en la Internet sobre el almacenamiento de los eventos dentro de las aplicaciones del gateway. El resultado de la investigación fue que las aplicaciones instaladas en el servidor Untangle se conectan a la base de datos UVM[30] (Untangle Virtual Machine) y cada una de ellas almacena los registros de configuración y eventos en múltiples tablas.

Después de llevar a cabo la investigación teórica, se da inicio a la parte práctica, la cual consistió en instalar algunas de las aplicaciones libres, con el fin de generar eventos. Estos registros se generaron desde la red interna, con el objetivo de analizar el comportamiento y la relación entre las tablas cuando los eventos se almacenaban. Esta labor puede necesitar del empleo de programas para la gestión de base de datos como Navicat y PgAdmin III.

Una vez identificada la manera cómo se almacenan los registros en la base de datos del servidor de Untangle, se investigó sobre herramientas que permitieron monitorizar archivos de log desde OSSIM. Para la realización de este paso, se consultó a personas expertas en el tema, quienes aconsejaron usar el HIDS OSSEC, porque es una de las herramientas más usadas para el monitoreo de logs y porque tiene compatibilidad con OSSIM. OSSEC hace parte de las herramientas utilizadas para la gestión de seguridad de esta consola.

El agente de OSSEC es una herramienta que se instaló en Untangle, la cual se encarga de monitorizar los cambios que ocurren sobre el archivo de eventos utilizado en la integración de este trabajo. Cuando se añade un registro de algún evento, el agente se encarga de enviar ese cambio hacia el servidor OSSEC, ubicado en OSSIM.

Luego de identificar la herramienta para la detección de intrusos en el equipo, se debía encontrar la manera para registrar los eventos que se almacenaron en la base de datos también a un archivo de texto plano. La solución encontrada fue emplear el uso de *triggers* en la base de datos que almacenan los eventos de las

[30] Núcleo de la plataforma

aplicaciones. El *Trigger* tiene la misión de elaborar una consulta después de registrar un evento, que tiene el formato de log definido en este trabajo y después pasar ésta consulta a una función que escribe sobre el archivo log. Para la escritura desde la base de datos en un archivo se empleó el lenguaje de programación para la base de datos pl/sh, que permite ejecutar comandos Shell.

Después se desarrollar el plugin para detectar y separar la información recibida por el agente de OSSEC. Una vez el servidor de OSSEC recibe el registro de log, es necesario tener una expresión regular, para determinar que la información recibida es de un evento del agente OSSEC de Untangle. Permitiendo así, generar información que luego pueda ser procesada para llevar estadísticas de lo ocurrido en la red de Untangle desde la consola de seguridad.

Finalmente, se realizó un cambio de permiso en el archivo que debe ser monitorizado, considerando que el usuario de Postgres escriba sobre él y el agente de OSSEC realice la lectura.

7. IMPLEMENTACIÓN

Los pasos para lograr la integración entre los eventos ocurridos en las aplicaciones del servidor Untangle y la consola OSSIM se encuentran a continuación. El orden en que aparecen permite obtener mayor claridad sobre el procedimiento a realizar.

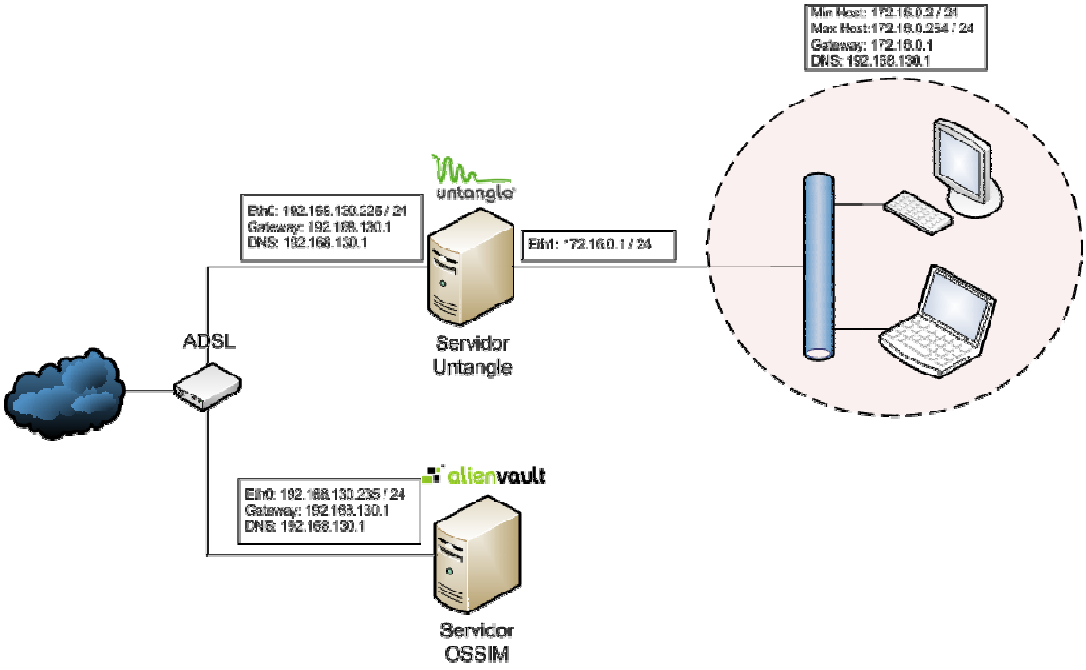
- Instalación y configuración de OSSIM y Untangle
- Creación del archivo de actividad en Untangle
  - Instalación de la aplicación a monitorear en Untangle
  - Configuración de las aplicaciones de Untangle para generar Eventos
  - Información relevante para el archivo de log
  - Relación entre las bases de datos de una aplicación en Untangle
  - Uso de triggers en la base de datos
  - Uso de una función almacenada en la base de datos para escritura del archivo
  - Cambiar permisos del Archivo
- Instalación y configuración del agente OSSEC
  - Instalación y configuración en OSSIM
  - Instalación y configuración en Untangle
- Creación del plugin para OSSIM
  - Inserción de la información del plugin en base de datos de OSSIM
  - Crear un decodificador personalizado en OSSEC
  - Creación de reglas para el plugin
  - Configurar el plugin con el detector

7.1 INSTALACIÓN Y CONFIGURACIÓN DE OSSIM Y UNTANGLE

La instalación y configuración de ambas distribuciones (Untangle y OSSIM) se realiza en máquinas virtuales[31], empleando el programa VMWARE SERVER. La configuración de estos equipos debe permitir el montaje que se encuentra en la Ilustración 7, considerando que estas máquinas se instalarón en equipos diferentes (Anexo A. Instalación de UNTANGLE y Anexo B. Instalación de OSSIM).

[31] Desarrollo de software que permite simular un equipo físico diferente

Ilustración 7 Diagrama de Montaje con las IP



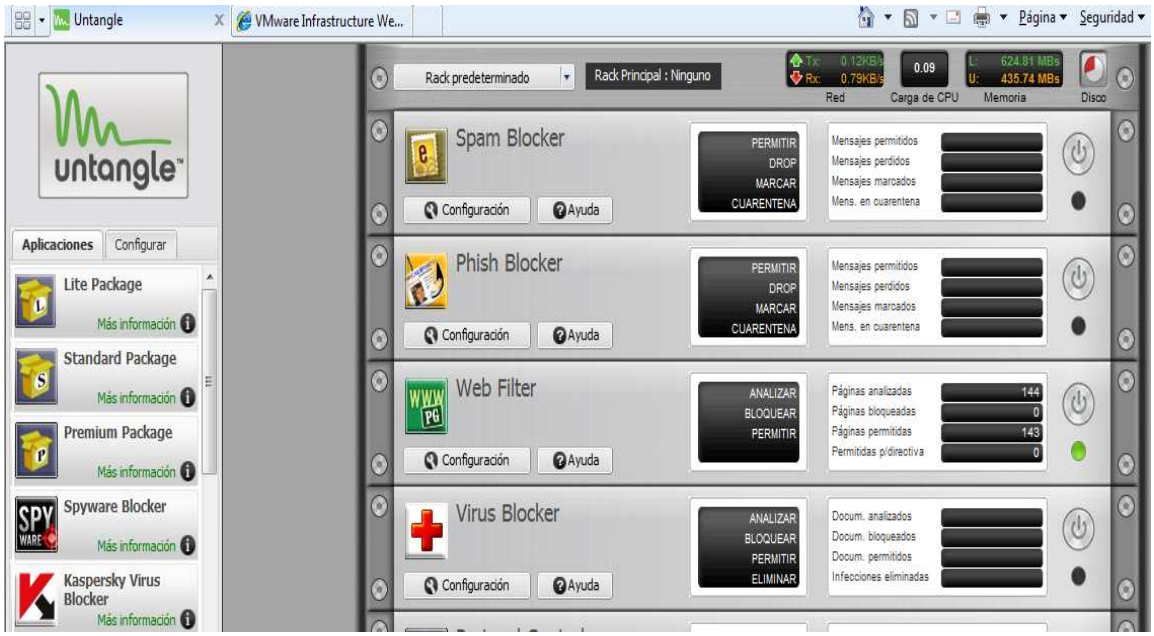
7.2 CREACIÓN DEL ARCHIVO DE ACTIVIDAD EN UNTANGLE

El archivo a monitorear (nombrado ossec-untangle.log) por parte del agente OSSEC es el resultado de una serie de pasos que involucran el estudio del manejo de la información de las aplicaciones de Untangle. Una vez se decide cuál será la información relevante para almacenar, se explica el método para la escritura directa desde la base de datos al archivo de eventos. Finalmente se deben cambiar los permisos del archivo.

7.2.1 Instalación de aplicaciones libres en Untangle.

En el presente trabajo se instalaron y trabajaron tres aplicaciones: Web Filter, Protocol Control y Firewall. El proceso de instalación de una de ellas, *firewall*, se encuentra en el Anexo A**.** Instalación de UNTANGLE, las otras se pueden instalar de manera similar. Este procedimiento consiste en seleccionar la aplicación del lado izquierdo (lista de aplicaciones) de la Ilustración 8 y esperar que se descargue, una vez esta lista para configurar, aparece en el rack (lado derecho).

Ilustración 8 Aplicaciones Untangle



7.2.2 Configuración de las aplicaciones de Untangle para generar Eventos.

Las aplicaciones de Untangle manejan los registros en una base de datos tipo *Postgres*. Cuando se ingresa a una aplicación por el servicio web se puede observar en todas las aplicaciones una tabla de eventos, con información que resulta de gran interés para el administrador del sistema y varía dependiendo de la aplicación.

El archivo de actividad será tratado como un log, y por lo tanto se actualiza cuando se genera un evento en algunas de las aplicaciones activas configuradas en el servidor. La estructura de este archivo tiene ciertos atributos que generaliza las otras aplicaciones de Untangle, cada una de ellas con el mismo nivel de detalle.

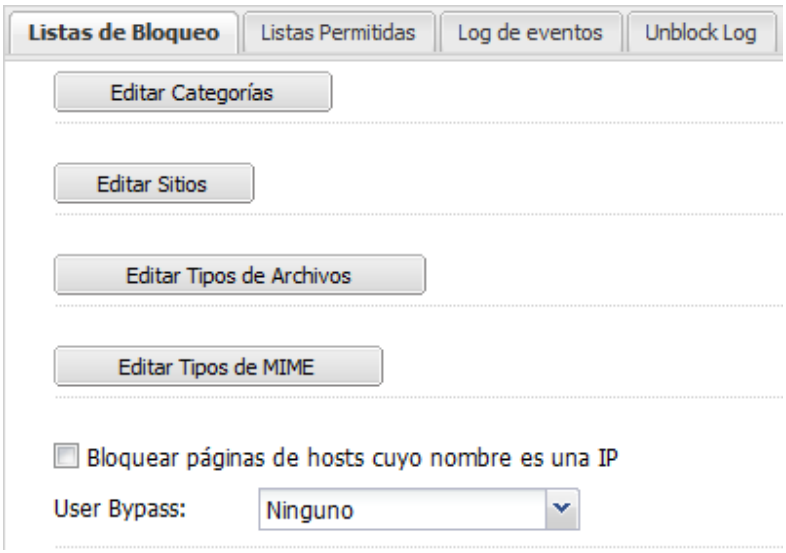
7.2.2.1 Web Filter.

La aplicación web filter fue descargada y se encuentra en el rack predeterminado de Untangle como se observa en la Ilustración 9. En esta ilustración se puede apreciar el botón que activa la aplicación, el funcionamiento está dado por la configuración actual que se encuentre almacenada en la base de datos por omisión.

Ilustración 9 Web Filter

Una vez se presiona el botón Configuración de la aplicación sale el menú de la Ilustración 10.

Ilustración 10 Configuración Web Filter



En la opción Listas de Bloqueo se encuentran las políticas de permisos para los sitios web. Las categorías hacen referencia al tipo de contenido que se encuentra en internet, por ejemplo Pornografía, Deportes, Violencia. En Editar Sitios se debe ingresar la url de la página web a la cual se desea hacer seguimiento o bloquear, por ejemplo se va a bloquear facebook.com, explicando la razón de bloqueo (sitio de red social) de esa página. También se puede hacer bloqueo por tipo de archivos que se pueden abrir desde una página web, por ejemplo jpg, pdf, exe.

También cuando se ha empleado en una página codificación MIME, se puede escoger la codificación para bloquear. Finalmente esta el bloqueo de las páginas que tienen como nombre el número de la dirección IP.

El user bypass sirve para determinar qué usuarios pueden acceder a los sitios web cuando éstos están bloqueados, funciona correctamente cuando se tiene la aplicación Policy Manager.

En el menú Listas Permitidas están los sitios permitidos y los clientes pueden acceder a los sitios bloqueados.

A continuación están los valores del menú eventos (Tabla 2), el cual se trabajará para la realizar el registro a monitorizar de esta aplicación.

Tabla 2 Valores del log de Eventos Web Filter

Timestamp	La fecha en que sucedió el evento
Action	La acción que realizó el servidor de Untangle.
Client	La IP del cliente que hizo la petición.
Request	La url de la petición.
reason for action	El motivo por el que se tomo la acción.
Server	La IP del equipo al cual intento acceder.

Finalmente está el menú Unblock Log, que es donde se registran los eventos que se han permitido, contiene la fecha y hora, el tipo de permiso permanente, el cliente y la solicitud URL.

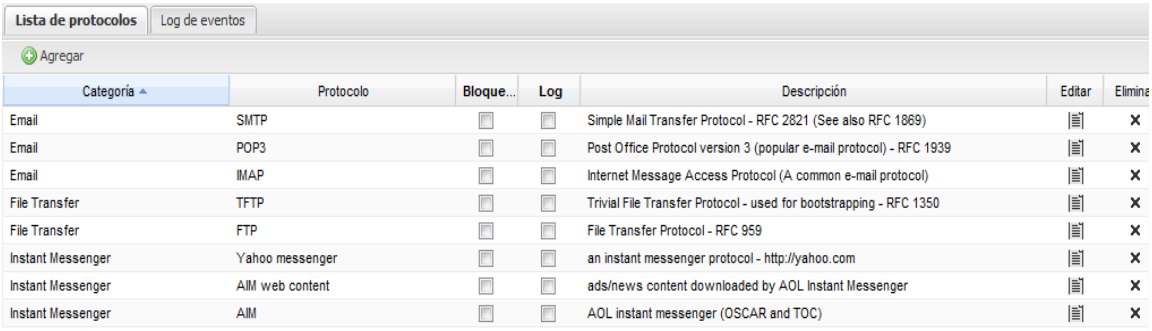
7.2.2.2 Protocol Control

La aplicación Protocol Control se encuentra en el rack predeterminado de Untangle como se observa en la Ilustración 11. Esta tiene un botón que activa la aplicación, el funcionamiento en ese momento estará dado por la configuración actual que se encuentre almacenada en la base de datos.

Ilustración 11 Protocol Control

Esta aplicación trae definida una lista de protocolos y también se pueden adicionar otros. En el presente trabajo se trabaja sobre los protocolos ya existentes, como el de Bittorrent[32], que están localizados en la categoría *PeertoPeer*[33]. La configuración de la aplicación se encuentra en el menú Listas de protocolos (Ilustración 12), en este se puede adicionar un protocolo y luego en la lista seleccionar si se bloquea o solo se guarda el registro (*Log*).

Ilustración 12 Listas del Protocolos de la aplicación Protocol Control



Categoría	Protocolo	Bloque...	Log	Descripción	Editar	Elimina
Email	SMTP			Simple Mail Transfer Protocol - RFC 2821 (See also RFC 1869)		×
Email	POP3			Post Office Protocol version 3 (popular e-mail protocol) - RFC 1939		×
Email	IMAP			Internet Message Access Protocol (A common e-mail protocol)		×
File Transfer	TFTP			Trivial File Transfer Protocol - used for bootstrapping - RFC 1350		×
File Transfer	FTP			File Transfer Protocol - RFC 959		×
Instant Messenger	Yahoo messenger			an instant messenger protocol - http://yahoo.com		×
Instant Messenger	AIM web content			ads/news content downloaded by AOL Instant Messenger		×
Instant Messenger	AIM			AOL instant messenger (OSCAR and TOC)		×

En el menú log de eventos, se encuentran los valores de la Tabla 3.

Tabla 3 Valores del registro de Eventos del Protocol Control

Timestamp	La fecha del evento
Action	La acción que toma el servidor de Untangle
Client	La dirección IP de la fuente del suceso
Request	El protocolo
reason for action	El motivo por el cual se aplicó la acción tomada por Untangle
Server	La dirección IP destino

[32] http://www.bittorrent.com/
[33] Red de nodos que se comportan iguales entre sí, permiten el intercambio de información.

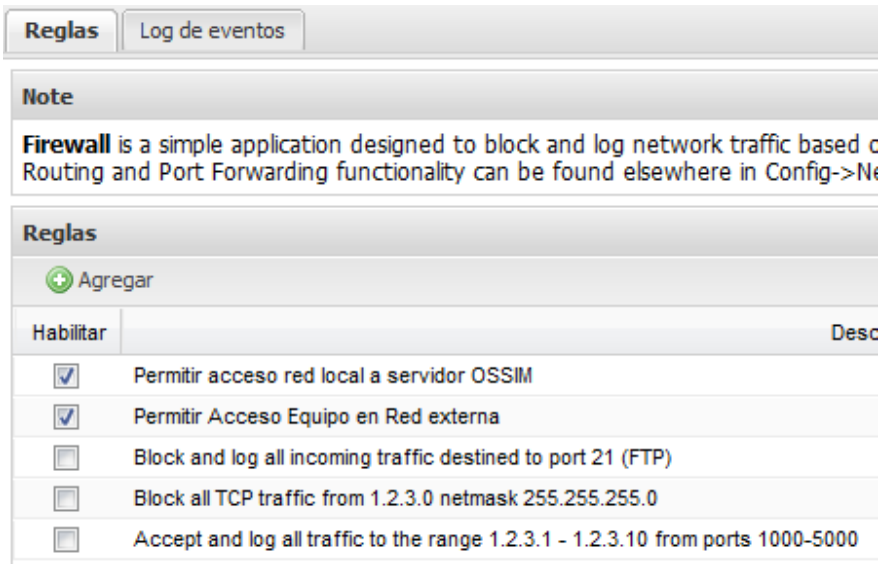
7.2.2.3 Firewall

La aplicación Firewall dentro del rack de Untangle se observa en la Ilustración 13. Esta tiene un botón que activa la aplicación, el funcionamiento en ese momento está dado por la configuración almacenada en la base de datos, siempre trae una por omisión pero que no están habilitadas.

Ilustración 13 Firewall

En la configuración de esta aplicación hay dos opciones en el menú (Ilustración 14), una permite configurar las reglas y la otra muestra los eventos de la aplicación.

Ilustración 14 Configuración de la aplicación Firewall de Untangle



Para adicionar una regla, se debe ir al menú Reglas y presionar el botón Agregar. Cuando se presiona este botón de la aplicación se despliega un menú parecido a la Ilustración 15, la diferencia es que esta vez se encuentra en blanco, porque los datos que aparecen en esta ilustración corresponden a una regla que se definió anteriormente para bloquear el acceso del servidor OSSIM desde la red interna de Untangle. Habilitar la regla, permite que ésta comience a funcionar en el momento de guardar la configuración. La descripción es el texto que sirve para guiar sobre la acción que se está efectuando. La medida es la acción a ejecutar en esta regla, es decir, si permite ó bloquea.

El Log permite que almacene el registro cuando se aplica la regla. Después están los parámetros básicos que se deben configurar en una regla de *firewall*, para controlar el sentido del tráfico permitido.

En la configuración del *firewall* es importante considerar el orden en que se escriben las reglas y si estás estan activas. La regla que este más abajo en la lista es la que primero se ejecuta. En la Ilustración 13, primero se evalúa la regla "Permitir Acceso Equipo en Red Externá", y después se evalúa "Permitir acceso red local a servidor OSSIM".

Ilustración 15 Creación de una regla en la aplicación Firewall

En el menú Event Log se pueden apreciar todos los eventos que han sido causados por una regla que tenga la casilla Log seleccionada. En la Tabla 4 están

los parámetros que se van a considerar en la creación del *log* a monitorear durante la integración con OSSIM con Untangle.

Tabla 4 Valores del Event Log del Firewall

Timestamp	La fecha cuando sucede el evento
Action	La acción que realiza el servidor de Untangle
Client	La IP del origen del tráfico
reason for action	Motivo por el cual se tomo una acción
Server	La ip del servidor a donde se accede

Fuente: Untangle Server User's Guide. Firewall.

7.2.3 Información relevante para el archivo de actividad

El archivo de actividad va tener la capacidad de reunir la información de cada uno de los sucesos que se registra en el menú Log de Eventos de las aplicaciones mencionadas anteriormente (Web Filter, Protocol Control y Firewall). Por este motivo se ha generalizado en un solo archivo la información que registran estas aplicaciones, mostradas en la Tabla 2 , la Tabla 3 y la Tabla 4, que son: fecha en que sucede el evento, acción tomada por el servidor de Untangle, dirección IP de origen, motivo de la acción tomada por el servidor y servidor destino.

De acuerdo con el párrafo anterior y considerando la información más relevante para un administrador de red, se propone que el nuevo archivo este constituido con la siguiente información: espacio, nombre servidor, barra vertical, espacio, nombre aplicación, espacio, barra vertical, espacio, fecha del evento, espacio, barra vertical, espacio, la acción que se tomo, espacio, barra vertical, espacio, la IP origen, espacio, barra vertical, espacio, espacio, barra vertical, espacio, un campo libre que describe el suceso, espacio, barra vertical, espacio, el motivo por el cual se bloqueo, espacio, barra vertical, espacio, IP destino y espacio (Ilustración 15).

Ilustración 16 Estructura propuesta para el registro Log

Servidor: corresponde a un mismo valor, “untangle”. Este valor permite distinguir que los registros corresponden a eventos que son reportados por el agente de ossec ubicado en el servidor de Untangle.

Aplicación: corresponde a la aplicación que genera el evento. Sirve para distinguir una aplicación cuando hay varias que están generando registros sobre el archivo log. Es necesario el nombre para diferenciarlas en la consola de monitorización OSSIM, en especial si se desea manejar la configuración de prioridad y confiabilidad de cada una de ellas por separado.

Fecha Evento: corresponde a la fecha y hora en la cual se generó el evento.

Acción Tomada: los administradores de red no siempre toman acciones restrictivas con respecto las peticiones que realizan sus usuarios, en muchas ocasiones, solamente les interesa tener un registro para monitorear el comportamiento de usuarios. Las acciones resultan ser de dos tipos: en el caso de web filter, Bloqueada (*Blocked* B), asumida como una petición negada, o permitida (*Allow*, A), asumida como aceptada, sin embargo para aplicaciones como firewall y protocol control, este campo corresponde a valores de verdadero (*true*, t) y falso (*false*, f) y la acción depende de la regla configurada descrita en el campo motivo. Por ejemplo:

Motivo: Permitir acceso a 192.168.130.0/24. Acción: true. Corresponde a un registro
Motivo: Bloquear acceso a 192.168.130.0/24. Acción: true. Corresponde a una acción de bloqueo.

IP Origen: corresponde a la IP del equipo, dentro de la red de untangle, para la cual fue registrado el evento.

Campo Libre: Este campo puede ser usado de manera distinta para cada una de las aplicaciones. En Web Filter registrará la url a la cual se hizo la petición de acceso (*www.facebook.com, ww.hi5.com, www.mysocialnetwork.com*). Para Firewall corresponderá al número de la regla que está configurada en el servidor de untangle (regla 1, regla 5, regla 4), recordando que este número corresponde al orden de la ejecución. Para Protocol Control, mostrará la descripción del protocolo que está siendo utilizado.

Motivo: Este campo señala una pequeña descripción, del por qué se está registrando el evento.

IP Destino: Corresponde a la dirección IP hacia la cual se hacen las peticiones desde los equipos de la red local, por lo general corresponderá a una IP por fuera de la red de Untangle.

A continuación se ilustra cómo se almacena el registro propuesto, que queda almacenado en el archivo de log por cada aplicación (Web Filter, Protocol Control y Firewall):

Ilustración 17 Log Web Filter Untangle

```
untangle | web filter | 2010-07-24 11:13:39.135 | B | 172.16.0.15 | www.goooogle.com | Proxy Sites | 72.14.253.104
untangle | web filter | 2010-07-24 11:16:24.879 | B | 172.16.0.15 | www.redtube.com | Pornography | 209.222.138.10
```

Ilustración 18 Log Protocol Control Untangle

```
untangle | protocolcontrol | 2010-11-15 18:51:15.821 | f | 172.16.0.45 | Solicitud:HTTP |  HyperText Transfer Protocol - RFC 2616 | 8.19.240.53
untangle | protocolcontrol | 2010-11-15 18:51:19.438 | f | 172.16.0.45 | Solicitud:HTTP |  HyperText Transfer Protocol - RFC 2616 | 8.19.240.53
```

Ilustración 19 Log Firewall Untangle

```
untangle | firewall | 2010-11-15 17:57:43.061 | t | 172.16.0.45 | regla:2 | Permitir Acceso red local a servidor OSSIM | 192.168.130.235
untangle | firewall | 2010-11-15 17:57:59.395 | t | 172.16.0.45 | regla:2 | Permitir Acceso red local a servidor OSSIM | 192.168.130.235
```

7.2.4 Relación entre las bases de datos de las aplicaciones en Untangle

Ossec es la aplicación que se encargará de chequear cambios del sistema (archivos de texto plano), por tanto se debe desarrollar una metodología que permita que los registros de las aplicaciones no sólo queden registrados en las bases de datos, sino también en un archivo de texto plano, para ser monitoreado por el agente de Ossec, llevándolos finalmente al servidor que se encuentra en la consola de Ossim.

Durante el desarrollo de este proyecto se emplearon dos programas con interfaz grafica, que permiten utilizar las bases de datos de los servidores, MySQL y PostgreSQL de forma fácil, uno es Navicat y el otro es Pgadmin III. Navicat permitió interactuar con ambos sistemas de gestión, mientras PgAdmin III facilitó el uso de Postgres.

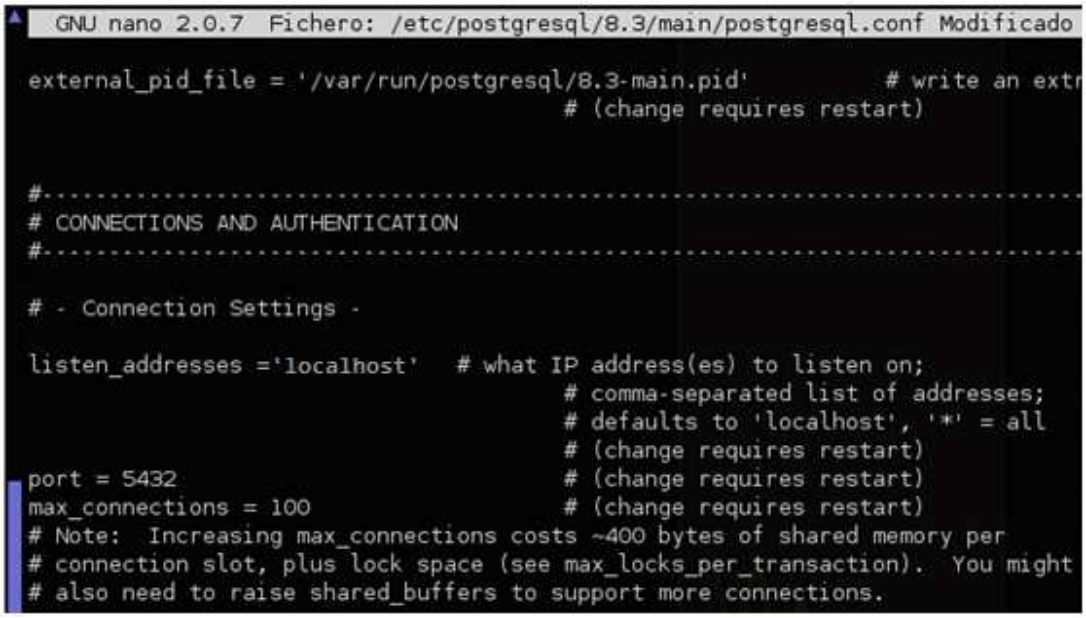
Para poder trabajar remotamente con la base de datos postgres del servidor Untangle se debe habilitar el acceso desde todas las IP[34], cambiando la configuración en los archivos **/etc/postgresql/8.3/main/pg_hba.conf** y

[34] **How to: Access the UVM database remotely. Clave internet:** http://forums.untangle.com/tip-day/12398-how-access-uvm-database-remotely.html

/etc/postgresql/8.3/main/postgresql.conf como se muestra a continuación (Ilustración 20 e Ilustración 21).

Ilustración 20 Nueva configuración archivo pg_hba.conf

Ilustración 21 Nueva Configuración del archivo postgresql.conf



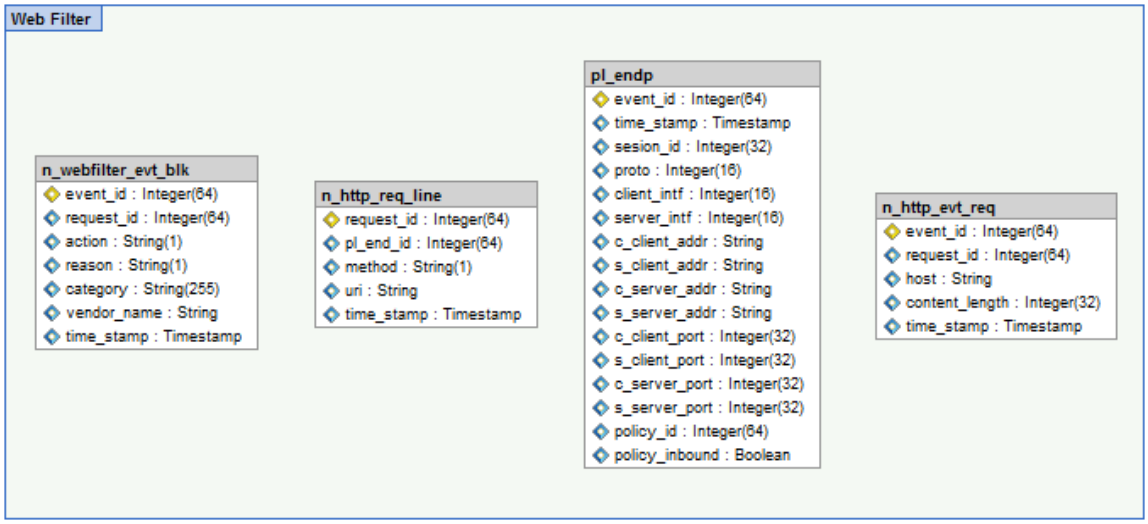
En caso de presentar problemas de conexión con la base de datos, debe revisarse que el puerto 5432 esté habilitado en el firewall del equipo servidor.

La base de datos de la distribución Untangle se encuentra organizada por esquemas. En el esquema llamado **events** se encuentran todos los registros de

tráfico que se han almacenado. En el esquema **settings** se encuentra la configuración almacenada de cada una de las aplicaciones. El nombre de las tablas puede convertirse en una guía para encontrar los registros que corresponden a la aplicación.

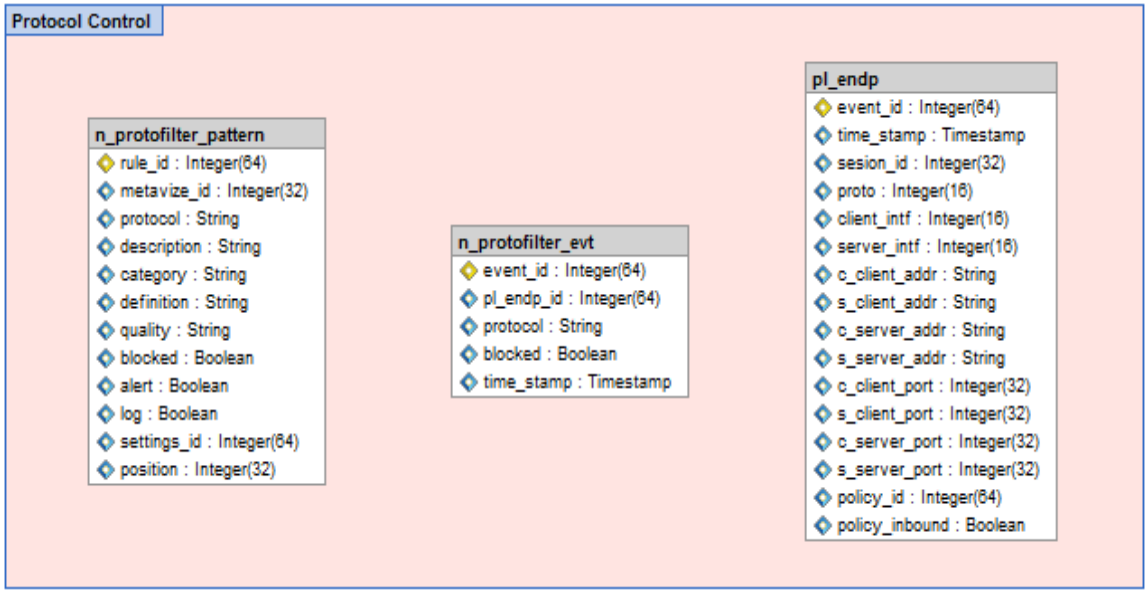
En la Ilustración 22 se encuentran las tablas que contienen la información que se emplea para la elaboración del archivo de log de la aplicación Web Filter. La tabla **n_webfilter_evt_blk** contiene todos los sucesos que han sido bloqueados por la aplicación Web Filter. La tabla **n_http_req_line** lleva el registro de todas las peticiones http que se han realizado, en esta se incluye la dirección url. La tabla **pl_endp** lleva el registro del tráfico, es decir, la dirección IP de origen, los puertos, hasta que llega al destino. La tabla **n_hhtp_req_line** almacena la dirección URI que permite reconocer a que recurso accedió en un host.

Ilustración 22 Tablas de Web Filter



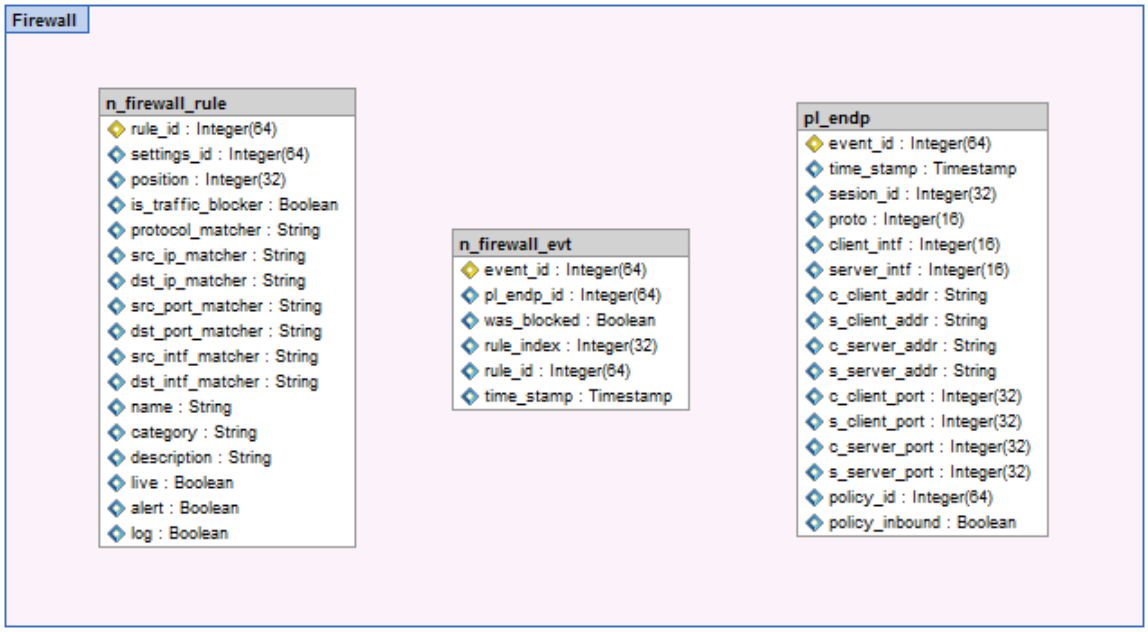
En la Ilustración 23 se encuentran las tablas empleadas para la generación del log de la aplicación Protocol Control de Untangle. La tabla **n_protofilter_pattern** contiene la información de los protocolos que se pueden configurar via web. La tabla **n_protofilter_evt** contiene los eventos registrados y la acción tomada. La tabla **pl_endp** contiene la información del trayecto que tomo el tráfico desde que se origina en el cliente.

Ilustración 23 Tablas de Protocol Control



En la Ilustración 24 se observan las tres tablas utilizadas para obtener la información del registro de la aplicación Firewall de Untangle. Las reglas que se han definido mediante la configuración del Firewall se registrán en la tabla **n_firewall_rule**. La tabla de los eventos que han sido bloqueados o permitidos se encuentra en **n_firewall_evt**. Finalmente la tabla en **pl_endp** se obtiene la información del tráfico que origino el evento.

Ilustración 24 Tablas del Firewall



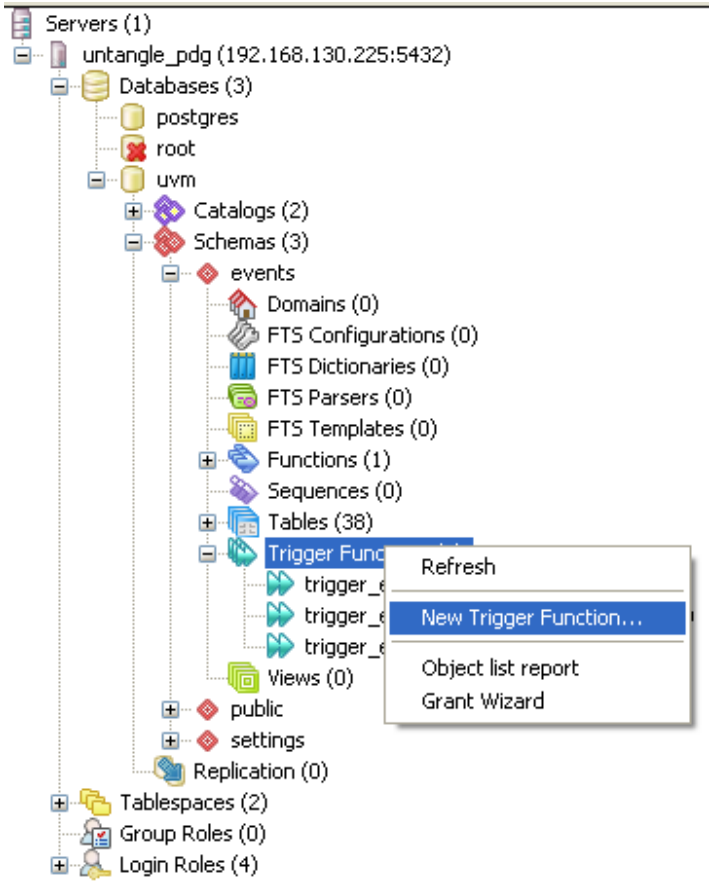
7.2.5 Uso de Triggers en la base de datos

Un trigger es una función almacenada en la base de datos que se ejecuta de forma automática como respuesta a eventos que ocurren en la base de datos. En el proyecto se utiliza un trigger cuando se almacenan eventos en las aplicaciones Web Filter, Protocol Control y Firewall, permitiendo llamar a la función que escribe sobre el archivo de texto plano (archivo de actividad).

Con el programa PgAdmin III se facilita la creación de una función Trigger (Ilustración 25). Se pueden emplear los siguientes valores para la creación de los tres triggers, uno por cada aplicación, variando la definición y el nombre.

Properties	Owner: postgres Language: plpgsql
Options	Volatility: VOLATILE Strict: X Security of definer:X Estimated cost: 100
*Los otros valores se dejan por omisión.	

Ilustración 25 Creación de una función trigger usando PgAdmin III



La declaración de una función en las bases de datos debe comenzar con la palabra declare, a continuación se escriben las variables que se van a utilizar. Las variables definidas son de tipo *text* porque permiten almacenar las cadenas de caracteres que retornan las consultas que se van a ejecutar dentro de la función.

El objetivo del procedimiento que se escribe dentro de los trigger es formar la cadena de log, que va ser escrita en el archivo de actividad ***ossec-untangle.log***. Es por esto que al inicio del código la variable aplicación toma el valor dependiendo de la tabla. En las sentencias SQL se busca asignar cada variable a un campo del registro que está determinado por la sentencia SELECT, las tablas mencionadas anteriormente por cada aplicación se deben poder relacionar en la sentencia FROM, esta relación entre las tablas se hace empleando los identificadores que aparecen en las tablas. Finalmente esta la validación WHERE para comprobar que los datos corresponden al id del nuevo registro que fue almacenado. Al final del procedimiento se concatenan las variables, dando el

formato de log establecido en este proyecto y se le pasa el texto a la función registro_log.

A continuación se encuentra el nombre y la definición que se utilizan para la función Trigger de cada una de las tablas de eventos a monitorizar.

Aplicación:
Web Filter

Nombre:
Trigger_event_webfilterblk_untangle

Definición:

```
declare
servidorQ  text default '';
aplicacionQ text default '';
fechaQ text default '';
accionQ text default '';
ip_origenQ text default '';
causa_eventoQ text default '';
motivo_eventoQ text default '';
ip_destinoQ text default '';
registro_log text default '';

begin

servidorQ = 'untangle';
aplicacionQ = 'web filter';

fechaQ = (SELECT  n_webfilter_evt_blk.time_stamp
FROM n_webfilter_evt_blk JOIN n_http_req_line USING (request_id) JOIN pl_endp ON (n_http_req_line.pl_endp_id=pl_endp.event_id) JOIN n_http_evt_req ON (pl_endp.event_id+1=n_http_evt_req.event_id)
WHERE n_webfilter_evt_blk.event_id=NEW.event_id);

accionQ = (SELECT  n_webfilter_evt_blk.action
FROM n_webfilter_evt_blk JOIN n_http_req_line USING (request_id) JOIN pl_endp ON (n_http_req_line.pl_endp_id=pl_endp.event_id) JOIN n_http_evt_req ON (pl_endp.event_id+1=n_http_evt_req.event_id)
WHERE n_webfilter_evt_blk.event_id=NEW.event_id);

ip_origenQ = (SELECT  pl_endp.c_client_addr
FROM n_webfilter_evt_blk JOIN n_http_req_line USING (request_id) JOIN
```

```
pl_endp ON (n_http_req_line.pl_endp_id=pl_endp.event_id) JOIN n_http_evt_req
ON (pl_endp.event_id+1=n_http_evt_req.event_id)
WHERE n_webfilter_evt_blk.event_id=NEW.event_id);

causa_eventoQ = (SELECT  n_http_evt_req.host
FROM n_webfilter_evt_blk JOIN n_http_req_line USING (request_id) JOIN
pl_endp ON (n_http_req_line.pl_endp_id=pl_endp.event_id) JOIN n_http_evt_req
ON (pl_endp.event_id+1=n_http_evt_req.event_id)
WHERE n_webfilter_evt_blk.event_id=NEW.event_id);

motivo_eventoQ = (SELECT  n_webfilter_evt_blk.category
FROM n_webfilter_evt_blk JOIN n_http_req_line USING (request_id) JOIN
pl_endp ON (n_http_req_line.pl_endp_id=pl_endp.event_id) JOIN n_http_evt_req
ON (pl_endp.event_id+1=n_http_evt_req.event_id)
WHERE n_webfilter_evt_blk.event_id=NEW.event_id);

ip_destinoQ = (SELECT  pl_endp.c_server_addr
FROM n_webfilter_evt_blk JOIN n_http_req_line USING (request_id) JOIN
pl_endp ON (n_http_req_line.pl_endp_id=pl_endp.event_id) JOIN n_http_evt_req
ON (pl_endp.event_id+1=n_http_evt_req.event_id)
WHERE n_webfilter_evt_blk.event_id=NEW.event_id);

registro_log = servidorQ||' | '||aplicacionQ||' | '||fechaQ||' | '||accionQ||' |
'||ip_origenQ||' | '||causa_eventoQ||' | '||motivo_eventoQ||' | '||ip_destinoQ;

perform registro_logs(registro_log);
return null;
end;
```

Aplicación:
Firewall

Nombre:
Trigger_event_firewall_untangle

Definición:

```
declare
servidorQ  text default '';
aplicacionQ text default '';
fechaQ text default '';
accionQ text default '';
ip_origenQ text default '';
```

```
causa_eventoQ text default '';
motivo_eventoQ text default '';
ip_destinoQ text default '';
registro_log text default '';

begin

servidorQ = 'untangle';
aplicacionQ = 'firewall';

fechaQ = (SELECT  n_firewall_evt.time_stamp
FROM n_firewall_rule JOIN n_firewall_evt USING (rule_id) JOIN pl_endp ON
(pl_endp.event_id=n_firewall_evt.pl_endp_id)
WHERE n_firewall_evt.event_id=NEW.event_id);

accionQ = (SELECT  n_firewall_evt.was_blocked
FROM n_firewall_rule JOIN n_firewall_evt USING (rule_id) JOIN pl_endp ON
(pl_endp.event_id=n_firewall_evt.pl_endp_id)
WHERE n_firewall_evt.event_id=NEW.event_id);

ip_origenQ = (SELECT  pl_endp.c_client_addr
FROM n_firewall_rule JOIN n_firewall_evt USING (rule_id) JOIN pl_endp ON
(pl_endp.event_id=n_firewall_evt.pl_endp_id)
WHERE n_firewall_evt.event_id=NEW.event_id);

causa_eventoQ = (SELECT  n_firewall_evt.rule_index
FROM n_firewall_rule JOIN n_firewall_evt USING (rule_id) JOIN pl_endp ON
(pl_endp.event_id=n_firewall_evt.pl_endp_id)
WHERE n_firewall_evt.event_id=NEW.event_id);

motivo_eventoQ = (SELECT  n_firewall_rule.description
FROM n_firewall_rule JOIN n_firewall_evt USING (rule_id) JOIN pl_endp ON
(pl_endp.event_id=n_firewall_evt.pl_endp_id)
WHERE n_firewall_evt.event_id=NEW.event_id);

ip_destinoQ = (SELECT  pl_endp.c_server_addr
FROM n_firewall_rule JOIN n_firewall_evt USING (rule_id) JOIN pl_endp ON
(pl_endp.event_id=n_firewall_evt.pl_endp_id)
WHERE n_firewall_evt.event_id=NEW.event_id);

registro_log = servidorQ||' | '||aplicacionQ||' | '||fechaQ||' | '||accionQ||' |
'||ip_origenQ||' | regla:'||causa_eventoQ||' | '||motivo_eventoQ||' | '||ip_destinoQ;

perform registro_logs(registro_log);
```

```
return null;
end;
```

Aplicación:
Protocol Control

Nombre:
Trigger_event_protocolcontrol_untangle

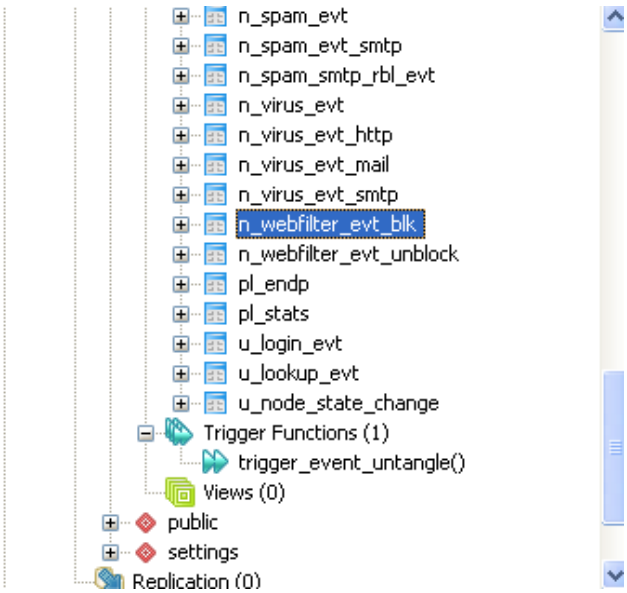
Definición:

```
declare
servidorQ  text default '';
aplicacionQ text default '';
fechaQ text default '';
accionQ text default '';
ip_origenQ text default '';
causa_eventoQ text default '';
motivo_eventoQ text default '';
ip_destinoQ text default '';
registro_log text default '';

begin

servidorQ = 'untangle';
aplicacionQ = 'protocolcontrol';

fechaQ = (SELECT  n_protofilter_evt.time_stamp
FROM n_protofilter_pattern JOIN n_protofilter_evt USING (protocol) JOIN
pl_endp ON (pl_endp.event_id=n_protofilter_evt.pl_endp_id)
WHERE n_protofilter_evt.event_id=NEW.event_id);

accionQ = (SELECT  n_protofilter_pattern.blocked
FROM n_protofilter_pattern JOIN n_protofilter_evt USING (protocol) JOIN
pl_endp ON (pl_endp.event_id=n_protofilter_evt.pl_endp_id)
WHERE n_protofilter_evt.event_id=NEW.event_id);

ip_origenQ = (SELECT  pl_endp.c_client_addr
FROM n_protofilter_pattern JOIN n_protofilter_evt USING (protocol) JOIN
pl_endp ON (pl_endp.event_id=n_protofilter_evt.pl_endp_id)
WHERE n_protofilter_evt.event_id=NEW.event_id);

causa_eventoQ = (SELECT  n_protofilter_pattern.protocol
```

```
FROM n_protofilter_pattern JOIN n_protofilter_evt USING (protocol) JOIN pl_endp ON (pl_endp.event_id=n_protofilter_evt.pl_endp_id)
WHERE n_protofilter_evt.event_id=NEW.event_id);

motivo_eventoQ = (SELECT  n_protofilter_pattern.description
FROM n_protofilter_pattern JOIN n_protofilter_evt USING (protocol) JOIN pl_endp ON (pl_endp.event_id=n_protofilter_evt.pl_endp_id)
WHERE n_protofilter_evt.event_id=NEW.event_id);

ip_destinoQ = (SELECT  pl_endp.c_server_addr
FROM n_protofilter_pattern JOIN n_protofilter_evt USING (protocol) JOIN pl_endp ON (pl_endp.event_id=n_protofilter_evt.pl_endp_id)
WHERE n_protofilter_evt.event_id=NEW.event_id);

registro_log =servidorQ||' | '||aplicacionQ||' | '||fechaQ||' | '||accionQ||' | '||ip_origenQ||' | Solicitud:'||causa_eventoQ||' | '||motivo_eventoQ||' | '||ip_destinoQ;

perform registro_logs(registro_log);
return null;
end;
```

Una vez se crea la función trigger se debe relacionar con la tabla correspondiente: n_webfilter_evt_blk, n_firewall_evt y n_protofilter_evt. Este procedimiento se puede realizar desde el programa pgAdmin III como se explica a continuación.

Ilustración 26 Utilización de PgAdmin III para asociar el trigger a la tabla



Haciendo clic derecho sobre una tabla se puede adicionar el Trigger (Ilustración 26), en la opción New Object --> New Trigger. Se indica cual es la función del trigger que se asocia, el nombre de la esa relación y que debe ejecutarse después de la inserción de un registro sobre la tabla (Ilustración 27). Este procedimiento se debe repetir por cada tabla que almacena los eventos de cada aplicación, por ejemplo la tabla de la aplicación web filter que se observa en la Ilustración 28.

Ilustración 27 Asociar función trigger a una tabla

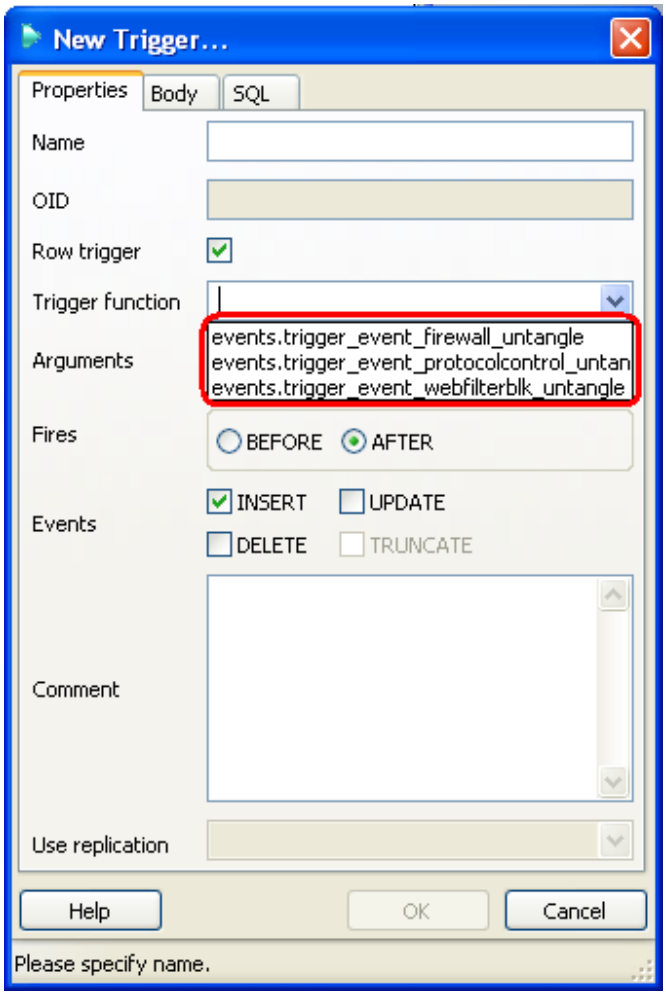
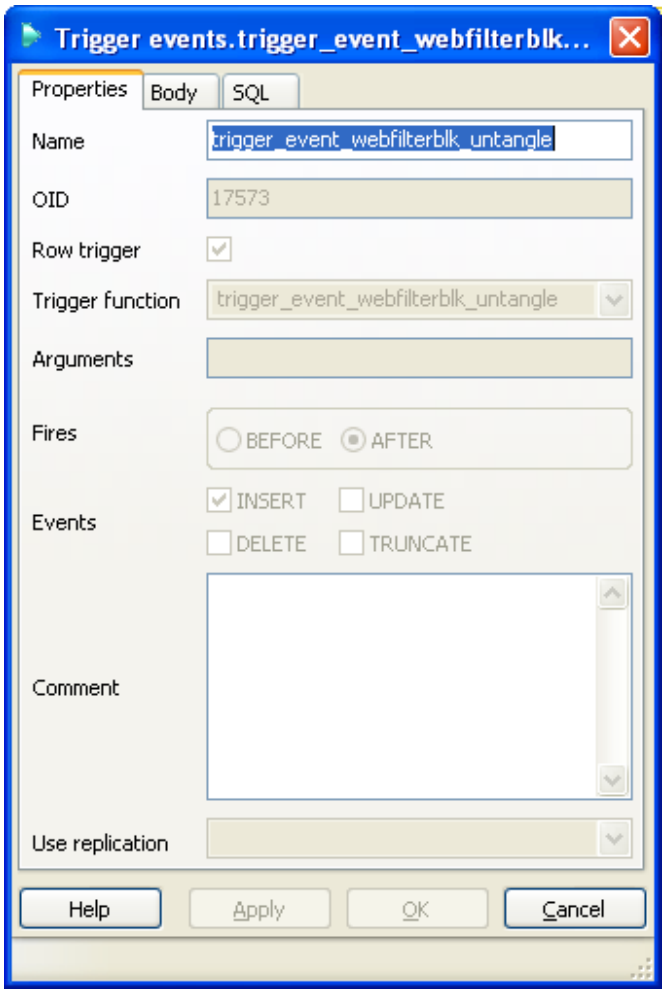


Ilustración 28 Asociación de la función del Trigger a la tabla n_webfilter_evt_blk



7.2.6 Función almacenada en la base de datos para escritura del archivo

Para almacenar en el archivo de registro de actividades la información de un evento que se almacena en la base de datos UVM de Untangle, se puede emplear el manejador de lenguaje llamado PL/sh. Este manejador permite escribir funciones en lenguaje de programación Shell, lo cual admite adicionar las variables por cada trigger de una tabla sobre el archivo.

Untangle por omisión no trae este lenguaje de programación habilitado en su base de datos UVM, por esto es necesario instalarlo. En el Anexo D. Instalación de pl/sh se explica el procedimiento.

Esta función es la encargada de recoger la información proveniente de los triggers definidos y escribirla sobre el archivode actividades. Los parámetros para crear la función utilizando el gestor Pgadmin III son:

Properties	Name: registro_logs Owner: postgres Return type: void Language: plsh
Options	Volatility: VOLATILE Estimated cost:100
Parameters	Type: character Mode: IN Name: evento
Definition	#!/bin/sh echo " "$1" ">> /var/log/untangle-ossec-eventos/ossec-untangle.log
Privileges	User/Group: public Privileges: X
*Los otros valores se dejan por omisión.	

7.2.7 Permisos del Archivo

Ahora se necesita que el log de eventos para las aplicaciones de Untangle quede con permisos de escritura y lectura. El usuario de la base de datos (Postgres) debe poder escribir, y el agente de OSSEC leer.

Para lo anterior se crea una carpeta en Untangle, donde se almacene el log que se necesitan para la integración (OSSIM y Untangle), llamada ***untangle-ossec-eventos*** en el directorio ***/var/log/***. En la nueva carpeta debe estar el archivo que se revisa constantemente (***ossec-untangle.log***). A continuación están los comandos que se deben ejecutar en el servidor donde está instalado Untangle.

cd /var/log/
mkdir untangle-ossec-eventos
cd untangle-ossec-eventos
touch ossec-untangle.log
chmod 755 ossec-untangle.log

7.3 INSTALACIÓN DEL AGENTE OSSEC

La comunicación entre el equipo de Untangle y OSSIM se realizará por medio del HID OSSEC. El cual consta de un agente que monitorea archivos en un equipo, y al detectar una anomalía la envía al servidor OSSEC. Este servidor se ubica en el equipo de OSSIM y el agente en el equipo de Untangle.

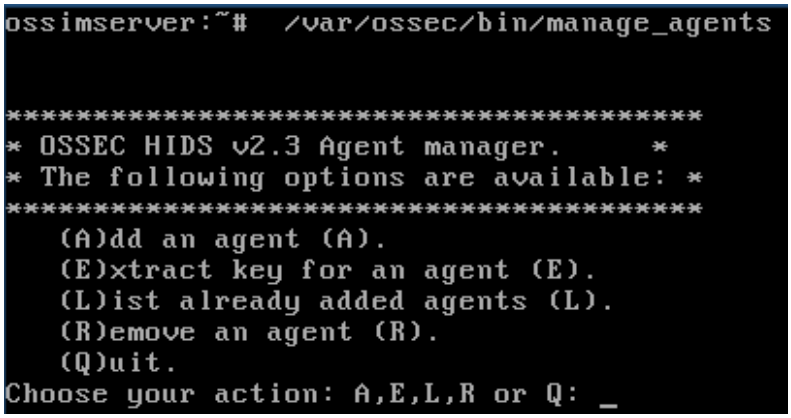
7.3.1 Adicionar un Agente OSSEC en el servidor OSSIM

Para adicionar un agente Ossec (Untangle) en la consola de Ossim, se debe utilizar el comando:

/var/ossec/bin/manage-agents

Este comando despliega un menú de opciones (Ilustración 29), seleccionar la opción A para adicionar un agente.

Ilustración 29 Menú Servidor OSSEC

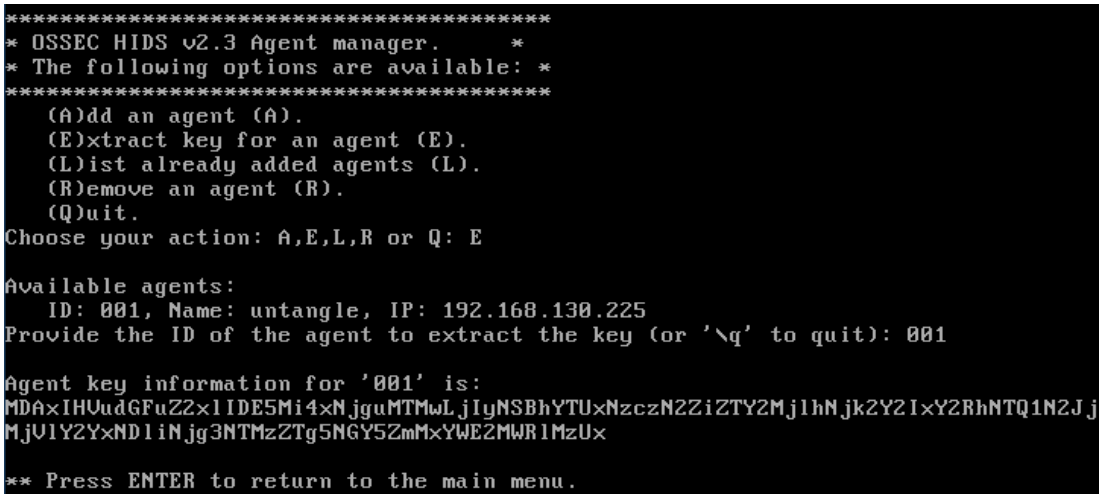


Ingresar la información del agente que será instalado en el servidor Untangle (IP, Nombre e Id) que permita identificarlo de otros agentes OSSEC que estén ejecutándose en la misma red. Finalmente confirmar la adición del nuevo agente.

El agente aún no está asociado con Ossim ni funcionando en Untangle, para esto es necesario realizar el procedimiento de instalación del agente en Untangle para después realizar el procedimiento de enlazar el agente al servidor por medio de una clave.

En el mismo menú de la Ilustración 29 seleccionar la opción E para extraer la clave del nuevo agente. Esta opción solicita el ID del agente al cual se le va generar una clave, en este caso 001. Esta clave (Ilustración 29) debe ser ingresada en el agente Ossec de Untangle para establecer la conexión.

Ilustración 30 Clave para el Agente OSSEC que será instalado en Untangle



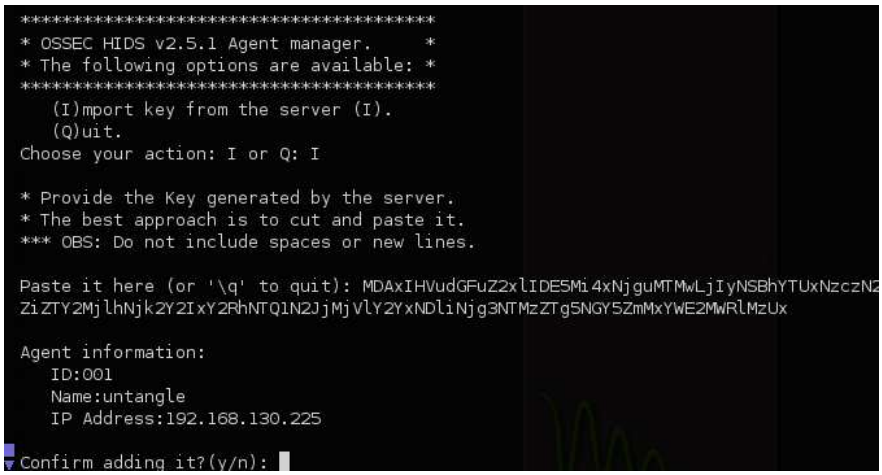
7.4 INSTALACIÓN DEL AGENTE OSSEC EN UNTANGLE

La instalación del agente OSSEC en Untangle se explica en el Anexo C. Instalación de OSSEC en Untangle. Una vez el servidor OSSIM da la clave de esté agente (Ilustración 30), se debe escribir en la consola de Untangle, después de ejecutar el siguiente comando:

/var/ossec/bin/manage_agents

Escoger la opción (I) para ingresar la clave mencionada en el párrafo anterior. Cuando la clave es correcta el sistema muestra la identificación del agente que estaba en el servidor (id, nombre e IP) con el fin de confirmar la adición del nuevo agente al sistema de OSSIM.

Ilustración 31 Ingreso de la clave del agente OSSEC en Untangle



Al confirmar la adición del agente en Untangle se puede iniciar por medio del comando:

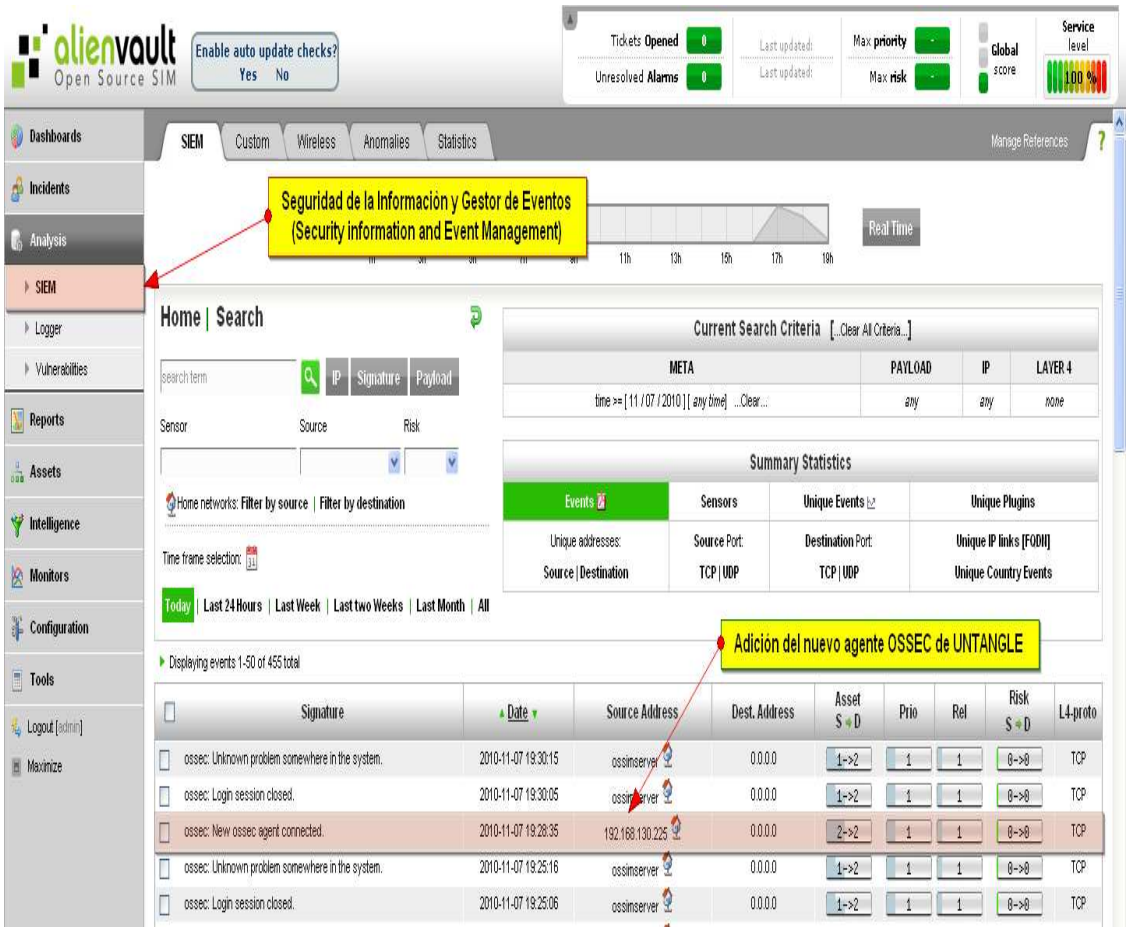
/var/ossec/bin/ossec-control start

También es necesario reiniciar el servidor de OSSEC (equipo OSSIM) luego de ingresar la clave en el agente Ossec de Untangle. Por medio del comando:

/var/ossec/bin/ossec-control start

Cuando la integración del agente de OSSEC se lleva sin errores, el inicio del agente se puede visualizar en la consola de OSSIM, por medio web, acceder en Analysis --> SIEM (Ilustración 32).

Ilustración 32 Visualización inicio de Agente OSSEC



Ahora se debe configurar el archivo de los eventos de las aplicaciones del servidor Untangle (ossec-untangle.log) para ser monitoreado por el agente OSSEC. Para esto, modificar el archivo de configuración del agente en Untangle que se encuentra en ***/var/ossec/etc/*** y se llama ***ossec.conf***. El nuevo código se puede insertar al final del archivo para llevar un orden, antes de </ossec_config> que es donde se encuentra toda la configuración, como se ilustra a continuación (Ilustración 33). Una vez terminado este proceso se pasa a la elaboración del plugin que interpreta ese log en OSSIM.

Ilustración 33 Modificación del archivo OSSEC.conf en Untangle

```
  <!-- Nuevo Untangle -->
  <localfile>
        <log_format>syslog</log_format>
        <location>/var/log/untangle-ossec-eventos/ossec-untangle.log</location>
  </localfile>

</ossec_config>
```

Es aconsejable reiniciar el agente de OSSEC en Untangle:

/var/ossec/bin/ossec-control restart

7.5 CREACIÓN DE UN PLUGIN PARA OSSIM

7.5.1 Inserción de información del plugin en base de datos

Para el desarrollo de un plugin de detección en OSSIM, es necesario conocer cuál es el formato de los logs del equipo externo (Untangle). El agente OSSEC instalado en Untangle, estará verificando las modificaciones que se realicen sobre el archivo de log y una vez modificado, con la adición de un nuevo registro, dicho agente enviara la línea del evento al servidor de OSSEC ubicado en OSSIM para ser interpretado.

Un plugin en OSSIM es un archivo que contiene reglas de selección definidas con expresiones regulares en lenguaje Python, las cuales permiten obtener información sobre el registro al separar los datos relevantes para la identificación del evento.

Para mantener los eventos de los plugins estandarizados, OSSIM define dos valores importantes para cada uno:

- Plugin ID: es el identificador del plugin que genera el evento.
- Plugin SID: es para identificar un evento en particular dentro del plugin.

El plugin que se desarrolló contiene tres de las aplicaciones del Gateway Untangle, por tanto, debe tener un ID y cada una de las aplicaciones se identificara a través de un SID.

Para el desarrollo de nuevos plugins, OSSIM cuenta con un rango de ID ya utilizados, desde el 1000 hasta el 7096, se hará uso de uno de los que se encuentre disponible, aunque se recomienda usar el rango entre 9000 y 10000.

El plugin ID será 7777 e identificará los eventos de las aplicaciones de Untangle como “Untangle: Gateway Appliance”. Dentro del plugin se distinguirán tres eventos particulares, correspondientes a cada una de las aplicaciones y también se debe definir el nivel de prioridad y confiabilidad, estos últimos dos parámetros podrán ser redefinidos posteriormente desde la consola web de OSSIM.

Una vez se tiene definido el identificador del plugin, sus eventos y los valores de prioridad y confiabilidad, se debe crear un script SQL con las sentencias que permitan ingresar la información del plugin en la base de datos de OSSIM (Tabla 5).

Tabla 5 Configuración eventos del plugin Untangle para untangle.sql

Aplicación	SID	Descripción	Prioridad	Confiabilidad
Evento genérico	100185	Evento genérico de Untangle.	1	1
Web Filter	100186	Navegación a sitios indebidos.	2	2
Firewall	100187	Registro de reglas firewall	2	2
Protocol Control	100188	Registro de protocolos	2	2

El directorio **/usr/share/doc/ossim-mysql/contrib/plugins** contiene los archivos con las sentencias de identificación de cada uno de los plugins de OSSIM. Para el proyecto se creará un archivo con nombre **untangle.sql** debe desarrollar las siguientes acciones:

- Eliminar el plugin ID de la tabla “plugin”.
- Eliminar los plugins SID de la tabla “plugin_sid”
- Insertar el nuevo plugin ID y la información en la tabla “plugin”.
- Insertar los nuevos plugins SID y la información en la tabla “plugin_sid”

A continuación se muestra la configuración del script **untangle.sql**

```
-- Untangle
--plugin_id: 7777

DELETE FROM plugin WHERE id = "7777";
DELETE FROM plugin_sid where plugin_id = "7777";

INSERT INTO plugin (id, type, name, description) VALUES (7777, 1, 'untangle', 'Untangle: Gateway Appliance');

INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, name, priority, reliability) VALUES (7777, 100185, NULL, NULL, 'Untangle: Evento generico de Untangle', 1, 1);

INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, name, priority, reliability) VALUES (7777, 100186, NULL, NULL, 'Untangle: Navegacion a sitios indebidos', 2, 2);

INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, name, priority, reliability) VALUES (7777, 100187, NULL, NULL, 'Untangle: Registro de reglas firewall.', 2, 2);

INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, name, priority, reliability) VALUES (7777, 100188, NULL, NULL, 'Untangle: Registro de protocolos.', 2, 2);
```

Una vez creado el script y ubicado en el directorio ***/usr/share/doc/ossim-mysql/contrib/plugins***, se debe ingresar a la base de datos para proceder a ejecutar las sentencias del script untangle.sql.

Para ingresar a la base de datos se ejecuta, en la consola de ossim, el comando:

ossim-db

Seguidamente se selecciona la tabla a utilizar:

use ossim;

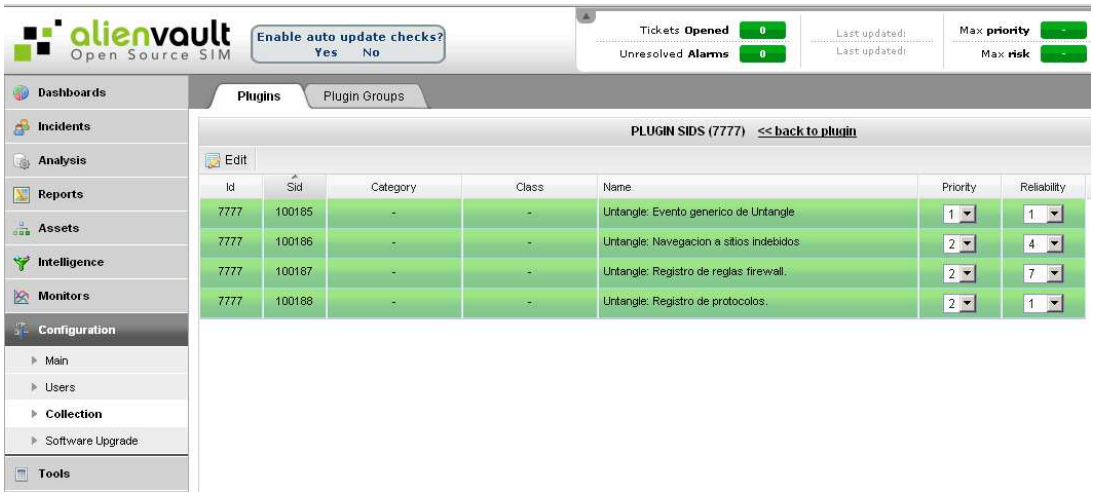
Se importa el archivo que contiene las sentencias SQL:

source /usr/share/doc/ossim-mysql/contrib/plugins/untangle.sql;

Una vez se ha ejecutado el script, se puede verificar su correcta inserción en la base de datos de plugins de OSSIM, accediendo a la consola web en

Configuration > Collections. Para este caso se debe buscar el ID 7777 que corresponde al ID del plugin de Untangle

Ilustración 34 Plugins de UNTANGLE



7.5.2 Crear un decodificador personalizado en OSSEC

El servidor de OSSEC (ubicado en OSSIM) utiliza los decodificadores para analizar los logs que son recibidos por el agente de OSSEC. Los decodificadores consisten en expresiones regulares, que una vez coinciden con un registro, envían la información a un archivo de reglas (**untangle_rules.xml**), también configurado en OSSEC, que se mostrará posteriormente.

La expresión regular empleada para procesar los registros de log generados por las aplicaciones de Untangle está a continuación, y se explica en el Anexo E. Expresiones regulares

```
^\s+untangle\s+\|\s+.*?\|\s+\d+-\d+-\d+\s+\S+\s+\|\s
+(\S+)\s+\|\s+ (\S+)\s+\|\s+(.*?)\|\s+(.*?)\|\s+(\S+)\s$
```

El decodificador, además de la expresión regular, debe tener designado un nombre, para este *plugin* se utilizará el nombre "**untangle-alert**".

Se debe adicionar el siguiente fragmento de código al final del archivo **/var/ossec/etc/decoder.xml**[35]**.** De coincidir un log con la expresión regular, se generará un evento con la información "**after_prematch**" que será examinado por las reglas (**untangle_rules.xml**) definidas para el plugin. Cada uno de los parametros escogidos en el decodificador corresponde a cada uno de los parametros que está en el registro de actividad.

```
<decoder name="untangle-alert">
<prematch>^\s+untangle\s+</prematch>
<regex offset="after_prematch">^\|\s+.*?\|\s+\d+-\d+-\d+\s+\S+\s+\|\s+(\S+)
\s+\|\s+(\S+)\s+\|\s+(\S+)\s+\|\s+(.*?)\|\s+(\S+)$</regex>
<order>action, srcip, url, extra_data, dstip</order>
</decoder>
```

Ilustración 35 Archivo decoder.xml

7.5.3 Creación de las reglas para el plugin

Una vez se ha generado un evento, a raíz del log reportado por el agente de OSSEC, se hace necesario pasar a un segundo nivel, en donde se comparará con las reglas definidas para diferenciar cada una de las aplicaciones. En el momento en que una de las reglas detecta que hay coincidencia con su definición, se generará una alerta que será normalizada para ser analizada por el agente de OSSIM.

[35] Create Custom decoder and rules. Clave internet: http://www.ossec.net/doc/manual/rules-decoders/create-custom.html

Se crea un archivo **untangle_rules.xml** con el contenido a continuación y se debe ubicar en el directorio **/var/ossec/rules/**

```
<!-- @(#) $Id: untangle_rules.xml,v 1.0 2010/08/01 16:21:07 dcid Exp $ -->
<!-- Untangle Log messages -->
<group name="untangle,">

  <rule id="100185" level="0">
    <decoded_as>untangle-alert</decoded_as>
    <description>mensajes de Untangle.</description>
  </rule>

  <rule id="100186" level="10">
    <if_sid>100185</if_sid>
    <match>web filter</match>
    <description>Navegacion a sitios indebidos.</description>
  </rule>

   <rule id="100187" level="10">
    <if_sid>100185</if_sid>
    <match>firewall</match>
    <description>Registro de reglas firewall.</description>
  </rule>

   <rule id="100188" level="10">
    <if_sid>100185</if_sid>
    <match>protocolcontrol</match>
    <description>Registro de protocolos.</description>
  </rule>

</group>
 <!-- Untangle --> <!-- EOF -->
```

Al crear el archivo de las reglas, es necesario incluirlas en el archivo de configuración de OSSEC. Para esto se adiciona la línea que se muestra a continuación al final de la lista de reglas del archivo **/var/ossec/etc/ossec.conf**.

<include>untangle_rules.xml</include>

Esto se puede ver en la siguiente ilustración:

Ilustración 36 Archivo de configuración OSSEC-OSSIM

7.5.4 Configurar plugin al detector

Para crear el plugin de Untangle en el agente de OSSIM se debe definir en la carpeta **/etc/ossim/agent/plugins/** un archivo con nombre **untangle.cfg**, este contendrá toda la información de configuración del plugin, la cual se muestra a continuación:

Nota: Las líneas precedidas por numeral (#) son comentarios, que se han agregado para explicar los contenidos.

```
[DEFAULT]
#El plugin_id corresponde al identificador del plugin ante el servidor.
plugin_id=7777

#El plugin untangle es un detector ya que constantemente estará
#leyendo el archivo de log, buscando patrones definidos en los
#nuevos registros.

[config]
type=detector
```

```
enable=yes

#El archivo a monitorear es un archivo de log, se debe indicar
#la ubicación de este archivo.

source=log
location=/var/ossec/logs/alerts/alerts.log

# La variable create_file se le indica si crea o no el archivo de
#log en caso de que no exista
# en caso de que no exista, el plugin no será procesado

create_file=false

process=ossec-logcollector
#Iniciar el proceso del plugin cuando el agente se Ossim se inicia
start=yes   ;
#Apagar el proceso del plugin cuando el agente se Ossim se detiene
stop=yes    ;

#Reiniciar el proceso del plugin en un interval de tiempo
restart=no  ; restart plugin process after each interval
#Intervalo de tiempo para reiniciar el plugin
restart_interval=_CFG(watchdog,restart_interval) ;

startup=/etc/init.d/ossec start
shutdown=/etc/init.d/ossec stop

#Cada uno de los eventos que puede generar un plugin
#debe tener una identificación única en la base de
#datos, a continuación se asocian cada uno de los eventos al
#plugin de untangle

[translation]
100185=7777
100186=7777
100187=7777
100188=7777

#La expresión regular que decodifica la alerta generada por el  #servidor de ossec, la parte señalada con gris coincide con la #expresión regular explicada en el ANEXO E.

[1-untangle-Untangle-regla]
```

```
event_type=event
regexp=Alert\s+\d+.*\n(?P<date>\d+\s+\w+\s+\d+\s+\d+:\d+:\d+)\s+.*\s(?P<sensor
>.*)-
\>.*\nRule:\s+(?P<plugin_sid>\d+)\s+.*\nSrc\s+IP:\s+(?P<src_ip>.*)\nUser:\s+(?P<
user>.*)\n\s+untangle\s+\|\s+(?P<aplicacion>\S+.*?)\|\s+(?P<fecha>\d+-\d+-
\d+\s+\S+)\s\|\s+(?P<accion>\S+?)\s\|\s+(?P<iporigen>\S.*?)\s\|\s+(?P<campouno
>\S+.*?)\|\s+(?P<motivolog>\S+.*?)\|\s+(?P<ipdestino>\S.*?)\s\n


#Ejemplo Alerta generadas por el OSSEC Server para logs de
#Untangle:

#** Alert 1292065240.22674: mail  - untangle,
#2010 Dec 11 06:00:40 (untangle) 192.168.130.225->/var/log/untangle-ossec-
eventos/ossec-untangle.log
#Rule: 100186 (level 10) -> 'Navegacion a sitios indebidos.'
#Src IP: (none)
#User: (none)
# untangle | web filter | 2010-12-11 21:26:34.402 | B | 172.16.0.181 | proxy.org |
Proxy Sites | 74.53.126.157

date={normalize_date($date)}
sensor={resolv($sensor)}
src_ip={$iporigen}
dst_ip={$ipdestino}
plugin_id={translate($plugin_sid)}
plugin_sid={$plugin_sid}
#username={$user}

#Los userdata son datos adicionales que puede incluir un plugin, y
#que son específicos al mismo. En esta ocasión se utilizan para
#ubicar  información contenida en los logs de Untangle.

userdata1={$aplicacion}
userdata2={normalize_date($fecha)}
userdata3={$accion}
userdata4={$iporigen}
userdata5={$campouno}
userdata6={$motivolog}
userdata7={$ipdestino}
```

Se debe aclarar que la expresión regular contenida en el plugin es la modificación a la estructura de la configuración del plugin de ossec ubicado en en el agente de OSSIM en **/etc/ossim/agent/plugins/ossec.cfg.**

El agente de OSSIM debe cargar la configuración del nuevo plugin, por tanto se debe modificar el archivo de configuración **/etc/ossim/agent/config.cfg** debajo de la etiqueta [plugins] se adiciona la siguiente línea:

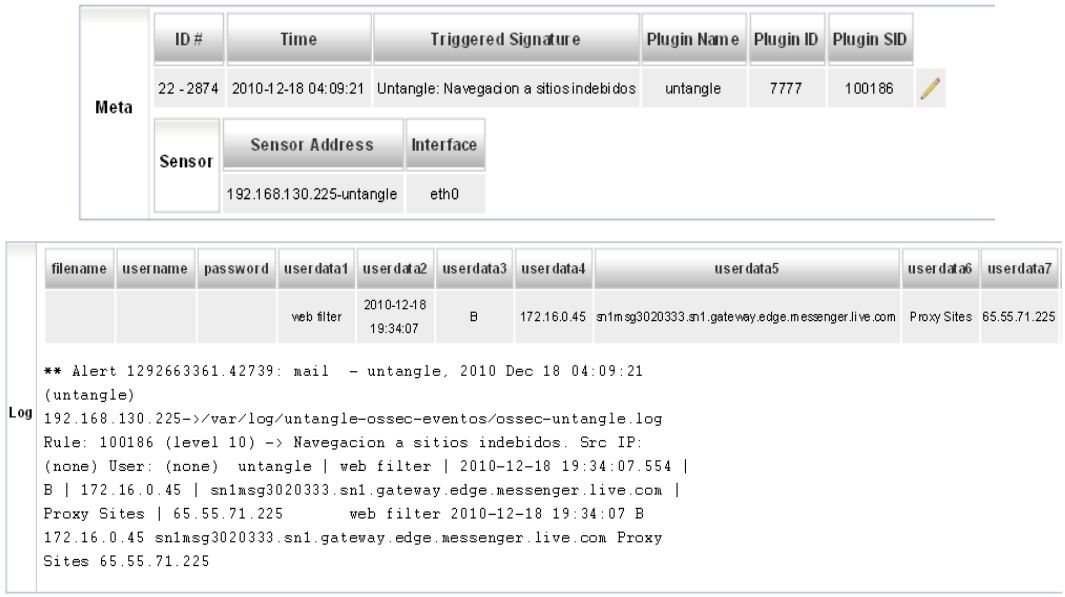
untangle=/etc/ossim/agent/plugins/untangle.cfg

Para finalizar la instalación se debe reiniciar el servidor de OSSIM, esto permitirá iniciar de nuevo todos sus servicios y la cargar la nueva información de configuración del nuevo plugin de Untangle. Se utiliza el comando:

shutdown 0

Cada vez que se genere un registro en las aplicaciones de Untangle, podrá visualizarse de la siguiente manera en la consola web de OSSIM.

Ilustración 37 Registro de log en la consola OSSIM



Meta

ID #	Time	Triggered Signature	Plugin Name	Plugin ID	Plugin SID
22 - 2874	2010-12-18 04:09:21	Untangle: Navegacion a sitios indebidos	untangle	7777	100186

Sensor	Sensor Address	Interface
	192.168.130.225-untangle	eth0

filename	username	password	userdata1	userdata2	userdata3	userdata4	userdata5	userdata6	userdata7
			web filter	2010-12-18 19:34:07	B	172.16.0.45	sn1msg3020333.sn1.gateway.edge.messenger.live.com	Proxy Sites	65.55.71.225

Log

```
** Alert 1292663361.42739: mail  - untangle, 2010 Dec 18 04:09:21
(untangle)
192.168.130.225->/var/log/untangle-ossec-eventos/ossec-untangle.log
Rule: 100186 (level 10) -> Navegacion a sitios indebidos. Src IP:
(none) User: (none)  untangle | web filter | 2010-12-18 19:34:07.554 |
B | 172.16.0.45 | sn1msg3020333.sn1.gateway.edge.messenger.live.com |
Proxy Sites | 65.55.71.225       web filter 2010-12-18 19:34:07 B
172.16.0.45 sn1msg3020333.sn1.gateway.edge.messenger.live.com Proxy
Sites 65.55.71.225
```

8. RESULTADOS

Al finalizar el trabajo se logró la centralización de la información sobre los eventos que reportan algunas de las aplicaciones Open Source de la plataforma Untangle en un solo equipo (OSSIM), el cual no es el encargado directo de la generación de dichos sucesos.

La integración llevada a cabo resulta ser transparente para las aplicaciones open source del software untangle, ya que solo se le adicionaron elementos (funciones y triggers) al funcionamiento de la base de datos sin afectar su rendimiento ni configuración alguna de la plataforma Untangle.

Con la implementación del plugin, se espera que otras personas puedan seguir desarrollando integraciones relacionadas con la seguridad informática con respecto a otras aplicaciones del Gateway Untangle u otros servidores de aplicaciones, encontrando en ese trabajo un paso a paso de los procedimientos que se deben seguir para lograr este tipo de integraciones.

CONCLUSIONES

La implementación de la solución no garantiza que el sistema resultante de la integración sea 100% seguro. Para aumentar ese porcentaje se debe emplear un sistema de gestión de seguridad de la información (SGSI), en donde se utilicen planes que respalden cada evento que pueda originarse en la consola de seguridad OSSIM.

El reporte de los registros de Untangle a OSSIM permite generar un sistema de seguridad de la información centralizado, facilitando la gestión de los registros de eventos de seguridad en compañías y ahorrando costos de tiempo y esfuerzo a los administradores de la red.

El uso del software Open Source permite adaptarse a las necesidades globales o particulares, haciendo pocas modificaciones. Esto hace que satisfaga todos los requisitos que exija una empresa sin entrar al dilema del costo vs. beneficio, que en muchas ocasiones hace que se abandonen proyectos.

El desarrollo de plugins para OSSIM y la versatilidad de sus herramientas, permiten la integración de casi cualquier sistema de información, facilitando la gestión de los administradores de red, puesto que se obtiene un sistema de gestión centralizado.

La integración implementada en este proyecto, puede ser fácilmente extendida a las otras aplicaciones de Untangle, siempre y cuando no sea modificada la estructura de logs generada por los aplicativos de Untangle.

La solución implementada permite ser adaptada a escenarios en producción, puesto que no requiere la modificación del código interno de los aplicativos de untangle(web filter, protocol control, firewall).

La metodología empleada, en este proyecto, para la integración de la herramienta Untangle, puede ser aplicada genéricamente para la integración de otras herramientas con la consola de seguridad OSSIM.

ANEXOS

Anexo A. Instalación de UNTANGLE

El servidor Untangle puede instalarse en una máquina virtual como VMware ó en un equipo real que cumpla con los requisitos mínimos que se mencionan en la Tabla 6. En la misma tabla se pueden apreciar los valores utilizados en el proyecto, que en este caso fueron un poco mayor.

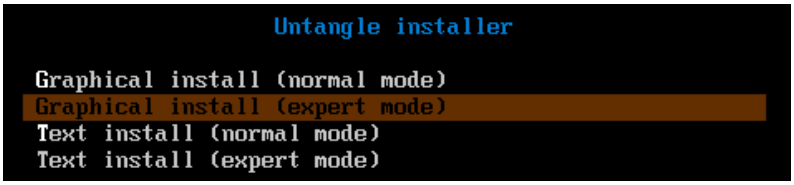
Nombre archivo: untangle_711.iso [36]
Nombre máquina: Untangle_PDG
Operating System: Linux operating system
Other Linux (32-bit)

Tabla 6 Requerimientos de Hardware Untangle

Recurso	Mínimo	Utilizado en el proyecto
CPU:	1 Procesador (1 GHz)	2 Procesadores (2 GHz)
Memoria:	512 MB	1 GB
Disco Duro:	20 GB	20 GB
Tarjetas de Red	2	2

Una vez se ha terminado la configuración del equipo en la máquina virtual con los requerimientos de hardware mencionados, se procede a la instalación de la distribución UNTANGLE. Por facilidad de instalación se escoge la forma grafica en el modo experto con el fin de modificar ciertos propiedades de particionamiento que puedan venir por omisión.

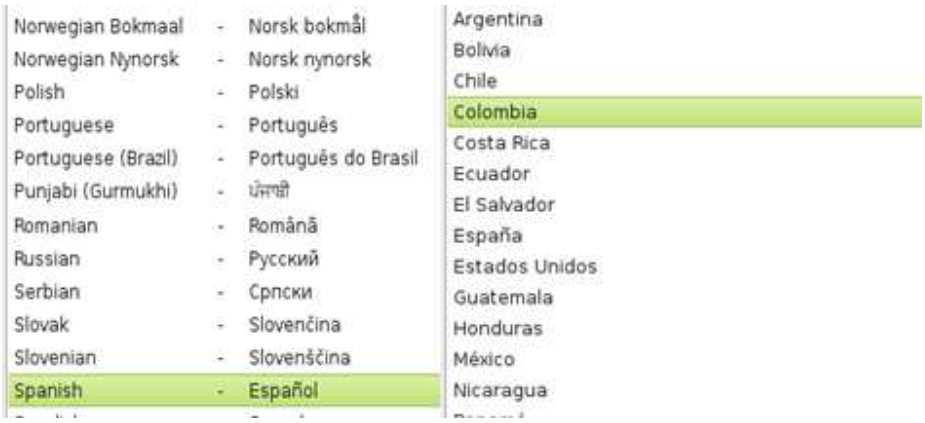
Ilustración 38 Modo de Instalación Untangle



Se selecciona el idioma (español) y la ubicación (Colombia).

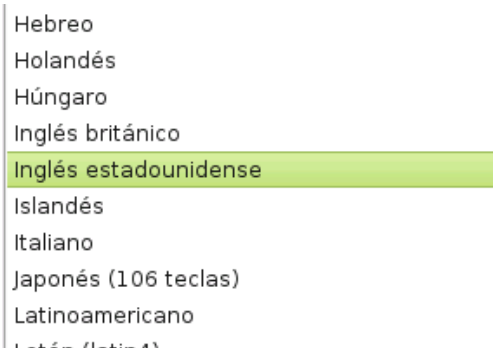
[36] http://www.untangle.com/Downloads/Download-ISO

Ilustración 39 Idioma y Ubicación Instalación Untangle



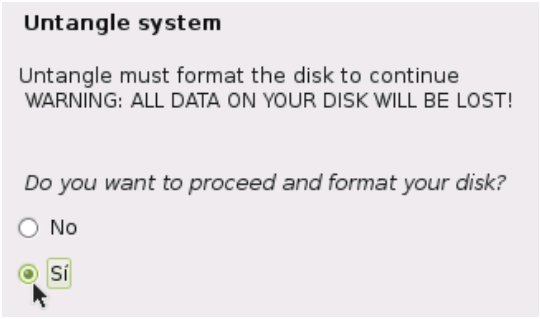
Se selecciona la distribución del teclado en Ingles porque se elaboró desde un portátil SONY con teclado estadounidense. Si su computador tiene la tecla ñ puede instalarlo en el modo LA ó ES.

Ilustración 40 Distribución Teclado Instalación Untangle



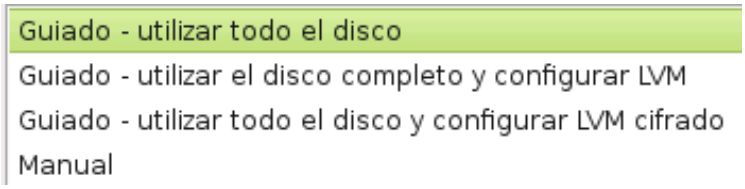
El sistema verifica que se cumplan con los requisitos para la instalación y luego procede a indicar si se desea darle formato al disco (Ilustración 41). Por ser una máquina virtual nueva no debe tener datos, pero es mejor formatear el disco para evitar tener almacenados datos innecesarios.

Ilustración 41 Dar Formato al disco de la Instalación Untangle



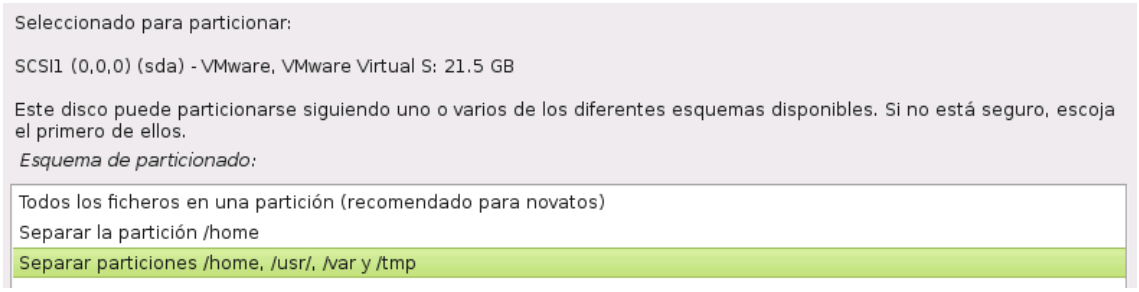
El particionado de disco será guiado, usando todo el disco y sin necesidad de configurar Volumen Lógico (Ilustración 42). El permitir que sea un particionamiento guiado da la posibilidad de revisar y confirmar los cambios en el tamaño de las particiones. Un volumen lógico da la posibilidad de tener el disco físico particionado de manera virtual.

Ilustración 42 Particionado de Discos Instalación Untangle



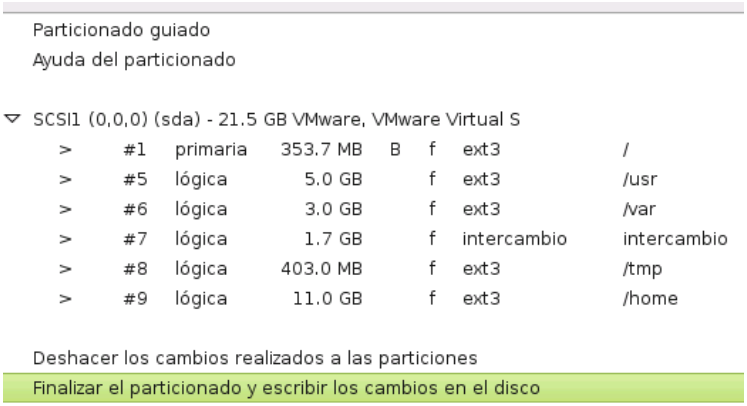
Solo hay un disco para particionar que fue creado desde el VMware. Se escoge la opción de separar el disco en /var, /tmp y /usr (Ilustración 43) con el fin de tener los archivos del sistema operativo clasificados de acuerdo al uso.

Ilustración 43 Particionado del Disco Instalación Untangle



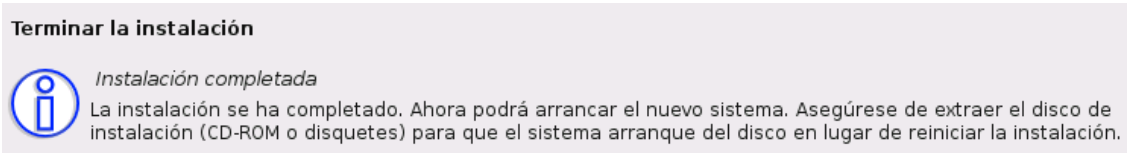
Luego el asistente de la instalación da los valores de las nuevas particiones y confirma si se desea escribir los datos sobre ellas (Ilustración 44). Por esta razón se había escogido el particionamiento guiado con el fin de poder modificar el valor de las particiones antes de finalizar el particionado.

Ilustración 44 Opción Modificación Particionado Guiado Instalación Untangle



Finalmente el asistente de instalación muestra el mensaje de finalización (Ilustración 45), no es necesario pero si aconsejable remover la referencia a la ISO para iniciar el sistema.

Ilustración 45 Instalación Completada Untangle



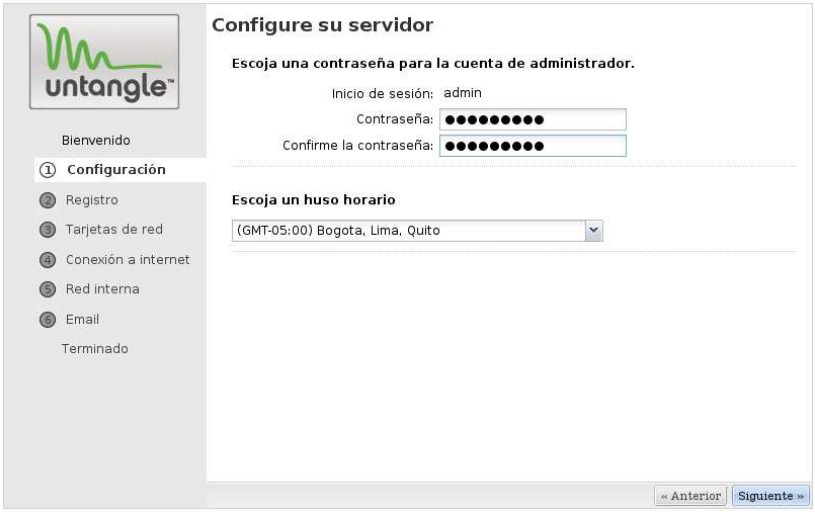
Configuración de Untangle

La primera vez que inicia el sistema operativo de Untangle se abre un asistente web para la configuración. Se escoge el idioma Español y se comienza a configurar el servidor.

El primer dato solicitado es para crear el usuario de la cuenta administradora (Ilustración 46).

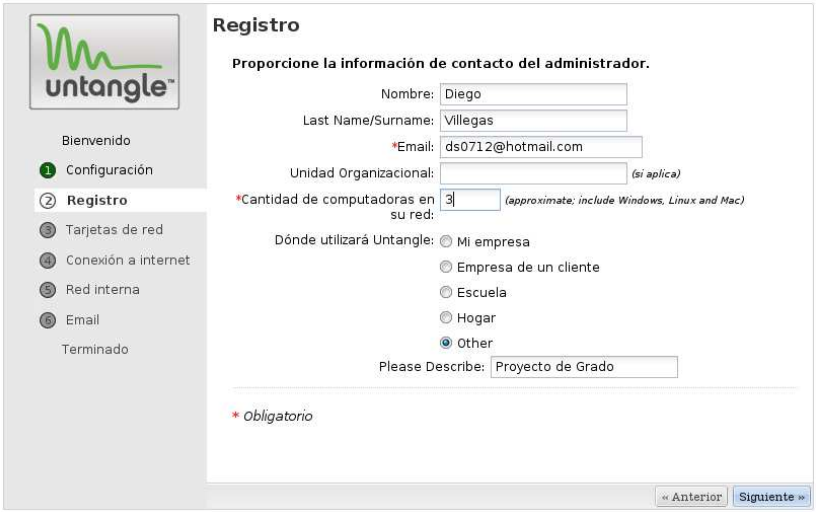
Inicio de sesión: admin
Contraseña: Password1
Zona Horaria: Bogotá

Ilustración 46 Usuario, Password y Zona horario Servidor Untangle



A continuación se debe ingresar la información del administrador de la red (Ilustración 47). Estos datos están disponibles cuando se cuente con un servidor de correo en la red.

Ilustración 47 Información del administrador de la red



Para el paso siguiente se debe verificar que los dos adaptadores de red estén conectados correctamente. Al usar la máquina virtual configurar las tarjetas de red en modo bridge (Ilustración 48) y verificar que estén conectadas (Ilustración 49).

Ilustración 48 Configuración en Modo Bridge tarjeta de red

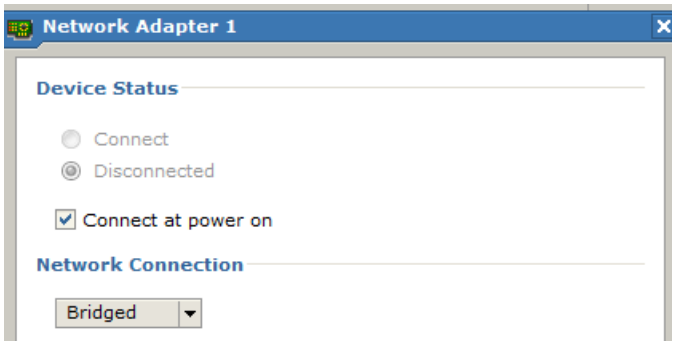


Ilustración 49 Verificación Funcionamiento tarjetas de red

- El adaptador 1 en VMware es Interfaz de red externa
- El adaptador 2 en VMware es Interfaz de red interna

A continuación configurar la IP de la tarjeta de red externa (Ilustración 50), con la cual se tiene acceso a internet. En el caso del proyecto, la IP del servidor Untangle

deberá tener conectividad con el servidor OSSIM, es decir, estar en la misma red. Al probar la conectividad sale un cuadro de dialogo con el mensaje: *Success!*

Ilustración 50 Configuración de la tarjeta de red externa

La red interna será: 172.16.0.0/24. Se escoge **Activar DHCP** por facilidad para la configuración de la tarjeta de red de un usuario que se conecta a la red.

Ilustración 51 Configuración de la tarjeta de red Interna

No se realiza la prueba para enviar un mail porque no se tiene un servidor de correo disponible dentro de la red.

Una vez termine la configuración de Untangle, se empiezan a descargar las aplicaciones para actualizar ciertos paquetes del sistema operativo, para la fecha de este montaje (Noviembre 7 de 2010) actualizó 37, con un tamaño total de 37 MB (Ilustración 52).

Ilustración 52 Actualización Paquetes Untangle

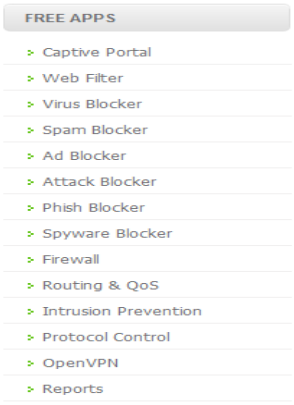
Después de la actualización es normal que el servidor quede no disponible durante unos minutos mientras la base de datos UVM inicia nuevamente.

Instalación de una aplicación Gratuita en Untangle

La instalación de una aplicación libre se realiza por medio de la interfaz web de Untangle, lo primero que se debe hacer es ingresar con el usuario de Untangle, en el caso del presente trabajo, es admin y la contraseña es Password1.

Para descargar una aplicación, se selecciona al lado izquierdo donde se encuentran disponibles. En este caso se va a trabajar con las aplicaciones gratuitas.

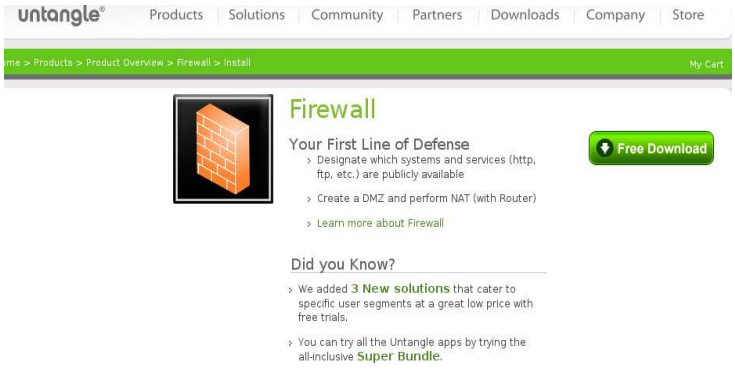
Ilustración 53 Aplicaciones Libres Untangle



Fuente: Control Web Content at the Gateway with our Free Web Filter. Clave internet: http://www.untangle.com/Web-Filter

Una vez se selecciona una aplicación, por ejemplo Firewall, direcciona a una página como la Ilustración 54 y se selecciona Free Download.

Ilustración 54 Página Descarga Aplicación Firewall Untangle



Mientras se descarga una aplicación se puede observar el progreso de la instalación. Una vez termina, se ubica automáticamente en el rack principal de las aplicaciones (Ilustración 55).

Ilustración 55 Firewall en rack del servidor Untangle

Anexo B. Instalación de OSSIM

En un escenario real de instalación de OSSIM, es necesario cumplir con los requerimientos mínimos de instalación de hardware (Tabla 7). Sin embargo para este proyecto de grado, en donde se va a monitorear principalmente un equipo correspondiente al servidor de Untangle, el tráfico que se registra es mínimo, al igual que la cantidad de datos procesados, por tanto se puede emplear menor cantidad de recursos.

Nombre de archivo: alienvault-ossim-installer-2.2.1.x86.iso[37]
Nombre de la Máquina: OSSIM_PDG
Sistema operativo: Linux operating systems
Other Linux (32-bit)

Tabla 7 Requerimientos de Hardware OSSIM

Recurso	Mínimo	Utilizado en el Proyecto
CPU:	2 Procesadores (1GHz) *	1 Procesador (1.8 GHz)
Memoria:	2GB, cantidad que debe ir incrementando en función del tráfico que se analice.	1.25 GB, El tráfico a analizar solo corresponde a un agente.
Disco Duro:	20GB *	15 GB
Tarjetas de Red:	1 en modo Promiscuo	1 en modo bridge VM, durante la instalación se configure en modo promiscuo.

* Depende en gran medida del número de eventos por segundo y del ancho de banda de la red a analizar.

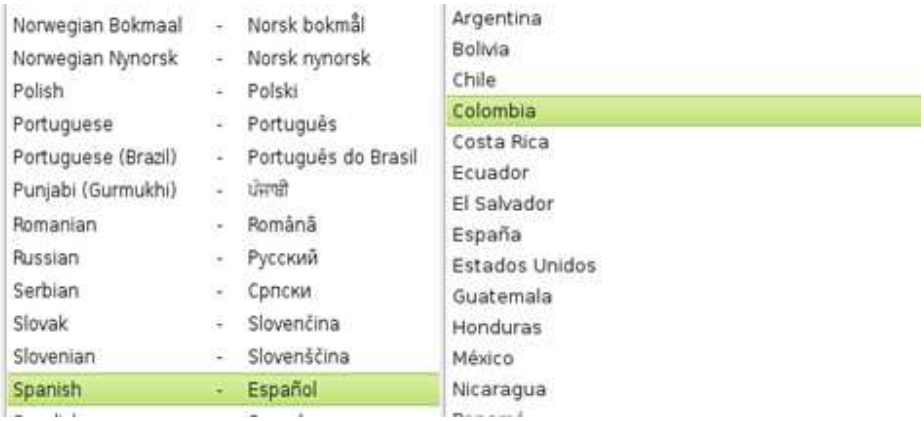
Una vez se ha configurado la máquina virtual, se procede a la instalación personalizada de OSSIM para modificar algunas de las propiedades que viene por omisión como se observa en la Ilustración 56.

Ilustración 56 Modo de Instalación de OSSIM

[37] http://data.alienvault.com/alienvault-ossim-installer-2.2.1.x86.iso

Se selecciona el idioma (español) y la ubicación (Colombia).

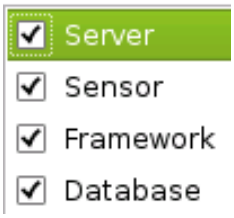
Ilustración 57 Selección de Idioma y Ubicación Instalación OSSIM



Luego se escoge la distribución del teclado según el computador.

Dependiendo de la función OSSIM puede configurarse con un determinado perfil de uso, este puede funcionar de manera distribuida instalando cada uno de sus componentes por separado, sin embargo para esta instalación se utilizaran todos los perfiles en un mismo equipo (para mayor información ver 5.1.1 Niveles).

Ilustración 58 Perfil de Instalación OSSIM

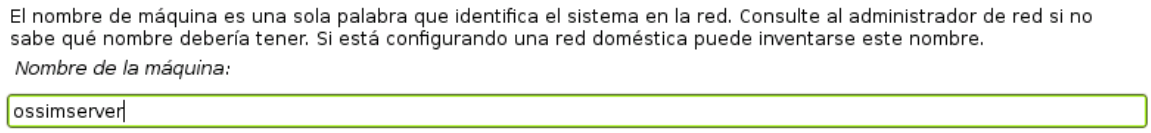


De acuerdo con el escenario, la tarjeta de red del equipo de OSSIM tendrá la dirección IP 192.168.130.235 con máscara de red 255.255.255.0, esta dirección también servirá para accede a la interfaz de gestión vía Web.

La pasarela, Gateway o puerta de enlace predeterminada nos permite enrutar los paquetes a la red, por lo general es la primera dirección de la red, en este caso 192.168.130.1. El DNS, servidor de nombres de dominio, para este montaje coincide con la dirección del Gateway 192.168.130.1, en caso de tener más de un servidor DNS se deberá ingresar cada dirección separada por un espacio.

El nombre del equipo de OSSIM que lo identificara dentro de la red y localmente es ossimserver (Ilustración 59). El dominio en el proyecto de grado no va ser utilizado, se empleara el manejo de las direcciones IP.

Ilustración 59 Configuración de Red Instalación OSSIM



Para el particionado de disco se seleccionó particionado guiado, utilizando todo el disco (Ilustración 60) que se creó para la máquina virtual, de esta manera los datos almacenados en el disco virtual serán eliminados, dejando toda la funcionalidad de este para el servidor de OSSIM. Al ser un particionamiento guiado se tiene la posibilidad de hacer cambios antes de confirmar el particionado.

Ilustración 60 Particionado de Disco Instalación OSSIM

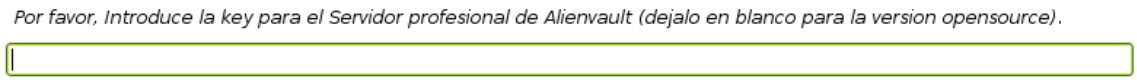
Luego de haber particionado el disco SCSI1 completamente, se escoge la opción para particionar el disco en **/home, /usr, / var y /tmp** (Ilustración 61) con el fin de ser organizados con el manejo de los archivos en el sistema operativo.

Ilustración 61 Esquema para la Partición Instalación OSSIM

Se despliega un resumen con cada una de las particiones que se va a realizar y su tamaño, se debe confirmar la partición seleccionando la opción Finalizar el particionado y escribir los cambios en el disco.

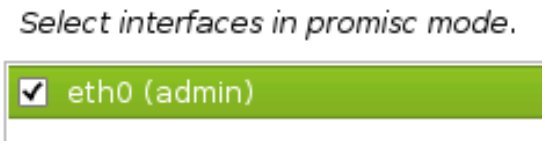
La versión utilizada para este montaje es Open source, por tanto se debe dejar en blanco el ingreso de claves para el servidor profesional de Alienvault (Ilustración 62).

Ilustración 62 Campo para introducir la clave en la versión comercial



En la máquina virtual solo se configuro una tarjeta de red para el servidor, esta debe estar en modo promiscuo para capturar todo el tráfico que circula en la red de ella, para así poder realizar los análisis de seguridad.

Ilustración 63 Selección Modo Promiscuo Tarjeta Instalación OSSIM



El equipo de Untangle está conectado a la red 192.168.130.0/24, por tanto se debe ingresar esa red para poder monitorizarla.

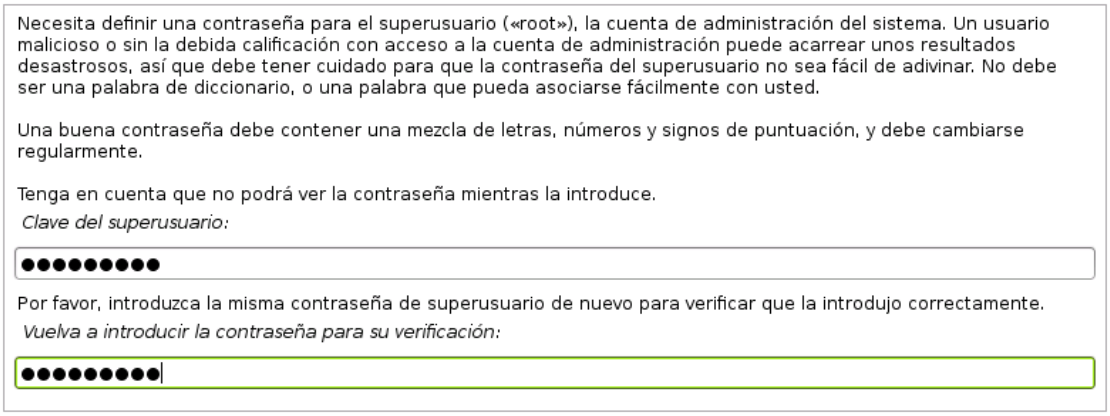
Ilustración 64 Selección de Red a Monitorear Instalación OSSIM

La configuración del servicio de correo no es necesaria, por ende se selecciona la opción ***Sin configuración*** y se da clic en continuar. Es importante anotar que en ambientes reales este servicio es necesario, dado que nos pueden alertar de manera automática acerca de un evento que ponga en riesgo la seguridad y confidencialidad de nuestra red.

A continuación ingresar la contraseña para el super usuario del sistema (root) (Ilustración 65). En el caso del proyecto se trabajará con la siguiente información:

Usuario: root
Clave: Password1

Ilustración 65 Configuración del Password del Súper Usuario Instalación OSSIM



Después pide conectarse a un servidor de actualización de Debían, esto con el fin de actualizar el sistema operativo al ser la primera vez que se va utilizar. Se puede escoger el servidor ftp de Estados Unidos. Cuando termine esto, va a pedir la información de un servidor proxy en caso de ser necesario para acceder a internet.

Ilustración 66 Replica del sistema Debían del servidor de Instalación OSSIM

Finalmente se obtiene un mensaje informando de que el sistema operativo fue instalado.

Configuración de OSSIM

Ahora se necesita configurar las propiedades del servidor OSSIM, los monitores y los detectores. Se dejan desactivados todos los monitores porque no se van a emplear.

Se dejaron activados los siguientes detectores con el fin de llevar registro sobre estos sucesos:

OSSEC
SSH
SUDO
SYSLOG

Luego de este proceso la máquina se reiniciará automáticamente cargando todos los servicios, detectorés y monitores configurados en la instalación. Se debe seleccionar YES cuando pida la instalación de actualizaciones disponibles con el fin de tener una máquina mejorada.

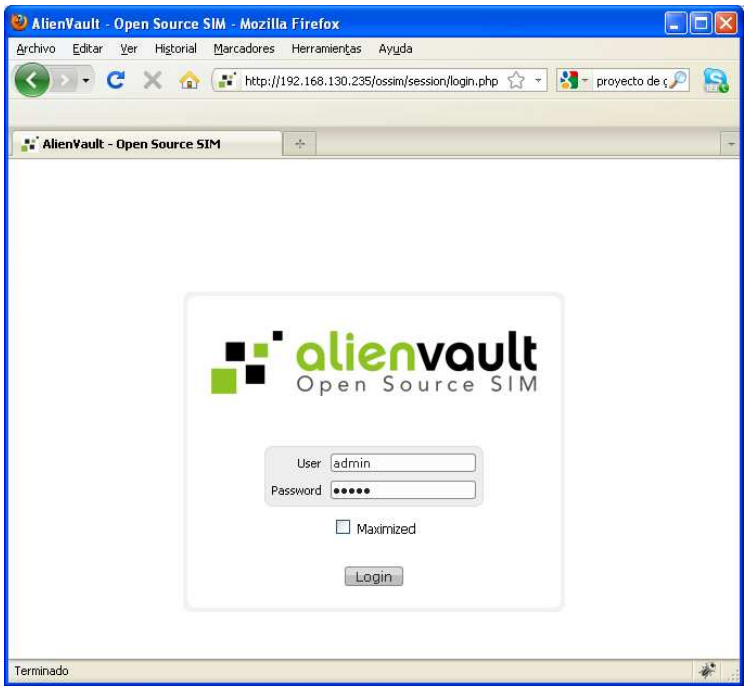
Ilustración 67 Terminar Instalación OSSIM

Se puede ingresar al servidor OSSIM de dos formas, vía web y desde la consola de comandos. Cuando se utilice la consola de comandos durante la realización del proyecto, hay que ingresar con el usuario root. La vía web se hará por medio del usuario admin.

Ilustración 68 Ingreso al Servidor OSSIM por medio de la consola

A través del navegador web se puede ingresar la dirección IP asignada al servidor de ossim, en este caso http://192.168.130.235.

Ilustración 69 Ingreso al Servidor OSSIM vía web



Anexo C. Instalación de OSSEC en Untangle

Antes de instalar el paquete OSSEC se debe instalar la librería **libc6-dev** en el servidor Untangle, para poder compilar lenguaje C. Si no realiza este paso sale error. También se debe habilitar en el firewall el puerto 1514(UDP) para permitir la comunicación entre el agente OSSEC de Untangle y el servidor OSSEC en OSSIM.

apt-get install libc6-dev

El código para instalar OSSEC en un sistema operativo mediante ventanas se encuentra en la página web http://www.ossec.net/main/downloads/; también lo podrá hacer desde un sistema operativo Linux[38], como el de Untangle, con permisos de superusuario (root) con el siguiente comando:

wget http://www.ossec.net/files/ossec-hids-latest.tar.gz

Descomprir el archivo descargado.

tar -zxvf ossec-hids-latest.tar.gz

Se ingresa al directorio de ossec que resultó después de descomprimir el archivo con extensión .tar.gz

cd ossec-hids-2.5.1

Se ejecuta el script de instalación install.sh que sirve de ayudar para guiar el proceso.

./install.sh

Empezar seleccionando el idioma español (es) para continuar con el proceso de instalación de OSSEC (Ilustración 70).

[38] OSSEC. Manual: installation. Clave internet: http://www.ossec.net/main/manual/manual-installation

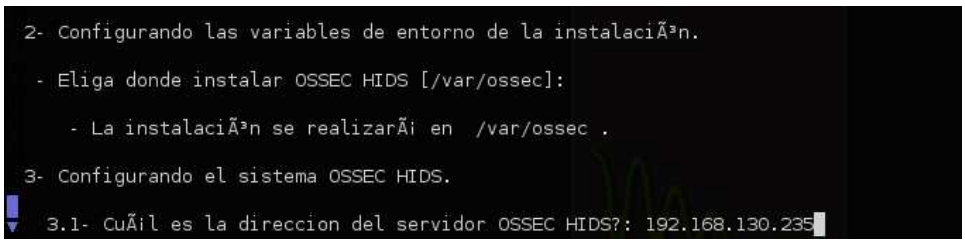
Ilustración 70 Selección de Idioma Instalación OSSEC

Cuando sale el mensaje que se necesita tener una librería que compile C para poder efectuar la instalación de OSSEC, se debe presionar Enter y a continuación se escoge el tipo de instalación para el sistema, el cual será tipo agente.

Ilustración 71 Tipo de instalación OSSEC para el sistema

En la ubicación de la instalación, está la ruta que viene por omisión /var/ossec y se ingresa la dirección IP del servidor OSSIM, que será también el servidor OSSEC.

Ilustración 72 Ubicación de Archivos y Dirección del servidor para OSSEC



A continuación pide si se desea agregar el servidor de integridad al sistema y el sistema de detección de *rootkit*, en el caso del proyecto se escogió la opción por

omisión s. El detector *rootkits* es un programa que verifica la integridad del *kernel* y los archivos principales de linux se encuentren bien e informa cuando se presenta un posible ataque.

Ahora se va a habilitar la respuesta activa, que será la que se origine una vez ocurra el evento, cuando se escribe sobre el archivo log. La ruta de este archivo debe queda en el archivo ossec.conf, agregando que sea una entrada de tipo localfile porque el log viene en una línea de texto.

Cuando se termina la instalación aparecen los comandos que se pueden emplear para comenzar el proceso de detección de intrusos basados en el Host y también la ruta para cambiar la configuración del agente OSSEC (Ilustración 73).

Ilustración 73 Mensaje de terminación Instalación OSSEC

Anexo D. Instalación de pl/sh[39]

Para la instalación de este manejador de lenguaje Shell en la base de datos Postgres de Untangle, agregar la siguiente línea en el archivo de fuentes **/etc/apt/sources.list** para instalación y actualización de programas.

deb http://ftp.us.debian.org/debian lenny main

A continuación se debe reiniciar el servidor para actualizar el repositorio que se acabo de escribir y ejecutar el comando de instalación:

Apt-get install postgresql-8.3-plsh

Este lenguaje de programación en la base de datos (pl/sh) se debe publicar donde se desea utilizar, en el caso del proyecto es en la base de datos uvm (untangle virtual machine) con el usuario postgres.

psql –U postgres –d uvm –f /usr/share/postgresql-8.3-plsh/createlang_pgplsh.sql

[39]Debian. Download Page for postgresql-8.3-plsh_1.3-1_i386.deb on Intel x86 machines. Clave internet: http://packages.debian.org/lenny/i386/postgresql-8.3-plsh/download

Anexo E. Expresiones regulares

Para la construcción de un plugin para OSSEC, es necesario desarrollar una expresión regular que permita analizar el contenido de la línea del log, registrada por las aplicaciones de untangle, y que además permita la separación de los campos o parámetros dentro del log que interesa sean controlados.

En el servidor de Untangle se creó un archivo que registraba los logs de las aplicaciones, el cual se ubico en (**/var/log/untangle-ossec-eventos/ossec-untangle.log**) y está siendo monitorizado frecuentemente por el agente de OSSEC. Si se presenta un nuevo evento, el agente se encarga de llevarlo de manera inmediata al OSSEC Server, ubicado en la consola de seguridad OSSIM.

Para entender la expresión regular, que permite filtrar los campos generados por las aplicaciones de untangle, se utilizará una tabla con un registro de cada aplicación.

Cada vez que se genera un log de las aplicaciones de untangle, se generará con la siguiente estructura:

Tabla 8 Símbolos utilizados en la estructura del log

^·Servidor·|·Aplicación·|·Fecha_Hora·|·acción·|·IP_Origen·|·Campo_Libre·|·Motivo_Log·|·IP_destino·¶

Simbología en cadena
"·" Corresponde a un carácter de espaciado "^" Corresponde a carácter de inicio de línea "¶" Corresponde a carácter de salto de línea

Es importante anotar que los caracteres mencionados en la Tabla 8 no corresponden a las secuencias utilizadas por regex en python.

Para escribir las expresiones regulares, de las aplicaciones que permiten obtener la información más relevante de los registros, se debe trabajar con los siguientes metacarácteres utilizados en python.

Tabla 9 Metacarácteres para expresión regular

Metacarácteres	
+	Repetición de una o más veces en un carácter
?	Repetición única de un carácter o tipo
*****	Repetición de cero o más veces en un carácter.
\d	Concuerda con dígitos numéricos
\s	Concuerda con caracteres de espaciado.
.	Concuerda con cualquier carácter por una vez.
\S	Delimitador de negación de \s
$	Concuerda con carácter de final de línea
^ ó \n	Concuerda con carácter de inicio de línea
( )	Delimitador de referencia a identificador.

Fuente: Creación de un plugin para OSSIM. Clave internet: http://ossimcolombia.blogspot.com/2010/05/creacion-de-un-plugin-para-ossim.html

La Ilustración 74 muestra los registros de cada una de las tres aplicaciones para las cuales se desarrolló el plugin.

Ilustración 74 Registros aplicaciones Untangle

untangle | web filter | 2010-11-15 17:07:12.069 | B | 172.16.0.15 | www.redtube.com | Pornography | 209.222.138.10

untangle | firewall | 2010-11-15 20:05:22.04 | t | 172.16.0.45 | regla:2 | Permitir Acceso red externa | 72.14.172.18

untangle | protocolcontrol | 2010-11-15 18:51:19.438 | f | 172.16.0.45 | MSN Messenger | Microsoft Network chat client | 8.19.240.53

Los 3 registros poseen campos similares, lo único que varía es su contenido, esto hace posible que solo se realice una expresión regular para la obtención de los patrones más relevantes en las tres aplicaciones. Para identificar fácilmente como es la representación de cada uno de los campos con expresiones regulares, se diseño una tabla que permitiera una fácil interpretación (Tabla 10).

Tabla 10 Ejemplo de expresiones regulares

		Servidor	|	Aplicación	|	Fecha y Hora	|	Acción	|
Campo No.	1	2	3	4	5	6	7	8	9
Aplicación 1	^·	untangle	·|·	web filter	|·	2010-11-15 17:07:12.069	·|·	B	·|·
Aplicación 2	^·	untangle	·|·	firewall	|·	2010-11-15 20:05:22.04	·|·	t	·|·
Aplicación 3	^·	untangle	·|·	protocolcontrol	|·	2010-11-15 18:51:19.438	·|·	f	·|·
Longitud Cadena		Fijo		Variable con espacios		Variable con estructura fija		Variable sin espacios	
RegExp	^\s+	untangle	\s+|\s+	.*?	|\s+	\d+-\d+-\d+\s+\S+	\s+|\s+	(\S+)	\s+|\s+

	IP Origen	|	Campo Libre	|	Motivo Log	|	IP Destino	
Campo No.	10	11	12	13	14	15	16	17
Aplicación 1	172.16.0.15	·|·	www.redtube.com	|·	Pornography	|·	209.222.138.10	·¶
Aplicación 2	172.16.0.181	·|·	regla:3	|·	Permitir Acceso red externa	|·	72.14.172.18	·¶
Aplicación 3	172.16.0.45	·|·	MSN Messenger	|·	Microsoft Network chat client	|·	8.19.240.53	·¶
Longitud Cadena	Variable sin espacios		Variable con espacios		Variable con espacio		Variable sin espacios	
RegExp	(\S+)	\s+|\s+	(.*?)	|\s+	(.*?)	|\s+	(\S+)	\s$

Los campos 8, 10 y 16, Acción, IP_Origen e IP_Destino respectivamente, poseen cadenas de longitud variable y sin espaciado. Teniendo en cuenta que la versión del protocolo IP puede variar a la versión 6, se asume como una cadena de texto continua, por otra parte, el campo Acción puede ser modificado para escribir la acción que se realizó con el evento, la representación para este tipo de cadena se asume como "\S+", "\S" concuerda con caracteres distintos al espacio y el "+" hace que se pueda repetir una o más veces la secuencia, a menos de que se presente un carácter espacio.

El campo 6, fecha y hora, posee una estructura que se presenta igual para los registros de fechas, su estructura coincide con el formato "\d+-\d+-\d+\s+\S+", para el formato de la fecha se tienen dígitos de longitud variable, separados por guiones, posteriormente la hora es separada de la fecha con un carácter de espacio "\s" y para simplificar la expresión se asume que sigue un cadena continua "\S+", hasta que se encuentre con el siguiente carácter de espacio "\s" ubicado en el campo No. 7.

Los campos 4, 12 y 14, Aplicación, Campo Libre y Motivo Log respectivamente, poseen combinaciones de texto con caracteres de espacio cuyo contenido y longitud es desconocida, por tanto se hace necesario utilizar un carácter especial de parada. La expresión regular para este tipo de cadenas coincide con ".*?". el "." Coincide con cualquier carácter, "*" permite la repetición entre 0 y muchas veces y finalmente "?" estará a la espera de un carácter de parada distinto, raya

horizontal y espacio “\|\s+”, que se puede ubicar en los campos subsiguientes (5, 13 y 15).

Los campos de separación de parámetros (3, 7 ,9 y 11) coinciden con la expresión regular “\s+\|\s+”, para evitar errores con cantidades de caracteres de espacio fijo, se asume múltiples caracteres de espacio seguidos a través de la expresión “\s+”.

Los campos 1 y 17 corresponden al inicio y el fin de la cadena, su expresión regular coincide con “^s+” y “\s$” marcando el inicio y el fin del registro.

Algunas de las expresiones regulares se pueden observar entre paréntesis “()” que son conocidos como identificadores regulares, estos permiten la asignación de variables dentro de las expresiones regulares, su utilización se mostrará en el siguiente capítulo (Parametrización de la expresión regular).

Finalmente la expresión regular que concuerda con los logs generados por las aplicaciones de untangle, es la siguiente:

^\s+untangle\s+\|\s+ .*?\|\s+\d+-\d+-\d+\s+\S+\s+\|\s+(\S+)\s+\|\s+(\S+)\s+\|\s+(.*?)\|\s+(.*?)\|\s+(\S+)\s$

Parametrización de la expresión regular

Una vez se tiene y se conoce la información de cada uno de los delimitadores contenidos en la expresión regular, se procede a parametrizar.

^\s+untangle\s+\|\s+.*?\|\s+\d+-\d+-\d+\s+\S+\s+\|\s+(\S+)\s+\|\s+(\S+)\s+ \|\s+(.*?)\|\s+(.*?)\|\s+(\S+)\s$

La parametrización consiste en adicionar variables dentro de los identificadores “()” de la expresión regular, que permitan la utilización en diferentes espacios de código.

En la expresión regular se pueden observar 5 identificadores, a cada uno se le asigna una variable adicionando después del paréntesis que abre “(” un fragmento de código de la forma ?P<nombreVariable>, quedando la expresión de la forma (?P<nombreVariable>\S+), la información capturada en el fragmento de la expresión regular quedará almacenada en la variable $nombreVariable.

Obsérvese la expresión regular parametrizada:

\s+untangle\s+\|\s+\S+.*?\|\s+\d+-\d+-\d+\s+\S+.*?\s\|\s+(?P<accion>\S+?)\s\|\s+(?P<iporigen>\S.*?)\s\|\s+(?P<campouno>\S+.*?)\|\s+(?P<motivolog>\S+.*?)\|\s+(?P<ipdestino>\S.*?)\s

La utilización de las variables se hará a través de $iporigen, $accion, $campouno, $motivolog, $ipdestino, por lo regular estas variables son asignadas a variables globales de OSSIM, que permiten desplegar la información en la consola WEB de OSSIM.

BIBLIOGRAFÍA

ANDRADE FONSECA, Roberto. Programación de funciones en PL/pgSQL para PostgreSQL, 2002. [Documento en Línea]. Disponible desde Internet en: <http://sdi.bcn.cl/desarrollo/doctos/PL_pgSQL.pdf> [Consulta agosto 21 de 2010]

CASAL, Julio. OSSIM – Descripción general del sistema, 2003. [Documento en Línea]. Disponible desde Internet en: <http://www.alienvault.com/docs/OSSIM-desc-es.pdf > [Consulta octubre 23 de 2010]

Cid, Daniel. Log Analysis using OSSEC, 2007. [Documento en línea]. Disponible desde internet en: < http://www.ossec.net/ossec-docs/auscert-2007-dcid.pdf> [consulta junio 27 de 2010].

CRN, 2007. SIEM: A Market Snapshot. . [Web en línea]. Disponible desde Internet en: <http://www.crn.com/news/security/197002909/siem-a-market-snapshot.htm> [consultado julio 28 de 2010].

MySQL. Capitulo 20: Disparadores (triggers). [Web en línea]. Disponible desde Internet en: <http://dev.mysql.com/doc/refman/5.0/es/triggers.html> [Consulta agosto 21 de 2010]

Openti. Detalles Untangle Red Segura Control de protocolos. [Web en línea]. Disponible desde Internet en: <http://openti.net/index.php?option=com_content&task=view&id=39&Itemid=45#bspy> [Consulta noviembre 14 de 2010]

Debian. Download Page for postgresql-8.3-plsh_1.3-1_i386.deb on Intel x86 machines. [Web en línea]. Disponible desde Internet en: <http://packages.debian.org/lenny/i386/postgresql-8.3-plsh/download> [Consulta agosto 14 de 2010]

Imtiaz, Fahmid. Intrusion Detection System Logs as Evidence and legal aspects. [Web en línea]. Disponible desde Internet en: http://www.forensicfocus.com/intrusion-detection-system-logs [consulta agosto 28 de 2010]

OSSEC. Manual: installation. [Web en línea]. Disponible desde Internet en: <http://www.ossec.net/main/manual/manual-installation> [consultado Junio 05 de 2010].

LATORRE, Cristian. Creación de un plugin para OSSIM, 2010. [Web en línea]. Disponible desde Internet en: <http://ossimcolombia.blogspot.com/2010/05/creacion-de-un-plugin-para-ossim.html> [consulta junio 13 de 2010].

Lavender, Brian. HOWTO for creating a simple plugin. [Web en línea].Disponible desde Internet en: <http://permalink.gmane.org/gmane.comp.security.ossim.support/612> [consultado marzo 13 de 2010].

Lucena López, Manuel J. El Protocolo SSL. [Web en línea]. Disponible desde Internet en: <http://web.ipsca.com/es/Certificados_ssl> [consultado noviembre 28 de 2010].

ROMAN, Alberto. OSSIM Management Server, 2008. [Web en línea]. Disponible desde Internet en: <http://www.ossim.net/dokuwiki/doku.php?id=documentation:serverd> [consultado octubre10 de 2010].

Rincón Informático. Seguridad Informática Norma ISO 27001, 2008. [Web en línea]. Disponible desde Internet en: <http://www.rinconinformatico.net/seguridad-informatica-norma-iso-27001> [consultado septiembre 18 de 2010].

Wiki Untangle, 2010. Server User's Guide: Web filter. [Web en línea]. Disponible desde Internet en: <http://wiki.untangle.com/index.php/Web_Filter> [consultado julio 10 de 2010].

Wiki Untangle, 2010. Server User's Guide: Virus blocker. [Web en línea]. Disponible desde Internet en: <http://wiki.untangle.com/index.php/Virus_Blocker> [consultado julio 10 de 2010].

Wiki Untangle, 2010. Server User's Guide: Spam Blocker. [Web en línea]. Disponible desde Internet en: <http://wiki.untangle.com/index.php/Spam_Blocker> [consultado julio 10 de 2010].

Wiki Untangle, 2010. Server User's Guide: Ad Blocker. [Web en línea]. Disponible desde Internet en: <http://wiki.untangle.com/index.php/Ad_Blocker> [consultado julio 10 de 2010].

Wiki Untangle, 2010. Server User's Guide: Firewall. [Web en línea]. Disponible desde Internet en: <http://wiki.untangle.com/index.php/Firewall> [consultado julio 17 de 2010].

Ossim Colombia, 2010. Creación de un plugin para OSSIM. [Web en línea]. Disponible desde Internet en:

<http://ossimcolombia.blogspot.com/2010/05/creacion-de-un-plugin-para-ossim.html> [consultado marzo 13 de 2010].

Instituto Colombiano de Normas Técnicas. Normas Colombianas para la presentación de Trabajos de Investigación, 2008. [Documento en línea] Disponible desde Internet en: <http://www.uceva.edu.co/ingenieria/images/norma/ntc1486.pdf> [Consulta noviembre 28 de 2010].