Universidad de las Ciencias Informáticas

Facultad 2

Título: Análisis y Configuración de la Herramienta de Seguridad Informática OSSIM en la Universidad de las Ciencias Informáticas

Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autores: Lázaro Esteban Arce Rodríguez

Sergio González Ginarte

Tutores: Ing. Yonis Del Pozo Rodríguez

Ing. Juan Carlos Rodríguez

Co-tutores: Ing. Nairys Morales Sosa

Ing. Dionner Polanco Noy

Ciudad de la Habana, 26 de junio del 2008

"Año 50 de la Revolución"

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Lázaro E. Arce Rodríguez

_____
Firma del Autor

Sergio González Ginarte

_____
Firma del Autor

Ing. Yonis Del Pozo Rodríguez

_____
Firma del Tutor

Ing. Juan Carlos Rodríguez

_____
Firma del Tutor

Ing. Nairys Morales Sosa

_____
Firma del Co-tutor

Ing. Dionner Polanco Noy

_____
Firma del Co-tutor

DATOS DE CONTACTO

Ing. Yonis Del Pozo Rodríguez.

Ingeniero en Ciencias Informáticas, recién graduado. Desde tercer año alumno ayudante de la asignatura Matemática. Durante el curso 06-07 fue miembro del proyecto de redes de la facultad. Actualmente impartiendo la misma asignatura para primer año y cursando un diplomado que constituye la antesala de una maestría.

Contacto: ydelpozo@uci.cu

Ing. Juan Carlos Rodríguez Gutiérrez.

Ingeniero en Ciencias Informáticas, graduado en el año 2006-2007, especialista en seguridad de la dirección de Redes y Seguridad Informática, profesor del Departamento de Sistema Digitales donde imparte las asignaturas de Sistema Operativo y Seguridad Informática.

Contacto: jrodriguezg@uci.cu

Ing. Nairys Morales Sosa.

Ingeniera en Ciencias Informáticas, recién graduada y profesora de la Facultad Regional de la UCI en Artemisa. Desempeña el cargo de asesora de calidad en dicha facultad.

Contacto: nmorales@uci.cu , nairys@hab.uci.cu.

Ing. Dionner Polanco Noy.

Ingeniero en Ciencias Informáticas, recién graduado. Participó en la Copa Pascal en 4to año obteniendo segundo lugar a nivel de faculta. Actualmente trabaja en la UCI como Administrador de Redes. Ha cursado tres postgrados entre ellos, el curso de certificación en servicios telemáticos sobre Linux.

Contacto: dpolanco@uci.cu

“El grado sumo del saber es contemplar el por qué”

Sócrates

“Vale más la sabiduría que las armas de guerra.”

Eclesiastés 9:18

AGRADECIMIENTOS

Lázaro

Agradezco a mis abuelos la paciencia y el amor con que me han educado. A Mima (Isela), por hacerme los mejores manjares del mundo, a ella agradezco su ternura, su quererme siempre, su apoyo durante las pruebas más difíciles de mi carrera, sus elogios, su beso eterno. A Papa (Alejandro), quien no dejó de asomarse nunca a mi cuarto cuando yo enfermaba, a él le debo ese deseo interminable de andar por mi pueblo, por mis raíces, y esa fuerza de vivir. A Ua (Gloria) la dedicación sin límites, la alegría de ser sencillo, la compañía en todo momento, por complacerme y ocuparse tan bien de mis cosas y de mí, aun recuerdo cuando ella me ponía las medias y los zapatos para salir a jugar, en los días más felices que solo se dan cuando eres niño. A Papo (Juvenal) la sabiduría con que me habla, el desinterés por lo vano, tantas respuestas a preguntas imposibles, su abrazo, su afán de trabajar, de hacer lo bueno, mi primera bicicleta de verdad; cuanto quisiera estar de nuevo en el asientico delantero de la suya para conocer del mundo. Agradezco a mi mamá por su constancia, su cariño y el serme fiel en todo momento. A su esposo Raúl por ser mi amigo, por correr con la mayoría de mis gastos de estudiante y joven. Fue a él y a mi mamá a quienes primero reconocí cuando salí del salón de operaciones y solo así dejé de llorar. A mi papá por el conocimiento que me transmite, a su esposa Ibel. A mis tíos Wilfredo y Luis quienes son padres para mí y a quienes les debo, desde que tengo conocimiento, las mejores experiencias que se hayan tenido jamás. Ellos son mi ejemplo y mi guía. Nunca estuve tan seguro de algo como del ruido del motor de mi tío Luis cuando llegaba a la casa, siempre con algún regalo, y por quien escondía el tete que usaba para dormir. A mi tío Wilfredo, el más alejado porque vive casi en otra provincia pero el más cercano pues es de todos el más cariñoso, a él mis mejores deseos y reconocimiento, cómo olvidar los asados que prepara y aquellas noches de pesca en el mar.

Son muchas mis deudas y mi gratitud para con mis abuelos, para con mi familia. Agradezco a mis hermanos Renier y David, a su mamá Dania por tenerme como un hijo, a mis tías Gisela y Dania, a mis primos y primas, a mis amigos y amigas, al piquete (Victor, Javier, Rahonel, Dennis y Erney), a Marlén y a Puchín, a mi novia Yudinela, a mis tutores y co-tutores, a toda aquella persona que haya tenido que ver conmigo o con alguno de los mencionados anteriormente, y a Dios. Me queda mucho por decir, me sobra más para entregarles. Quede a todos mi amor, mi sí con ellos, mi oración continua pues son mi razón de ser. Dios los bendiga a todos como me ha bendecido con tenerlos a mi lado. No existe razón mayor que la de ser amado, ni satisfacción más plena que conocerla.

Sergio

A nuestros tutores, Yonis, Juan Carlos, Nairys, y Dionner. Por dedicarnos tiempo para que lográramos este sueño.

A la gente del barrio. A los de mi grupo los que llegaron y los que llegarán.

A mis suegros: Los de allá y los de más pa allá, a los cuatro por ser conmigo como si fueran mis padres.

A mi familia: Mi mamá la que siempre esta ahí dando su ejemplo para mí, mi papá por todo lo que ha hecho por mí. A los dos por las cosas que sé que están dispuestos a hacer por mí, por el cariño que me han demostrado, por la educación que me han dado. Por enseñarme a soñar y a vivir con la frente en alto de una manera honrada y honorable, incluso en los momentos mas difíciles. Por las lágrimas que sin merecer les he hecho derramar. A mis tíos: Migdalia, Papito, Pedro, Piro, y a todos los demás. A mis primos, los que están cerca y los que están lejos.

A mi otra familia; La de Morón, a toda.

A Nairet. A Lisi. A Jinet.

A mis amigos: Considero mis amigos a todos los del apto, a Mario, a Yohan, a Willy (a pesar de su ausencia), a la gente de la Horda, a la gente del Pre, a Marzabal, a Ludiel, al Gordo, a Osvaldo, Kusitin, Oscar, el Dino, el Colorao, Kamilo, a René, a Luis, a Yanedi, el Flaco, a Heidy, Dori, Carlos, Gabriel, Alioth, Ariel, Pedro, Zuleivy, Ben, y en especial para Adonis, por ser siempre un ejemplo para mí.

A mi esposa: Por aparecer en el momento indicado y quedarse para siempre, por dedicarme más tiempo a mí que a ella misma, por hacerme feliz y apoyarme siempre. Para ti en especial.

A mi hermano. De ti no se ni que decir, si me pongo a decir podría llenar un libro. Tenías razón, yo siempre te he querido imitar. Pero es porque veo en ti un modelo de persona el cual hubiese querido ser, espero algún día ser al menos parecido a tí.

DEDICATORIA

A mis cuatro abuelos, por ser los mejores ángeles que he tenido y tendré. Nadie como ellos me ha enseñado tanto. Más que mi esfuerzo es el milagro de tantas horas de amor infinito.

Lázaro

A mis padres.

Sergio

RESUMEN

En la UCI se desarrollan gran cantidad de proyectos en el área del software. Además de esto cuenta con un amplio número de activos informáticos que constituyen una prioridad a la hora de proteger los bienes de la Universidad. Los métodos empleados para gestionar la seguridad son insuficientes y no se utilizan de forma centralizada. En el presente trabajo se ha hecho una recopilación de la información fundamental acerca de las herramientas utilizadas para la seguridad informática, sobre todo en lo referente a OSSIM, con el cual se pretende centralizar y correlacionar las diferentes aplicaciones de seguridad que soporta. Esto permitirá un nivel más elevado de abstracción y una mejor administración de los servidores y equipos en general.

Se explican los mecanismos utilizados por OSSIM, su funcionamiento y pasos a seguir para ser instalado y configurado. Esta poderosa herramienta es una de las mejores de su tipo, cuenta con numerosas ventajas y facilidades que contribuyen al trabajo eficiente de los administradores de sistemas y redes haciendo a una organización más segura y eficiente.

SUMARY

In the UCI develop large number of projects in the area of software. In addition to this account with a large number of IT assets that are a priority when it comes to protecting the property of the University. The methods used for managing security are insufficient and are not used in a centralized manner. In this work has been done a compilation of key information about the tools used for computer security, especially as regards OSSIM, which seeks to centralize and correlate the different security applications it supports. This will allow a higher level of abstraction and better management of servers and computers in general.

It explains the mechanisms used by OSSIM, its operations and steps to be installed and configured. This powerful tool is one of the best of this kind, has many advantages and facilities which contribute to the efficient working of the system and network administrators to make an organization more secure and efficient.

PALABRAS CLAVE

Seguridad Informática, centralizar, monitorizar, correlación, protección, ataque, control de activos, herramienta de seguridad, Software Libre, administración de redes, OSSIM.

TABLA DE CONTENIDO

INTRODUCCIÓN

La informática, y la seguridad informática dentro de este campo, es la rama que actualmente se ha convertido en un tema crítico en la sociedad moderna mundial. De ahí que hoy en día, para combatir los problemas de Seguridad de la Información que se presentan, los administradores se han visto en la necesidad de implementar herramientas para la Gestión de la Seguridad como antivirus, monitores de red y sistemas, detectores de vulnerabilidades, analizadores de logs, proxies, firewall, IDS, por citar las más destacadas por su utilidad y popularidad; pero a pesar de las medidas tomadas los sabotajes informáticos, los accesos no autorizados y el uso indebido de los activos se siguen dando.

La solución que han encontrado muchos de los usuarios es poner un firewall, el cual cierra todos aquellos puertos que no usa, reduciendo de esta forma la posibilidad de ataque. Esto puede estar bien intencionado, pero si embargo, no es suficiente pues el firewall representa una puerta que prohíbe el paso a todos aquellos servicios no autorizados, pero que deja pasar aquellos que el usuario detrás del Firewall necesita usar. Y ahí está el problema, pues aunque sólo se permitan los servicios básicos y teóricamente seguros, existen agujeros que se pueden aprovechar, ante los cuales el Firewall no puede hacer nada. Todo esto suponiendo, claro está, que la integridad del Firewall no haya sido comprometida.

Para suplir algunos de estos problemas aparece una herramienta que conduce a la optimización de los procesos, sentando las bases en redes seguras y proporcionando así un mejor servicio en las empresas. Esta herramienta, cuyo objetivo es ofrecer un marco para centralizar, organizar y mejorar las capacidades de detección y visibilidad en la monitorización de eventos de seguridad de la organización es la llamada OSSIM.

Si se da una mirada al pasado, se puede ver y comprobar cómo ha ido evolucionando el campo de la seguridad informática, desde los primeros cortafuegos hasta los más avanzados IDS. De todo ello ha surgido una visión común basada en la necesidad de asegurar los sistemas y así proteger los datos. Poco a poco, pero de forma efectiva, el tema de la seguridad ha ido ocupando un espacio importante en las empresas y eso ha hecho que la tecnología en cuanto a esto, evolucione de manera vertiginosa y eficiente. Es aquí cuando tiene cabida un proyecto como OSSIM, una consola de seguridad central con 22 herramientas de open source, que no sólo pone a trabajar juntos estos programas, sino que recoge y ordena la información que cada uno de ellos genera, y la cruza, para hacer valoraciones sobre el estado de la red o buscar patrones que sirvan para detectar si está siendo atacada. Además de esta, sin duda, otra de las mayores ventajas de dicha herramienta, es que es un software de open

source, y que dado esto, las empresas no tienen que utilizar tantos recursos ni en cada nuevo dispositivo tecnológico que sale al mercado, ni en personal especializado para poder afrontar tal cantidad de eventos.

La informática en Cuba no está exenta a estos cambios. Para incidir positivamente en su desarrollo y lograr establecer en ella parámetros de excelencia es imprescindible implantar también estos métodos de seguridad adaptándolos creativamente a las condiciones concretas del país. Es por ello que surge la importancia de desarrollar este trabajo, ya que OSSIM es una distribución de productos de open source integrados para construir una infraestructura de monitorización de seguridad y que va a ayudar a erradicar los grandes problemas que existen en el país en cuanto al desarrollo del campo que encierra a la seguridad informática, ya que esta es una rama de dicha ciencia a la que le falta mucho por ampliar en conocimientos y tecnologías de avanzada.

La Universidad de las Ciencias Informáticas (UCI), como líder del proceso de desarrollo de software en el país, desarrolla una gran cantidad de proyectos, por lo cual se plantea la necesidad de emplear una seguridad máxima en sus redes informáticas. La red de la UCI cuenta en estos momentos con el sistema de detección de intrusos Snort, firewalls y programas antivirus pero aún así la seguridad no alcanza el nivel necesario para mantener la integridad y confiabilidad de la información que se maneja. Por tanto, es necesario recurrir a una herramienta que controle las aplicaciones de seguridad que se utilizan en la UCI con la visión futura de poner en funcionamiento, y centralizar además, otras aplicaciones de seguridad que en estos momentos no son empleadas en la universidad. Para ello se propone utilizar la herramienta OSSIM.

Para lograr poner en práctica satisfactoriamente la herramienta OSSIM dentro de la UCI se deben establecer las configuraciones exactas y óptimas que den como resultado un buen desempeño de esta herramienta en las redes de la UCI. En estos momentos, dichas configuraciones constituyen un problema a resolver puesto que no se cuenta con la información necesaria para hacer funcionar, atendiendo a las necesidades específicas, la herramienta OSSIM. De no realizarse este estudio se debería pagar, una alta suma, a alguna compañía extranjera por prestar este servicio de configuración y luego de mantenimiento a la Universidad.

OSSIM constituye un problema cuando a configuraciones y mantenimiento se refiere, pues esos servicios los presta la empresa española IT Deusto. El software OSSIM es gratis y se puede descargar desde su página principal www.ossim.net, pero el soporte técnico hay que solicitarlo y pagarlo a la empresa IT Deusto.

Esta situación problémica, unida a que en la UCI no se cuenta con especialistas o ingenieros expertos en el uso y configuración de OSSIM, es el motivo principal de esta investigación, con la cual se pretende resolver el problema de configuración y personalización de OSSIM para su posterior utilización en la Universidad. A partir de los planteamientos anteriores se plantea el siguiente problema: ¿Cómo configurar y personalizar la herramienta OSSIM para su perfecta y eficiente utilización en la UCI?

En general, la solución propuesta facilitará el trabajo con herramientas de seguridad y sobre todo ayudará en la obtención de conocimientos que estarán dados por los resultados que se obtendrán al utilizar OSSIM. De ahí la importancia del presente trabajo.

El objeto de estudio de este trabajo lo constituye: el proceso de análisis y configuración de las herramientas de monitoreo y seguridad de redes en la UCI y el campo de acción que abarca es: el proceso de análisis y configuración de la herramienta de monitoreo y seguridad de redes OSSIM en la UCI.

La idea a defender que se plantea es emplear la herramienta OSSIM utilizando una ayuda práctica de configuración y personalización, para aumentar así la calidad y el desempeño de la seguridad informática en la UCI.

Se propone como objetivo general analizar y configurar la herramienta de monitoreo OSSIM para aumentar la seguridad informática en la UCI. Este objetivo general se ha proyectado en varios objetivos específicos como son:

- Configurar las aplicaciones que va a integrar OSSIM en la UCI respecto a las necesidades y características de la misma.
- Organizar y desarrollar procedimientos para utilizar, configurar y personalizar la herramienta OSSIM como lo requieren las condiciones específicas de la UCI.

Entre los aportes prácticos esperados del trabajo se encuentran:

- Una ayuda que le permita a todos los trabajadores de seguridad informática de la Universidad conocer los procesos que se deben realizar para utilizar la herramienta.
- Métodos y procedimientos detallados de cómo configurar la herramienta OSSIM en la Universidad para utilizarla en el trabajo de forma más efectiva y eficiente.

- Adquirir experiencia con la herramienta OSSIM para aplicar como solución libre en la seguridad de redes de la UCI.

Para alcanzar los objetivos planteados se trazaron las siguientes tareas:

- Investigar toda la información existente sobre la herramienta OSSIM.
- Investigar las características de la red en la UCI.
- Estudio de diferentes enfoques sobre la utilización de la herramienta OSSIM.
- Seleccionar el sistema operativo para instalar OSSIM.
- Definir las aplicaciones soportadas por la consola de OSSIM que se van a utilizar en la UCI.
- Configurar y aplicar la herramienta OSSIM en una pequeña subred de un laboratorio de proyecto para realizar pruebas.

1 CAPÍTULO
FUNDAMENTACIÓN TEÓRICA

1.1 INTRODUCCIÓN

La seguridad informática desempeña un rol significativo en el avance científico-técnico del mundo actual. Es por ello que hablar de seguridad informática hoy en día, no supone un alarde de modernidad y novedad; pues con el desarrollo de los ordenadores personales, la vertiginosa evolución de Internet, la implantación del comercio electrónico y el impulso de la denominada "Sociedad de la Información", todo el mundo habla, sabe y se preocupa de la seguridad en estos ámbitos. De una estructura informática basada en sistemas propietarios y grandes servidores manejada por personal técnico, con una formación muy específica y alejada del conocimiento común del resto de los mortales, se ha evolucionado a otra más amigable y cercana al usuario final. Ello ha supuesto que los niveles iniciales de conocimiento sean rápidamente adquiridos por cualquier persona interesada, sin especiales conocimientos técnicos en la materia. Se entiende que los sistemas anteriores no eran más seguros que los actuales, tan sólo eran mucho más desconocidos.

La necesidad de asegurar los medios informáticos y la información ha dado lugar a una alta demanda de herramientas que controlen y monitoreen las redes informáticas. Los virus, hackers y toda clase de programas malignos, ataques o acceso no autorizado a información o recursos informáticos son, en la actualidad, el punto de atención y debate de todas o casi todas las empresas o instituciones del planeta, dando lugar al amplio debate de los expertos que se plantean y crean aplicaciones para asegurar la confidencialidad, integridad y la disponibilidad de la información. A raíz de esto surgen las herramientas de seguridad y monitoreo de redes informáticas como OSSIM, que centraliza a varias aplicaciones de este tipo, recogiendo y ordenando la información que generan y cruzándola, para hacer valoraciones sobre el estado de la red o buscar patrones que sirvan para detectar si está siendo atacada.

Basándose en estas ideas, en este capítulo se desglosan los elementos teóricos relacionados con este tema, se abordan los aspectos vinculados con la seguridad informática a nivel mundial, en Cuba y en la universidad. Se exponen las principales características de las herramientas más utilizadas y se

presentan los diferentes enfoques sobre la importancia, ventajas y beneficios de utilizar la seguridad en el mundo informático de hoy en día.

1.2 SEGURIDAD INFORMÁTICA

Se puede entender como seguridad un estado de cualquier sistema que nos indica que el mismo está libre de peligro, daño o riesgo. Se entiende como peligro o daño todo aquello que pueda afectar su funcionamiento directo o los resultados que se obtienen del mismo. Por lo tanto, la seguridad informática consiste en asegurar que los recursos del sistema de información de una organización sean utilizados de la manera que se decidió y que el acceso a la información allí contenida así como su modificación sólo les sea posible a las personas que se encuentren acreditadas y dentro de los límites de su autorización. Para la mayoría de los expertos el concepto de seguridad en la informática es utópico porque no existe un sistema 100 por ciento seguro.

La seguridad a nivel informático no se limitará entonces a la posibilidad de evitar la adulteración de la información o intromisión no autorizada a lugares restringidos de acceso, sino también a que los equipos donde se opera y almacena la información sean confiables, siendo la seguridad general establecida, tan buena como la menor seguridad de cualquier componente.

Pero las estadísticas mundiales indican que los usuarios están preocupados más por la probabilidad que tiene un experto en filtrarse dentro de la información existente y adulterarla, que por saber si el sistema se va a detener y no funcionar por un período de tiempo por problemas de hardware.

Estas tendencias mundiales han llevado a que la seguridad informática sea acotada sólo a lo referente a la violación de claves de acceso, redes, sistemas, protecciones contra copias, en general a la seguridad del software.

Un concepto general de seguridad informática sería aquel definido como el conjunto de procedimientos y acciones encaminados a obtener la garantía de funcionamiento del sistema de información, logrando eficacia, entendida como el cumplimiento de la finalidad para el que estaba determinado, manteniendo la integridad, confidencialidad y consistencia de los datos, el sistema y los recursos informáticos. Si se consigue todo esto, labor nada fácil, se podrá decir que se cuenta con un sistema seguro.

Para que un sistema se pueda definir como seguro debe tener estas cuatro características:

- ***Integridad:*** *La información sólo puede ser modificada por quien está autorizado.*

- ***Confidencialidad:*** *La información sólo debe ser legible para los autorizados.*
- ***Disponibilidad:*** *Debe estar disponible cuando se necesita.*
- ***Irrefutabilidad:*** *Que no se pueda negar la autoría.*

Dependiendo de las fuentes de amenaza, la seguridad puede dividirse en seguridad lógica y seguridad física.

En estos momentos la seguridad informática es un tema de dominio casi obligatorio para cualquier usuario de la Internet, para no permitir que su información sea robada. (1)

1.2.1 Principales conceptos

Sobre la seguridad informática existen muchos términos y conceptos que son necesarios conocer para adentrarse en el conocimiento de este mundo.

- **Activo:** recurso del sistema de información o relacionado con éste, necesario para que la organización funcione correctamente y alcance los objetivos propuestos.
- **Amenaza:** es un evento que pueden desencadenar un incidente en la organización, produciendo daños materiales o pérdidas inmateriales en sus activos.
- **Impacto:** medir la consecuencia al materializarse una amenaza.
- **Riesgo:** posibilidad de que se produzca un impacto determinado en un Activo, en un Dominio o en toda la Organización.
- **Vulnerabilidad:** posibilidad de ocurrencia de la materialización de una amenaza sobre un Activo.
- **Ataque:** evento, exitoso o no, que atenta sobre el buen funcionamiento del sistema.
- **Desastre o Contingencia:** interrupción de la capacidad de acceso a información y procesamiento de la misma a través de computadoras necesarias para la operación normal de un negocio.

1.2.2 Objetivos de la seguridad informática

La seguridad informática esta dirigida a proteger los activos. Para ello se divide en tres grupos los cuales se tratan de diferentes formas a la hora de su aseguramiento. El objetivo de la seguridad informática también puede tratarse en dos áreas de forma general: el área del hardware y el área del software.

1. **La información:** Es el objeto de mayor valor para una organización, el objetivo es el resguardo de la información, independientemente del lugar en donde se encuentre registrada, en algún medio electrónico o físico.
2. **Equipos que la soportan:** Software, hardware y organización.
3. **Usuarios:** Individuos que utilizan la estructura tecnológica y de comunicaciones que manejan la información.

Dentro del área del hardware es priorizada la atención fundamentalmente a servidores, clientes y líneas de comunicaciones.

Los servidores, especialmente en instalaciones intermedias y grandes, suelen estar situados en nodos centrales, agrupados y en dependencias específicas como Centros de Procesos de Datos.

Los clientes son aquellos equipos remotos que interactúan entre sí o con los servidores.

Las líneas de comunicaciones son las redes que por varias vías enlazan la comunicación.

Dentro del área de software los objetos de atención son sistemas operativos, bases de datos y aplicaciones.

1.2.3 Importancia de gestionar la seguridad informática

El propio avance tecnológico de la sociedad impone la gestión de la seguridad informática como norma obligatoria. Junto con dicho avance se hacen mas frecuentes cada día los casos de fraude, robo, y acciones poco éticas a través de los medios de comunicación. Es por eso de gran importancia mantener un adecuado control sobre los posibles fallos de seguridad que pueden existir en los sistemas, los cuales son potenciales riesgos que deben ser mitigados.

Por otra parte existen varios niveles de seguridad los cuales están incluidos en una escala que dicta que para alcanzar cada uno de ellos es necesario tener implícito a los niveles inferiores.

El activo más importante que posee una empresa es la información y, por lo tanto, deben existir técnicas que la aseguren, más allá de la seguridad física que se establezca sobre los equipos en los cuales se almacena. Estas técnicas las brinda la seguridad lógica que consiste en la aplicación de barreras y procedimientos que resguardan el acceso a los datos y sólo permiten acceder a ellos a las personas autorizadas para hacerlo. Para desarrollar dichas técnicas es necesario evaluar los riesgos y trabajar en base al análisis de los mismos.

El análisis de riesgos consiste en listar los riesgos y la probabilidad de ocurrencia de cada uno de ellos. Lo cual arroja como resultado una base sobre la cual hay que apoyarse para mitigarlos.

Existe un viejo dicho en la seguridad informática que dicta: *"lo que no está permitido debe estar prohibido"* (2) y ésta debe ser la meta perseguida. Y los medios para conseguirla son:

- Restringir el acceso (de personas de la organización y de las que no lo son) a los programas y archivos.
- Asegurar que los operadores puedan trabajar pero que no puedan modificar los programas ni los archivos que no les estén permitidos cambiar (sin una supervisión minuciosa).
- Asegurar que se utilicen los datos, archivos y programas correctos en y por el procedimiento elegido.
- Asegurar que la información transmitida sea la misma que reciba el destinatario al cual se ha enviado y que no le llegue a otra persona.
- Asegurar que existan sistemas y pasos de emergencia alternativos de transmisión entre diferentes puntos.
- Organizar a cada uno de los empleados por jerarquía informática, con claves distintas y permisos bien establecidos, en todos y cada uno de los sistemas o aplicaciones empleadas.
- Actualizar constantemente las contraseñas de accesos a los sistemas de cómputo.

El no uso de este tipo de precauciones puede exponer a las empresas a la ocurrencia de un desastre, el cual puede ser más costoso que los recursos invertidos en hacer un trabajo de análisis de riesgos.

1.2.4 Necesidad de utilizar la seguridad informática

La creciente demanda en el aseguramiento de la información esta dado por el nivel de importancia que tiene para cada empresa el asegurar que sus recursos informáticos sean utilizados de la manera que se decidió, y por la proliferación cada vez más amplia de programas malignos, ataques y violaciones dentro de empresas u organizaciones, ya sea por agentes externos como internos. De esta manera, la necesidad de utilizar la seguridad informática viene dada principalmente por las amenazas que podrían afectar el buen funcionamiento de una organización.

Una vez que la programación y el funcionamiento de un dispositivo de almacenamiento o transmisión de la información se consideran seguros, todavía deben tenerse en cuenta las circunstancias "no informáticas" que puedan afectar los datos, las cuales son a menudo imprevisibles o inevitables, de modo que la única protección posible es la redundancia y la descentralización, por ejemplo, mediante estructura de redes.

Estos fenómenos pueden ser causados por:

- **El usuario:** puede que no este comprometido con el cuidado de los medios que utiliza, no se da cuenta de los errores que comete, o porque a propósito ejecuta una violación de las normas de seguridad establecidas.
- **Programas maliciosos:** Es instalado en el ordenador abriendo una puerta a intrusos, o bien modificando los datos. Estos programas pueden ser: un virus informático, un gusano informático, un troyano, una bomba lógica o un programa espía o Spyware.
- **Un intruso:** persona que consigue acceder a los datos o programas de los cuales no tiene acceso permitido.
- **Un siniestro (robo, incendio, inundación):** una mala manipulación o una mala intención derivan a la pérdida del material o de los archivos.
- **El personal interno de Sistemas:** Las pujas de poder que llevan a disociaciones entre los sectores y soluciones incompatibles para la seguridad informática.

1.2.5 Políticas de seguridad

Las políticas de seguridad son las reglas y procedimientos que regulan la forma en que una organización previene, protege y maneja los riesgos de diferentes daños. Tienen como objetivo

informar al mayor nivel de detalle a los usuarios, empleados y gerentes de las normas y mecanismos que deben cumplir y utilizar para proteger los componentes de los sistemas de la organización.

Están compuestas por varias políticas específicas que regulan un área o función en especial. Por lo general se puede hablar de una política de seguridad de una empresa u organización, aunque esta esté compuesta por varias políticas de seguridad, pues en su conjunto contribuyen a un mismo fin: regular y asegurar los bienes, usuarios y servicios de la misma. Estos componentes son:

- Una política de privacidad.
- Una política de acceso.
- Una política de autenticación.
- Una política de contabilidad.
- Una política de mantenimiento para la red.
- Una política de divulgación de información.

Una política de seguridad informática es una forma de comunicarse con los usuarios, ya que las mismas establecen un canal formal de actuación del personal, en relación con los recursos y servicios informáticos de la organización.

No se puede considerar que una política de seguridad informática es una descripción técnica de mecanismos, ni una expresión legal que involucre sanciones a conductas de los empleados, es más bien una descripción de lo que se desea proteger y él por qué de ello, pues cada política de seguridad es una invitación a cada uno de sus miembros a reconocer la información como uno de sus principales activos así como, un motor de intercambio y desarrollo en el ámbito de sus negocios. Por tal razón, las políticas de seguridad deben concluir en una posición consciente y vigilante del personal por el uso y limitaciones de los recursos y servicios informáticos.

1.3 TÉCNICAS DE ASEGURAMIENTO DEL SISTEMA

Para asegurar un sistema informático hay que usar mecanismos o métodos desarrollados por personal con experiencia. En la mayoría de los casos es aceptable técnicas como:

- *Codificar la información: usando Criptología, Criptografía, Criptociencia, o contraseñas difíciles de averiguar a partir de datos personales del individuo.*
- *Vigilancia de red.*
- *Tecnologías repelentes o protectoras: cortafuegos, sistema de detección de intrusos - anti-spyware, antivirus, llaves para protección de software, etc. Mantener los sistemas de información con las actualizaciones que más impacten en la seguridad.*

Se deben tener consideraciones en cuanto a software y a red. Cada caso enfocado a disminuir la posibilidad de ocurrencia de desastres y aseguramiento de la recuperación después de la ocurrencia de un desastre.

Tener instalado en la máquina únicamente el software necesario reduce riesgos. Así mismo, tener controlado el software asegura la calidad de la procedencia del mismo. En todo caso un inventario de software proporciona un método correcto de asegurar la reinstalación en caso de desastre. El software con métodos de instalación rápidos facilita también la reinstalación en caso de contingencia.

Las consideraciones de una red son las decisiones tomadas por los administradores respecto a que interesa monitorizar y controlar dentro de la misma.

Los puntos de entrada en la red son generalmente el correo, las páginas web y la entrada de ficheros desde discos, o de ordenadores ajenos, como portátiles.

Mantener al máximo el número de recursos de red sólo en modo lectura, impide que ordenadores infectados propaguen virus. En el mismo sentido se pueden reducir los permisos de los usuarios al mínimo.

Controlar y monitorizar el acceso a Internet puede detectar, en fases de recuperación, cómo se ha introducido el virus. (3)

1.4 PRINCIPALES HERRAMIENTAS PARA LA SEGURIDAD INFORMÁTICA

Una herramienta de seguridad informática es una aplicación, un conjunto de ellas o un sistema que se encarga de gestionar la seguridad informática de un ordenador, de una red o un sistema de redes garantizando la detección y protección de ataques, intrusiones, invasiones y/o violaciones de tipo informático, por parte de personas o programas no autorizados. Estas herramientas deben reforzar la

permanencia del sistema informático, en condiciones de actividad adecuadas para que los usuarios accedan a los datos con la frecuencia y dedicación que requieran.

Según su calificación ya sea un antivirus, IDS, detector de anomalías, firewall, proxie o monitor de red, disponibilidad o de otro tipo, se dedica a detectar y eliminar virus o todo tipo de programas malignos, detectar intrusos, monitorear una red, detectar anomalías, inventariar hardware y software.

El uso de estás herramientas constituyen métodos para asegurar los recursos informáticos de diferentes tipos de contingencias. Estos métodos se clasifican en activos si se trata de vigilancia y monitoreo, y pasivos si se refiere solamente a garantizar la recuperación después de un desastre. Por ejemplo:

Activos:

Antivirus: Detectan y eliminan virus y otros programas informáticos maliciosos.

Filtros de ficheros: Genera filtros para ficheros dañinos si el ordenador está conectado a una red. Estos filtros se denominan cortafuegos. Un cortafuego o firewall es un elemento de hardware o software utilizado en un equipo o red de ordenadores previniendo intrusiones no deseadas. Crea un punto de control de entrada y salida de tráfico de datos de un equipo o una red.

Anti Espías: Programa que evita que un virus propague información de nuestro ordenador por la red.

Anti Spam: Evita que un buzón de correo se sature de e-mails no deseados.

Pasivos:

Copias de seguridad: mantener una política de copias de seguridad garantiza la recuperación de los datos y la respuesta cuando los métodos activos no han funcionado.

1.4.1 Herramientas top de seguridad

Las herramientas que dentro de su clasificación se encuentran entre las más usadas y distribuidas a nivel mundial, y que por el uso de sus funcionalidades han obtenido mejores resultados son las llamadas herramientas top. A continuación se presenta un resumen de las mismas:

Nessus: Es una herramienta de auditoría de seguridad. Posibilita evaluar módulos de seguridad buscando puntos vulnerables que deberían ser reparados.

Nagios: Es un sistema open source para monitorizar una red. Ofrece un informe sobre el estado los hosts y servicios que se especifiquen, alertando cuando el comportamiento de la red no es el deseado y nuevamente cuando vuelve a su estado correcto.

Snare: Es un programa de open source que soporta el registro de eventos de seguridad, de aplicación y del sistema así como los del DNS. Envía la información registrada a un servidor SNARE remoto o a un servidor Syslog remoto.

Netcat: Es una herramienta multiuso para TCP/IP. Es una utilidad para Unix, que lee y escribe datos a través de conexiones de red usando los protocolos TCP o UDP. Es útil para depurar y explorar, ya que puede crear casi cualquier tipo de conexión que puedas necesitar.

Tcpdump: Es una herramienta para el monitoreo y la adquisición de datos en redes. Puede ser usado para imprimir los encabezados de los paquetes en una interfaz de red que concuerden con una cierta expresión.

Snort: Es un Sniffer/logger de paquetes flexible que detecta ataques. Está basado en la biblioteca libpcap y puede ser usado como un "sistema de detección de intrusiones" (IDS) de poco peso.

P0f: Es una herramienta para monitorear sistemas operativos de forma pasiva, ya que puede identificar los sistemas de las máquinas que se conectan a tu servidor y de las máquinas a las que te conectas desde tu servidor.

Saint: Es una herramienta de evaluación de seguridad basada en SATAN. Incluye escaneos a través de un firewall, chequeos de seguridad, 4 niveles de severidad (rojo, amarillo, marrón y verde) y una interfaz HTML abundante en características.

Internet Security Scanner: Un escáner de seguridad comercial muy popular.

Abacus Portsentry: Este demonio de detección de escaneo de puertos tiene la habilidad de detectar estos tipos de acciones en las interfaces de red de la máquina.

DSniff: Es un comprobador de integridad de archivos y directorios.

CyberCop: Es un escáner comercial. Esta herramienta es bastante costosa y no viene con el código fuente. Una poderosa versión de demostración está disponible para prueba.

SARA: Es una herramienta de análisis de seguridad de tercera generación que está basada en el modelo de SATAN y distribuida bajo una licencia del estilo de la GNU GPL.

Hping2: Es una herramienta de red capaz de enviar paquetes ICMP/UDP/TCP hechos a medida y de mostrar las respuestas del host destino de la misma manera en la que lo hace la herramienta `ping' para las respuestas de ICMP. Puede manejar fragmentación y tamaños y cuerpo arbitrarios de paquetes.

Sniffit: Una herramienta de monitoreo para paquetes de TCP/UDP/ICMP.

SATAN: Herramienta de Auditoría de Seguridad para Analizar Redes.

IPFilter: Es un filtro de paquetes de TCP/IP, adaptable para uso en un ambiente de firewall.

Strobe: Es un escáner de puertos de TCP de alta velocidad.

L0pht Crack: Es una herramienta de auditoría de contraseñas para NT. Computa las contraseñas de usuarios de NT a partir de las hashes criptográficas que son guardadas por el sistema operativo NT.

John The Ripper: Es una herramienta activa para crackear contraseñas. Útil para encontrar contraseñas débiles de tus usuarios.

Hunt: Es un sniffer de paquetes y un intruso avanzado en conexiones que puede intercalarse en la misma, observarla y reiniciarla.

OpenSSH / SSH: En una herramienta derivada de la versión de ssh de OpenBSD, que a su vez deriva del código de ssh pero de tiempos anteriores a que la licencia de ssh se cambiara por una no libre. SSH es un programa para autenticarse en una máquina remota y ejecutar comandos en la misma.

Ntop: Muestra la utilización de la red al estilo de la herramienta `top'. Muestra un sumario del uso de la red por parte de las máquinas, en un formato que recuerda a la utilidad de Unix `top'.

traceroute/ping/telnet: Estas son utilidades que básicamente todas las máquinas con UNIX ya tienen.

NAT (NetBIOS Auditing Tool): La herramienta de auditoría de NetBIOS está diseñada para explorar los servicios de NetBIOS que ofrece un sistema que permiten compartir archivos.

Sam Spade: Herramientas online para investigar direcciones de IP y para seguir el rastro de spammers.

NFR: Una aplicación de sniffing comercial para crear sistemas de detección de intrusión (IDS).

Logcheck: Envía al administrador mensajes por correo electrónico informando de las anomalías en los archivos de registro del sistema.

Cheops: Es una herramienta basada en GTK. Ofrece una interfaz simple a la mayoría de las utilidades de red, mapea redes locales o remotas y puede mostrar tipos de sistemas operativos de las máquinas en la red.

Cerberus Internet Scanner: Es un escáner gratis desarrollado y mantenido por la empresa `Cerberus Information Security, Ltd' y está diseñado para ayudar a los administradores a localizar y reparar agujeros de seguridad.

OpenBSD: El proyecto OpenBSD produce un sistema operativo libre, multi-plataforma del estilo del UNIX.

LSOF: Es una herramienta de diagnóstico específica de Unix. Lista información acerca de cualquiera archivo abierto por procesos que están actualmente corriendo en el sistema.

Lids: Es un sistema de detección y defensa de intrusión en Linux. El objetivo es proteger a sistemas con Linux para prevenir intrusiones a nivel de root, deshabilitando algunas llamadas a sistema en el kenel mismo.

IPTraf: Es un monitor de IP en LAN que genera varias estadísticas de red incluyendo información sobre TCP, conteos de UDP, información de ICMP y OSPF; información sobre Ethernet y estadísticas por nodo.

IPLog: Es una herramienta de registro de tráfico TCP/IP.

Fragrouter: Apunta a probar la exactitud de un NIDS de acuerdo a ataques específicos a TCP/IP.

Nmap: Es un programa de código abierto que sirve para efectuar rastreos de puertos TCP y UDP. Se usa para evaluar la seguridad de sistemas informáticos, así como para descubrir servicios o servidores en una red informática.

1.5 LA TECNOLOGÍA SIM

A causa de la gran cantidad de herramientas informáticas, y de lo difícil que se hace el trabajo con las alarmas que emite cada una de ellas surge una tecnología que integra los resultados de los sistemas de seguridad en una consola más comprensible para los administradores de redes. Dicha tecnología es conocida como SIM.

Un SIM es un sistema que de forma centralizada colecta, normaliza, correlaciona y prioriza los eventos de seguridad. Estos presentan los datos de seguridad de manera tal que puedan ser "digeridos" por el personal de seguridad. Su propósito es aumentar la eficiencia y la eficacia, pero además dar una vista global del estado general de seguridad, un excelente punto de partida para analizar el estado de la red, que antes no existía.

Diagrama simplificado del sistema:

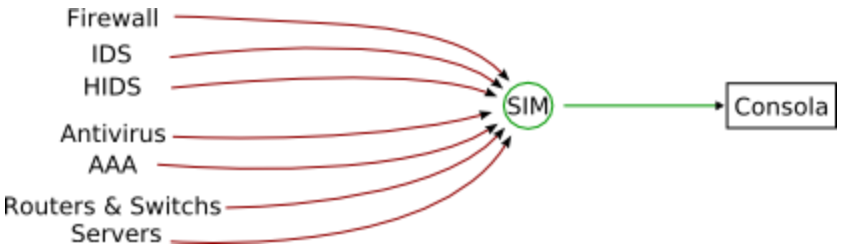


Figura #1: Sistema SIM.

Los SIMs propietarios no son productos baratos, su precio puede variar desde 50.000 a 200.000 dólares. Esto puede parecer una cantidad exorbitante, pero mantener el mismo nivel de seguridad sin disponer de esta herramienta, implicaría por los menos disponer del doble de personal en seguridad, dedicado a la sola tarea de chequear los eventos de seguridad, y de todas formas puede ser que se nos escapen cosas. Además con lo difícil que es conseguir profesionales en seguridad, es un desperdicio asignarlos a filtrar eventos poco relevantes. (4)

Muchos desarrolladores de SIMs son propietarios pero también existen proyectos de software libre que implementan un sistema este tipo de sistemas como son OSSIM y Prelude, ambos con un grado importante de desarrollo.

Los SIM son una tecnología muy reciente y aunque algunos analistas consideran que las versiones actuales son más bien "administradores de eventos", aún así la mínima funcionalidad de tener

disponible la información de seguridad en un solo lugar; puede aliviar el trabajo de los administradores de seguridad, haciéndoles disponer de más tiempo para otras tareas.

Algunas de las características que debe cumplir o tener un SIM son:

- Colectar y normalizar las alarmas de un gran número de dispositivos al menos de los que se tienen instalados en la actualidad.
- Posibilitar el desarrollo de agentes, a través de algún lenguaje de scripting, para dar soporte a sistemas de seguridad no soportados.
- Disponer de un formato sencillo para agregar reglas de correlación.
- Correlacionar los eventos de seguridad en tiempo real.
- Disponer de un inventario con información de los dispositivos de red, servidores, PC de escritorio, relevancia de los dispositivos y ubicación. Además debe utilizar esta información para priorizar las alertas.
- Generar reportes los cuales deben ser muy personalizables y el trabajo con ellos también debe ser sencillo.

1.5.1 Herramientas que utilizan la tecnología SIM

Algunas de las herramientas más prometedoras que implementan este tipo de tecnología son: ArcSight ESM, Cisco Works SIMS, Cisco MARS que son privativas y por tanto comerciales cuyo precio no es nada barato. Por parte del software libre están OSSIM cuya distribución es gratis y Prelude. No por ser libre y gratis carece OSSIM de calidad pues es considerado por los expertos como una potente herramienta de seguridad y dan fe de ello las muchas y reconocidas empresas e instituciones que lo utilizan. Por tanto OSSIM es una solución libre a los problemas de seguridad sobre todo a la hora de centralizar las aplicaciones de monitoreo de redes, detección de intrusos, entre otras que se desarrollan en software libre y que se utilizan en la mayoría de las redes en el mundo.

Prelude (Hybrid Intrusion Detection System): *Está caracterizado como un IDS híbrido, porque trabaja como un IDS basado en Host y en Red al mismo tiempo, es un producto que permite colectar los reportes de todas las aplicaciones de seguridad en un sistema centralizado. A diferencia de*

OSSIM, Prelude usa el estándar IDMEF (Intrusion Detection Message Exchange Format) para la comunicación de los eventos.

Prelude soporta Snort, Nessus, Samhaim y más de 30 tipos de log de sistema. Sin embargo, al motor de correlación le falta bastante desarrollo, aunque es un proyecto muy activo. (4)

1.6 EL SOFTWARE LIBRE EN EL MUNDO DE LA SEGURIDAD

El mundo del open sourse, como también se le conoce al software libre gracias a las comunidades de desarrollo que posee, ha avanzado mucho también el campo de la seguridad informática. De ahí que muchas de las llamadas herramientas top son de código libre.

Se entiende como software libre, aquel programa o conjunto de ellos de los cuales el usuario puede disponer del código fuente, sin restricciones. (5) También puede modificar dicho código y redistribuirlo. Estas libertades garantizadas al usuario del software no son contrarias a los derechos legítimos del autor del programa, es decir, el autor del programa no pierde todos sus derechos sobre el mismo.

1.6.1 Fallos de seguridad en la utilización del software libre

Las herramientas de software libre pueden presentar distintos tipos de fallos de seguridad, de los cuales se puede hacer un análisis agrupándolos según el tipo. Este Como resultado de este análisis se puede enfocar cómo distintos tipos de software ayudan a solventarlos. De forma simple, se pueden dividir en tres bloques:

- *Fallos debidos a errores desconocidos en el software, o conocidos sólo por terceras entidades hostiles.*
- *Fallos debidos a errores conocidos pero no corregidos en la copia en uso del software.*
- *Fallos debidos a una mala configuración del software, que introduce vulnerabilidades en el sistema.*

El primero de ellos se puede achacar a la calidad del código, el segundo a la capacidad y celeridad de arreglo de los errores descubiertos en el código por parte del proveedor del mismo y a la capacidad del administrador de recibir e instalar nuevas copias de este software actualizado. El tercer tipo de vulnerabilidades puede achacarse, sin embargo, a una falta de documentación del software o una falta

de formación adecuada de los administradores para hacer una adaptación correcta del mismo a sus necesidades. (5)

1.6.2 Ventajas del software libre

Si se analiza la descripción realizada previamente de la definición de software libre se derivan una serie de ventajas principales de este tipo de software sobre el software propietario, algunas de las cuales son muy adecuadas para el mundo de la seguridad. Estas ventajas son:

- Al disponer del código fuente de los programas en su completitud, éste puede ser analizado por terceras personas ajenas a sus autores en busca de fallos de diseño o de implementación. Es decir, cualquiera con los conocimientos necesarios puede realizar una auditoría de dicho código.

- La posibilidad de realizar modificaciones libremente al código fuente y distribuirlas permite que cualquiera pueda ofrecer mejoras sobre éste. Estas mejoras podrán ser nuevas funcionalidades que se incorporen al mismo o parches que corrijan problemas detectados anteriormente.

- Las características del software libre hacen que no sea lógico cargar costes sobre el software en sí, lo que permite que este tipo de software pueda ser utilizado por organizaciones y personas con menos recursos económicos. Esto se presenta como una ventaja cuando se compara con los precios de lo que cuesta el software de seguridad propietario.

- De igual forma, la posibilidad de modificar libremente el software permite que las organizaciones lo adapten a sus propias necesidades, pudiendo eliminar funcionalidades que no le sean de interés.

1.6.3 Desventajas del software libre

El uso de software libre no está exento de desventajas. Así se podrían enumerar las siguientes:

- La posibilidad de una generación más fácil de troyanos, dado que el código fuente también puede ser modificado con intenciones maliciosas. Si el troyano logra confundirse con la versión original puede haber problemas graves.

- Al no tener un respaldo directo, la evolución futura de los componentes software no está asegurada o se hace demasiado despacio.

En mayor o menor grado, algunas de estas desventajas pueden tener solución. Por ejemplo, la difusión de troyanos se limita mediante el uso de técnicas de firma digital para garantizar la inviolabilidad del código o binarios transmitidos. De igual forma, los problemas de evolución futura parecen quedar resueltos con un cambio de paradigma por parte de las compañías de software. Es el cambio de un modelo de negocio de cobro por productos a cobro por servicios. Ya se observan, en el mundo de software libre, compañías que contratan a personal cualificado para hacer mejoras sobre proyectos libres para cubrir sus intereses pero haciendo públicas las modificaciones realizadas.

1.7 ESTADO DEL ARTE DE LA SEGURIDAD INFORMÁTICA Y EL USO DE OSSIM

Para dar solución a los problemas de seguridad en la uci se hizo un análisis cuantitativo y cualitativo del estado de la seguridad informática, de la utilización de herramientas de tipo SIM, y de OSSIM a nivel mundial, nacional, y en la universidad.

1.7.1 A nivel mundial

Además de lo expresado en los puntos anteriores podemos agregar que un mayor número de ejecutivos de todas las empresas privadas y organismos públicos continúan mejorando en la implementación de políticas y tecnologías de seguridad, a pesar que el ritmo de las mejoras es más lento que en años anteriores.

Lo que es más difícil de ignorar son las noticias de grandes organizaciones perdiendo laptops con datos personales sin encriptar de millones de clientes.

Una gran parte de los ejecutivos de seguridad admiten que no están cumpliendo con las regulaciones que dictan las medidas de seguridad que su organización debe tomar o se arriesgan a sanciones severas, pudiendo llegar a tiempo en prisión para los ejecutivos.

La disciplina de la seguridad informática todavía sufre del problema fundamental, hacer un caso de valor para la seguridad. La seguridad todavía esta vista y calculada como un costo, no como algo que podría añadir valor estratégico, transformándose en ingresos o ahorros.

Existe evidencia de que las organizaciones que cumplen con las leyes de seguridad son las que integraron y alinearon la seguridad con la estrategia de negocios y procesos de la compañía, lo que en retorno reduce el número de ataques exitosos y las pérdidas financieras que resultan de éstos. En breve, la seguridad puede crear un valor si es parte del plan de negocios de la organización y si el

ejecutivo en cargo es parte del equipo ejecutivo que toma estas decisiones estratégicas y de política. Sin embargo, la vasta mayoría de las compañías en el mundo todavía no han creado puestos de seguridad nivel C.

Manejar la seguridad estratégicamente, y a nivel ejecutivo, puede tener sentido en teoría, pero con el tiempo se está convirtiendo en un punto de debate en la sala de reuniones. Se necesitan pruebas para demostrar que el planeamiento de estrategias de seguridad funciona y así convencer al lado de negocios de la organización a que le hagan un lugar en la mesa ejecutiva.

Las compañías del sector financiero son las que tienen mejor porcentaje de contratar especialistas en seguridad informática. Los presupuestos de seguridad en el sector financiero son típicamente mayores que los presupuestos de IT como un todo y se aumentan más rápido también que en otros sectores. Esto puede ser causado porque las compañías financieras son las más cercanas al concepto de unir las políticas de seguridad con los procesos de negocios. Estas compañías son proactivas, instituyen los procesos formales de la seguridad informática como monitoreo de logs y pruebas periódicas de intrusión. La mayoría de sus empleados siguen las políticas de seguridad de la compañía. Como si esto fuera poco, las compañías financieras también han implementado más tecnología de seguridad, como detección de intrusos, herramientas de encriptación y soluciones para el manejo de identidad.

Consecuentemente, las empresas financieras han reportado menos pérdidas, menos caídas de la conectividad y menores incidentes de datos privados robados que cualquier otra industria.

La razón por todo esto es obvia. El producto en la industria financiera es el dinero, y el dinero es el objetivo principal de los criminales cibernéticos, incluyendo las mafias, hasta los terroristas. Proteger el dinero es la preocupación más crítica de la industria. En los años pasados se habían visto grandes crecimientos en crímenes informáticos. Cuando sea que un ejecutivo de seguridad pueda demostrarle a sus jefes que invertir en seguridad puede proteger e incluso aumentar el valor de sus acciones, él seguramente convencerá a la junta directiva a que inviertan y que le hagan a la seguridad un lugar en el plan estratégico de la organización.

Hoy en día el panorama de las amenazas es muy dinámico, lo que hace necesario adoptar nuevas medidas de seguridad. Los sistemas operativos tienen menos vulnerabilidades que pueden conducir a ataque masivos de virus de Internet.

Se ha venido reportando un crecimiento significativo en el número de vulnerabilidades del lado del cliente, incluyendo vulnerabilidades en los navegadores de Internet, programas de ofimática, en los reproductores de multimedia y en otras aplicaciones de escritorio.

Los usuarios que tienen permitido por sus empleadores navegar en Internet se han convertido en una fuente mayor de riesgo de seguridad para sus organizaciones. Unos años atrás, asegurar los servidores y servicios era visto como una tarea primaria para asegurar una organización. Hoy en día es igualmente importante, quizás aun más importante, prevenir que las computadoras de los usuarios sean expuestas mediante páginas web maliciosas u otros ataques dirigidos al lado del cliente.

La configuración por defecto para muchos sistemas operativos y servicios continúan siendo débiles y se siguen usando las contraseñas por defecto. Como resultado de esto, muchos sistemas han sido expuestos usando ataques de diccionario y fuerza bruta en el 2007.

Los hackers están encontrando más formas creativas para obtener información sensible de las organizaciones. Por lo tanto, ahora es crítico chequear la naturaleza de cualquier información que sale de los límites de una organización.

En el mundo en general se utilizan varias herramientas de seguridad informática con tecnología SIM como: ArcSight ESM, Cisco Works SIMS, Cisco MARS, todas estas de software propietario. Por parte del software libre están OSSIM y Prelude, este último con menos funcionalidades.

Como es evidente se utilizan una gran cantidad de aplicaciones y métodos como Antivirus, IDS´s como Snort, detectores de intrusos y vulnerabilidades como Nessus, cortafuegos, encriptación, analizadores de logs, proxies y contraseñas. También existe el caso del sistema TAIPS-net que es una combinación de los proyectos Honeywall y Nepehentes. El objetivo que persigue este sistema es integrar sistemas de potes de miel de baja interacción con funcionalidad de IDS/IPS, captura y análisis de tráfico en una sola computadora. TAIPS-net es un sistema de código abierto que se mantiene bajo licencia GPLv2.

A pesar de la gran cantidad de herramientas de seguridad que existe en el mundo no se ha logra eliminar problemas de seguridad.

Con OSSIM ganamos dinero (ahorro en licencias y hardware), y sobre todo ganamos en gestión de la seguridad.

OSSIM actualmente es conocido como una de las mejores plataformas de seguridad. Entre las organizaciones que lo han descargado se encuentran gobiernos, militares e incluso la propia NASA.

El proyecto fue creado y desarrollado por un grupo de especialista en seguridad informática de la empresa Ipsoluciones. Entre sus miembros principales se encuentran: Julio Casal, Dominque Kang, David Gil, Fabio Ospitia, y un largo número de analistas de seguridad dedicados a OSSIM.

Actualmente Ipsoluciones ha sido adquirida por IT Deusto, desde donde el grupo de seguridad realiza instalaciones, configuraciones y explotación de OSSIM en empresas de toda España, tanto privadas como importantes instituciones públicas.

Referencias de OSSIM

La lista de empresas u organizaciones que usan OSSIM crece cada día; esto no implica que la implantación haya sido realizada por la empresa OSSIM o una de sus partners. Algunas de ellas se mencionan a continuación:

Fhilips	HSBC	LANCASTER University
Siemens	At&t	BME-X
NASA	Kpn	ICEX (Instituto Español de Comercio Exterior)
U.S. Army	COMMERCEBANK	RSI
Gobierno de Chile	Citibank	GENERALITAT VALENCIANA
Alcantel	BARCLAYS	Autoritat de Certificació de la Comunitat Valenciana
Orange & france telecom Vodafone	CHRONO ESPRÉS TATA	Junta de Comercio de Castilla-La Mancha
Telefonica	Total	Isofotón
MCI	COLCIENCIAS (Colombia)	People Trabajo Temporal
Telecom (Italia)	China TELECOM	
Telmex	Universidad de Edinburgh	Caixanova

Tabla #1: Organizaciones que utilizan OSSIM.

1.6.2 En Cuba

La seguridad informática se ha instrumentado en el país mediante una política integral, estableciendo las adecuaciones legales pertinentes y las responsabilidades institucionales y personales de los diferentes actores, con lo cual se ha obtenido un bajo índice de afectación en el país.

En el marco normativo y regulatorio se han establecido por los organismos rectores diversas disposiciones e instrumentos jurídicos que ordenan y aseguran la participación de las diferentes entidades que intervienen en la provisión de los servicios de las TIC, promueven la modernización y desarrollo de las infraestructuras necesarias mediante la introducción de tecnologías de avanzada y garantizan el desarrollo y cumplimiento de los programas y entidades sociales priorizadas por el Gobierno, asociados a la informatización de la sociedad.

El desarrollo de la seguridad informática se ha visto impulsado por la promulgación de algunas de las normas que se han elaborado al respecto. En este caso, es el Ministerio del Interior quien ostenta la labor primordial de estudiar y elaborar normas sobre este aspecto, auxiliado por el Ministerio de Justicia y el Ministerio de la Informática y las Comunicaciones.

La Resolución 6 del Ministerio del Interior, promulgada el 18 de noviembre de 1996, puso en vigor el Reglamento sobre la seguridad informática, estableciendo las normas básicas que permiten implementar un sistema de medidas administrativas, organizativas, físicas, técnicas y legales que garanticen la confidencialidad, integridad y disponibilidad de la información que se procese, intercambie, reproduzca y conserve mediante el uso de las tecnologías de la información.

Por su parte, el Ministerio de la Industria Sidero Mecánica y la Electrónica, mediante la Resolución 204, de 20 de noviembre de 1996, pone en vigor el Reglamento sobre la Protección y Seguridad Técnica de los Sistemas Informáticos. En él se establecen las medidas de protección y seguridad técnica que se deben aplicar en el trabajo con las tecnologías informáticas, las que, por definición de la misma norma, incluyen los medios técnicos y los programas.

Un avance en esta materia constituye la promulgación por el Consejo de Estado, en noviembre de 1999, del Decreto Ley 199 sobre la seguridad y protección de la información oficial. En dicho Decreto Ley se regula el sistema para la seguridad y protección de la información oficial aplicable a los órganos, organismos, entidades o a cualquier otra persona natural o jurídica residente en el territorio nacional, como las representaciones cubanas en el exterior. Se establece, además, que el sistema para la seguridad y protección de la información oficial comprende la seguridad informática, la protección criptográfica y el conjunto de regulaciones, medidas, medios y fuerzas que eviten el conocimiento o divulgación no autorizados de esta información.

La Oficina de Seguridad para las Redes Informáticas, adscrita al Ministerio de la Informática y las Comunicaciones, es creada por medio de la Resolución 64, de 21 de mayo de 2002. De este modo,

dicho Ministerio pone en manos de una entidad específica la gestión de la seguridad en el ámbito de la informática.

Finalmente se puede mencionar que en Cuba se desarrollan proyectos como TAIPS-net de software libre, por la Empresa Segurmática que ha diseñado y desarrollado herramientas como SAVLinux, SABCOR, AAInternet, SAVMailer; y se utilizan herramientas, tanto de código abierto como propietario, como Antivirus preferentemente el Kaspersky, IDS´s como Snort, Snare, detectores de vulnerabilidades como Nessus, monitores de disponibilidad como Nagios, monitores de red como Ntop, antiespías como el Ad-Aware, cortafuegos como el propio del Kaspersky Anti-Virus o el del sistema operativo Windows, analizadores de logs, proxies, contraseñas y encriptación. También se utilizan otros antivirus como el NOD32, Norton Antivirus, Pandion Antivirus, MacAfee y muchos otros programas con los que los usuarios aseguran sus ordenadores como: Keylogger_Killer, Network_Security_Scanner, SpyRemover, Spyware Doctor, NetSentinel, Second Copy, BackupExpress, File Scavenger para recuperar ficheros borrados, HDD Regenerator, por solo decir los más comunes. No obstante no se utiliza OSSIM en ninguna de las redes nacionales por lo que su puesta en práctica requiere de un esfuerzo mucho mayor.

1.7.2 En la Universidad de las Ciencias Informáticas

El la UCI, como parte del panorama cubano referido a la seguridad informática, se aplican herramientas de monitoreo específicas como el Snort y el Nagios que son utilizados en los nodos para el control y detección de intrusos en la red y medir la disponibilidad de los equipos respectivamente. Se emplean también, como política de seguridad, los antivirus de los cuales el Kaspersky Anti-Virus cuenta con la actualización disponible a todos los usuarios de la universidad, los firewalls donde el más utilizado es el propio del sistema operativo Windows aunque algunos usuarios utilizan el firewall del Kaspersky Anti-Virus. También son empleados los proxies para controlar la navegación fundamentalmente con el uso de contraseñas y métodos de encriptación.

La UCI cuenta con un sistema detección y control en los servidores de su red, que detecta cuando determinado sitio está haciendo demasiadas peticiones a una PC y por tanto asume que esta ha sido contaminada con programas espías y desconecta el servicio a esa PC por una hora. Pasado este tiempo se revisa para ver si ya no está contaminada y en caso de no estarlo se habilita de nuevo la conexión.

Existen otra serie de programas muy diversos que aseguran o contribuyen a mantener, restablecer o activar la seguridad en cada una de las computadoras u ordenadores presentes en la UCI como son los Adware que se utilizan mayormente para desinfectar a los ordenadores que hayan sido contaminados por algún Spyware.

La universidad pone en práctica políticas de seguridad de las cuales algunas son parte de las políticas de seguridad a nivel nacional como la restricción del acceso a los servicios de correo de servidores como Yahoo, Hotmail, G-Mail, entre otros. Además establece que todas las PC deben estar configuradas en el domino UCI.

Se cuenta con un departamento encargado de asegurar los bienes informáticos que trabaja en conjunto con las empresas de soporte técnico tanto de software como de hardware. Los softwares son actualizados constantemente tanto para el entorno de Windows como para el software libre. En este caso, como parte del sistema de redes que existen en Cuba, tampoco en la UCI se utiliza OSSIM como herramienta de seguridad y aunque se ha intentado usarlo mediante pequeñas pruebas de dicho software no se ha llegado a una configuración confiable para mantenerlo en funcionamiento.

1.8 CONCLUSIONES

Mientras la seguridad informática es un concepto subjetivo, es decir propio al sujeto, la inseguridad informática es objetiva, es decir propia al objeto. Se hace imposible evitar la inseguridad informática pues es una propiedad inherente a los objetos. Por ello, es necesario examinar en profundidad dicha propiedad, pues mientras mejor se entienda la inseguridad, más facilidad se tendrá para comprender la seguridad informática de las organizaciones.

Si se conoce más sobre la inseguridad informática con mayor facilidad se comprenderán las acciones y resultados de la seguridad en las organizaciones. En este sentido, la detección de posibles problemas de seguridad no generaría valor sin una adecuada respuesta. Esta respuesta debe reconocer la inseguridad informática como insumo y el ataque de seguridad como una variante a considerar en la protección de los activos.

Finalmente se puede decir que existen varias formas de tratar la inseguridad, ya sea mediante aplicaciones destinadas a alertar sobre fallos o eventos de seguridad que con políticas a cumplir en una determinada empresa. La inseguridad informática es pues una estrategia de reflexión y acción para reflexionar la seguridad informática como una disciplina que es, al mismo tiempo, concepto y realidad.

2

CAPÍTULO
OSSIM, HERRAMIENTA DE SEGURIDAD

2.1 INTRODUCCIÓN

Sin duda alguna en el mercado actual se pueden encontrar todo tipo de cortafuegos, IDS, detectores de vulnerabilidades, programas de monitorización, detectores de anomalías, y un gran surtido de programas.

No obstante, esto es un mercado que ninguna empresa puede soportar, porque no sólo requiere un gasto económico en cada dispositivo tecnológico, sino que además se necesita afrontar un gasto humano en personal especializado para poder contraponer tal cantidad de eventos de seguridad, de manera individual. No es posible revisar todas las alertas generadas debido a la cantidad y a la poca fiabilidad

En otras palabras, se generan demasiadas alertas de las cuales no todas son fiables, se obtienen demasiados falsos positivos. Se obtiene así mismo información muy detallada, pero parcial y sin capacidad de abstracción, no se es capaz de detectar ataques definidos por comportamientos más complejos, es el problema que se conocen como falsos negativos.

Es aquí donde OSSIM juega un rol significativo, integrando más de 22 herramientas de open source, sin ningún costo para las empresas.

2.2 ELECCIÓN DEL SISTEMA OPERATIVO

Generalmente los motivos por los que un individuo elije un sistema operativo u otro tienen más que ver con la facilidad de uso o la gama de aplicaciones disponibles que con la seguridad. Sin embargo, la elección del sistema operativo determinará en gran medida la garantía de seguridad que se podrá conseguir al acceder a servicios ofrecidos a través de Internet. Por tanto, cuando se necesite esta garantía deben conocerse bien las características que tienen los distintos sistemas operativos en lo que a seguridad se refiere.

Muchas veces se ha dicho que GNU/Linux es un sistema muy seguro, pero esta afirmación hay que matizarla: la verdad exacta es que GNU/Linux tiene el potencial para convertirse en enormemente

seguro. Pero de entrada no tiene porque serlo. Se trata de un sistema operativo pensado para entornos de red y por ello tiene grandes capacidades de conexión con otros ordenadores y de ofrecerles servicios. En la mayoría de las distribuciones, tras realizar la instalación inicial, se dejan activos una serie de servicios como puede ser un servidor web, un servidor de ficheros o servicios de bajo nivel. Desde el punto de vista de seguridad conviene desactivar todos los servicios que no se deseen ofrecer para reducir el número de oportunidades que se le dan a un posible atacante para encontrar una vulnerabilidad.

Este sería el primer paso para configurar el sistema para que sea más seguro. Sólo mediante una configuración cuidadosa y exhaustiva del sistema operativo y sus aplicaciones se podrá decir que el ordenador es seguro. GNU/Linux ofrece las herramientas y la capacidad de configuración necesaria para hacer esto posible.

Se decidió utilizar el sistema operativo Debian GNU/Linux basados en que OSSIM y el conjunto de aplicaciones que incluye se desarrollan en software libre. Es un requerimiento de software utilizar un sistema operativo GNU/Linux para instalar OSSIM, aunque este puede monitorizar sistemas operativos Mac, Windows, Unix, Linux, entre otros. Debian es un sistema operativo altamente recomendado por las comunidades de software libre para el desarrollo de proyectos respecto a otras distribuciones de Linux por lo tanto es más fiable. La recomendación de los desarrolladores de OSSIM es que se use Debian como sistema operativo para el mejor funcionamiento de OSSIM, aunque este puede instalarse en otros como Redhat, Gentoo, Fedora y Ubuntu los cuales son menos recomendados. Además la última versión de OSSIM viene con el sistema Debian incluido.

Debian se toma la seguridad muy en serio. Se hace cargo de todos los problemas de seguridad que se le reclaman y los corrigen en un plazo razonable. Muchos avisos se coordinan con otros agentes del software libre y se publican el mismo día que se hace pública la vulnerabilidad. También tienen un equipo de auditoría de seguridad que revisa el archivo en busca de errores de seguridad nuevos o no corregidos.

2.3 LA HERRAMIENTA OSSIM

OSSIM es una consola de seguridad central, que permite gestionar y saber el nivel de seguridad que tiene una empresa. Se trata de un proyecto Open Source, con lo que todo el mundo puede disfrutar de él sin ningún coste, además de poder colaborar en su código para formar parte de su evolución.

OSSIM engloba más de 22 herramientas de seguridad, entre ellas: IDS (Snort), detector de vulnerabilidades (Nessus), firewall (Iptables), detector de sistemas operativos (Pof), monitorización en tiempo real y estadísticas (Ntop), detector de anomalías (RRD), escaneadores (Nmap), OCS-NG Inventory para el inventario de hardware y software, OSSEC para la gestión de los logs entre otros.

Todas estas utilidades son las más usadas en su categoría y todas ellas son Open Source, con lo que se contará con un gran número de personas mejorando y actualizando cada una de ellas. Pero además OSSIM no sólo consigue englobar estas herramientas, sino que la fuerza real de OSSIM reside en su motor de correlación, gracias al cual podemos tener una red o varias con millones de alertas de diferentes dispositivos y mediante su potente motor disponer de alarmas reales, sin falsos positivos y de manera centralizada.

Con esto se conseguirá que una empresa pueda, no sólo tener un gran número de dispositivos tecnológicos, sino que además sepa en cada momento el nivel de seguridad que tiene y el que desea tener.

Gracias al motor de correlación de OSSIM se conseguirá detectar entre otras cosas, virus antes incluso que los propios fabricantes de antivirus, ya que al trabajar de manera centralizada con muchas herramientas, es capaz de detectar anomalías en el funcionamiento de las máquinas, detectando nuevos virus que aún no han sido identificados por nadie.

Su objetivo es ofrecer un marco para centralizar, organizar y mejorar las capacidades de detección y visibilidad en la monitorización de eventos de seguridad de una organización.

2.3.1 Componentes de OSSIM

OSSIM esta formado por cuatro componentes:

- **Servidor Central o Servidor.** Es el componente principal de OSSIM. Se encarga de recibir los eventos enviados por los distintos agentes, también realiza las funciones de priorización y correlación.

- **Agente Central.** En algunas bibliografías es nombrado Sensor o simplemente Agente, para la realización de este trabajo se tomará el término Agente Central para evitar posibles confusiones. Son hosts distribuidos en diferentes segmentos de red, para monitorear los distintos eventos. Esta distribución es en base a los servicios que se va a monitorear. Cada

Agente Central tendrá configurado un conjunto de detectores o monitores, que generan eventos para que el Agente Central los recolecte y reporte al Servidor Central. Algunas aplicaciones necesitan de clientes, también llamados agentes, que deben ser instalados en cada host que se quiera monitorizar. Dicha aplicación cliente envía la información recogida hacia el Agente Central donde se encuentra su aplicación servidora correspondiente.

- **Framework.** Es el intermediario entre el Servidor Central y el usuario. Es la herramienta de administración utilizada para configurar y organizar los diferentes módulos tanto externos como propios que integra OSSIM. Mediante este se puede definir una topología, inventariar activos, definir políticas de seguridad, definir reglas de correlación y unir las diferentes herramientas integradas.

- **Base de datos.** Es el lugar donde se almacenan los diferentes eventos recolectados por los agentes, y las configuraciones de las distintas herramientas y OSSIM. Los componentes Servidor, Framework y la Base de Datos se encuentran ubicados en un equipo que se desempaña como Servidor Central de OSSIM y los Agentes Centrales (Sensores) pueden estar distribuidos en los distintos equipos.

2.4 HERRAMIENTAS DE CÓDIGO ABIERTO INTEGRADAS EN OSSIM

El sistema constará de las siguientes Herramientas de Monitorización:

a. Cuadro de Mandos para visibilidad a alto nivel.

b. Monitores de Riesgo y Comportamiento para la monitorización a nivel medio.

c. Consola Forense y Monitores de Red para el bajo nivel.

Estas herramientas se alimentarán de las nuevas capacidades desarrolladas en el "postproceso" y cuyo objeto es aumentar la fiabilidad y sensibilidad de la detección:

a. Correlación.

b. Priorización.

c. Valoración de Riesgos.

El postproceso a su vez es alimentado por los preprocesadores, estos son un número de detectores y monitores ya conocidos, por la mayoría de los administradores, que se integrarán en OSSIM:

d. IDS (detectores de patrones).

e. Detectores de anomalías.

f. Firewalls.

g. Monitores varios.

Por último se debe tener una herramienta de administración que configure y organice los diferentes módulos tanto externos como propios que integrará OSSIM, esta herramienta será el Framework. (6)

2.4.1 *Productos que están integrados*

Snort: IDS.

Nessus: Detector de Vulnerabilidades.

Ntop: Monitor de Red.

Nagios y OpenNMS: Monitores de Disponibilidad.

Osiris y Snare: Host IDS's.

Arpwatch, Spade Y HW Aberrant Behaviour o RRD Aberrant Behaviour: Detectores de anomalías.

P0f, Pads y Fprobe: Monitores Pasivos. P0f y Pads son monitores pasivos de anomalías.

Nmap: Análisis de red.

Acid/Base: Analizador Forense.

Oinkmaster, PHPAcl, fw1logcheck, ScanMap3D.

OSVDB Base de datos de vulnerabilidades.

OCS-NG: Aplicación para el inventario de hardware y software.

OSSEC: Aplicación cliente-servidor utilizada para gestionar los logs.

Cisco IDS: IDS de la familia de Cisco, esta basado en firmas. Puede tomar una respuesta contra un ataque, como bloquear la IP comprometida.

Cisco PIX: Solución de seguridad ofrecida por Cisco Systems, permite controlar el tráfico entre la red interna y externa. Integra un syslog para registrar eventos como conexiones establecidas, conexiones fallidas, errores en las configuraciones, entre otros.

Iptables: Filtra el tráfico utilizando reglas que examinan el origen/destino de los paquetes y el protocolo. Recolecta los eventos de las iptables.

Router Cisco: Tiene la capacidad de enviar mensajes a un servidor Syslog, para reportar los eventos que observa el router.

Tcptrack: Sniffer que despliega información sobre las conexiones TCP observadas en la interfaz de red, como: dirección origen/destino, puerto origen/destino y estado de la conexión.

Ntsyslog: Analizador de logs para Windows, similar al Syslog de Linux. Recoge información sobre el sistema, la seguridad y los eventos de las aplicaciones. Para luego enviar a un servidor Syslog.

Pam Unix (Pam Unix authentication mechanism), Sudo y SSHD (Secure Shell daemon): Analizan los logs para Unix y Linux. Recogen información sobre el sistema y los eventos de las aplicaciones.

Syslog: Es un estándar para la transferencia de mensajes de eventos y alertas. Los mensajes son enviados por el sistema operativo, al inicio o fin de una aplicación, o reporte actual de un proceso.

Snarewindows: Trabaja como un manejador de logs y reporte de incidentes.

Apache: Servidor HTTP de código abierto, puede ser utilizado en plataformas Unix y Windows. En los Log File se registra: la actividad, rendimiento y los problemas que puedan ocurrir en el servidor HTTP.

IIS: Servidor de páginas Web de Microsoft, tiene la capacidad de registrar los eventos del servidor, peticiones y errores.

Pueden ser algunos más indistintamente aunque estos son los más utilizados. OSSIM puede utilizar un servidor web no libre aunque se recomienda utilizar uno que sea de software libre.

2.5 INFRAESTRUCTURA OPEN-SOURCE DE MONITORIZACIÓN DE SEGURIDAD

2.5.1 Solución vs producto

OSSIM no quiere ser un producto, sino una solución, un sistema personalizado para las necesidades de cada organización, formado por la conexión e integración de varios módulos de especialistas.

En esta solución, tan importante como el código, son los conceptos o definiciones de:

La arquitectura.

Los Modelos y Algoritmos de Correlación.

La definición del Entorno y el Framework.

El Modelo de Gestión de la Seguridad Perimetral

El interés del proyecto OSSIM es tanto ofrecer el código como generar la discusión y el conocimiento de estos modelos y algoritmos.

Arquitectura Abierta

OSSIM es una arquitectura de monitorización abierta pues integra diversos productos del mundo libre, intentando seguir siempre los estándares y las tendencias del mundo open source (los cuales se cree que en soluciones de monitorización serán los estándares en todos los entornos).

Solución Integral

Es una solución integral pues es capaz de ofrecer las herramientas y funcionalidad para monitorización de todos los niveles desde el más bajo (firmas detalladas de un IDS, dirigido al técnico de seguridad), hasta el más alto (El Cuadro de Mandos dirigido a la Dirección Estratégica), pasando por: Consolas Forenses, niveles de Correlación, Inventariado de Activos y Amenazas, y Monitores de Riesgos.

Software de Open source

OSSIM se propone como un proyecto de integración, la intención no es desarrollar nuevas capacidades sino aprovechar las riqueza ("joyas") del software libre, programas desarrollados por la inspiración de los mejores especialistas del mundo (como pueden ser Snort, RRD, Nmap, Nessus, o Ntop) integrándolas en una arquitectura abierta que heredará todo su valor y capacidades. El proyecto de desarrollo de OSSIM será el encargado de integrar y relacionar la información de estos productos.

Estas herramientas de open source son, por la naturaleza de este, probadas y mejoradas por decenas o centenas de miles de instalaciones en el mundo convirtiéndose en elementos robustos y altamente probadas.

Por el hecho de ser código abierto son así mismo confiables y exentas de cualquier duda de posibles puertas traseras al ser auditables por cualquiera que lo desee.

2.5.2 El proceso de detección

Para resumir en una frase lo que busca el proyecto OSSIM se podría decir: Aumentar la capacidad de detección.

Se llamará “Proceso de Detección” al proceso global desarrollado por el SIM, incluyendo tanto los diferentes detectores y monitores de la organización como los realizados por el sistema para procesar esta información.

Detectores

Se definirá un detector como cualquier programa capaz de procesar información en tiempo real, información normalmente a bajo nivel como tráfico o eventos de sistema y lanzar alertas ante la localización de situaciones previamente definidas.

La definición de estas situaciones se puede hacer de dos formas:

- A través de patrones, o reglas definidas por el usuario.
- A través de grados de anomalía.

La Capacidad de Detección

La capacidad de detección ha aumentado enormemente en los últimos años, por ejemplo en su máximo exponente los IDS, capaces de detectar patrones al nivel más bajo de detalle.

Para discutir sobre la capacidad de un detector se definirán 2 variables:

- *Sensibilidad*, o la capacidad de análisis, en profundidad y complejidad, que posee nuestro detector a la hora de localizar un posible ataque.

- *Fiabilidad*, como su nombre indica es el grado de certeza que nos ofrece nuestro detector ante el aviso de un posible evento.

La Incapacidad de Detección

Pese al desarrollo "en la profundidad de detección" de estos sistemas, se está muy lejos de que su capacidad sea aceptable.

De la incapacidad de los detectores de afrontar estas dos propiedades tenemos los dos principales problemas de la actualidad:

- **Falsos Positivos.** La falta de fiabilidad en nuestros detectores es el causante de los falsos positivos, es decir alertas que realmente no corresponden con ataques reales.
- **Falsos Negativos.** La incapacidad de detección implicaría que un ataque es pasado por alto.

Se pueden resumir los anteriores puntos en la siguiente tabla:

	Propiedad	Efecto ante su ausencia
Fiabilidad	El grado de certeza que nos ofrece nuestro detector ante el aviso de un posible evento.	Falsos Positivos
Sensibilidad	La capacidad de análisis, en profundidad y complejidad, que posee nuestro detector a la hora de localizar un posible ataque	Falsos Negativos

Tabla #2: Capacidad de los Detectores.

2.5.3 Fases del proceso de detección

El Proceso de Detección implica normalmente tres fases bien distinguidas:

Preproceso: La detección en si misma, la generación de alertas por los detectores y la consolidación previa al envío de información.

Colección: El envío y recepción de toda la información de estos detectores en un punto central.

Postproceso: El tratamiento que se realizará una vez se tenga toda la información centralizada.

Postproceso: El preproceso y la colección son capacidades ya clásicas que no aportan nada nuevo, pero en el postproceso, una vez que se tenga toda la información en un mismo punto, se podrán implementar mecanismos para poder mejorar la sensibilidad y fiabilidad de la detección. Se aumentará la complejidad del tratamiento incluyendo métodos que se encargarán de descartar falsos positivos o al contrario priorizar o descubrir patrones más complejos que los detectores han pasado por alto.

En OSSIM se desarrollarán 3 métodos en el postproceso:

1. Priorización: Donde se priorizarán las alertas recibidas mediante un proceso de contextualización desarrollado a través de la definición de una Política Topológica de Seguridad y el Inventariado de los sistemas.

2. Valoración de Riesgo: Cada evento será valorado respecto del Riesgo que implica, es decir, de una forma proporcional entre el activo al que aplica, la amenaza que supone y la probabilidad del evento.

3. Correlación: Donde se analizarán un conjunto de eventos para obtener una información de mayor valor.

El siguiente cuadro muestra como afectan estos procesos a las anteriores propiedades:

	Proceso	Efecto
Priorización	Valoración de la amenaza mediante contextualización de un evento	Aumenta la Fiabilidad
Valoración de Riesgo	Valoración del Riesgo respecto del valor de activos	Aumenta la Fiabilidad
Correlación	Relación de varios eventos para obtener una información de mayor valor	Aumenta la Fiabilidad, Sensibilidad, y Abstracción

Tabla #3: Postproceso.

El sistema se alimentará por lo tanto de “alertas” ofrecidas por los detectores y producirá tras el tratamiento de las mismas lo que se llamará “alarmas”.

Una alarma será normalmente el resultado del proceso de varias alertas, tendrá normalmente un mayor grado de abstracción permitiendo localizar patrones más complejos, y ofrecerá un mayor grado de fiabilidad.

2.6 FUNCIONALIDAD DE OSSIM

Para entender que ofrece OSSIM se puede definir la funcionalidad del sistema de una forma gráfica y simplificada con los 9 siguientes niveles:

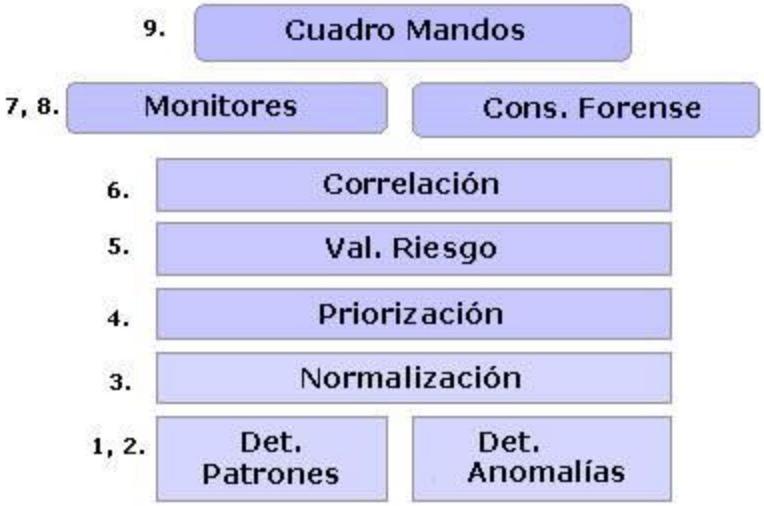


Figura #2: Funcionalidad de OSSIM.

Se comentará cada uno de estos niveles para ir introduciendo en los detalles del sistema:

2.6.1 Detectores de patrones

La mayoría de los detectores clásicos funcionan con patrones, el ejemplo más claro es el IDS o sistema de detección de intrusos, sistema capaz de detectar patrones definidos a través de firmas o reglas.

Existen otra serie de detectores de red incluidos en la mayoría de los dispositivos como routers o Firewalls, capaces de detectar por ejemplo scaneo de puertos, intentos de spoofing, o posibles ataques por fragmentación.

Se tiene además detectores para los eventos de seguridad de un sistema operativo, casi todos ellos incluyen su propio logger como el de Unix llamado syslog, siendo capaces de alertar de posibles problemas de seguridad.

En definitiva cualquier elemento de la red (router, computadora, firewall) incluye la capacidad de detección en mayor o menor medida, en OSSIM se van a recibir los eventos de todos los sistemas críticos para de esta forma obtener uno de los objetivos principales: la Visibilidad de la red.

2.6.2 Detectores de anomalías

La capacidad de detección de anomalías es más reciente que la de patrones. En este caso al sistema de detección no habrá que decirle que es bueno o que es malo, él es capaz de "aprender" por sí solo y alertar cuando un comportamiento difiera los suficiente de lo que ha aprendido como normal.

Esta nueva funcionalidad ofrece un punto de vista diferente y complementa la detección de patrones.

La detección de anomalías puede ser especialmente útil para prevenir ataques perimetrales, estos son en sí una anomalía continua, en la dirección, el sentido de las comunicaciones, y el camino que definen, en el flujo de datos, el tamaño, el tiempo, el horario, el contenido, etc.

Otros ejemplos en los que estos detectores serían útiles:

- Un nuevo ataque del cual no existen todavía firmas podría traspasar los sistemas de detección de patrones pero producir una anomalía clara.
- Los ataques internos, de empleados desleales desde dentro de una red, no implican la violación de ninguna política ni la ejecución de ningún exploit. Implican sin embargo una anomalía en el uso de un servicio.
- Un gusano que se ha introducido en la organización, un ataque de spamming, o el mismo uso de programas P2P, generarían un número de conexiones anómalas fácilmente detectable.

Se podrán detectar así mismo:

- Usos de servicios anormales por origen y destino.
- Usos en horario anormal.
- Exceso en el uso de tráfico o conexiones.
- Copia anormal de ficheros en la red interna.

- Cambios en el sistema operativo de una máquina.

Se puede pensar que como efecto negativo estos detectores generarán un número de nuevas alertas, amplificando la señal y empeorando el problema (el objetivo es limitar el número de alertas), sin embargo si se toman como información adicional que acompaña a las clásicas alertas de patrones permitirá cualificar y por lo tanto diferenciar aquellas que puedan implicar una situación de mayor de riesgo.

2.6.3 Centralización y normalización

La normalización y centralización o agregación tiene como objetivo unificar en única consola y formato los eventos de seguridad de todos los sistemas críticos de la organización.

Todos los productos de seguridad poseen normalmente la capacidad de gestión centralizada a través de protocolos estándar, la agregación es por lo tanto sencilla utilizando estos protocolos. En futuras versiones de OSSIM se tendrá en cuenta una arquitectura de cifrado e identificación mediante firma la cual aumentará la confidencialidad y autenticación para entornos que lo requieran. Por ahora habrá que conformarse con protocolos más sencillos y garantizar la confidencialidad a través de una topología apropiada como se comentará en epígrafes posteriores.

La normalización implica la existencia de un “parser” o traductor que conozca los tipos y formatos de alertas de los diferentes detectores, se necesita desarrollar un trabajo de organización de la base de datos y adaptación de la Consola Forense para homogenizar el tratamiento y la visualización de todos estos eventos.

De esta forma se podrán observar en la misma pantalla y con un mismo formato los eventos de seguridad de un determinado momento, ya sean del router, del firewall, del IDS, o del servidor Unix.

Al tener centralizados en la misma base de datos todos los eventos de la red se obtendrá una gran “visibilidad” de lo que ocurre en ella, a partir de ese momento se podrán desarrollar procesos que permitan detectar patrones más complejos y distribuidos.

2.6.4 Priorización

La prioridad de una alerta debe ser dependiente de la Topología y el Inventario de sistemas de la organización donde se vaya a utilizar OSSIM, las razones son bastante claras como muestran estos ejemplos:

Si una alerta que se refiere a un ataque al servicio IIS de Microsoft llega a una máquina con sistema operativo Unix y servidor Apache, la alerta debe ser despriorizada.

Si existe una conexión sospechosa de un usuario sobre un servidor, el sistema debe:

- Darle máxima prioridad si el usuario es externo y ataca a la base de datos de clientes.
- Darle prioridad baja si el usuario es interno y ataca a una impresora de red.
- Descartarla si es un usuario que normalmente hace pruebas contra un servidor de desarrollo.

Se llamará priorización al proceso de contextualización, es decir la evaluación de la importancia de una alerta respecto del escenario de nuestra organización. Este escenario está descrito en una base de conocimiento sobre la red del cliente formada por:

- Inventario de Máquinas y Redes (identificadores, sistema operativo, servicios, etc.).
- Política de Accesos (desde donde a donde está permitido o prohibido).

Para realizar estas tareas (así como la valoración de riesgos explicada en el siguiente epígrafe) se dispone de un Framework donde se podrá configurar:

- Política de Seguridad o valoración de parejas activo-amenazas según la Topología y flujo de los datos.
- Inventario.
- Valoración de activos.
- Valoración de amenazas (priorización de alertas).
- Valoración de Fiabilidad de cada alerta.
- Definición de Alarmas.

A través de la cualificación se realizará una de las partes más importante del filtrado de alertas recibidas por los detectores, la cual debe realizarse a través del proceso continuo de tunning y realimentación de la situación de la organización.

2.6.5 Valoración de riesgo

La importancia que se debe dar a un evento debe ser dependiente de estos tres factores:

El valor del Activo al que el evento se refiere.

La Amenaza que representa el evento.

La Probabilidad de que este evento ocurra.

Riesgo Intrínseco

Con ellos se construye la definición clásica de riesgo: el valor del posible impacto de una amenaza sobre un activo ponderado con la probabilidad de que este ocurra.

La valoración de riesgos se ha referido clásicamente a riesgos intrínsecos, o riesgos latentes, es decir riesgos que soporta una organización derivados del hecho de "ser" (los activos que posee para desarrollar su negocio) y "estar" (las amenazas circunstanciales que existen sobre estos activos).

Riesgo Instantáneo

En este caso, debido a la capacidad de medir en tiempo real, se puede medir el riesgo asociado a la situación actual, en términos instantáneos.

Por tanto el riesgo será medido como la medida ponderada del daño que produciría y la probabilidad de que este ocurriendo en este momento la amenaza.

Esta probabilidad, derivada de la imperfección de nuestros sensores, no será más que el grado de fiabilidad de estos en la detección de la posible intrusión en curso.

Se llamará Riesgo Instantáneo a la situación de riesgo producida por la recepción de una alerta, valorada de forma instantánea como la medida ponderada entre el daño que produciría el ataque y la fiabilidad del detector que lo reporta.

OSSIM calculará el Riesgo Instantáneo de cada evento recibido que será la medida objetiva que se utilizará para valorar la importancia que un evento puede implicar en términos de seguridad, sólo a través de esta medida se valorará la necesidad de actuar.

Se incluye en el sistema (OSSIM) un Monitor de Riesgos que valorará el riesgo acumulado en el tiempo, de redes y grupos de máquinas relacionados en un evento.

2.6.6 Correlación

Se define una función de correlación como un algoritmo que realiza una operación a través de unos datos de entrada y ofrece un dato de salida.

Se pensará en la información recogida por los detectores y monitores como información específica pero parcial, dibujando pequeñas zonas del espectro de toda la información que interesaría tener.

Se puede pensar en la capacidad de correlación como la de aprovechar estos sistemas y a través de una nueva capa de proceso llenar otras zonas de ese espectro infinito de toda la información que podría existir de una red.

En contra de esta idea podía intentar instalarse un sistema único con un detector capaz de localizar toda la información posible de la red, pero para ello se necesitaría una visibilidad total desde un punto único y una capacidad de almacenamiento y de memoria casi ilimitada.

Los sistemas de correlación son por tanto artificios que suplen la falta de sensibilidad, fiabilidad y la visibilidad limitada de los detectores.

Entrada y Salida.

En esta arquitectura, de una forma simplificada, se puede decir que se tiene dos elementos claramente diferenciados para ofrecer información a las funciones de correlación:

- Los Monitores, que ofrecerán normalmente indicadores.
- Los Detectores, que ofrecerán normalmente alertas.

Como salida se obtendrá también uno de estos dos elementos: alertas o indicadores. Las funciones se habrán convertido en nuevos detectores o monitores.

Modelo de Correlación

OSSIM desarrolla un modelo de correlación bien ambicioso, con el que se podrá:

- Desarrollar *patrones específicos* para detectar lo conocido y detectable.

- Desarrollar *patrones ambiguos* para detectar lo desconocido o no detectable.
- Poseer una *máquina de inferencia* configurable, a través de reglas relacionadas entre sí, capaz de describir patrones más complejos.
- Permitir enlazar Detectores y Monitores de forma recursiva para crear cada vez objetos más abstractos y capaces.
- Desarrollar algoritmos que ofrezcan una visión general de la Situación de Seguridad.

Métodos de correlación

Para lograr estos objetivos se utilizarán dos métodos de correlación muy diferentes que se intentarán describir mediante sus diferencias principales:

Correlación mediante Secuencias de Eventos. Focalizado en los ataques conocidos y detectables relaciona, a través de reglas que implementarán una máquina de estados, los patrones y comportamientos conocidos que definen un ataque.

Correlación mediante Algoritmos Heurísticos. Tomando una aproximación opuesta se implementan algoritmos que mediante funciones heurísticas intentan detectar situaciones de riesgo. Este método detectará situaciones sin conocer ni ofrecer detalle de los mismos, intenta suplir pues la incapacidad de los anteriores métodos y será útil para detectar ataques no conocidos.

A) Método 1: Correlación mediante Algoritmos Heurísticos

En OSSIM se implementa un sencillo algoritmo heurístico de correlación por acumulación de eventos con el objetivo de obtener un indicador o una fotografía general del estado de seguridad de la red.

El primer objetivo de este es recibir lo que se ha definido previamente como “riesgo instantáneo” recibiendo como resultado un valor que se puede definir como el Nivel Acumulado de Riesgo.

Se obtiene una monitorización a alto nivel que servirá como “termómetro” de situaciones de riesgo, sin conocer en ningún momento detalles de las características del problema.

Por hacer un símil se construye un termómetro que será sensible y sumará la cantidad de riesgo acumulado en una ventana de tiempo, el termómetro subirá proporcionalmente a la cantidad y lo

"calientes" que sean los últimos eventos recibidos, y se enfriará con el paso del tiempo en caso de no recibir nuevos eventos.

Este método de correlación quiere suplir con un punto de vista opuesto a la correlación mediante secuencias de eventos, donde se intenta caracterizar al máximo nivel de detalle los posibles ataques.

Su interés es pues doble:

- Ofrecer una visión global y rápida de la situación.
- Detectar posibles patrones que el resto de los sistemas de correlación puedan pasar por alto, ya sea por tratarse de ataques desconocidos o por falta de capacidad.

CALM

CALM (Compromise Attack Level Monitor) es un algoritmo de valoración por acumulación de eventos con recuperación en el tiempo. Recibe como entrada un alto volumen de eventos y como salida un único indicador del estado general.

La acumulación se realiza para cualquier sujeto de la red, entendiendo como tal a cualquier máquina, grupo de máquinas, segmento de red y camino que se interese monitorizar.

Acumulación de Eventos

La acumulación se realiza a través de la simple suma del riesgo instantáneo de cada evento en dos variables de estado:

La "C" o el Nivel de Compromiso, que mide la posibilidad de que una máquina se encuentre comprometida.

La "A" o el Nivel de Ataque al que está sometido un sistema, que mide el posible riesgo debido a los ataques recibidos.

¿Por que separar estas dos variables en la monitorización? En primer lugar porque caracterizan situaciones diferentes: el Nivel de Ataque indica la posibilidad de estar recibiendo un ataque, ataque que podrá o no tener éxito. El Nivel de Compromiso ofrece evidencia directa, como su nombre indica, que ha habido un ataque y ha tenido éxito.

En segundo lugar la importancia de cada una de las dos variables será dependiente de la situación de la máquina. Principalmente debido a la exposición de las redes perimetrales, expuestas a multitud de ataques la mayoría de ellos automatizados y para las cuales desgraciadamente un alto valor del Nivel de Ataque ha de ser una "situación normal". Para estas redes sin embargo el indicador de Compromiso, o movimiento que pueda hacer pensar que hay un atacante alojado en ellas debe ser inmediatamente notificado y revisado.

Al contrario, hay casos en los que una máquina que por su función genera, anomalías en la red como un scanner de seguridad, un servicio con puertos pasivos aleatorios, entre otros, tendrá normalmente una C alta y una A baja.

La asignación del valor a las variables C o A de una máquina de la red se produce a través de 3 reglas:

Cualquier posible ataque que se produzca desde una máquina 1 a una máquina 2 aumentará la A (el nivel de ataques recibidos) de 2 y la C (el nivel de compromiso o acciones sospechosas que normalmente haría un hacker) de 1.

El caso de tratarse de una respuesta de ataque los ya comentados "attack responses" (respuestas que pueden implicar que el ataque a tenido éxito), en este caso aumentará el nivel de C tanto en 1 como en 2.

En caso de ser eventos internos aumentará únicamente la C de la máquina originaria. Por último existe una excepción para:

Acumulación en el Tiempo

CALM está pensado para la monitorización en tiempo real, por lo que su interés es una ventana de tiempo en el corto plazo, es decir, interesa la valoración de eventos de un espacio de tiempo cercano, el algoritmo debe tener una memoria en el corto plazo primando los eventos más recientes y caducando los más antiguos.

La implementación actual es una simple variable de recuperación en el tiempo. El sistema irá rebajando con un valor constante de forma periódica los niveles de C y A de cada máquina.

B) Método 2: Correlación mediante Secuencias de Eventos

La idea inicial de la detección de una secuencia de patrones es sencilla pues sería simplemente realizar una lista de reglas “si ocurre el evento A y luego B y luego C, haz la acción D”.

Esto se realiza a través del Panel de Secuencias, donde se definen listas de reglas para cada secuencia de eventos que se quieran definir.

La complejidad del panel dependerá de la capacidad de abstracción que permitan sus reglas y la posibilidad de analizar diferentes entradas en nuestras funciones.

En OSSIM el panel será capaz de realizar secuencias con las siguientes características:

- Posibilidad de definir orígenes y destinos variables.
- Tomar como entrada tanto patrones procedentes de detectores como indicadores procedentes de monitores.
- Definir el nivel de *prioridad y fiabilidad* de las nuevas alertas.
- Utilizar variables "elásticas" o capaces de medir el grado para definir la prioridad o fiabilidad (ejemplo: Denegación de servicio: total -> prioridad grave, 50 por ciento -> prioridad media, 15 por ciento -> prioridad baja).
- Arquitectura recursiva: se podrá crear objetos a través de la correlación de reglas que se podrán incluir en nuevas reglas como en detectores o monitores.

Niveles de correlación

Debido a la recursividad de este modelo se podrá crear una jerarquía de niveles casi infinita, para poder centrar este estudio se definirá una jerarquía de 3 niveles tal y como se muestra en el siguiente gráfico:

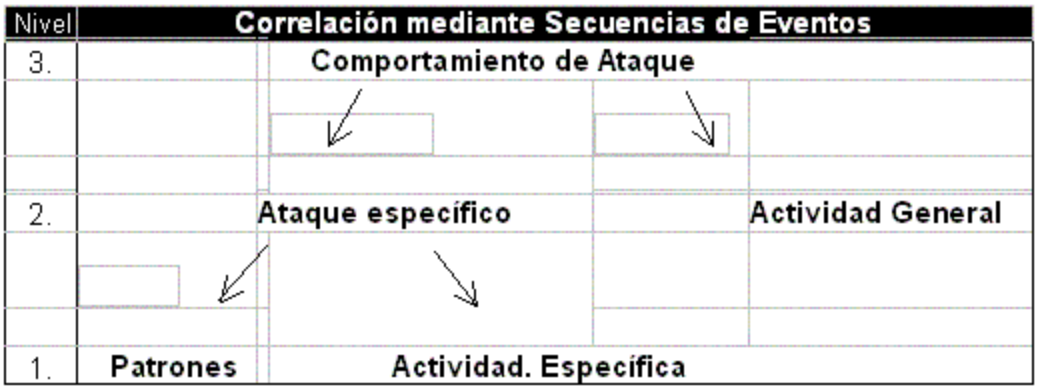


Tabla #4: Niveles de correlación.

O sea, se tienen los siguientes niveles: nivel 1 especificado en 2 secuencias: la 1.1 referida a patrones y la 1.2 referida a actividad específica, nivel 2 especificado en 2 secuencias también: la 2.1 referida a ataque específico y la 2.2 referida a actividad general y por último el nivel 3 referido a comportamiento de ataque.

Se recorrerá cada uno de estos niveles desordenadamente para su mejor entendimiento:

Nivel 2.1. Ataque Específico

Este nivel trata directamente con los detectores y monitores, se intentará relacionar tanto las firmas como la actividad que se refiera a un ataque concreto, un ataque con nombre y apellidos tal y como lo conocen los detectores (por ejemplo: “compromiso mediante ftp, cwd, overflow”).

El principal objetivo del nivel de ataque específico es el de aumentar la *fiabilidad* de las detecciones, o sea, no bastará con la firma de la posibilidad de un ataque, sino que se buscará más evidencias que demuestren que se está produciendo el ataque o clarifiquen que es únicamente un intento fallido.

Esta *cualificación* es la que hará la diferencia a la hora de limitar falsos positivos y priorizar ataques reales en un sistema de detección de seguridad, ya que como se ha visto la fiabilidad de un evento afecta directamente al cálculo del riesgo.

Un ejemplo sencillo de correlación de un detector de patrones y un monitor sería el siguiente:

El IDS detecta mediante una firma un posible ataque de denegación de servicio mediante *“synflood”,* la alerta de este arrancará una pregunta al Monitor de Servicio para ver si este ha sufrido un decremento

de indisponibilidad y en que grado. De esta forma se podrá añadir un grado de fiabilidad mayor a la alerta “Denegación de servicio por “Synflood”.

Normalmente se tendrán secuencias más complejas, donde se correlacionarán las alertas producidas por firmas con los comportamientos específicos que caracterizan un ataque. Por ejemplo en la detección de un Caballo de Troya, las operaciones que se es capaz de detectar a través de firmas de IDS son varias:

Connect, active, get info, access, server2client_flow, client2server_flow, response, traffic_detect

La detección de una operación *connect* probablemente no es una información de gran valor, en entornos perimetrales se reciben decenas al día, pero si se detecta cualquier otra operación y en especial de respuesta al intento de conexión se debe enviar una alerta con prioridad alta.

Nivel 1.2. Detección por Actividad Específica

Aprovechando el ejemplo del Caballo de Troya para explicar el concepto de”*actividad específica”*, en un caso simple: tras un intento de conexión, si el Caballo de Troya opera a través del puerto P, simplemente si este puerto tiene actividad, es decir transmite datos. Si esto ocurre se tiene como antes una confirmación de que el intento de conexión probablemente ha tenido éxito, pero en vez de ser una firma de un IDS se ha localizado monitorizando la actividad propia del Troyano.

La Actividad Específica implica la utilización de los monitores para atender una pregunta concreta sobre la actividad asociada a un posible Ataque Específico, serán consultas que arrancará y terminará el motor de correlación para un caso concreto.

Nivel 1.1. Detección Mediante Patrones

Este nivel ya se ha comentado y viene proporcionado directamente por los detectores de patrones. El sistema de correlación será capaz de procesar cualquier alerta detectada por estos.

Respuestas de Ataque

Se hará un alto para sacar una conclusión de los dos puntos anteriores: la importancia de las repuestas de ataque (“attack responses” en los IDS) como comprobación de la existencia de un evento, o lo que es lo mismo aumento de la fiabilidad de una alerta.

El motor de correlación está diseñado para buscar continuamente estas respuesta de ataque, tras recibir la primera información de un posible ataque se pondrá todo el sistema a localizar evidencias de que el ataque realmente se está produciendo. Esto permitirá diferenciar ataques fallidos de los que realmente han tenido éxito.

Gráficamente se podría representar este ejemplo de la siguiente forma:

Sucesión de Eventos	Tipo de Alerta	Fiabilidad
1. Intento, compromiso o conexión caballo Troya	Firma: conexión, client2server, access, getinfo	Baja
	Act. Específica: flujo atacante -> víctima	
2. Detección de respuesta típica de éxito del ataque o respuesta a operación de caballo Troya	Firmas: response, server2client	Alta
	Activ. Específica: flujo víctima->atacante	

Tabla #5: Aumento de la fiabilidad para Caballo de Troya.

Se llamará detección por Actividad General a las reglas destinadas a localizar ataques no conocidos o no detectables pues no se conocen los patrones que caracterizan este ataque.

La localización de estos ataques será gracias a la generación de actividad anómala por parte del atacante, para ello se monitorizarán parámetros generales de cada usuario tales como los puertos o servicios, el tráfico, el horario, etc.

Mediante esta técnica se podrá caracterizar ataques con cierto detalle en algunos casos, pero generalmente se detectarán comportamientos sospechosos, con un nivel menor de precisión que en la detección de Ataques Específicos y moviéndose muchas veces en la frontera entre lo que es un ataque, un problema de red o el mal uso por parte de los usuarios.

Ejemplos de esta detección serían:

- Detección de un gusano desconocido. Que generará un tráfico anormal, un número de conexiones atípico con puertos y destinos que hasta ahora no habían sido usados.

- Detección de acceso sospechoso. Al localizar un usuario conectado de forma persistente a un puerto de administración, que hasta ahora no lo había hecho.

- Exceso de uso de tráfico. A través de anomalías de destinos y utilización.

Nivel 3. Comportamiento de Ataque

Este tercer nivel de correlación se alimenta y es principalmente la correlación de varios Ataques Específicos o Comportamientos Generales localizados en el primer nivel.

La arquitectura del sistema de correlación es recursiva y en sus reglas se pueden incluir nuevos objetos que funcionarán como detectores (enviando alertas) o monitores (ofreciendo un valor) que están formados por un conjunto de reglas del nivel inferior.

Pero la caracterización de los nuevos niveles no debe hacerse por el hecho de que los objetos de entrada sean procedentes del nivel inferior, de hecho esto no será así siempre y podremos mezclarlos según nos convenga.

La caracterización de cada nivel debe ser debida al nivel de abstracción al que se refiera, en este caso se intentará localizar patrones de comportamiento que caractericen cual es el objetivo, el camino trazado, el comportamiento y el método del atacante. Para ello se definirán Comportamientos de Ataque o la secuencia de ataques y comportamientos desarrollada por el usuario sobre una o varias máquinas comprometidas.

Ejemplos de estos comportamientos podrían ser:

- Ataque distribuido. Al encontrar relación de varios atacantes y ataques recibidos.

- Acceso a red crítica desde Internet. Siguiendo el flujo de un ataque perimetral que desde Internet llega en varios saltos a una red crítica.

- Compromiso por un usuario Interno Malicioso. A través de la localización de varios comportamientos anormales de un usuario interno.

2.6.7 Monitores

Los monitores se encargan de observar, registrar y analizar la red, así como los procesos de las máquinas que en esta se encuentren. Estos monitores de procesos constituyen funcionalidades que valen la pena destacar, como son:

Monitor de Riesgos

OSSIM posee un monitor de riesgos llamado RiskMeter que dibujará los valores producidos por el algoritmo CALM, valores que miden el nivel de riesgo de compromiso (C) y de ataque (A) derivados de la recepción de alertas que indican la posibilidad de que una máquina ha sido comprometida o está siendo atacada.

Monitor de Uso, Sesiones y Perfiles

En OSSIM se da mucha importancia, a la monitorización detallada de cada máquina y perfil.

Se diferencian 3 tipos de monitorización para ello:

- Monitor de Uso: Ofrece datos generales de la máquina como el número de bytes que transmite al día.
- Monitor de Perfiles. Ofrece datos específicos del uso realizado por el usuario y permite establecer un perfil, por ejemplo: usa correo, pop, y http, es un perfil de usuario normal.
- Monitor de Sesiones. Permite ver en tiempo real las sesiones que está utilizando el usuario. Ofrece una foto instantánea de la actividad de una máquina en la red.

Se cree que cualquiera de estos 3 son imprescindibles para un sistema de seguridad, en caso contrario, el administrador de seguridad estará “ciego” ante eventos pasados, no podrá valorar lo normal de lo anormal y no será capaz de ver su red, sería semejante a un policía de tránsito en una carretera completamente oscura.

Aquí la frontera de seguridad se confunde con la administración de redes, pero este solape es inevitable pues la saturación de una red o el comportamiento anómalo de una máquina puede significar tanto un problema de red como un problema de seguridad.

En OSSIM se incluyen las 3 capacidades de monitorización anteriormente expuestas a través de productos capaces de actuar como “sniffers” y ver al máximo nivel de detalle la situación de la red.

Monitor de Caminos

Este monitor es capaz de dibujar en tiempo real los caminos trazados en la red entre las diferentes máquinas que realizan comunicaciones o enlaces entre ellas.

El dibujo se realiza en un intervalo de tiempo creando un grafo cuyas ramas irán caducando en el tiempo.

El monitor obtiene sus datos de otros dos monitores: el de sesiones donde están localizados cada uno de los enlaces del momento, y del monitor de riesgo de donde obtiene el nivel de riesgo de cada máquina para dibujar cada una con un color diferente y calcular el riesgo agregado de cada uno de estos grafos.

La monitorización de enlaces tiene a su vez dos métodos:

- **Hard Link Analysis (TCP Link Analysis):** Mediante el cual se dibuja únicamente sesiones TCP persistentes. Es un método desarrollado con la intención de localizar ataques de red que implican la intrusión de varias máquinas de forma continuada, situación típica de una intrusión perimetral.
- **Soft Link Analysis:** Mediante el cual se dibujan todos los enlaces percibidos en la red, tanto UDP, TCP como ICMP, lo cual puede implicar en muchos casos un mapa de red caótico.

2.6.8 Consola forense

La Consola Forense permite acceder a toda la información recogida y almacenada por el colector.

Esta consola es un buscador que accede a la base de datos de eventos, y permite al administrador analizar a posteriori y de una forma centralizada los eventos de seguridad de todos los elementos críticos de la red.

Al contrario que el Monitor de Riesgos, esta consola permitirá profundizar al máximo detalle sobre cada uno de los eventos ocurridos en el sistema.

2.6.9 Cuadro de mandos

La última de las funcionalidades es el Cuadro de Mandos. Mediante este se puede ofrecer una visión de alto nivel de la situación de la red puesto que monitoriza una serie de indicadores que miden el estado de la organización respecto a la seguridad.

El cuadro de mandos permitirá definir una serie de umbrales u objetivos que debe cumplir una organización. Estos umbrales serán definidos de forma absoluta o relativa como un grado de anomalía.

Se podrá asignar el envío de alarmas cuando se superen estos umbrales o la ejecución de cualquier procedimiento automático.

Es importante así mismo la forma de visualización de la información en este cuadro de mandos pues debe ser lo más concisa y simple posible. Para ello se necesita una configuración versátil que muestre únicamente la información relevante en ese momento.

El cuadro de mandos debe ser el termómetro general de todo lo que ocurre en la red. A través de él se enlazará cada una de las herramientas de monitorización para profundizar sobre cualquier problema localizado.

Como ejemplo se podrían visualizar los siguientes datos:

- Monitorización permanente de los niveles de riesgo de las principales redes de la organización.
- Monitorización de las máquinas o subredes que superan el umbral de seguridad.
- Monitorización permanente de parámetros generales de red, sistema y niveles de servicio:
    - ✓ Throughput y Tráfico de principales redes.
    - ✓ Recursos de la base de datos principal.
    - ✓ Latencia de Servicios críticos.
    - ✓ Número de transacciones de servicios críticos.
- Monitorización de aquellos parámetros de red o niveles de servicio que superen el umbral establecido:
    - ✓ Número de correos, virus y accesos externos.

- ✓ Latencia de los servicios y uso de tráfico por servicios.

- Monitorización de perfiles que superen los umbrales por:
  - ✓ Uso de tráfico.
  - ✓ Uso de servicios críticos.
  - ✓ Uso de servicios anómalos.
  - ✓ Cambios en configuración.
  - ✓ Cualquier otra anomalía de comportamiento.

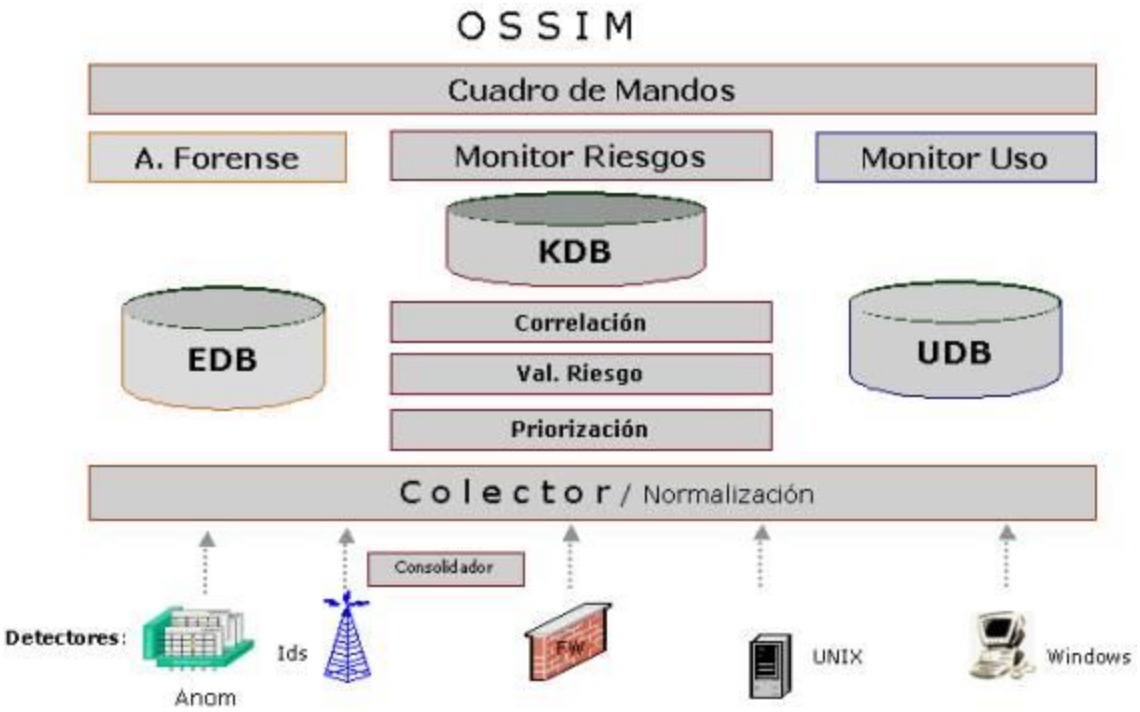
El cuadro de mandos debe ser algo completamente personalizado y hecho a medida. Al contrario que para el resto de funcionalidades, OSSIM sólo incluirá un ejemplo base sobre el cual trabajar.

2.7 ARQUITECTURA

El sistema contará como es común con dos partes diferenciadas, en ellas se desarrollan dos momentos diferentes del proceso:

- **Preproceso:** que se realizará en los propios monitores y detectores
- **Postproceso:** que se realizará en una consola centralizada

El dibujo general de la arquitectura según los procesos realizados es el siguiente:



Figura# 3: Arquitectura de OSSIM. (7)

En ella se perciben cada una de las funcionalidades anteriormente descritas. Así mismo se ven 3 bases de datos:

- EDB, la base de datos eventos, la más voluminosa pues alojará todos los eventos individuales recibidos de los detectores.
- KDB, la base de datos del Framework, en la cual se parametrizará el sistema para que conozca la red y se definirá la política de seguridad.
- UDB, la base de datos de perfiles, que almacenará todos los datos aprendidos por el Monitor de Perfiles.

2.8 FLUJO DE LOS DATOS

Para entender la integración de cada uno de los productos se hará un recorrido del flujo desde la generación de un evento:

- *Los eventos son procesados por los detectores hasta que, bien por la localización de un patrón o una anomalía, se produce una alerta.*

- *Las alertas son procesadas, en caso de ser necesario, por los consolidadores antes de ser enviadas. Estos se encargarán de enviar la información agrupada para ocupar el mínimo ancho de banda.*
- *Las alertas son recibidas por el colector a través de diferentes protocolos abiertos de comunicación.*
- *El parser se encarga de normalizarlas y guardarlas si proceden de la base de datos de eventos.*
- *El parser se encarga así mismo de cualificarlas determinando su prioridad según la política de seguridad definida en el framework y los datos sobre el sistema atacado, localizados en el Inventario de Sistemas.*
- *El parser valora el riesgo instantáneo que implica la alerta y en caso de ser necesario envía una alarma al Cuadro de Mandos.*
- *Las alertas cualificadas son enviadas a cada uno de los procesos de correlación que actualizarán sus variables de estado y eventualmente lanzarán nuevas alertas con una información más completa o fiable. Estas alertas son enviadas de nuevo al parser para su almacenamiento, priorización, valoración del riesgo, etc.*
- *El monitor de riesgos visualizará periódicamente la situación de cada uno de los índices de riesgo según han sido calculados por CALM.*
- *El cuadro de mandos mostrará las alarmas recientes, actualizará el estado de cada uno de los índices, los comparará respecto a los umbrales y lanzará nuevas alarmas o realizará las acciones correspondientes en caso de ser necesario.*
- *El administrador podrá desde el cuadro de mandos enlazar y visualizar, a través de la consola forense, todos los eventos ocurridos en el momento de la alerta.*
- *Podrá además comprobar el estado instantáneo de la máquina a través de los monitores de uso, de perfiles, y de sesiones.*

El siguiente gráfico muestra el flujo de los datos:

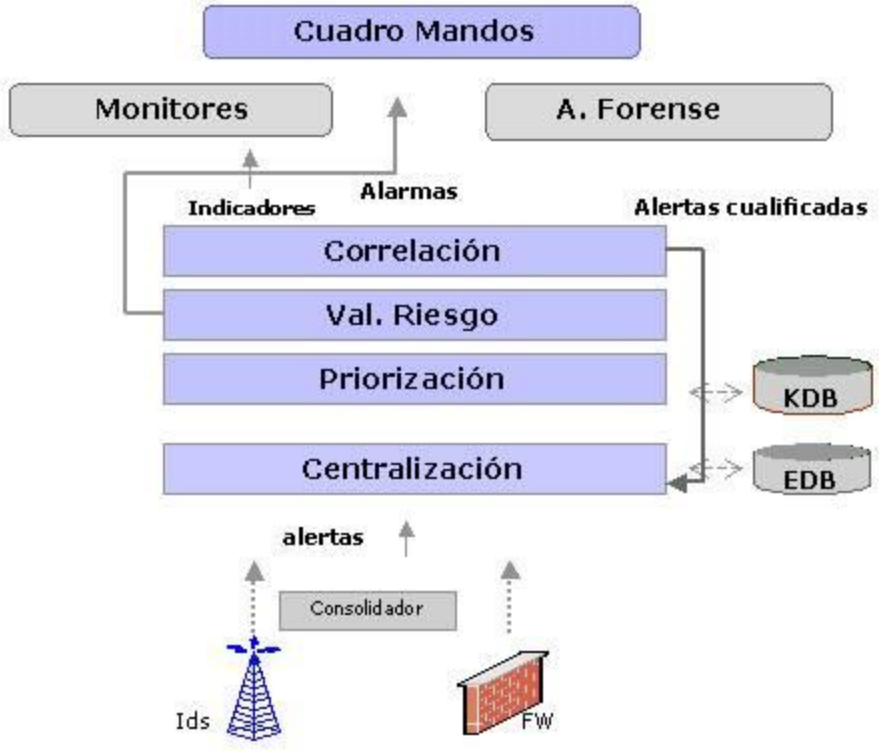


Figura #4: Flujo de datos. (7)

2.9 ARQUITECTURA DE MONITOREO

OSSIM posee una arquitectura formada por cuatro componentes: servidor (Consola de Gestión), framework (Interacción entre Módulos), base de datos (Eventos) y agentes centrales (Sensores Colectores).

Para diseñar una adecuada arquitectura de OSSIM dentro de la red, es importante tomar en cuenta algunos aspectos, como:

- El servidor OSSIM debe estar ubicado en un punto central de la red, con la protección adecuada para prevenir accesos no autorizados.
- Los distintos agentes centrales pueden estar distribuidos en cada segmento de red, o de forma en que el administrador de seguridad crea más conveniente. Se puede tener un solo agente central junto con el servidor OSSIM que analice toda la red o parte de ella.
- De acuerdo a la distancia de los agentes centrales, determinar las herramientas que conformarán cada uno de ellos, esto es en base a los servicios que va a monitorear. Con esto

determinamos, según la distancia de los agentes centrales, las herramientas a instalar en cada uno de ellos que integrarán OSSIM.

A continuación se presenta un diagrama de la arquitectura de monitoreo.

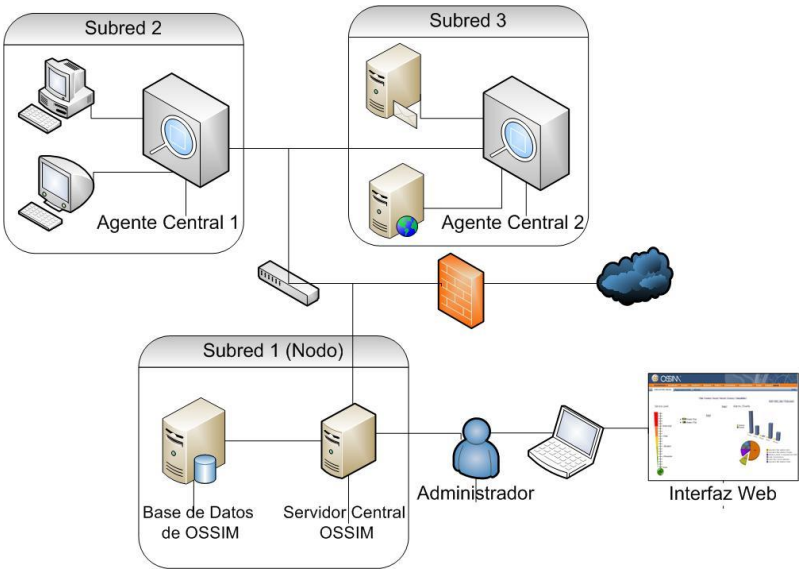


Figura #5: Distribución del servidor central y los agentes centrales.

2.10 CARACTERÍSTICAS DE LA RED UCI

La red de la Universidad de las Ciencias Informáticas (UCI) es una red LAN que brinda servicios a un gran número de usuarios. La misma garantiza un buen desempeño en el proceso docente, investigativo y productivo de la universidad, que dependen en gran medida del uso de las Tecnologías de la Información y las Comunicaciones (TIC). Por esta razón resulta imprescindible un eficiente funcionamiento de la red, la cual se ve sometida a un crecimiento constante en función de satisfacer las crecientes demandas de la universidad.

2.10.1 Descripción general de la red

La red de la Universidad de las Ciencias Informáticas (UCI) cuenta actualmente con alrededor de 7500 computadoras y poco más de 13 000 usuarios, los cuales tienen acceso a distintos servicios que brinda la universidad, como son: correo electrónico, navegación por Internet, intranet y mensajería instantánea, los cuales son soportados por servidores localizados en el nodo central por un personal capacitado. Los switches capa 3 de los nodos nivel 1 como el nodo central, docencia, parque tecnológico y residencia están conectados mediante fibra óptica con una velocidad de 10Gbps. El nodo

de parque tecnológico se conecta con sus módulos a una velocidad de 1 Gbps, al igual que el nodo de docencia con los nodos de los docentes 2, 3, 4 y 5. El nodo central con el rectorado, el docente 1 viejo y producción también se conectan a esa velocidad. El nodo de residencia se conecta con el nodo Biblioteca, Edificio 58 y Edificio 123 a una velocidad de 2 Gbps. Las conexiones entre máquinas de los edificios de residencia, laboratorios u oficinas se conectan a un switch capa 2 a una velocidad de 100 Mbps mediante cable UTP. Estos switches se conectan a sus respectivos subnodos a una velocidad de 100 Mbps pero mediante fibra óptica.

En la siguiente figura se muestra lo explicado anteriormente:

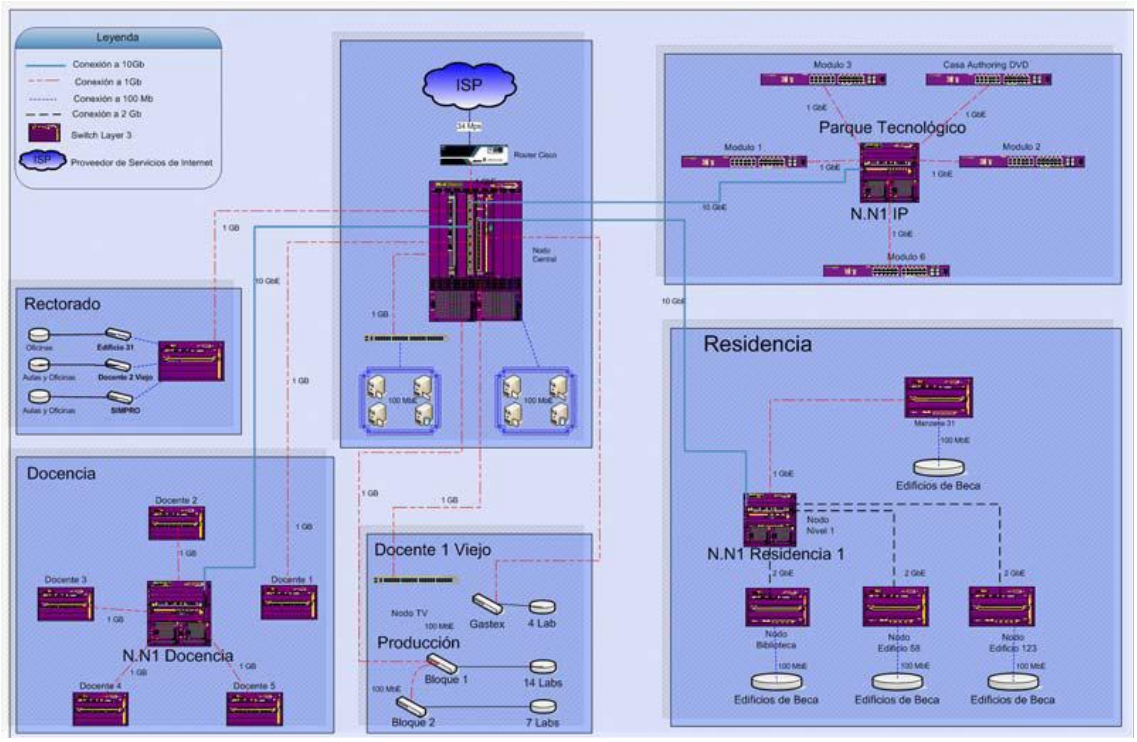


Figura # 6: Red UCI.

Se espera que la red de la UCI siga aumentando, puesto que existen nuevas facultades regionales en Artemisa, Ciego de Ávila y Manzanillo y hay expectativas de que existan en cada provincia del país, por lo que la cantidad de usuarios va a aumentar al igual que el tráfico en la red.

2.10.2 Topología y protocolo de enrutamiento en la UCI

La topología o forma lógica de una red se define como la forma de tender el cable a estaciones de trabajo individuales; por muros, suelos y techos del edificio. Existe un número de factores a considerar para determinar cuál topología es la más apropiada para una situación dada.

La topología en una red es la configuración adoptada por las estaciones de trabajo para conectarse entre sí. En la universidad la red se encuentra distribuida en topología en estrella.

Topología en estrella

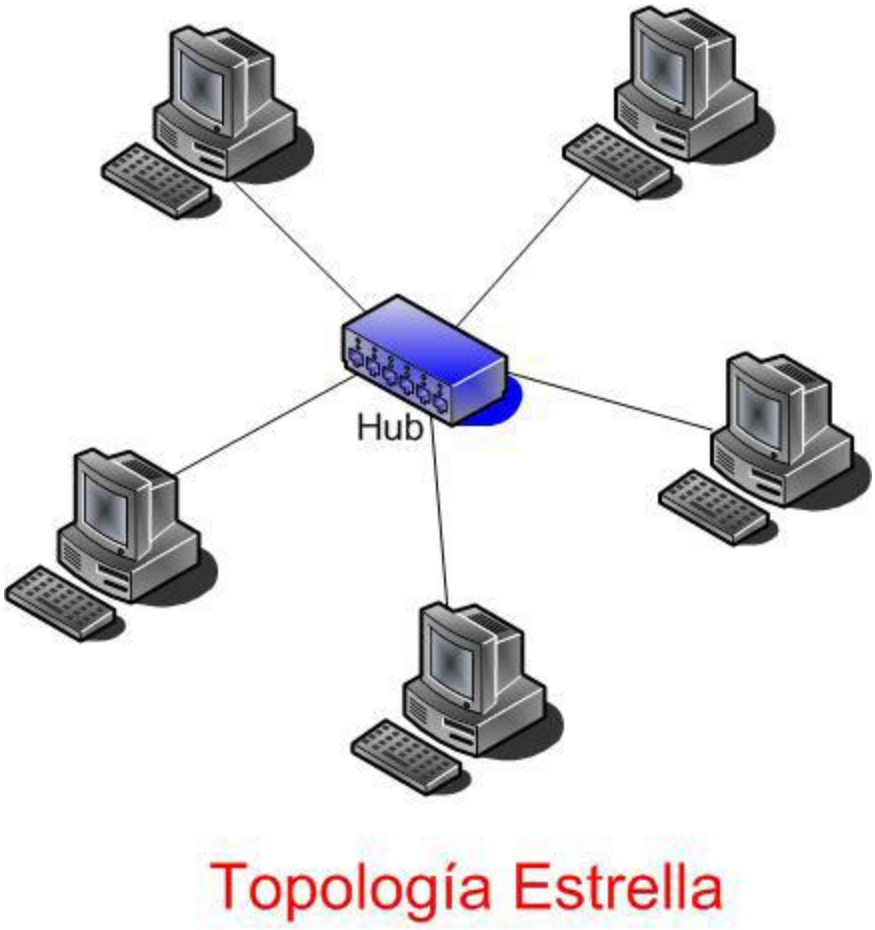
Reduce la posibilidad de fallo de red conectando todos los nodos a un nodo central. Todos los nodos periféricos se pueden comunicar con los demás transmitiendo o recibiendo del nodo central solamente. Un fallo en la línea de conexión de cualquier nodo con el nodo central provocaría el aislamiento de ese nodo respecto a los demás, pero el resto de sistemas permanecería intacto.

Los datos en estas redes fluyen del emisor hasta el concentrador, este realiza todas las funciones de la red, además actúa como amplificador de los datos.

La red se une en un único punto, normalmente con un panel de control centralizado, como un concentrador de cableado. Los bloques de información son dirigidos a través del panel de control central hacia sus destinos. Este esquema tiene una ventaja al tener un panel de control que monitorea el tráfico y evita las colisiones y una conexión interrumpida no afecta al resto de la red.

Debido a que la topología estrella utiliza un cable de conexión para cada computadora, es muy fácil de expandir, sólo dependerá del número de puertos disponibles en el hub o switch. La desventaja de esta topología en la centralización de la comunicación, ya que si el hub falla, toda la red se cae.

Topología en estrella de una red:



Figura# 7: Topología en Estrella.

Protocolo de Información de Enrutamiento, RIP

RIP calcula el camino más corto hacia la red destino usando el algoritmo del vector de distancias. La distancia o métrica está determinada por el número de saltos de ruteador hasta alcanzar la red de destino.

RIP no es capaz de detectar rutas circulares, por lo que necesita limitar el tamaño de la red a 15 saltos. Cuando la métrica de un destino alcanza el valor de 16, se considera como infinito y el destino es eliminado de la tabla (inalcanzable).

La métrica de un destino se calcula como la métrica comunicada por un vecino más la distancia en alcanzar a ese vecino. Teniendo en cuenta el límite de 15 saltos mencionado anteriormente. Las métricas se actualizan sólo en el caso de que la métrica anunciada más el coste en alcanzar sea

estrictamente menor a la almacenada. Sólo se actualizará a una métrica mayor si proviene del enrutador que anunció esa ruta.

En la Universidad este protocolo es ineficiente puesto que al enviar copias periódicas de las tablas de enrutamiento de un ruteador a otro, genera mucho tráfico en la red. Escoge el camino más corto, sin tener en cuenta el ancho de banda, ya que puede enviar los paquetes por un camino que tenga bajo ancho de banda. Además, como la red de la universidad está en topología en estrella un paquete para ir de una fuente a su destino tiene que dar muchos saltos y pudiera llegar a dar 16 saltos y el destino volverse inalcanzable. El algoritmo que utiliza este protocolo acumula información acerca de las distancias de la red, permitiéndole mantener una base de datos de la topología de la red. Sin embargo, los algoritmos de vector-distancia no permiten que un ruteador conozca la topología exacta de una red, ya que solo ven a sus ruteadores vecinos.

2.11 POSIBLES UBICACIONES DE OSSIM

Las aplicaciones que integrará OSSIM (sensores y aplicaciones servidoras(colectores)) pueden estar instaladas conjuntamente en la misma computadora o máquina donde tendremos el servidor OSSIM pero también pueden estar en otras computadoras (Agentes Centrales) para aligerar los procesos de la computadora donde tengamos el servidor OSSIM y porque tal vez no sea necesario tener todas las herramientas de seguridad para todas las subredes sino que se implementen según la necesidad en cada subred para optimizar las funciones. Con esto se tendrán Agentes Centrales con determinadas herramientas (aplicaciones) de seguridad para una subred o con las mismas aplicaciones para toda la red que se quiera monitorizar. Incluso se puede disponer de todas las herramientas para la red en general (todas las subredes) distribuidas en diferentes Agentes Centrales.

Si se cuenta con una máquina bien potente y excelentemente equipada se pueden tener instaladas las aplicaciones a integrar, es decir, un solo Agente Central conjuntamente con el servidor OSSIM en la misma computadora. Algunas herramientas necesitan de aplicaciones servidoras y agentes o clientes locales. Las aplicaciones servidoras serán instaladas en los Agentes Centrales o en la misma máquina donde este OSSIM. Los agentes o clientes que necesitan algunas de las herramientas integradas, como es el caso de OCS-NG Inventory, OSSEC, Osiris, Python, fw1-loggrabber, OpenVPN y Snare serán instalados en cada máquina (host) de cada segmento de red que se quiera monitorizar con dichas herramientas. Por lo anteriormente mencionado se escogerá el nodo central para la instalación del servidor OSSIM.

2.12 CARACTERÍSTICAS DE LA MÁQUINA DONDE SE INSTALARÁ OSSIM

La máquina teórica para una red de tráfico bajo puede ser de 300MHz con 128MB de RAM en adelante. Siempre se querrá tener una cantidad justa de espacio de la unidad de disco duro y una tarjeta de la red 1Gbps. Actualmente se utiliza una computadora con una memoria RAM de 512 MB, un disco duro de 160Gb, un Micro de 3.0 GHz y se cuenta con una tarjeta de red de 100Mbps. Con la distribución de Linux que se trabaja es el Debian 4.0

Se recomienda para un mayor aprovechamiento del sistema y una mayor rapidez en el funcionamiento de los procesos que estarán corriendo, instalar OSSIM en una computadora con 1Giga de RAM o superior, Micro a 3.0 GHz o superior, 160Gb de disco duro o más y con una tarjeta de red de 1Gbps o superior. Preferentemente se deberá utilizar un servidor con bastante espacio en disco duro para poder almacenar la inmensa cantidad de eventos registrados y con suficiente memoria RAM para no sobrecargar el equipo.

Las computadoras donde se instalen los Agentes Centrales deberán contar, como mínimo, con un Micro a 300MHz, 128MB de RAM y una tarjeta de red de 100Mbps cada una. Los requerimientos de disco duro no se especifican pues por lo general los Agentes Centrales requieren de muy poco espacio en disco, a penas unos Megas. Puede ser la misma computadora donde se instale el servidor OSSIM.

2.13 HERRAMIENTAS QUE INTEGRARÁ OSSIM EN LA UCI

Todas las incluidas en la versión 1.0.4 excepto la herramienta OpenVPN que no se configurará pues no se han establecido las configuraciones óptimas para su uso en la UCI. Esto deberá quedar como un caso de estudio para posteriores investigaciones sobre este tema ya que esta herramienta además de crear redes privadas virtuales es capaz de realizar balanceo de carga para los servidores.

2.14 VENTAJAS Y DESVENTAJAS DE OSSIM

Ventajas

- Integra diferentes aplicaciones para la seguridad informática.
- Disminuye los falso positivos y falsos negativos con la ayuda de la correlación.
- Permite realizar análisis forenses con los eventos almacenados.

- Tiene soporte de una comunidad abierta mundial que se encuentra en crecimiento constante.

Desventajas

- Solo se encarga de almacenar los eventos y reportarlos, pero no realiza ninguna acción para detener los ataques, excepto que puede definírsele que en casos previamente especificados levante una aplicación adicional o externa. Dicha aplicación será desde ese momento la encargada de realizar acciones para contrarrestar un ataque o dificultad. Claro está, el control, efectividad y funcionamiento de la aplicación llamada quedará fuera del dominio de OSSIM.

2.15 CONCLUSIONES

En este capitulo se han expuesto las características de OSSIM que lo hacen ventajoso frente a otras herramientas de seguridad de acuerdo a los requerimientos de la red de la universidad. Se han descrito el funcionamiento y los métodos que utiliza, así como las aplicaciones de seguridad integradas.

Esta descripción es sumamente importante para entender como funciona este SIM. Por todo lo expuesto anteriormente se concluye como propuesta a desarrollar en la presente tesis la configuración del SIM: OSSIM como mejor opción para elevar el nivel de seguridad de la red UCI.

3

CAPÍTULO
UTILIZAR OSSIM EN LA UNIVERSIDAD

3.1 INTRODUCCIÓN

OSSIM es una buena solución a los problemas de seguridad presentes en la UCI. Su aplicación traería como resultado más eficiencia en la seguridad de las redes, así como un ahorro de tiempo del personal de seguridad informática en el centro. Para su adecuada instalación es necesario tener cuidado con ciertos componentes, los cuales estando mal configurados o instalados pueden traer consigo mas perjuicios que beneficios. En este capitulo se describirá paso a paso la adecuada instalación de los componentes de OSSIM.

3.2 INSTALACIÓN DE OSSIM USANDO PAQUETES PRECOMPILADOS

Esta es una forma muy conveniente de instalar OSSIM pues así se puede definir cuales son las herramientas que se usaran en la empresa según las necesidades y las condiciones que presente la misma.

3.2.1 Pre-requisitos

Primero hay que hacer un análisis de las condiciones de la empresa, y se debe cumplir con determinados requisitos. Seguidamente se escogerá el sistema operativo y luego se configurará correctamente.

3.2.1.1 Instalación del sistema operativo

Instalar un Debian reciente, preferentemente Debian 4.0 (Etch). Se puede descargar el instalador desde (www.debian.org) o del repositorio de la Universidad (http://debian.prod.uci.cu/ etch main contrib non-free) Los paquetes Debian proporcionados por el equipo de desarrollo de OSSIM están compilados en un sistema Debian etch, por lo que habrá que instalar un Debian etch. Entre los paquetes elegidos ha de encontrarse solamente el sistema estándar, de forma que en los ordenadores del cliente quede instalado lo mínimo imprescindible.

3.2.1.2 Configuración de apt

Se editará el fichero */etc/apt/sources.list* para ver que repositorios se tienen. Se ha de eliminar aquellos que no sean de la distribución etch y añadir la siguiente línea para descargar los paquetes desde Internet:

deb http://www.ossim.net/download/ debian/

deb-src http://www.ossim.net/download/debian/

Si se quisieran instalar los paquetes desde una máquina o repositorio local hay que especificar el repositorio.

De esta manera se indicará al sistema de donde obtener las fuentes para instalar OSSIM. También es necesario crear el fichero */etc/apt/preferences* y dejarlo como se verá a continuación para indicarle al sistema que en caso de encontrar los paquetes a descargar en varios repositorios opte por el de OSSIM:

*Package: **

Pin: release o=ossim

Pin-Priority: 995

Estos pasos de instalación son para la versión 0.9.9.

3.2.1.3 Sincronización por NTP

Debido a que se tienen las aplicaciones generalmente instaladas en diferentes máquinas, los timestamps (la hora de los eventos) a la hora de insertarlos en las BDs (Bases de Datos) suelen ser diferentes. Diferentes máquinas tienden a desincronizarse rápidamente, lo cual no es bueno si se quiere correlar sucesos en diferentes redes y/o máquinas. Generalmente se instalará el servidor de tiempo en la misma máquina donde se tenga instalado el servidor OSSIM.

 # apt-get install ntp ntp-server ntp-simple

Las máquinas que pueden hacer consultas al servidor de tiempo deberían ser solo los Agentes Centrales y algunas veces la BD (Base de Datos). Para instalar el cliente en las máquinas que se deseen se deberá ejecutar:

 # apt-get install ntpdate

Se editará el fichero */etc/default/ntpdate*, asignando un valor al servidor, para que sepa el cliente donde consultar.

NTPSERVERS="IP_SERVIDOR"

Por último se configurará el crontab (el gestor de tareas a ejecutar periódicamente) para que ejecute la consulta cada hora:

*00 * * * * root /etc/init.d/ntpdate reload >> /dev/null 2>&1*

3.2.1.4 Configuración de debconf

Para configurar el resto de paquetes que se instalarán más adelante, estos paquetes se basan en debconf para hacer preguntas de configuración. Se debe configurar esta aplicación para que pida el nivel más alto de detalle:

 # dpkg-reconfigure -plow debconf

Se escogerá el nivel bajo y un modo de respuesta que no sea gráfico (no se han instalado las X se supone).

3.2.2 Instalar la Base de Datos de OSSIM

En la consola del Sistema Operativo se pondrá el comando:

 # apt-get install ossim-mysql

A continuación se muestran las preguntas que hará el sistema con las respuestas que se darán:

Contraseña para el usuario root de mysql (ojo que no es el mismo que el root de Debian). Es conveniente usar la misma contraseña a partir de ahora, debido a que deberemos usar muchas y podemos fácilmente equivocarnos más tarde cuando le debamos pasar a un programa la contraseña del usuario de otro para que el primero pueda usar al segundo, dando lugar a que nuestro sistema no funcione

¿Debemos establecer conexiones desde sistemas que ejecutan Sarge? No

En caso de querer modificar el puerto en el que escucha mysql se puede cambiar editando el fichero */etc/mysql/my.cnf*, aunque no es aconsejable (muchos programas de OSSIM esperan el puerto estándar y habría que reconfigurar esta característica) en principio.

Para crear las BDs para OSSIM se tecleará:

 # mysql -u root -p

(La contraseña que pide es la que se puso anteriormente para el mysql)

mysql> create database ossim;

mysql> create database ossim_acl;

mysql> create database snort;

mysql> create database osvdb;

mysql> exit;

Ahora se tiene que cargar las tablas en las bases de datos:

 # zcat /usr/share/doc/ossim-mysql/contrib/create_mysql.sql.gz \

/usr/share/doc/ossim-mysql/contrib/ossim_config.sql.gz \

/usr/share/doc/ossim-mysql/contrib/ossim_data.sql.gz \

/usr/share/doc/ossim-mysql/contrib/realsecure.sql.gz | \

mysql -u root ossim -p

 # zcat /usr/share/doc/ossim-mysql/contrib/create_snort_tbls_mysql.sql.gz \

/usr/share/doc/ossim-mysql/contrib/create_acid_tbls_mysql.sql.gz \

| *mysql -u root snort –p*

Las barras "\" se usan para decir a Debian que no ejecute aun el comando, que continua en la siguiente línea.

Los plugins de OSSIM estan situados en */usr/share/doc/ossim-mysql/contrib/plugins/*. Se podrán:

- Cargar los plugins que se necesiten, por ejemplo:

 # cd /usr/share/doc/ossim-mysql/contrib/plugins

 # zcat snort.sql.gz | mysql -u root ossim -p

 # cat arpwatch.sql p0f.sql pads.sql pam_unix.sql rrd.sql ssh.sql

*# sudo.sql *

nmap-monitor.sql ossim-monitor.sql | mysql -u root ossim –p

- O si no se sabe cuales instalar, se instalarán todos:

 # cd /usr/share/doc/ossim-mysql/contrib/plugins

*# zcat *.sql.gz | mysql -u root ossim -p*

*# cat *.sql | mysql -u root ossim -p*

3.2.3 Instalar el servidor OSSIM

Se deberá teclear en la consola:

 # apt-get install ossim-server

Se formularán las siguientes preguntas:

Nombre del servidor ossim: ossimserver (no es importante)

Dirección IP donde el programa servidor de OSSIM escuchará (podría haber más de una interfaz con su dirección IP en el sistema)

Nombre del framework: ossimframework (El framework se instalará más adelante)

Dirección IP del framework: la que sea (recordar que en cliente los programas estarán en diferentes equipos o quizás no).

Puerto del framework: presionar enter (lo coge por defecto)

Nombre de la BD de ossim: ossim

Nombre del host que alberga a la BD: se pondrá la dirección IP en lugar del nombre.

Nombre del usuario de la BD: root

Contraseña del usuario de la BD: el que se haya puesto al instalar la BD

Nombre la BD para snort que se va a usar: snort

Nombre del host que alberga la BD de snort: poner la dirección IP.

Nombre usuario de la BD snort: root

Contraseña del usuario de la BD de snort: la que se haya puesto antes.

(* Con esto se refiere a que se ha instalado en la misma máquina)

En caso de equivocación o cambio, y esto es valido para las siguientes instalaciones el comando a llamar es dpkg-reconfigure + nombre del paquete:

 # dpkg-reconfigure ossim-server

3.2.4 Instalar el Agente Central de OSSIM

Este agente o cliente del servidor, irá instalado en cada máquina que vigile una subred del cliente. Más adelante se verá como instalar las aplicaciones de seguridad que lo acompañan. Se deberá teclear:

 # apt-get install ossim-agent

Preguntas que se harán por el sistema:

Dirección IP del agente central: puede haber más de una interfaz en la máquina en la que se instala.

Interfaz por defecto del sistema: la que sea (generalmente suele ser eth0).

Dirección IP del servidor de ossim: la que se haya escogido anteriormente.

Pluggins que el agente central cargará: en principio los que vienen por defecto que son: arpwatch, ntop, pads, snort y syslog). En cada cliente se verá si se necesita alguno más.

Hay que editar manualmente el archivo */etc/ossim/agent/config.cfg*. Aún no se ha podido configurar adecuadamente el debconf para esto, con lo que si no pregunta nada hay que introducir las entradas a mano.

Se ha sustituido el password en la linea siguiente:

ossim_dsn=mysql:localhost:ossim:root:yoursecretpassword.

3.2.5 Instalar el framework de OSSIM

3.2.5.1 Instalar el Servidor Web

Hay que instalar el paquete apache con soporte para php. Se puede usar apache, apache2 o apacheSSL y php4 o php5. Se usará Apache2 y PHP5. Para esto se deberá teclear:

 # apt-get install apache2 php5 libapache2-mod-php5

Si se desea ir directamente a la instalación de OSSIM cuando se acceda a la página, una de las opciones es por ejemplo editar el fichero */etc/apache2/sites-available/default* y cambiar el RedirectMatch a “ossim”:

<Directory /var/www/>

Options Indexes FollowSymLinks MultiViews

AllowOverride None

Order allow,deny

allow from all

 # This directive allows us to have apache2's default start page

 # in /apache2-default/, but still have / go to the right place

RedirectMatch ^/$ /ossim/

#RedirectMatch ^/$ /apache2-default/

</Directory>

3.2.5.2 *Instalar phpGACL*

Se tecleará:

 # apt-get install phpgacl

El sistema generará las siguientes preguntas que se responderán a continuación:

Guardar las contraseñas de administración de ls BD en dbconf: SI

¿Se utilizará este servidor para acceder a BDs remotas? En cliente casi siempre si. La primera vez que se instala suele ser todo en la misma máquina por lo que se responderá no.

Configurar la BD para phpgacl con dbconfig-common: si

Tipo de BD para phpgacl: mysql

Método de conexión a la BD mysql de phpgacl: conexión socket. Si se ha instalado todo en el mismo ordenador, sino conexión tcp

Nombre usuario de administración de BD: root

Contraseña del usuario de administración de la BD: la que se haya puesto antes

Nombre de usuario para phpgacl: root

Contraseña: la que queramos. Recomendable que sea igual que las anteriores.

Nombre BD para phpgacl: ossim_acl

Versión Apache: All

Se deberá permitir la red editando */etc/phpgacl/apache.conf*, y descomentar la linea que dice *allow from all* quitando el # *y*a que se podrán añadir redes o cambiarlas en el futuro y editar cada vez el fichero es muy tedioso.

3.2.5.3 *Instalar el framework*

Si se trabaja con una versión vieja de Debian, al haber escogido php5 antes de instalar el framework, hace falta instalar varios paquetes que no vienen por defecto en debian (si vienen sus versiones para php4), aunque existen un par de casos como se verá más adelante que aun no tienen versión para php5 por lo que se usará la de php4.

 # apt-get install libphp-jpgraph

 # apt-get install php5-cli

 # apt-get install php5-gd

 # apt-get install php5-mysql

El paquete libphp-jpgraph depende de PHP4, por lo que desinstalará, al menos, libapache2-mod-php5 y php5. Luego en realidad se va a acabar teniendo una instalación en PHP4. Se tecleará:

 # apt-get install ossim-framework

El sistema preguntará lo siguiente:

Configurar BD acidbase con dbconfig-common: si

Tipo conexión con BD: socket unix si tenemos todo instalado en el mismo ordenador, sino conexión tcp

Tipo BD: mysql

Nombre usuario administración de la BD: root

Contraseña del usuario de administración: La que se quiera, preferiblemente la misma que las anteriores.

Nombre usuario para acidbase: root (no snort)

Contraseña para la aplicación mysql para acidbase: la misma de antes.

Nombre de la BD para acidbase: snort

Versiones Apache: All

BD a usar con ossim: ossim

Nombre del servidor de la BD a usar: usar la dirección IP en lugar del nombre.

Nombre del usuario de la BD a usar: root

Contraseña: la que se hubiese puesto antes.

Nombre de la BD de ossim_acl: ossim_acl

Nombre del host donde está alojada la BD: usar la dirección IP.

Nombre del usuario de administración de esta BD: root

Contraseña para el usuario: la que se hubiese puesto antes.

Servidor web a usar: All

En caso de que se haya configurado la BBDD con clave, hay que modificar en el fichero */etc/ossim/framework/ossim.conf* las variables *ossim_pass* y *phpgacl_pass*.

Se deben configurar los paquetes php con el siguiente comando (ver que hay varios que ponen php4; estos son los que no tienen aun versión en php5). Para esto se tecleará en la consola de Linux:

 # dpkg-reconfigure php5-cli php4-domxml php5-gd php5-mysql php4-xslt

Ahora se accederá al framework para comprobar que todo esté correcto. Ir a http://yourhost//ossim La primera vez saldrá que hay que hacer el setup de phpgacl, por lo que se deberá ir debajo de la página y pinchar en Let's get. Después dirá que hay que configurar las listas de acls y aparecerá una página con ellas. Abajo se presionará en back y ya se estará en la consola de administración de OSSIM. Para entrar el user y contraseña por defectos son admin ambas. Luego se deberá ir a Configuration/Menú para ver si hay que cambiar alguna contraseña. Para inicializar el demonio del framework en caso de ser necesario, se deberá teclear:

 # /etc/init.d/ossim-framework start

3.2.6 Instalar el paquete ossim-utils

Dependiendo de donde se tenga instalado el framework puede que no se tengan instaladas, por lo que se procederá a instalarlas, De estar en el sistema avisará. Se tecleará:

 # apt-get install ossim-utils

Quizás haya que reconfigurar con dpkg-reconfigre el paquete para los accesos a la BD de algunos scripts

3.2.7 Instalar Snort

Se tecleará:

 # apt-get install snort-mysql

Preguntas que se deberán responder:

Modo de arranque de snort: manual

Interfaz en la que escucha: any (eth0 da problemas)

Intervalo que monitorizará snort: any (generalmente el cliente no da los rangos)

Deshabilitar modo promiscuo: no

Cambiar orden reglas: no

Opciones adicionales: presionar enter

Enviar resumen por email: no

Configurar una BD a la que snort-mysql envie los registros: si

Nombre del servidor de la BD que se va a utilizar: usar la dirección IP

Nombre BD a utilizar: snort

Usuario: root

Contraseña usuario de la BD: contraseña que se pusiese antes

La estructura de la BD ha sido creada anteriormente por lo que se podrá borra el fichero */etc/snort/db-pending-config* usando rm –f fichero. Se debe editar el fichero */etc/snort/snort.conf* teniendo en cuenta que de las siguientes líneas el .. implica que puede haber líneas intermedias. El fichero debería parecerse a:

..

var HOME_NET [192.168.0.0/16]

Aquí debe ir la red donde esté snort instalado (si varios clientes varias subredes)

var EXTERNAL_NET !$HOME_NET

..

 # Variables needed by the bleeding snort rules

include $RULE_PATH/bleeding-all.rules

Se deberá descomentar la línea del incluye quitando el #

..

 # splitted in two lines for readability

 # replace values with your database settings

 # OSSIM output example:

output database: alert, mysql, user=root password=yourdbpass dbname=snort

host=yourdbhost sensor_name=your_sensor_ip logfile=alert

..

 # if you want spade support obtain a valid spade.conf file

 # (for example from ossim source or from ossim-contrib package)

include spade.ossim.conf

..

El resto ya está bien (incluye el pass correcto, el sensor, etc)

Se va a instalar reglas actualizadas para el snort. Se tecleará:

 # cd /etc/snort/rules/

 # wget http://www.bleedingsnort.com/bleeding-all.rules

Se actualizará la base de datos de ossim con las reglas del sistema (hace falta el paquete ossim-utils). Se tecleará:

 # /usr/share/ossim/scripts/create_sidmap.pl /etc/snort/rules

mysql -u root ossim –p

Ahora se iniciará el snort con: */etc/init.d/snort start* se hace: *tail -30 /var/log/syslog* para ver si al final da un fatal error. Si es por una regla, se va a */etc/snort/snort.conf* y se comenta la línea *include \$RULE_PATH/bleeding-all.rules* En cliente se deberá refinar las reglas para que funcionen sin quitarlas todas. Por ello se instalarán. Esto sucede porque se usa una versión de snort un poco antigua y estas reglas son las más modernas. Si el error es de que no se ha podido acceder con el usuario o parecido eso significa que el usuario no tiene permisos de escritura en la BD, lo que puede ser porque el usuario o la contraseña estén incorrectas (por eso se ha insistido tanto en usar siempre la misma contraseña). Otro problema frecuente por lo que puede que no correle es que esté saturado de eventos. En este caso lo mejor es refinar los eventos de snort. Las reglas se guardan en */etc/snort/rules/*. Primero se verá que tipo de regla es (en el directorio se tienen varios ficheros según el tipo de regla) y se toma su id (esa información viene dada por OSSIM) y se busca en el fichero correspondiente. Se podrá comentar la regla añadiendo un # al inicio de la línea, pero cuidado con eliminar reglas que sean necesarias. Otro problema que puede dar snort es con el formato de los logs que manda al agente central de OSSIM. Si no están en el formato correcto se puede tener problemas y que no correle. Para ello se verá el archivo */etc/snort/snort.conf* y que líneas se deben asegurar que estén comentadas y cuales no.

 # alert_syslog: log alerts to syslog esta línea debe estar comentada.

output alert_syslog: LOG_AUTH LOG_ALERT y esta descomentada

#output database: log, mysql, user=root password=temporal dbname=snort host=localhost esta debe ir comentada

output database: alert, mysql, user=root password=temporal dbname=snort host=localhost sensor_name=10.222.55.32 logfile=alert esta también irá descomentada

3.2.8 Instalar Ntop

Se istalará ntop tecleando:

apt-get install librrd2 ntop

Se preguntará lo siguiente:

Nombre de la interfaz que escuchará: el cliente puede tener más de una; generalmente será eth0, pero depende del cliente.

Nombre del usuario que correrá ntop: ntop

Se tendrá que definir la contraseña para el usuario y se inicia el servicio:

ntop -u ntop

>> Please enter the password for the admin user:

^C

/etc/init.d/ntop start

En un navegador de internet se va a http://yourhost:3000/ y se verá al ntop en acción. Se debe ir a plugins que está arriba, pinchar en round_robin databases (rrd) y pinchar en “configure”. En el apartado “Data to Dump”, activar “Hosts” y desactivar “interfaces”. Guardar preferencias (”Save preferences”). Una vez guardado, aparecerá un nuevo apartado “Hosts Filter”, en el que se deberá especificar la red en el formato especificado (ej: 172.22.0.0/255.255.0.0,192.168.5.0/255.255.255.0). Además se deberá editar el fichero /etc/default/ntop y en GETOPT=”” entre las comillas añadir —no-mac para que el rrdplugin trabaje con direcciones IP en lugar de macs para y además para que la dirección IP sea la clave principal en ntop y no las mac, sobre todo en sesiones.

El agente central se encargará de levantar ntop automáticamente. Se pueden ver los posibles errores en */var/log/ossim/agent.log*

Si se accede desde otra máquina diferente a la que tiene el framework instalado, modificar la configuración de Configuration→main la variable ntop_link para que coincida con la IP de donde está instalado:

ntop_link: http://dirección_IP_de_ossim:3000

3.2.9 Instalar Osiris

A pesar de que puede requerir alguna otra configuración adicional, para instalar Osiris se deberá ejecutar en la consola de Linux:

En el Agente Central:

 # apt-get install osiris osirismd

En cada host monitorizado:

 # apt-get install osirisd

3.2.10 Instalar otros plugins

Se configurará el p0f, el arpwatch, pads y tcptracker tecleando:

 # apt-get install p0f arpwatch pads tcptrack

No se debe dejar que arpwach ni pads se arranquen solos, sino que lo haga ossim:

 # update-rc.d -f arpwatch remove

 # update-rc.d -f pads remove

3.2.11 Trucos y problemas comunes

Con el comando tail –f y un fichero de log se pueden ver los logs en tiempo real de los diferentes programas de OSSIM. Es conveniente tenerlos en varias terminales abiertas a la vez para comprobar

que un suceso llegue desde snort al agente central y de ahí al servidor, o que el formato de los logs es el correcto, etc. Los diferentes logs son:

snort: /var/log/snort/alert

agente: /var/log/ossim/agent.log

servidor: /var/log/ossim/server.log

framework: /var/log/ossim/frameworkd.log

En /var/log/ossim/ existen también para el p0f, el arpwatch, etc.

Uno importante es /var/log/syslog debido a que registra todos los eventos del sistema, como por ejemplo fallo al arrancar algún programa, un programa ha dado fallo en ejecución, etc.

El tema de las direcciones de loopback (127.0.0.1) es problemático. Si se ha instalado todo en el mismo equipo o se tiene un Server-sonda (cliente y servidor en el mismo equipo por ejemplo para redes pequeñas) en teoría debería funcionar para comunicar las aplicaciones que estén en la misma máquina. Por lo general funciona bien para las direcciones IP de las BDs y no tan bien con el servidor y el agente central, aunque depende de la instalación. En caso de fallo se puede probar con ambas IPs en las configuraciones.

Puede que de este error o uno similar cuando se acceda a OSSIM la primera vez:

Warning: mysql_connect() [function.mysql-connect]: Access denied for user 'root'@'localhost' (using password: YES) in /usr/share/php/adodb/drivers/adodb-mysql.inc.php on line 358"

La solución es configurar bien las contraseñas en los campos snort_pass, osvdb_pass y backup_pass de la tabla “config” de la db ossim. Se puede hacer fácilmente utilizando el panel de control de OSSIM (Configuration→Main). No hará falta cambiarlas todas para que funcione correctamente, pero es aconsejable hacerlo para no perder nada de funcionalidad, obviamente.

Si el Plug-in RRD no carga: editar “*/etc/ossim/agent/plugins/rrd.cfg*” y sustituir la variable “*interfaces*” en la linea “*startup*”. Es decir, se deberá comentar la que viene por defecto y añadir una nueva. Quedaría así:

#startup=/usr/share/ossim/scripts/rrd_plugin.pl -d %(ossim_dsn)s -i %(interfaces)startup=/usr/share/ossim/scripts/rrd_plugin.pl -d %(ossim_dsn)s -i eth0

Si AcidBase (http://ossim/acidbase/) da el error: "*You don't have permission to access /acidbase//base_stat_alerts.php on this server*". Se deberá modificar el fichero "*/etc/acidbase/apache.conf*" y darles permiso a todos los hosts que se necesiten:

 # deny from all

 # allow from 127.0.0.0/255.0.0.0

allow from all

Finalmente reiniciar Apache:

 # /etc/init.d/apache2 restart

Si aparece el siguiente error:

Detected schema version greater than ossim version, please upgrade Ossim.

Refiriendose a la versión rc4 de OSSIM, se deberá ir al fichero: */usr/share/ossim/include/classes/About.inc* y cambiar la linea del *schema* por:

$this->version = "0.9.9rc5";

3.3 INSTALACIÓN Y CONFIGURACIÓN DE OSSIM UTILIZANDO UN ISO

Esta instalación y configuración son para la versión 1.0.4 de OSSIM.

3.3.1 Pre-requisitos

Tener OSSIM 1.0.4 en un CD o DVD para instalarlo. De no tenerlo se deberá obtener el ISO, esto se puede hacer descargándolo del sitio web www.ossim.com, y quemarlo en un CD o DVD.

La máquina donde se va a instalar debe contar con un lector de CD o DVD.

Se deberá crear 2 particiones en la máquina: una con formato Linux ext3 de 10 Gigas como mínimo y otra con formato Linux swap con el doble de tamaño que la memoria RAM de la máquina. La partición

Linux swap no debe ser menor a 1 G. Si la máquina donde se va a instalar OSSIM tiene menos de 512 de RAM se deberá crear una partición Linux swap de 1 Giga.

3.3.2 Iniciando la instalación

- Se deberá introducir el CD o DVD con OSSIM en la máquina a instalarlo. En esta máquina se instalarán tanto el Servidor de OSSIM como el Agente Central. En las máquinas que se quieran monitorizar solo habrá que instalar los clientes o agentes de las aplicaciones servidoras. Dichos clientes se conectarán al Agente Central que contiene las aplicaciones servidoras correspondientes.

- Se reiniciará la máquina para que botee desde el lector de disco, o sea, desde donde está el CD o el DVD.

3.3.3 Pasos para instalar Debian y OSSIM

- Se deberá escoger el idioma para los pasos de instalación. Para esto se seleccionará Spanish para ver la información en español o English para verla en inglés.
- Se deberá elegir la distribución del teclado. Para esto se seleccionará inglés estadounidense o español según el tipo de teclado que se este usando.
- Se pasará a configurar la red donde se tecleará el IP que tendrá la máquina. La mascará de subred será 255.255.255.0, la pasarela será la misma dirección IP pero cambiando el ultimo de los 4 espacios por 254 (o sea, *.*.*.254), los servidores de dominio serán 10.0.0.3 y 10.0.0.4 para la red de la UCI. El nombre de la máquina puede ser cualquiera, se recomienda, para mayor claridad a la hora de identificarla, llamarla OSSIM. El nombre de dominio será UCI.CU o bien puedo inventar uno.

Aquí el signo “*” significa que puede ser cualquier número que esté entre 0 y 254.

- Se realizará el particionado de discos escogiendo la partición donde se va a instalar el Debian con el OSSIM. Para esto se escogerá el particionado manual y se seleccionará la partición F ext3 presionando la tecla Enter en esa opción. En “utilizar como” se pondrá “sistema etx3 transaccional”. En “formatear la partición” se pondrá “sí, formatear”. En “punto de montaje” se pondrá “/ -sistema de ficheros raíz”. Se bajará hasta la opción donde dice “se ha terminado de definir la partición” y se pulsará la tecla Enter. Se seleccionará la partición “F intercambio intercambio” y se verificará que en “utilizar como” aparezca “área de intercambio”. Se debe

pulsar la tecla Enter y bajar hasta la opción: “finalizar el particionado y escribir los cambios en el disco” donde se dará Enter también. Se verificará que estén las particiones que fueron seleccionadas y se dará Enter en la opción: “Sí”.

- Para configurar el súper-usuario (root) se deberá teclear una clave que será la contraseña del root de Debian. Es recomendable ponerle admin a la clave para no olvidarla o confundirla con la de OSSIM que en principio será admin también. O sea Se recomienda tener la misma contraseña o password para el root de Debian, para OSSIM y para sus componentes. Otra opción recomendada es tener una clave para el root de Debian y otra para OSSIM y sus componentes.

Si se levanta un error cuando se esté configurando el gestor de paquetes diciendo que: “no se puede acceder a las actualizaciones de seguridad” no hay por qué preocuparse, solo se deberá dar Enter en “continuar” y listo.

- Al terminar estos pasos se reiniciará la máquina

A la primera vez que se están inicializando los procesos el ossim-server da un error por lo que se reiniciará de nuevo la máquina pulsando Ctrl+Alt+Delete y el ossim-server se iniciará correctamente.

- Se debe correr la actualización de OSSIM: */home/ossim/dist/ossim-update.pl*. Si no se tiene la actualización para la versión 1.0.4 de OSSIM se deberá descargar del sitio web www.ossim.com y correrla.

3.3.4 Actualizar OSSIM

Si la versión instalada necesita de alguna actualización se deberá correr la misma.

Mientras se realizó esta investigación se utilizó la versión 1.0.4 de OSSIM la cual necesita actualizarse. Para ello se descargará la actualización: Update.pl desde el sitio web www.ossim.com. El archivo Update.pl deberá correrse una vez instalado OSSIM o escribir en la consola de Linux:

/home/ossim/dist/ossim-update.pl

3.3.5 Configurar el servidor Apache

Para poder acceder a la consola (interfaz visual) de OSSIM desde cualquier máquina de la red se deberán configurar los puertos del servidor Apache.

Pasos a seguir:

- Abrir una consola de Linux y autenticarse como root.
- Teclear *nano /etc/apache2/ports.confs*
- Incluir los puertos por donde queremos conectarnos.
- *Listen 5800*
- *Listen 5900*
- Salir con Ctrl+x dando enter para salvar los cambios.
- Reiniciar el servidor Apache con: */etc/init.d/apache2 restart*
- Si se prefiere se puede reiniciar la máquina pero no es necesario.

Una vez configurado el servidor Apache ya se podrá acceder a OSSIM desde cualquier máquina de la red. Para ello se deberá abrir un navegador web en la computadora desde donde se quiera acceder y poner en la barra de direcciones:

http://IP_del_servidor_ossim:5800/ossim

También puede ser por el otro puerto configurado:

http://IP_del_servidor_ossim:5900/ossim

El IP_del_servidor_ossim no es más que la dirección IP de la máquina donde se tenga instalado el OSSIM.

3.3.6 Instalación y configuración de los agentes

En cada máquina que se quiera monitorizar y examinar con mayor detalle, principalmente los servidores, se deberá instalar los agentes de las herramientas integradas en OSSIM que lo requieran. Estos agentes locales, clientes o simplemente agentes se encargarán de enviar a las distintas aplicaciones servidoras toda la información recogida por los mismos. En síntesis, los agentes envían la información que se necesita de cada host donde se encuentren instalados hasta el servidor donde será procesada.

Las herramientas o aplicaciones que requieren agentes son: OSSIM, FW1-Loggrabber, OCS Inventory, OpenVPN, OSSEC, Python25, Snare. Dichos agentes se deberán descargar desde la consola de OSSIM, la ubicación está en la pestaña titulada Download, o se pueden descargar desde el sitio oficial de cada herramienta. La siguiente figura muestra donde se encuentran los agentes requeridos:

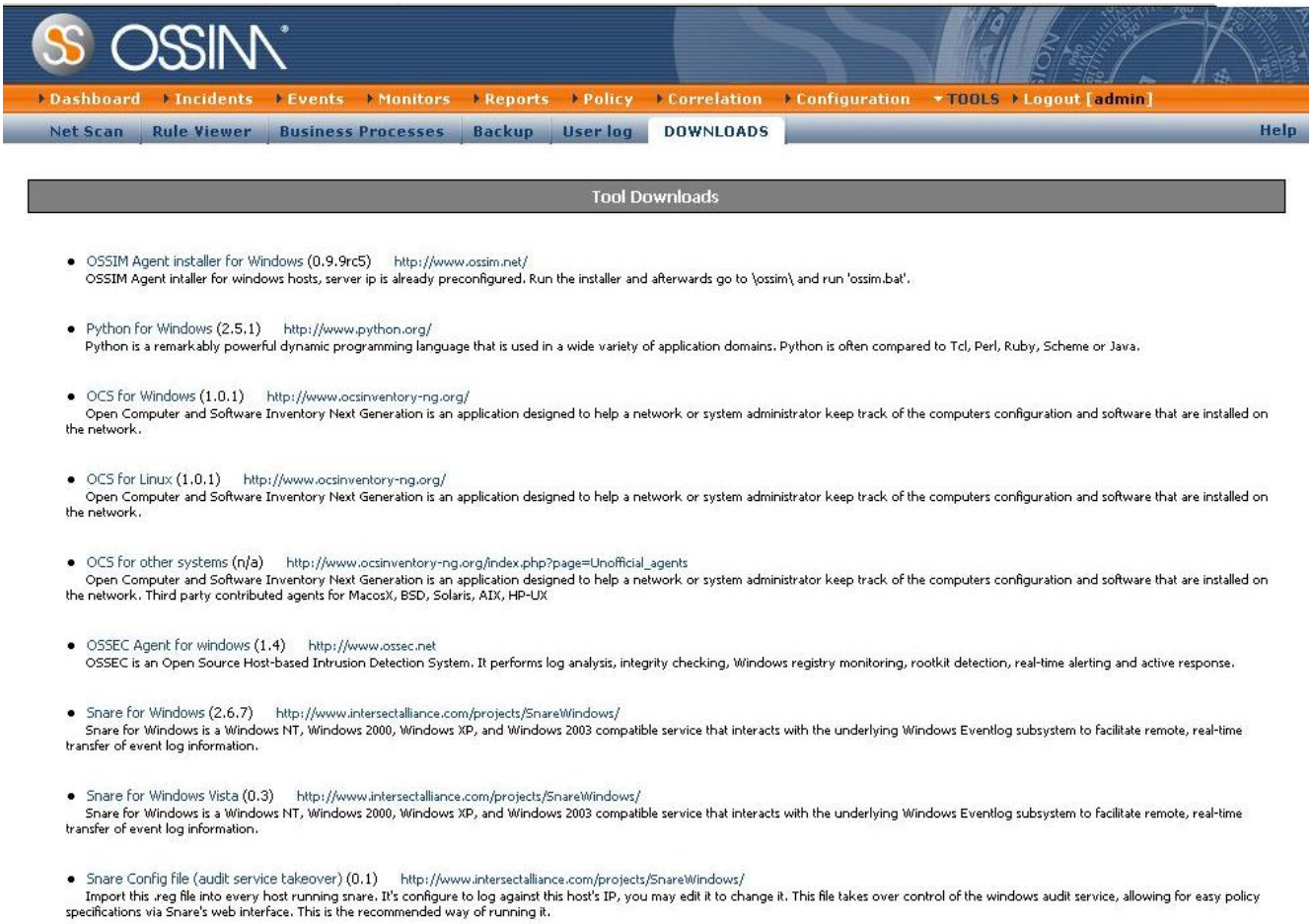


Figura # 8: Agentes.

3.3.6.1 Instalar ossim-agent

a) En sistema operativo Windows:

Una vez descargado el archivo llamado “ossim-install” se correrá, o sea, se ejecutará el mismo. Después se deberá ir a la dirección donde se instaló el agente y ejecutar el archivo llamado “ossim.bat”.

Se deberá ir a la siguiente dirección: “*C:\\Archivos de Programas\etc\ossim\agent”* y verificar que en el archivo llamado: “config.cfg” esté bien especificado el IP del servidor OSSIM en la línea de código perteneciente a “[output_server]”. También de cederá especificar, en el mismo archivo, la dirección IP del sensor que en este caso será la misma IP del servidor OSSIM.

En el archivo llamado config.xml se deberá especificar la dirección IP del servidor OSSIM en las siguientes lineas:

<!-- Replace 127.0.0.1 with your sensor Ip -->

<!ENTITY sensor "IP-Servidor-OSSIM" >

<!-- Replace localhost with your server Ip -->

<!ENTITY serverip " IP-Servidor-OSSIM " >

b) En sistema operativo Linux:

OSSIM no dispone de un agente en su web de descargas por lo que no será necesario instalar ninguno.

3.3.6.2 Instalar FW1-Loggrabber

a) En sistema operativo Windows:

Una vez descargado el archivo llamado “FW1-Loggrabber” se correrá el mismo. Después se deberá ir a la carpeta donde se instaló dicho agente y abrir al archivo llamado “lea.conf-sample” y guardarlo como “lea.conf” en esa misma carpeta. Lo mismo se hará con el archivo llamado “fw1-loggrabber.conf-sample” que se guardará como “fw1-loggrabber.conf”. Posteriormente se modificará el archivo “lea.conf” especificando el IP del servidor OSSIM.

b) En sistema operativo Linux:

Se deberá descargar el archivo de instalación del sitio http://www.logreport.org.

A partir de la versión 1.11, fw1-loggrabber incluye un instalador script muy básico para las distribuciones Linux y Solaris. Para utilizar este script de instalación: INSTALL.sh, se deberá modificar la configuración de PATH en INSTALL.sh de acuerdo a su entorno. Por defecto, el script instala fw1-loggrabber en / usr/local/fw1-loggrabber. El script de instalación preguntará si se desea adicionar dos variables de entorno al fichero de configuración de shell que se explicarán en la sección siguiente:

Variables de entorno:

fw1-loggrabber hace uso de dos variables de entorno, que debe definirse en el archivo de configuración de shell:

LOGGRABBER_CONFIG_PATH: esta variable define un directorio donde fw1-loggrabber busca sus archivos de configuración (fw1-loggrabber.conf, lea.conf). Si esta variable no está definida, fw1-loggrabber busca en el directorio actual de estos archivos de configuración.

LOGGRABBER_TEMP_PATH: esta variable define un directorio donde fw1-loggrabber almacena sus archivos temporales. Si esta variable no está definida, fw1-loggrabber almacenará estos archivos en el directorio actual.

3.3.6.3 Instalar OCS-NG Inventory-agent

a) En sistema operativo Windows:

Una vez descargado el archivo llamado OCSNG_WIN32_AGENT_1.01_repack se descomprimirá el mismo y se correrá el archivo llamado OcsAgentSetup.

Se deberá poner como servidor el IP o el nombre de la computadora donde se tenga instalado el OSSIM y marcar *no usar proxy.* El puerto de comunicación será el 80 que viene por defecto.

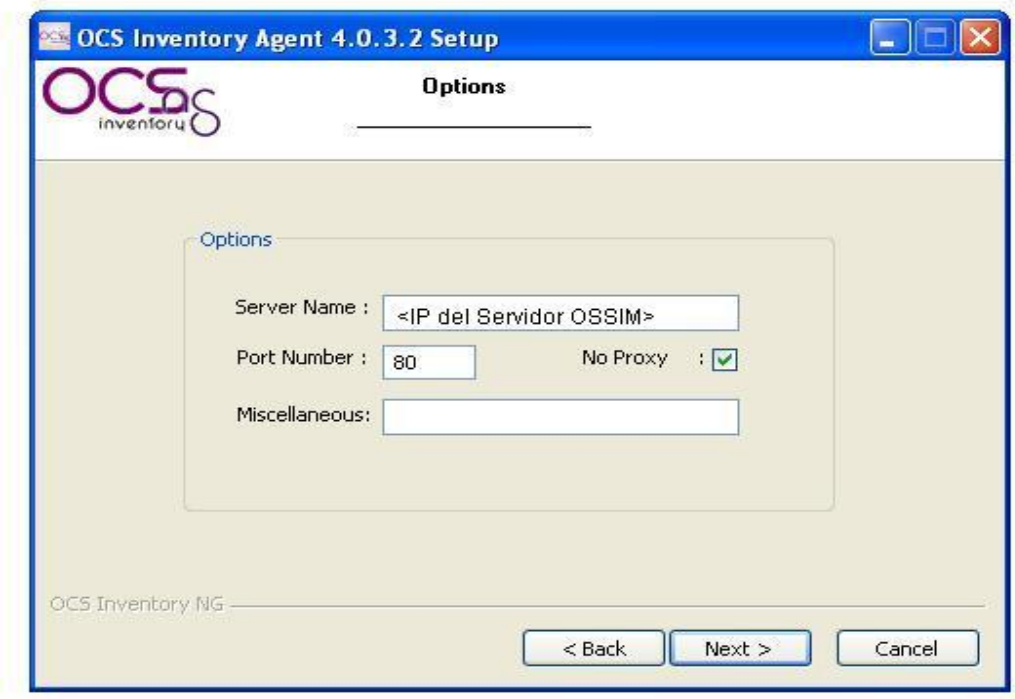


Figura #9: Instalación de OCS-NG Inventory.

b) En sistema operativo Linux:

El agente OCS Inventory NG para Linux sólo puede ser establecido a nivel local, o sea, se debe ir computadora por computadora para instalarlo. No se puede desplegar el agente a través de la red como es posible con el agente de OCS Inventory NG para Windows. Sin embargo, se puede elegir durante la instalación activar la actualización automática del agente si ha elegido HTTP como método de inventario.

Se deberá tener privilegios de root para configurar el servidor de administración.

Requerimientos:

El Agente OCS Inventory NG para Linux requiere:

- dmidecode versión 2.2 o superior.
- PERL 5.6 o superior.
- Perl module XML::Simple versión 2.12 o superior.
- Perl module Compress::Zlib version 1.33 o superior.
- Perl module Net::IP version 1.21 o superior.
- Perl module LWP::UserAgent version 5.800 o superior.
- Perl module Digest::MD5 version 2.33 o superior.
- Perl Module Net::SSLeay version 1.25 o superior.
- Make utility.
- Compilador C/C++ como GNU GCC.

Es mejor para la integridad del sistema utilizar paquetes precompilados para la distribución de Debian si están disponibles. En Linux, como Debian, se puede usar "apt-get" herramienta para configurar los módulos necesarios:

apt-get install libxml-simple-perl

apt-get install libcompress-zlib-perl

apt-get install libnet-ip-perl

apt-get install libwww-perl

apt-get install libdigest-md5-perl

apt-get install libnet-ssleay-perl

El nuevo instalador script "setup.sh" es capaz de instalar estas dependencias en caso de que no estén disponibles. Sin embargo, nunca se actualizará un módulo instalado. Si un módulo tiene versión inferior al previsto, se debe actualizar manualmente.

El instalador no establece los componentes necesarios para las dependencias. Por ejemplo, Net::SSLeay requiere openssl para ser instalado. Si no se instala, la configuración de Net::SSLeay fracasará y la configuración del agente de OCS Inventory NG también fracasará.

Por otra parte, el instalador script produce un archivo de registro "setup.log". Si se encuentra algún error durante la instalación del agente de OCS Inventory NG, se deberá referirse a este archivo para tener un mensaje de error detallado.

Instalación del agente de forma interactiva:

Descargar "OCSNG_LINUX_AGENT_1.01.tar.gz" del sitio oficial de OCS Inventory NG y desempaquetarlo.

tar-xvzf OCSNG_LINUX_AGENT_1.01.tar.gz

Se deberá ejecutar el instalador "setup.sh". Durante la instalación, por defecto se presenta la elección entre corchetes “[ ]”. Por ejemplo, [y] / n significa que "y" (sí) es la opción predeterminada, y "n" (no) es la otra opción.

• CD OCSNG_LINUX_AGENT_1.01

• sh setup.sh

El instalador escribe un archivo de registro "ocs_agent_setup.log" en el mismo directorio. Si se encuentra algún error se deberá consultar este registro para conocer con detalle el mensaje de error.

Entonces se tendrá que elegir entre 2 métodos para la generación del inventario:

1. http: La computadora donde se está instalando el agente está conectada a la red y es capaz de llegar a la comunicación con el servidor mediante el protocolo HTTP.

2. local: La computadora no está conectada a la red y el inventario se generará en un archivo para enviarlo manualmente al servidor de OCS Inventory NG.

Se deberá escoger "http" si su equipo puede llegar a comunicarse con el servidor OCS Inventory NG, o de lo contrario se escogerá "local".

Después se deberá especificar la dirección IP del servidor de OCS Inventory NG y el puerto por el que se establecerá la comunicación, que por lo general es el puerto 80.

Se deberá también especificar un valor para el TAG que no es más que el nombre del grupo al que pertenecerá el agente que se está instalando. Este valor puede estar creado anteriormente. En resumen se puede decir que el los TAG son los grupos en que se tendrán agrupadas las computadoras que tienen instalados los clientes de OCS Inventory NG.

El sistema comprobará lo siguiente:

- dmidecode binary.
- Compress::Zlib PERL module
- XML::Simple PERL module
- Net::IP PERL module
- LWP::UserAgent PERL module
- Digest::MD5 PERL module
- Net::SSLeay PERL module

Si no se encuentra alguno de estos componentes, se preguntará si se desea instalarlo. Se deberá entrar "y" o validar para permitir la instalación de los componentes necesarios. Si se introduce "n", la configuración se detendrá. Con esto quedará instalado el agente.

Aquí se muestra un ejemplo del fichero de configuración “ocsinv.conf” para una computadora conectada a una red del agente de OCS Inventory NG para Linux:

<CONF>
<DEVICEID> Computer.domain.tld-2006-02-27-13-59-47 </ DEVICEID>

<DMIVERSION> 2,2 </ DMIVERSION>

<IPDISCOVER_VERSION> 3 </ IPDISCOVER_VERSION>

<OCSFSERVER> My_ocs_com_server.domain.tld: 80 </ OCSFSERVER>

</ CONF>

Desplegar agente a través de guiones de instalación (script) sin la interacción del usuario:

Se deberá descargar "OCSNG_LINUX_AGENT_1.01.tar.gz" del sitio web oficial de OCS Inventario NG y desempaquetarlo.

• tar-xvzf OCSNG_LINUX_AGENT_1.01.tar.gz

Después se deberá ejecutar el instalador "setup.sh" con los siguientes argumentos de línea de comandos:

• cd OCSNG_LINUX_AGENT_1.01

• sh setup.sh <SETUP DEPENDENCIES> <server ADDRESS> [<server PORT> (Tag VALUE>]

Donde los valores de los parámetros son los siguientes:

• <SETUP DEPENDENCIES> debe ser "1" si desea habilitar la configuración automática de dependencias faltantes, "0" para desactivar (configuración de fallará si hay desaparecidos dependencia).
• <SERVER ADDRESS> debe ser la dirección IP o el nombre DNS del servidor OCS Inventory NG. Si se planea establecer el agente en modo local pues no se tiene la computadora conectada a la red, se deberá configurar < SERVER ADDRESS> definiéndole "local".

• < SERVER PORT> puede ser el puerto 80 si no se está utilizando para otro servicio.

• <Tag VALUE> puede ser el valor de TAG, puesto entre comillas.

Los parámetros de línea de comando <SETUP DEPENDENCIES> y <server ADDRESS> son obligatorios. Los demás parámetros son opcionales, pero si se desea cambiar el <TAG VALUE>, se deberá también especificar el anterior parámetro opcional <server PORT>.

El instalador escribe un archivo de registro "ocs_agent_setup.log" en el mismo directorio. Si se encuentra algún error se deberá consultar este registro para conocer con detalle el mensaje de error.

3.3.6.4 Instalar OSSEC-agent

a) En sistema operativo Windows:

La comunicación entre el servidor y el agente de OSSEC es segura (encriptada y autenticada). Para cada agente instalado se deberá crear una llave de autenticación (authentication key) en el servidor de la aplicación OSSEC.

- Primeramente se deberá adicionar el agente dentro del servidor de OSSEC que está en la máquina donde se tenga instalado, o sea, en la computadora donde tenemos el OSSIM. Para esto se deberá correr el comando *"manage_agents"* abriendo una consola de Linux en dicha máquina y una vez autenticado como root teclear:

/var/ossec/bin/manage_agents

Los pasos a seguir son los siguientes:

```
****************************************
* OSSEC HIDS v0.8 Agent manager.       *
* The following options are available: *
****************************************
   (A)dd an agent (A).
   (E)xtract key for an agent (E).
   (L)ist already added agents (L).
   (R)emove an agent (R).
```

(Q)uit.

Choose your actions: A,E,R or Q: **A *(Se pondrá la opción A que es para adicionar un agente).***

- Adding a new agent (use 'q' to return to main menu).

Please provide the following:

* A name for the new agent: **nombre_del_agente *(puede ser el nombre de la computadora donde se quiera instalar el agente).***

* The IP Address for the new agent: **x.x.x.x *(IP de la computadora donde se vaya a instalar el agente).***

* An ID for the new agent[001]: ***dar Enter en esta opción para que el programa lo escoja por defecto.***

Agent information:

ID:001

Name: nombre_del_agente

IP Address:x.x.x.x

Confirm adding it?(y/n): **y**

Added.

1. Después que se haya adicionado el agente se deberá extraer la llave de autenticación que usará para establecer la comunicación con el servidor. Para esto en el "manage_agents" seleccionaremos la opción "E" y proveer el ID del agente. La llave generada será copiada y usada a la hora de instalar el agente. Quedaría de la siguiente manera:

****

* OSSEC HIDS v0.8 Agent manager. *

* The following options are available: *

****

(A)dd an agent (A).

(E)xtract key for an agent (E).

(L)ist already added agents (L).

(R)emove an agent (R).

(Q)uit.

Choose your actions: A,E,R or Q: **E *(Esta opción es para extraer la llave de autenticación)***

Available agents:

ID: 001, Name: nombre_del_agente, IP: x.x.x.x

ID: 002, Name: obsd1, IP: 192.168.2.10

Provide the ID of the agent you want to extract the key: **001 *(Aqui se selecciona a que agente va generarsele la llave).***

Agent key information for '001' is:

CDAxIGxpbnX4MSAxOTIuMTY4LjAuMzIgOWM5MENlYzNXXXYYYZZZZZ== ***(Esta es la llave generada)***

** Press ENTER to continue

La llave generada se deberá copiar.

2. Después de estos pasos se deberá ir a la computadora donde se quiera instalar el agente cuya clave se generó y correr el archivo llamado ossec-agent-win32-1.4. Se deberá especificar el IP del servidor que será el IP de la computadora donde se tenga instalado el OSSIM y se pegará la llave de autenticación generada para el agente.

b) En sistema operativo Linux:

Se deben seguir los pasos 1 y 2 igual que para el sistema operativo Windows. Además se deberá descargar el agente para Linux del sitio oficial de OSSEC: *http://www.ossec.net* o utilizar los siguientes comando para descargarlo del sitio web:

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

wget http://www.ossec.net/files/ossec-hids-latest_sum.txt

Cuando se trabaja de forma cliente-servidor se deberá tener en cuenta que: si existen uno o más cortafuegos entre el servidor y el cliente hay que abrir el puerto 1514 UDP, para que OSSEC pueda comunicarse sin problemas con los clientes.

A continuación, se deberán ejecutar los siguientes comandos en el siguiente orden:

md5 ossec-hids-latest.tar.gz y sha1 *ossec-hids-latest.tar.gz*

Se debe tener en cuenta que los comandos md5 y sha1 no existen en algunos sistemas operativos en ese caso se deberán utilizar los comandos md5sum y sha1sum respectivamente.

Lo siguiente es descomprimir el archivo y entrar en el directorio recién creado, esto se podrá hacer con:

tar -zxvf ossec-hids-.tar.gz (o se podrá utilizar gunzip -d;tar -xvf de ser preciso)*

*cd ossec-hids-**

Seguidamnete se podrá ejecutar el script de instalación:

./install.sh

El sistema formulará las siguientes preguntas, las cuales se muestran a continuación con la respuesta correspondiente que se deberá teclear por quien esté llevando a cabo la instalación:

Que tipo de instalación Usted desea (servidor, agente, local ó ayuda)? ***agente***

Elija donde instalar OSSEC HIDS [/var/ossec]: ***Presionar la tecla ENTER***

*Cuáles la dirección del servidor OSSEC HIDS****?: IP_servidor_OSSIM***

Desea Usted agregar el servidor de integridad del sistema? (s/n) [s]: ***s***

Desea Usted agregar el sistema de detección de rootkit? (s/n) [s]: ***s***

Desea Usted habilitar respuesta activa? (s/n) [s]: ***s***

Con esto ya estará listo el OSSEC para ejecutarse, para ello los comandos son:

Para comenzar OSSEC HIDS:

/usr/share/seguridad/ossec/bin/osseccontrol start

Para detener OSSEC HIDS:

/usr/share/seguridad/ossec/bin/osseccontrol stop

La configuración puede ser leída ó mofificada en

/usr/share/seguridad/ossec/etc/ossec.conf

Con estos pasos ya queda instalado el agente de OSSEC.

3.3.6.5 Instalar Snare-agent

a) En sistema operativo Windows:

Una vez descargado el archivo llamado SnareSetup-2.6.7-MultiArch se correrá el mismo.

Después se deberá establecer la comunicación con el servidor OSSIM. Para esto se va a la siguiente dirección: “http://localhost:6161/network” y en “Destination Snare Server address” se deberá especificar el IP del servidor OSSIM.

b) En sistema operativo Linux:

Aunque se pueden instalar agentes de Snare en sistemas operativos Linux, no hay necesidad de hacerlo puesto que estos sistemas vienen con su propio syslog que debe ser configurado para que envíe los log hacia el servidor syslog de OSSIM. Evidentemente, OSSIM utiliza también este servidor syslog para recibir los log enviados por los agentes de Snare para Windows.

3.3.6.6 Instalar Osiris-agent

Tras la instalación y puesta en funcionamiento de un sistema operativo, es altamente recomendable realizar una revisión periódica para determinar si éste ha sido modificado. Una modificación no controlada de un sistema puede suponer bien que algún usuario o administrador de éste ha realizado cambios no controlados ni documentados, o bien que se ha producido una intrusión en el mismo. Un servicio de control de integridad nos permitirá descubrir de forma rápida si alguna parte del sistema se ha visto modificada.

a) En sistema operativo Windows:

Una vez descargado el archivo llamado osiris-4.2.3-win32 se correrá el mismo. La primera pantalla que se presenta muestra el Acuerdo de Licencia de Usuario Final.

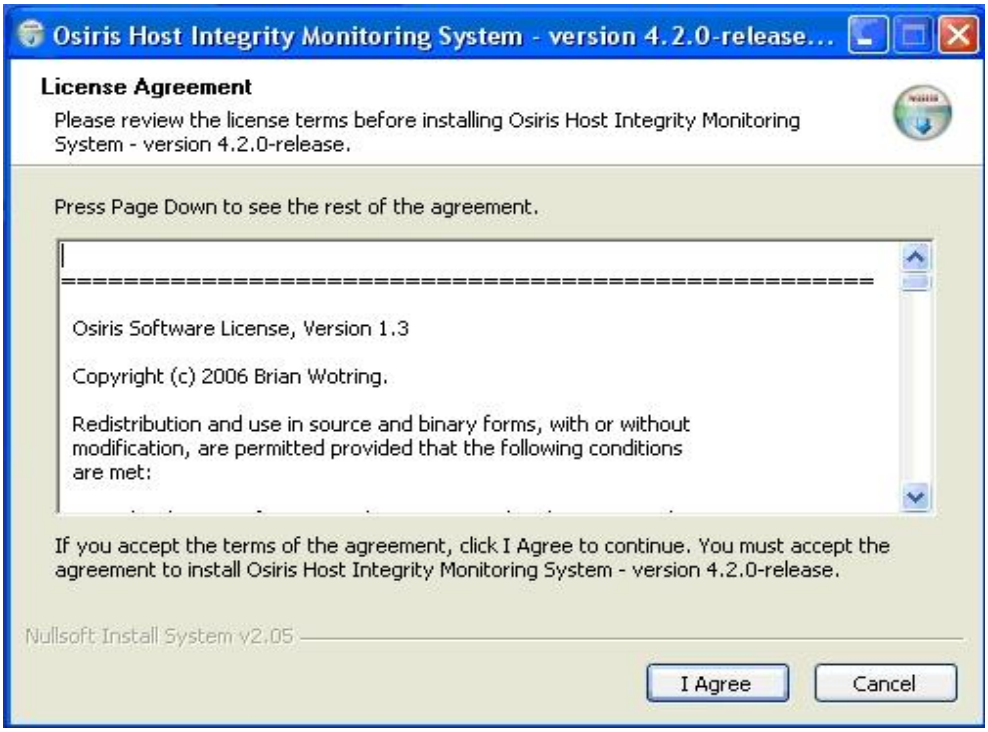


Figura #10: Instalación de Osiris (paso 1).

Se deberá hacer click en "I Agree" y a continuación se presentará una pantalla en la cual seleccionar los componentes a instalar, en este caso únicamente se debe seleccionar el agente "Scanning Service".

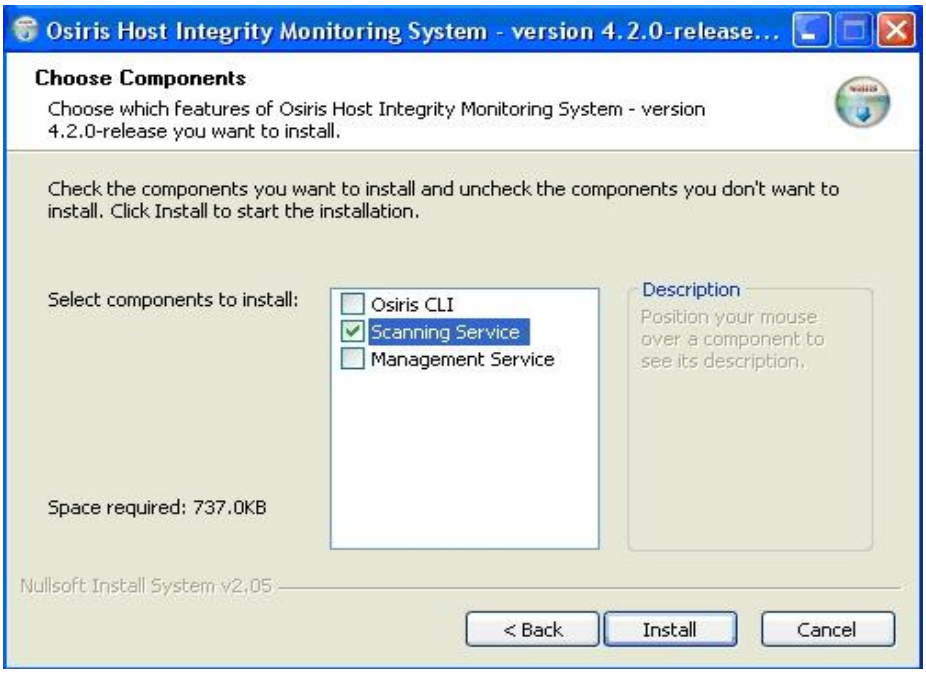


Figura #1: Instalación de Osiris (paso 2).

A continuación se deberá hacer click en "Install" con lo que comenzará el proceso de instalación. Una vez finalizado éste, la pantalla mostrará la frase "Setup was completed sucessfully" en la parte superior izquierda del cuadro de diálogo.

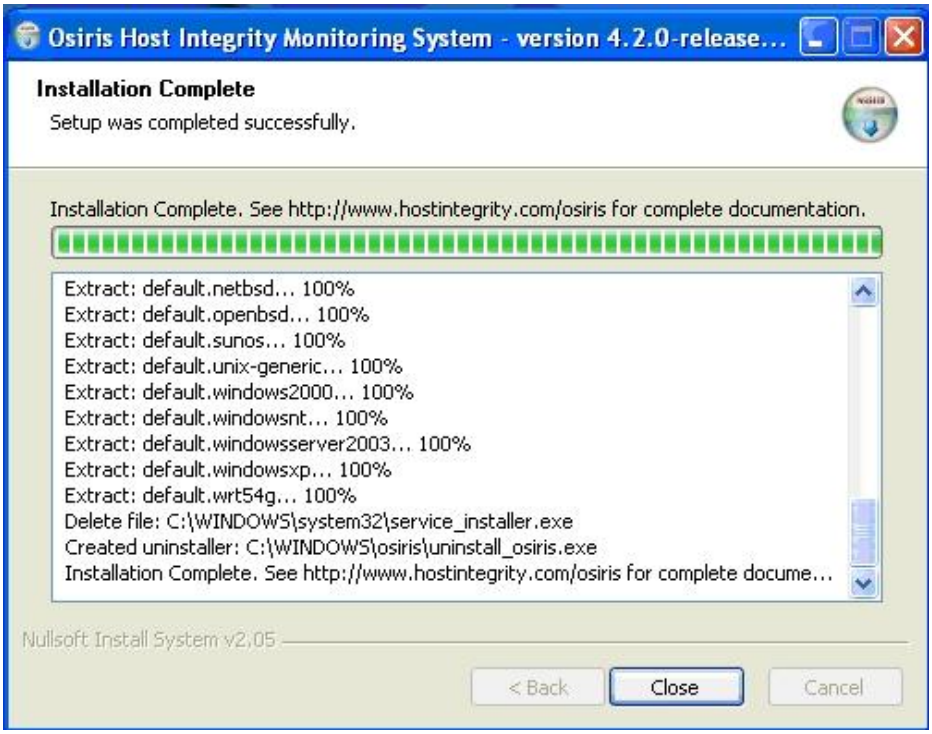


Figura #12: Instalación de Osiris (paso 3).

Por último, se deberá hacer clic en el botón "Close" con ello el cuadro de diálogo se cerrará y habrá terminado el proceso de instalación. En este momento, el agente se estará ejecutando como un servicio y únicamente aceptará conexiones procedentes de la consola de gestión.

b) En sistema operativo Linux:

OSSIM no dispone de ningún agente de Osiris para Linux por lo que no será necesario instalar ninguno. En este caso se utiliza otra aplicación llamada Syslog.

3.3.6.7 Instalar Python

a) En sistema operativo Windows:

Una vez descargado el archivo llamado python-2.5.1 se correrá el mismo y seguir los pasos hasta terminar.

b) En sistema Operativo Linux

En el caso de los sistemas Linux se deberá verificar que ya lo tenga instalado, de no tenerlo se deberá instalar del repositorio.

3.3.7 Configuración de Nagios

Nagios es un monitor de disponibilidad que verifica el estado de los host que esté monitoreando así como de los servicios que estén corriendo en los mismos. Principalmente el Nagios es utilizado para informar sobre el estado de servidores y equipos en la red que sean de un alto valor ya sea por sus servicios o por otro motivo importante.

La configuración de esta herramienta es muy fácil. Primeramente se deberá ir a la siguiente dirección, se podrá usar el comando “mc” para disponer de la consola MC de Linux, pero se deberá autenticarse como root:

/etc/nagios2

Una vez allí se deberán crear, pueden crearse con el comando nano, los siguientes archivos:

hots.cfg

En este archivo se especificarán los host (equipos) que va a monitorear el Nagios, deben incluirse todo los que se requieran.

hostgroups.cfg

En este archivo se organizarán los host de acuerdo a su clasificación como por ejemplo: servidores web, servidores DNS, servidores de base de datos, swiches o gateway y todos los servidores y equipos importantes.

services.cfg

En este archivo se especificarán que servicios queremos monitorear para verificar su estado.

Después de estos pasos se pasará a configurar como tal cada archivo creado.

Para modificar el archivo *hosts.cfg:*

Se deberá abrir el archivo hosts.cfg creado y modificarlo. Para esto una vez que se esté dentro del archivo habrá que incluir las siguientes líneas:

define host{

use *generic-host* *; Template to use*

host_name aquí debe incluirse el nombre del equipo a monitorear.

alias aquí se pondrá el nombre del equipo igualmente.

address aquí incluimos el IP de del equipo a monitorear.

}

Ejemplo:

define host{

use *generic-host* *; Template to use*

host_name *ossim.uci.cu*

alias ossim

address 10.31.20.121

}

Esto debe hacerse para cada host que se quiera monitorear.

Para modificar el archivo *hostgroups.cfg:*

Se deberá abrir el archivo hostgroups.cfg creado y modificarlo. Para esto una vez que se esté dentro del archivo habrá que incluir las siguientes líneas:

define hostgroup {

hostgroup_name aquí se debe especificar el tipo de host que incluirá este grupo.

alias aquí se debe especificar el tipo de host que incluirá este grupo como en el hostgroup_name.

members aquí incluimos los hosts o equipos que pertenezcan a este grupo separados por comas.

}

Ejemplo:

define hostgroup {

hostgroup_name http-servers

alias HTTP servers

members ossim.uci.cu,ucixs12.uci.cu,pepito.dominio.com,fulanito

}

Esto debe hacerse para cada grupo de hosts en los que se decida organizar los host a monitorear.

Para modificar el archivo *services.cfg:*

Se deberá abrir el archivo services.cfg creado y modificarlo. Para esto una vez que se esté dentro del archivo habrá que incluir las siguientes líneas:

define service {

hostgroup_name aquí se debe poner el nombre del grupo, previamente creado, que brinda el servicio a monitorear.

service_description aquí se deberá incluir el tipo de servicio a monitorear.

check_command aquí se debe poner el comando para chequear el servicio.

use generic-service ; Template to use

notification_interval 0

}

Ejemplo:

define service {

hostgroup_name http-servers

service_description HTTP

check_command check_http

use generic-service ; Template to use

notification_interval 0

}

Esto debe hacerse para cada servicio que se quiera monitorear.

Después de estos pasos se deberá abrir el archivo *nagios2.cfg* para modificarlo. Este archivo está en la siguiente dirección:

/etc/nagios2/nagios.cfg

Una vez dentro del archivo se deberán descomentarear, quitando el signo de número (#), las siguientes líneas:

cfg_file=/etc/nagios2/hostgroups.cfg

cfg_file=/etc/nagios2/hosts.cfg

cfg_file=/etc/nagios2/services.cfg

Por último se deberá reiniciar el servicio de Nagios para que adopte las nuevas configuraciones. Esto puede hacerse tecleando la siguiente línea en la consola de Linux:

/etc/init.d/nagios2 restart

3.3.8 Configuración de Pam_Unix

El Pam_Unix es un detector que analiza los logs para Unix y Linux. Este detector registra una gran cantidad de eventos de los cuales los de tipo “pam_unix: authentication successful” cubren un por ciento excesivamente grande, estos son entre un 80 y 84 por ciento del total. Esto traería como consecuencia que la base de datos de OSSIM aumentaría en un breve tiempo hasta escasearse el espacio en disco disponible. En las pruebas realizadas al software se reportaron 2873 de tipo pam_unix: authentication successful, de 3433 eventos en total en 4 días para un 84 por ciento. Esto trajo como consecuencia que el espacio en disco, que para realizar las pruebas fue de 31 Gigabyte, fuera insuficiente y se reportó una alerta por parte del Nagios con dicha advertencia. Este tipo de evento se refiere a autenticaciones internas de OSSIM y sus aplicaciones que en caso de ser realizadas satisfactoriamente no existe ningún tipo de problema por lo que no son relevantes. Lo alarmante sería, en caso contrario: que falle alguna de estas autenticaciones pues entonces se sabría que hubo problemas al intentar abrir una sesión por parte del sistema. Para remediar esta dificultad de espacio de almacenamiento se tienen las siguientes variantes:

- Disponer de un espacio casi ilimitado en disco duro del servidor donde se encuentre la base de datos para poder almacenar todos los eventos de seguridad.
- Realizar salvas periódicas en determinadas fechas con antelación a que se consuma todo el espacio de disco duro disponible y borrar de la base de datos todos los eventos anteriores a dichas fechas.

- Parar el funcionamiento del detector Pam_Unix.
- Deshabilitar el registro de los eventos de tipo “pam_unix: authentication successful” y mantener el resto de las funcionalidades del detector en funcionamiento.

De estas variantes solo la segunda y la última son convenientes pues la primera es casi imposible, además de requerir una inversión mucho mayor en hardware, y la tercera sería una deficiencia del sistema a la hora de detectar otro tipo de evento como lo sería un fallo en la autenticación. De las dos variantes válidas se escogió la última porque no requiere de ninguna funcionalidad adicional y se seguiría contando con todos los eventos en la base de datos por un período de tiempo mucho mayor. Además de esto seguiría el funcionamiento de todas las funcionalidades del detector Pam_Unix excepto la de registrar los eventos de tipo “pam_unix: authentication successful” y con esto se estaría eliminando información innecesaria de la consola forense. El registro de fallos en la autenticación, en caso de ocurrir, sí seguiría reportándose como un evento. A continuación se explica como realizar este cambio:

Se deberá ir al archivo de configuración del Pam_Unix en la siguiente dirección y modificar el archivo llamado *pam_unix.cfg*:

/etc/ossim/agent/plugins/

Se deberá buscar donde dice [pam_unix-autentification-successful] y comentar todo el bloque tecleando el signo # delante de cada línea.

3.3.9 Mapa de Nagios del panel de control

El mapa que se encuentra en la página principal de OSSIM no es más que una imagen estática almacenada en:

/usr/share/ossim/www/nagios.png

Esta imagen debe de cambiarse por otra que se haya generado previamente, ya sea en Nagios u otro programa, con el estado actual de la red que se esté monitorizando.

3.4 POLÍTICAS

A continuación se mencionan las políticas contenidas en el Plan de Seguridad Informática de la UCI, de las cuales se han seleccionado las necesarias para la configuración de las políticas de seguridad de OSSIM.

- Las computadoras con sistema operativo *Windows* deben estar dentro del dominio **uci.cu** y antes de efectuar una nueva instalación del sistema operativo, en los casos en que sea posible, se deben retirar del dominio.

- El nombre que identifica a las computadoras debe estar en correspondencia con la localización geográfica o la función a la que se encuentran destinadas.

- Todas las computadoras deberán tener instalado el programa antivirus Kaspersky y este debe estar actualizado constantemente.

- El sistema operativo de las computadoras debe estar correctamente actualizado con todos los parches de seguridad.

- Se prohíbe brindar sin autorización algún servicio telemático a la comunidad universitaria desde las estaciones de trabajo, estos solo se podrán configurar en los servidores destinados para estos fines.

- No se deben realizar ataques informáticos, activos o pasivos, a sistemas locales o remotos, por ejemplo: escaneo de puertos, detección de vulnerabilidades, instalación de *keyloggers*, captura de paquetes en la red, suplantación de identidad, violación de la cuenta de administración de las estaciones de trabajo, entre otros.

- No deben enviarse mensajes masivos con información intrascendente, de supuestas alertas y cartas cadenas, cuyo único propósito es el congestionamiento de las redes y la sobrecarga de los servidores.

- El grupo Domain Admins debe permanecer en el grupo de administradores locales de las estaciones de trabajo, con vistas a facilitar las tareas de soporte técnico o la realización de auditorias.

- Los usuarios deberán leer y firmar el Código de Ética de la UCI antes de solicitar los servicios telemáticos.

- ✓ No se empleará el correo electrónico, Internet u otros servicios de la red para transmitir, acceder, o difundir información pornográfica, terrorista, contrarrevolucionaria, o en general con fines lesivos a los intereses de la sociedad, la Institución, la Revolución o de terceros.

- ✓ El uso de servidores de correo electrónico en el exterior de nuestro país como Yahoo, Hotmail, Gmail, entre otros; desde la red de la UCI, debe estar autorizado por la dirección de la Universidad.

- ✓ Se realizará el monitoreo de la actividad desarrollada por los usuarios autorizados a utilizar los servicios del correo electrónico e Internet, con vistas a determinar si se cumplen con las regulaciones establecidas. Se suspenderán los servicios de aquellos que incumplan con lo establecido en el Código de Ética.

3.5 CONCLUSIONES

La instalación de OSSIM resulta complicada, pues lleva consigo implícita la configuración de varios componentes, los cuales son los pilares de su funcionamiento. Es decir, que sin estos componentes no seria posible el buen funcionamiento de la herramienta. Cada empresa con sus características específicas necesita una configuración precisa y dedicada a las necesidades de la misma. Existen dos formas diferentes de instalar OSSIM. Una es mediante un ISO la cual debe adaptarse para que sea eficiente; otra para todo tipo de redes, la cual es funcionalmente más personalizada, es a través de paquetes pre-compilados.

Para la UCI por la dimensión de sus redes se recomienda la última ya que favorece el uso de sensores en diferentes lugares de la red, aunque la otra variante también es una buena solución pero debe combinarse con la instalación de sensores que se conecten al servidor central. El uso de OSSIM es útil para toda la red de la UCI pero su mayor eficiencia se lograría al monitorizar de manera especial los servidores de la Universidad.

CONCLUSIONES

Mientras se tiene en cuenta la seguridad informática no se puede perder de vista el término inseguridad informática, este último más objetivo que el anterior. No se puede hablar de un sistema totalmente seguro debido a la existencia de este fenómeno el cual trae consigo el mejoramiento de los sistemas de seguridad.

Para el mejoramiento de la seguridad se deben instalar y perfeccionar los sistemas de forma tal que se elimine la mayor cantidad de huecos de seguridad posibles y que se reduzca la cantidad de violaciones de seguridad ocurridas. Para esto se propone la herramienta de seguridad OSSIM como solución.

En esta investigación se han expuesto las características de OSSIM que lo hacen ventajoso frente a otras herramientas de seguridad de acuerdo a los requerimientos de la red de la universidad, teniendo en cuenta que para el desarrollo de este trabajo no se cuenta con ningún tipo de financiamiento. Se han descrito el funcionamiento y los métodos que utiliza, así como las aplicaciones de seguridad integradas. Se han analizado y configurado las herramientas propuestas para que OSSIM integre en la UCI, así como su localización y alcance. Se lograron desarrollar y describir los procedimientos para una correcta instalación de OSSIM, y se mencionaron algunas de las políticas de seguridad a configurar en este SIM.

RECOMENDACIONES

Si ocurre algún fallo al iniciar el servidor de OSSIM de la versión 1.0.4: ossim-server, se debe reiniciar el sistema. Esto generalmente ocurre la primera vez que se inicia este servicio en el proceso de instalación. Una vez reiniciado el sistema después de la instalación este problema queda resuelto.

Incluir al servidor de OCS-NG Inventory de OSSIM los cambios realizados en la UCI a la herramienta OCS-NG Inventory. Estos cambios fueron desarrollados por los tesistas Annioldys Uranga González y Erik Machado Cano con la tesis correspondiente al curso 2007-2008.

Realizar un estudio sobre la herramienta OpenVPN que es una de las que integra OSSIM dentro de sus funcionalidades. Esta herramienta es muy útil pues con ella se pueden crear redes privadas virtuales y realizar balanceo de carga para los servidores.

BIBLIOGRAFÍA

Referencias:

1. **Latinos.us.** Seguridad.us. *Seguridad.us.* [En línea] [Citado el: 20 de enero de 2008.] http://www.seguridad.us.

2. **Guzmán Rosas, Ing. Oswaldo.** Seguridad de la Información. *Relaciones Comerciales.* [En línea] 2007. [Citado el: 22 de enero de 2008.] http://www.relacionescomerciales.com.mx/articulos/10.pdf.

3. **Latinos.us.** Técnicas de Aseguramiento del Sistema. *Seguridad.us.* [En línea] 21 de febrero de 2006. [Citado el: 22 de enero de 2008.] http://www.seguridad.us/privada/empresas/tecnicas-de-aseguramiento-del-sistema

4. **Blogspot.com.** Blog de Redes y Seguridad de la Información. *Blogspot.com.* [En línea] 06 de mayo de 2006. [Citado el: 06 de febrero de 2008.] http://brsi.blogspot.com/2006/05/security-information-managment.html.

5. **Ferrer, Jorge y Fernández Sanguino, Javier.** El Sofware Libre en el Mundo de la Seguridad. *ibiblio.* [En línea] 02 de septiembre de 2005. [Citado el: 08 de febrero de 2008.] www.ibiblio.org/pub/linux/docs/LuCaS/Presentaciones/200103hispalinux/ferrer/html/sw-libre.html

6. **Seguridad Digital.** OSSIM (Open Source Security Information Management). *seguridaddigital.info.* [En línea] 28 de julio de 2006. [Citado el: 07 de febrero de 2008.] http://www.seguridaddigital.info/index.php?option=content&task=view&id=108.

7. **OSSIM.** Whatis_es. *OSSIM.* [En línea] [Citado el: 05 de diciembre de 2007.] http://www.ossim.net/whatis_es.php.

Otras fuentes:

8. **Belt Iberica S.A.** Noticias Profesionales. *Belt.es.* [En línea] 05 de abril de 2005. [Citado el: 05 de diciembre de 2007.] http://www.belt.es/noticias/2005/abril/05/osimm.htm.

9. **BULMA.** Instalando y Configurando Nagios. *Bulma.net. [En línea] 05 de agosto de 2004. [Citado el: 15 de abril de 2008.] http://bulma.net/body.phtml?nIdNoticia=2075.*

10. **Nagios.** Home. *Nagios.org*. [En línea] [Citado el: 23 de abril de 2008.] http://www.nagios.org.

11. **Insecure.** Nmap. *Insecure.org.* [En línea] [Citado el: 15 de abril de 2008.] http://www.insecure.org/nmap.

12. **Ubuntu.** Guía-Ubuntu. *Ubuntu.org.* [En línea] 11 de junio de 2007. [Citado el: 15 de abril de 2008.] http://guia-ubuntu.org/index.php?title=Ntop.

13. **Intersect Alliance.** Snare. *Intersectalliance.com.* [En línea] [Citado el: 15 de abril de 2008.] http://www.intersectalliance.com/projects/SnareWindows/index.html.

14. **Zona Gratuita.** 50 Herramientas Top de Seguridad. *Zonagratuita.com.* [En línea] [Citado el: 10 de enero de 2008.] http://www.zonagratuita.com/a-cursos/utilidades/50_herramientas_top.htm.

15. **Casal, Julio.** Documentation_es:gd_es. *Ossim.net.* [En línea] 14 de junio de 2006. [Citado el: 19 de marzo de 2008.] http://www.ossim.net/dokuwiki/doku.php?id=documentation_es:gd_es.

16. **Seguridad Informática UCI.** Políticas de Seguridad Informática de la UCI. *Seguridad.uci.cu*. [En línea] 10 de febrero de 2006. [Citado el: 20 de mayo de 2008.] https://seguridad.uci.cu/mambots/editors/doc_rel/Politicas%20de%20Seguridad%20Informatica%20de%20la%20UCI.pdf.

17. **Microsoft Corporation**. Directorio Activo. *Microsoft.com. [En línea] [Citado el: 27 de mayo de 2008.]* http://www.microsoft.com/spain/technet/productos/directorioactivo/default.mspx.

18. **IT Deusto.** Portal Home. *ITDeusto.com.* [En línea] [Citado el: 6 de diciembre de 2007.] http://www.itdeusto.com/itdeusto/WebApp?Resource=IdealPortal.Home.

19. **Panamacom.** Glosario de Informática. *Panama.com.* [En línea] 23 de mayo 2008. [Citado el: 27 de mayo de 2008.] http://glosario.panamacom.com/letra-a.html.

20. **Del Pozo, Yonnis y Guerra, Lester.** *Propuesta de una infraestructura computacional basada en Tecnología Grid para la UCI.* Ciudad de la Habana: Universidad de las Ciencias Informáticas, junio de 2007. MIS-005672.

21. **Chirino, Yevgeni y Díaz, Dayrena.** *Análisis y Configuración del Sistema de Detección de Intrusos Snort en la Universidad de las Ciencias Informáticas.* Ciudad de la Habana: Universidad de las Ciencias Informáticas, junio 2007. MIS-005665.

ANEXOS

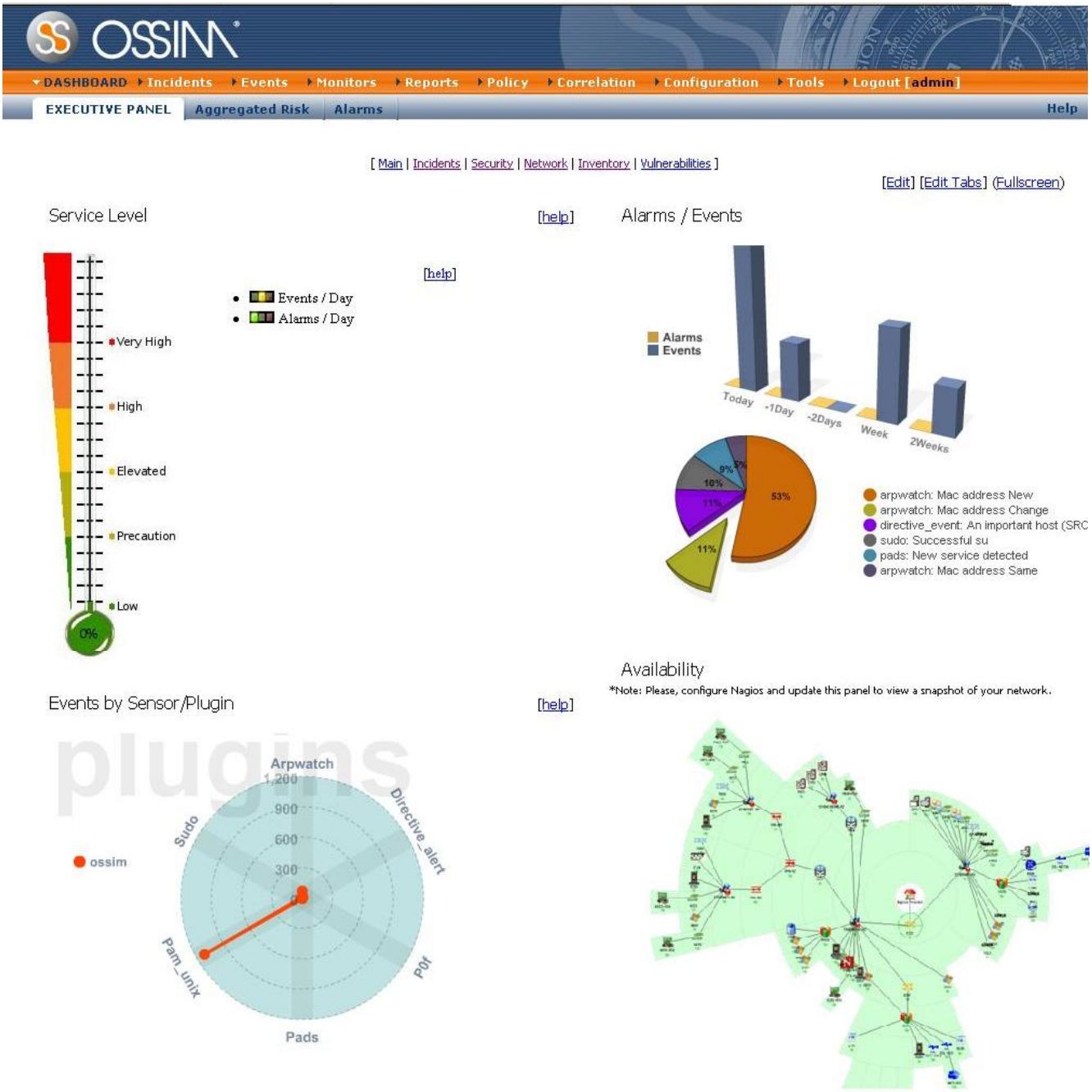


Figura # 13: Panel Principal.

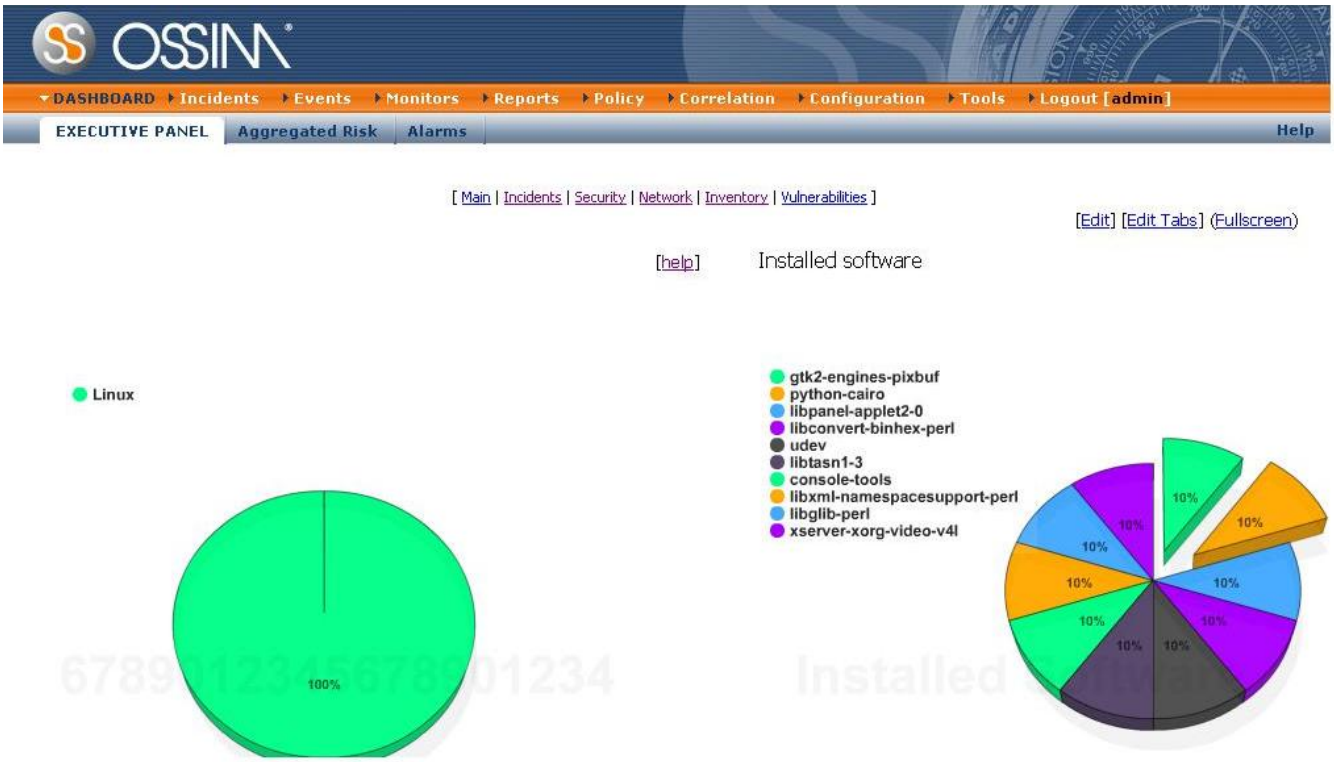


Figura # 14: Panel Principal. Inventario.

	OSSIM	ARCSIGHT	RSA	Net IQ	IBM - ISS	Symantec	LogLogic
General							
Costos de Licencia	No	Muy altos	Altos	Altos	Muy altos	Altos	Moderado
Sim/SIEM	Sí	Sí	Sí			Sí	No
Interfaz web	Sí			No (Win32)		Sí	Sí
Gestión de logs	Sí	Sí	Sí	Sí		Sí	Sí
Correlación de logs	Sí	Sí	Sí	Sí		Sí	Sí
Indencias Mng	Sí	Sí	Sí	Sí	Sí	Sí	No
Reportes de DataMart	Sí	Solo Reportes	Solo Reportes	Sí		Sí	Sí
IDS de red	Sí con Snort	No	No	No	Sí	Si con Symantec IDS	No
Detectores de Vulneravilidades	Sí con Nessus	No	No	No	Sí	Si	No
Monitoreo de redes	Sí con Ntop	No	No	No	No		No.
Deteción de Anomalías	Sí con Spade	No	No	No	Sí		No
Host IDS	Sí con Snare & Osiris	No	No	No	Sí	Sí con Symantec IDS	No
Inventario	Sí con OCS	No	No	No	No		No
Antivirus	Sí con ClamAv	No		No	No	Sí con Norton	No.
Hardware							
Aplicaciones	Sí	No	No		Sí	Sí	No

Tabla # 6: Comparación de productos.

GLOSARIO

Ancho de banda: Es la cantidad de bits que pueden viajar por un medio físico (cable coaxial, par trenzado, fibra óptica, etc.) de forma que mientras mayor sea el ancho de banda más rápido se obtendrá la información. Se mide en millones de bits por segundo (Mbps).

Aplicación de sniffing: Sniffer.

Ataques CGI: Ataques realizados mediante un CGI.

Ataque informático: Es un acto que se realiza de forma no autorizada para dañar o acceder a algún sistema informático.

Autentificación o autenticación: Es la comprobación de la identidad de una persona o de un objeto.

Base de datos (database): es un conjunto de datos pertenecientes al un mismo contexto y almacenados sistemáticamente para su posterior uso.

BIOS: Es el sistema básico de entrada/salida **Basic Input-Output System (BIOS).** Es un código de interfaz que localiza y carga el sistema operativo en la RAM; es un software muy básico instalado en la placa base que permite que ésta cumpla su cometido. Proporciona la comunicación de bajo nivel, y el funcionamiento y configuración del hardware del sistema.

Bomba lógica: Es un programa informático que se instala en un ordenador y permanece oculto hasta cumplirse una o más condiciones pre-programadas para entonces ejecutar una acción.

Buffer: Es una ubicación de la memoria en una computadora o en un instrumento digital reservada para el almacenamiento temporal de información digital, mientras que está esperando ser procesada.

Buffer overflows (desbordamiento de buffer): Es un error de software que se produce cuando se copia una cantidad de datos sobre un área que no es lo suficientemente grande para contenerlos, sobrescribiendo de esta manera otras zonas de memoria.

Canal: Es un punto de entrada a los servicios Web que lleva las peticiones y respuestas entre los servicios web y el gateway.

CD-ROM: Del inglés ***C****ompact* ***D****isc* - ***R****ead* ***O****nly* ***M****emory*, "Disco Compacto de Memoria de Sólo Lectura". Es un disco de plástico plano con información digital codificada en una espiral desde el centro hasta el borde exterior.

CGI (Common Gateway Interface): Interfaz de Acceso Común. Son programas usados para hacer llamadas a rutinas o controlar otros programas o bases de datos desde una página Web.

Chat (charla): Es un anglicismo que usualmente se refiere a una comunicación escrita a través de internet entre dos o más personas que se realiza instantáneamente.

Checksum: Valor numérico utilizado para verificar la integridad de un bloque de datos.

Cliente o agente: Es una aplicación informática que se utiliza para acceder a los servicios que ofrece un servidor, normalmente a través de una red de telecomunicaciones.

Consola (*Intérprete de comandos*): Es básicamente cualquier interfaz capaz de controlar algún dispositivo o programa. Es un programa informático que actúa como Interfaz de usuario para comunicar al usuario con el sistema operativo mediante una ventana que espera órdenes escritas por el usuario en el teclado. Por extensión también se llama *Intérprete de comandos* a algunas interfaces de programas que comunican al usuario con el *software* o al Cliente de un Servidor.

Código fuente: Es un conjunto de líneas que conforman un bloque de texto, escrito según las reglas sintácticas de algún lenguaje de programación destinado a ser legible por los seres humanos.

Comando: Es una instrucción o mandato que el usuario proporciona a un sistema informático.

Consola de Seguridad: Consola de programas dedicados a la seguridad informática.

Conexión remota: Consiste en conectarse por la red a otro ordenador como si se accediera desde el propio ordenador.

Cracker: Es alguien que viola la seguridad de un sistema informático de forma similar a como lo haría un hacker, sólo que a diferencia de este último, el cracker realiza la intrusión con fines de beneficio personal o para hacer daño.

Crackear Software: Es una actividad realizada, generalmente, por los crackers. Consiste, básicamente, en modificar el código fuente de un software para fines variados. Generalmente se realiza un **Crackeo** para permitir el uso de un software que debe ser comprado.

Criptología: La *criptología* es el estudio de los criptosistemas: sistemas que ofrecen medios seguros de comunicación en los que el emisor oculta o cifra el mensaje antes de transmitirlo para que sólo un receptor autorizado o nadie pueda descifrarlo.

Criptografía o Criptociencia: Perteneciente al griego *kryptos*, «ocultar», y *graphos*, «escribir», literalmente «escritura oculta». Es el arte o ciencia de cifrar y descifrar información utilizando técnicas que hagan posible el intercambio de mensajes de manera que sólo puedan ser leídos por las personas a quienes van dirigidos.

Datos: Información propia de una empresa o persona que en su generalidad puede incluir cualquier tipo de archivo.

Demonio o **daemon:** (de sus siglas en inglés ***Disk And Execution Monitor***): Es un tipo especial de proceso informático que se ejecuta en segundo plano en vez de ser controlado directamente por el usuario (es un proceso no interactivo).

Directorio Activo (Active Directory): es el servicio de directorio incluido en Windows 2000 Server y posterior.

Dirección IP: Es el número que identifica a cada dispositivo dentro de una red con protocolo IP.

Dispositivo tecnológico: El término dispositivo se utiliza como sinónimo de aparato. En Informática, se utiliza para referirse a los componentes del ordenador.

E-mail (correo electrónico): Es un servicio de red que permite a los usuarios enviar y recibir mensajes rápidamente, también denominados mensajes electrónicos o cartas electrónicas, mediante sistemas de comunicación electrónicos.

Enrutador (router), ruteador o encaminador: Es un dispositivo de hardware para la interconexión de la red de computadoras.

Estándar de facto: Es aquel patrón o norma que se caracteriza por no haber sido consensuada ni legitimada por un organismo de estandarización al efecto.

Ethernet: Es el nombre de una tecnología de redes de computadoras de área local (LANs) basada en tramas de datos.

Evento: Es un suceso que ocurre en un lugar y tiempo particular. Es un fenómeno observable que puede ocurrir dentro de una red de computadoras o en algún sistema informático.

Exploit: Es un programa informático malicioso, parte del programa o técnica que trata de forzar alguna deficiencia o vulnerabilidad de otro programa (llamadas bugs).

Firewall: Es una herramienta utilizada en las redes de computadoras para controlar las comunicaciones, permitiéndolas o prohibiéndolas. Tiene la capacidad de filtrar paquetes.

FTP (*File Transfer Protocol*): Es un protocolo de transferencia de archivos entre sistemas conectados a una red TCP basado en la arquitectura cliente-servidor.

Gestor de base de datos: Es un programa que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada.

GNU GPL: Es la licencia pública general de GNU o mas conocida por su nombre en inglés GNU General Public License o simplemente su acrónimo del inglés GNU GPL, es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

GTK: Es una biblioteca del equipo GTK+, la cual contiene los objetos y funciones para crear la interfaz gráfica de usuario.

GTK+ o **The GIMP Toolkit:** Es un grupo importante de bibliotecas o rutinas para desarrollar interfaces gráficas de usuario (GUI).

Gusano: es un virus informático que tiene la propiedad de duplicarse a sí mismo.

Hacker: Es el neologismo utilizado para referirse a los expertos relacionados con la informática, también para referirse a cualquier profesional que está en la cúspide de la excelencia en su profesión, ya que en la descripción más pura, un hacker es aquella persona que le apasiona el conocimiento, descubrir o aprender nuevas cosas y entender el funcionamiento de éstas.

Hardware: Dispositivo electrónico o tecnológico.

Hashing: Se refiere a la función o método para generar claves o llaves que representen de manera casi unívoca a un documento, registro o archivo, resumir o identificar un dato a través de la probabilidad, utilizando una función hash o algoritmo hash. Un **hash,** o los **hashes,** son el resultado de dicha función o algoritmo.

HIDS: s*istema de detección de intrusos en un Host*. Busca detectar anomalías que indican un riesgo potencial, revisando las actividades en la máquina (host).

Hipertexto: Es el nombre que recibe el texto que en la pantalla de una computadora conduce a su usuario a otro texto relacionado. La forma más habitual de hipertexto en documentos es la de hipervínculos o referencias cruzadas automáticas que van a otros documentos.

Host: Es todo ordenador de la red que ofrece servicios a otros ordenadores conectados a dicha red.

Hypertext Transfer Protocol (HTTP): Es un protocolo de Internet que se utiliza para recuperar objetos de hipertexto de hosts remotos.

ICMP (Protocolo de Mensajes de Control de Internet): Por sus siglas el *Internet Control Message Protocol* es el subprotocolo de control y notificación de errores del Protocolo de Internet (IP).

IDEA (International Data Encryption Algorithm): Es un algoritmo de encriptación.

IDS (Intrusion Detection System): Es un programa usado para detectar accesos no autorizados a una computadora o a una red.

Interfaz: Es el conjunto de comandos y/o métodos que permiten la intercomunicación del programa con cualquier otro programa o usuario.

IP (Internet Protocol): Protocolo de comunicación utilizado para establecer comunicaciones entre equipos de una red informática. Cada computadora de una red debe tener una dirección IP única.

IPS (Intrusion Prevention System): Es un programa usado para prevenir a los sistemas de accesos no autorizados.

ISO: Es la Organización Internacional para la Estandarización o International Organization for Standardization, que nace después de la segunda guerra mundial en 1946, es el organismo encargado de promover el desarrollo de normas internacionales de fabricación, comercio y comunicación para todas las ramas industriales a excepción de la eléctrica y la electrónica. Su función principal es la de

buscar la estandarización de normas de productos y seguridad para las empresas u organizaciones a nivel internacional.

ISO/IEC: es un estándar de International Organization for Standardization (ISO) u Organización Internacional para la Estandarización que se refiere a la seguridad de la información.

IT: Es la tecnología de información, también llamada TI, según lo definido por la asociación de la tecnología de información de América (ITAA) es el estudio, diseño, desarrollo, implementación, soporte o dirección de los sistemas de información computarizados, en particular de software de aplicación y hardware de computadoras.

IT Deusto: Es una de las principales compañías españolas independientes de consultoría y servicios, especializada en tecnologías de la información.

ITSEC/ITSEM: Information Technology Security Evaluation Criteria (ITSEC - White Book - Criterios Europeos). Es un criterio de certificación de productos de seguridad.

Kernel: Es el núcleo de un sistema operativo. Es el software responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora.

LAN: Es una red de área local, conocida por sus siglas en inglés LAN (*Local Area Network*). Es una red de ordenadores menor a una WAN.

Lenguaje de scripting: Es un lenguaje de programación que fue diseñado para ser ejecutado por medio de un intérprete, en contraste con los lenguajes compilados. También se les conoce como lenguajes de ***script***.

Licencia GPLv2: Es la versión 2 de la conocida GNU Public License (GPL).

Líneas de comunicaciones: Es el medio físico que se utiliza para conectar dos o más dispositivos con el propósito de transmitir y recibir datos.

Linux: Versión de libre distribución (gratis) del sistema operativo Unix, desarrollada inicialmente por Linus Torvalds, y mejorada gracias a las contribuciones de programadores de todo el mundo.

Lista de Control de Acceso o **ACL** (del inglés, ***Access Control List***): Es un concepto de seguridad informática usado para fomentar la separación de privilegios. Es una forma de determinar los permisos de acceso apropiados a un determinado objeto.

Log: Es un registro de actividad de un sistema, que generalmente se guarda en un fichero de texto, al que se le van añadiendo líneas a medida que se realizan acciones sobre el sistema.

Logger: Es un programa que interactúa con los registros de mensajes (logs) de un determinado sistema o componente de aplicación.

Malware: Programa maligno que puede dañar los sistemas informáticos.

Modelo OSI: Es el modelo de referencia de Interconexión de Sistemas Abiertos (OSI, Open System Interconnection) lanzado en 1984 fue el modelo de red descriptivo creado por ISO; esto es, un marco de referencia para la definición de arquitecturas de interconexión de sistemas de comunicaciones.

Monitor de Red: Es una herramienta que monitorea el tráfico de una red.

Monitor de disponibilidad: Es una herramienta que monitorea el estado de los equipos conectados en una red informática así como el estado de los servicios que brindan los mismos.

Multiplataforma: Es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.

NIDS (Network Intrusion Detection System): Sistema de detección de intrusos en una Red. Busca detectar anomalías que inicien un riesgo potencial, tales como ataques de denegación de servicio, escaneadores de puertos o intentos de entrar en un ordenador, analizando el tráfico en la red en tiempo real.

NetBIOS (*Network Basic Input/Output System*): Es una especificación de interfaz para acceso a servicios de red, es decir, una capa de software desarrollado para enlazar un sistema operativo de red con hardware específico.

Niveles de Seguridad: Los niveles de seguridad son clasificaciones que de manera ascendente y complementaria describen diferentes tipos de seguridad del Sistema Operativo y se enumeran desde el mínimo grado de seguridad al máximo. Sistema de niveles más utilizado es el de TCSEC Orange Book.

Nodo: Ordenador o conjunto de ordenadores que reciben la llamada del usuario y la dirigen hacia el servicio solicitado Lugar o área donde se encuentran establecidos los servidores de una entidad.

Open source (Código abierto o software libre): Es el término con el que se conoce al software distribuido y desarrollado libremente. Fue utilizado por primera vez en 1998 por algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (*free software*). Su máximo exponente es el sistema operativo Linux.

OSPF (Open Shortest Path First): Protocolo de enrutamiento avanzado y escalable basado en el algoritmo Link State de Dijkstra.

Outsourcing (Subcontratación): Es también llamado **tercerización** o **externalización**. Es el proceso económico en el cual una empresa determinada mueve o destina los recursos orientados a cumplir ciertas tareas, a una empresa externa, por medio de un contrato.

Paquete de red: Es cada uno de los bloques en que se divide, en el nivel de Red, la información a enviar.

Password o contraseña: Es una clave de acceso que permite un sistema sea utilizado por un usuario.

PC (Personal Computer): También denominada como **ordenador**, **computador, máquina o equipo**, es una máquina electrónica que recibe y procesa datos para convertirlos en información útil.

Ping (Packet Internet Grouper): Se trata de una utilidad que comprueba el estado de la conexión con uno o varios equipos remotos por medio de los paquetes de solicitud de eco y de respuesta de eco, definidos en el protocolo de red ICMP, para determinar si un sistema IP específico es accesible en una red.

Plataforma: Es precisamente el principio, en el cual se constituye un hardware, sobre el cual un software puede ejecutarse y desarrollarse.

Probes: Término perteneciente al idioma Inglés que significa pruebas. En Informática se utiliza para mencionar los diferentes tipos de pruebas que pueden realizarse en una red o sistema. Un ejemplo son las pruebas de SMB o SMB probes.

Protocolo de comunicación: Protocolo de red o también Protocolo de Comunicación es el conjunto de reglas que especifican el intercambio de datos u órdenes durante la comunicación entre las entidades que forman parte de una red.

Puerto: Es una forma genérica de denominar a una interfaz por la cual diferentes tipos de datos pueden ser enviados y recibidos.

RAM: Sigla de *R*andom *A*ccess *M*emory (Memoria de acceso directo). La RAM se usa para mantener los programas mientras se están ejecutando, y los datos mientras se los procesa.

Red (network): Una red de computadoras o red informática es un conjunto de equipos, computadoras y/o dispositivos conectados por medio de cables, señales, ondas o cualquier otro método de transporte de datos, que comparten información, recursos y servicios.

Registro del sistema o registro de Windows: Es una base de datos que almacena las configuraciones y opciones del sistema operativo Microsoft Windows.

RIP: Son las siglas de Routing Information Protocol (Protocolo de encaminamiento de información). Es un protocolo de puerta de enlace interna o IGP (Internal Gateway Protocol) utilizado por los routers (enrutadores), aunque también pueden actuar en equipos, para intercambiar información acerca de redes IP.

Root (raíz): Es el nombre de la cuenta de ingreso que da acceso completo y total a todos los recursos de un sistema.

RSA: Es un sistema de criptografía de clave pública. Su nombre viene de los apellidos de sus inventores: Rivest, Shamir y Adleman.

Rutina: Es el procedimiento, en este caso un conjunto de código, que es usado cada vez que se le llame.

Samba: Es una implementación libre del protocolo de archivos compartidos de Microsoft Windows (antiguamente llamado SMB, renombrado recientemente a CIFS) para sistemas de tipo UNIX. De esta forma, es posible que ordenadores con Linux y Unix en general se vean como servidores o actúen como clientes en redes de Windows.

Script: Es un tipo de programa que consiste de una serie de instrucciones que serán utilizadas por otra aplicación.

Seguridad Física: Consiste en la aplicación de barreras físicas y procedimientos de control, como medidas de prevención y contramedidas ante amenazas a los recursos e información confidencial.

Seguridad Lógica: consiste en la "aplicación de barreras y procedimientos que resguarden el acceso a los datos y sólo se permita acceder a ellos a las personas autorizadas para hacerlo.

Servidor: Es una aplicación informática o programa que realiza algunas tareas o servicios en beneficio de otras aplicaciones llamadas clientes.

Servidor proxy: Es un servidor que sirve para permitir el acceso a Internet a todos los equipos de una organización cuando sólo se puede disponer de un único equipo conectado, esto es, una única dirección IP.

Servidor web (webserver): Es un programa que implementa el *protocolo HTTP* (*hypertext transfer protocol*). Este protocolo está diseñado para transferir lo que llamamos páginas web.

Shell: Consola.

Sistema informático: es el conjunto de hardware, software y soporte humano.

Sistema Operativo NT: Es una familia de sistemas operativos de tecnología NT (Nueva Tecnología) producidos por Microsoft, de la cual la primera versión fue liberada en julio de 1993. Al principio fue diseñado para ser un poderoso sistema operativo multiusuario, basado en lenguaje de alto nivel, independiente del procesador, con rasgos comparables con Unix.

Sistema operativo (SO): Es un software de sistema, es decir, un conjunto de programas de computadora destinado a permitir una administración eficaz de sus recursos.

SMB: Son las siglas de *S*erver *M*essage *B*lock (Bloque de mensajes de servidor), SMB es el protocolo de comunicación que usan los sistemas operativos basados en MS-Windows para permitir los recursos compartidos a través de la red.

Sniffer o packet sniffer (analizador de tráfico): Es un programa de captura de las tramas de red.

Sniffing: Es el arte o acción de analizar el tráfico de una red con sniffers.

Sociedad de la información: Es una sociedad en la que la creación, distribución y manipulación de la información forman parte importante de las actividades culturales y económicas.

Software: Programa informático.

SPAM: Un **spam**, **correo basura** o **sms basura** es un mensaje no solicitado, habitualmente de tipo publicitario, enviado en grandes cantidades y de forma masiva , que perjudican de alguna o varias maneras al receptor.

Spammers: Se puede definir a un *spammer* como aquella persona o programa que roba o compra direcciones de correo electrónico y realiza el envío de correos no solicitados a estas direcciones.

Spywares o Programas Espía: Son aplicaciones que se dedican a recopilar información del sistema en el que se encuentran instaladas para luego enviarla a través de Internet, generalmente a alguna empresa de publicidad.

Stealth Port Scans: Término perteneciente al idioma Inglés utilizado para describir los escaneos indetectables de puertos.

Subredes: En redes de computadoras, una subred es un rango de direcciones lógicas. Cuando una red de computadoras se vuelve muy grande, conviene dividirla en subredes.

Syslog: Es un estándar de facto para el envío de mensajes de registro en una red informática IP. Por *syslog* se conoce tanto al protocolo de red como a la aplicación o biblioteca que envía los mensajes de registro.

TCP/IP: Familia de protocolos de Internet es un conjunto de protocolos de red en la que se basa Internet y que permiten la transmisión de datos entre redes de computadoras. TCP significa Protocolo de Control de Transmisión.

TCP Wrapper ("*Envolvedor de TCP*"): Es un sistema de red ACL que trabaja en terminales y que se usa para filtrar el acceso de red a servicios de protocolos de Internet que corren en sistemas operativos UNIX, como Linux.

TCSEC Orange Book: Trusted Computer Security Evaluation Criteria (TCSEC - Orange Book - Criterios Norteamericanos). Es un criterio de certificación de productos de seguridad creado en 1985 que propone 7 niveles de seguridad informática: D (Sin Protección), C1 (Protección Discrecional), C2 (Protección de Acceso Controlado), B1 (Seguridad Etiquetada), B2 (Protección Estructurada), B3 (Dominios de Seguridad) y A (Protección Verificada).

Tecnología NT (Nueva Tecnología): Es la tecnología que fue desarrollada por Microsoft para superar los obstáculos impuestos por la vieja arquitectura de sus sistemas operativos.

Telnet (TELecommunication NETwork): Es el nombre de un protocolo de red, y del programa informático que implementa el cliente, que sirve para acceder mediante una red a otra máquina, para manejarla como si se estuviera sentado delante de ella.

TIC: Tecnologías de la Información y de las Comunicaciones.

TI: Tecnologías de la Información.

Topología: Es una disciplina Matemática que estudia las propiedades de los espacios topológicos y las funciones continuas.

Traceroute: Es una herramienta de diagnóstico de redes que permite seguir la pista de los paquetes que van desde un host (punto de red) a otro.

Traceroute (comando): Es un comando utilizado en los sistemas Linux y Unix para seguir la ruta de un paquete desde un host a otro.

Tracert (comando): Es un comando utilizado en los sistemas Windows para seguir la ruta de un paquete desde un host a otro.

Tráfico: Es la cantidad de datos enviados y recibidos en una red de computadoras.

Throughput: Es el rendimiento final de una conexión. Volumen de datos que una conexión brinda como resultante de la suma de su capacidad y la resta de los overheads que reducen su rendimiento.

Transferencia Real. Cantidad de datos que son transmitidos a algún punto de la red.

UDP: Son las siglas de Protocolo de Datagrama de Usuario (en inglés *User Datagram Protocol*) un protocolo no orientado a la conexión que, como TCP, funciona en redes IP.

Unix: Sistema operativo creado por los Laboratorios Bell en 1969, se considera a Ken Thompson y a Dennis Ritchie como sus creadores, fue el primer SO escrito en C. Es un sistema multiusuario y multitarea.

Virus: Programa o código malicioso que puede dañar los sistemas informáticos. Es capaz de reproducirse por si solo.

VPN (Virtual Private Network) o Red Privada Virtual (**RPV**): Es una tecnología de red que permite una extensión de la red local sobre una red pública o no controlada, como por ejemplo Internet.

WAN (Wide Area Network): Es una red de área amplia. Es una red de computadoras de gran tamaño, dispersa por un país o incluso por todo el planeta.

Windows: Sistema operativo de Microsoft basado en el uso de ventanas virtuales para las distintas aplicaciones o documentos.

www (World Wide Web): Es el sistema de información distribuido, que utiliza el protocolo HTTP para enlazar páginas mediante mecanismos de hipertexto. Posiblemente sea el servicio más conocido de Internet.