Universidad del Azuay

Facultad de Ciencias de la Administración

Escuela de Ingeniería de Sistemas

Sistema de Detección de Intrusos para un ISP

Trabajo de graduación previo a la obtención del título de

Ingeniero de Sistemas

Autores: **Augusto Cabrera Duffaut**

Esteban Fajardo Moscoso

Director: **Ing. Fabián Carvajal**

Cuenca, Ecuador

2006

DEDICATORIA

Queremos dedicar nuestro esfuerzo a lo largo de la carrera en primera instancia a DIOS porque en El encontramos la fuerza para seguir adelante y no rendirnos ante las adversidades que surgieron en el camino y de una manera muy especial a las personas que siempre estuvieron allí dándonos su apoyo y comprensión ya que a través de su consejo pudimos alcanzar las metas propuestas.

Gracias Padres por creer en nosotros.

AGRADECIMIENTO

Nuestro agradecimiento va dedicado a todas las personas que estuvieron involucradas en el proceso de nuestra formación como son nuestros profesores, amigos, compañeros de trabajo, familiares, y de manera muy sincera a la empresa ETAPATELECOM S.A. que nos abrió sus puertas para poner en practica nuestros conocimientos y desarrollarnos como profesionales adquiriendo la experiencia necesaria que nos lleva a ser mejores cada día.

INDICE DE CONTENIDOS

RESUMEN

Un IDS o Sistema de Detección de Intrusiones para un ISP, es una herramienta de seguridad que monitorea el tráfico de la red. A diferencia de un firewall el objetivo no es el de bloquear las intrusiones sino detectarlas mediante reglas provocando alertas que avisan al administrador de una intrusión o alguna anomalía en el trafico de la red del ISP, que al ser un proveedor de servicio se ve expuesto a ataques desde el Internet.

Con la elaboración de esta monografía se pretende configurar un IDS para que sea parte de las seguridades en el ISP de ETAPATELECOM S.A., elaborando para ello reglas especiales para este propósito. El IDS se implementará en sistema operativo Linux Fedora Core 3 utilizando la aplicación snort y en base de datos MYSQL donde guardaremos los registros de las intrusiones, logrando de esta manera mantener una bitácora de alertas que nos permitan sacar estadísticas o reportes del tipo de intrusiones que se detectan tomando así las precauciones del caso y evitando futuros ataques, ya que, para poder protegernos primero debemos conocer a nuestro atacante o nuestras vulnerabilidades.

Cabe anotar que un sistema de detección de intrusos surge como una medida preventiva, nunca como una medida para asegurar nuestros sistemas, ya que, para ello existen herramientas como Iptables o Firewalls que tienen como objetivo realizar esta tarea, un IDS es un complemento de seguridad muy necesario para proteger nuestros sistemas y no estar a ciegas del tráfico en nuestra red.

ABSTRACT

An IDS or Intruder Detection System for an ISP is a tool used for security purposes that monitors the traffic-flow of information on the net. Unlike the firewall, its objective is not to block intrusions but to detect them by set rules provoking alerts that notify the user of an intrusion or of some abnormality with the traffic-flow of information on the net of the ISP, which being a service provider it is exposed to attacks from the net.

The purpose of elaborating this monograph is to make an IDS that will be part of the security system on the ISP of ETAPATELECOM S.A. and to achieve this, special set rules must be made for it. The IDS will be installed on the operating system Linux Fedora Core 3 using the program Snort and with data base mysql where the intrusions will be saved and registered, achieving by this a record alert of what is being done which will enable us to have stadistics or reports of the type of intrusions that are detected, helping us take the necessary precautions and avoid future attacks because to protect ourselves we must know our attackers and our weaknesses first.

Its worthwhile to mention that a system that detects intruders comes as a preventive measurement and not as a measurement to assure our systems because for this, programs such as iptables or firewalls exist which have as objectives to carry out this task. An IDS is additional security very necessary to protect our systems and not to be unaware of the traffic-flow of information in our net.

INTRODUCCIÓN

Día a día estamos viendo que el número de usuarios que acceden a Internet desde sus hogares, trabajos, o centros de formación (bien sean Escuelas, Institutos o Universidades), está experimentando un gran aumento. Esto se debe al gran avance que se ha producido en los últimos 12 años en cuanto a las tecnologías de transmisión de datos, y en el abaratamiento de las mismas.

Así mismo se aprecia que desde los medios de comunicación, se está dando un gran apoyo a estas tecnologías.

Este aumento del uso de Internet no acaba aquí, sino también gran cantidad de empresas están abriendo sus páginas en la Red, puesto que han encontrado en ella un modo económico y rápido de hacer llegar información sobre sus productos a los usuarios.

No solo el aumento del uso de la Red de Redes se da en lo referente a usuarios domésticos o en lo que atañe a publicidad. La introducción de las tecnologías de la información en las empresas está provocando que también sean cada vez más las empresas que comienzan a abrir sus redes internas para que sus empleados puedan trabajar desde sus propias casas, dándoles acceso a través de Internet a sus aplicaciones y a sus datos confidenciales.

Todo esto conlleva a que las posibilidades de intrusiones en nuestros sistemas sean más altas cada día teniendo que proteger las vulnerabilidades de nuestras redes con sistemas de seguridad mas seguros, uno de los elementos de un sistema de seguridad es lo que se conoce como IDS (Sistema de Detección de Intrusos) el IDS monitorea todo el trafico de la red generando alarmas que advierten al administrador de una intrusión esto es una gran ayuda a la hora de ser victima de algún intento de infiltración en nuestros sistemas.

PARTE I:

INTRUSIONES Y SISTEMA DE DETECCCIÓN

CAPITULO 1

INTRUSIONES

Introducción.-

Desde los albores del tiempo, hemos visto que las seguridades se incrementan en vista de que cada vez las amenazas crecen, en el mundo digital esto aumenta considerablemente cada día tomando en cuenta que los hackers o intrusos debido a su propia naturaleza se reinventan continuamente a si mismos y a sus técnicas, convirtiéndose con frecuencia en fantasmas del ciberespacio casi imposibles de perseguir, es por este motivo que la mejor manera de combatirlos es conociéndolos mas a fondo y teniendo claro quienes son y que tipos de técnicas utilizan para realizar sus ataques a nuestros sistemas y así estar alertas para contrarrestar su efectividad.

1.1. Qué es una intrusión.

Definimos intrusión como cualquier intento de comprometer la confidencialidad, integridad, disponibilidad o evitar los mecanismos de seguridad de una computadora o red. Las intrusiones se pueden producir de varias formas: atacantes que acceden a los sistemas desde Internet, usuarios autorizados del sistema que intentan ganar privilegios adicionales para los cuales no están autorizados y usuarios autorizados que hacen un mal uso de los privilegios que se les han asignado.

1.2. Tipos de intrusiones.

En si una intrusión es atacar una vulnerabilidad de seguridad ingresando a los sistemas, los intrusos pueden originarse dentro de los sistemas o ser personas que conocen las herramientas de seguridad implementadas logrando evitarlas con mayor facilidad. Se pueden citar varios tipos de intrusiones como:

- “Intentos de entrada: Una persona ajena a nuestro sistema intenta acceder de forma no autorizada al mismo. Se detectan normalmente por modelos de comportamiento atípicos, o violaciones de las restricciones dadas por la política de seguridad.

- Ataque enmascarado: A partir de un usuario del sistema se intenta un ataque al mismo, es detectado por modelos de comportamiento atípico o violaciones de constraints de seguridad.
- Penetraciones en el sistema se control: Que son normalmente detectadas a partir de la observación de modelos especiales de actividad.
- Fuga: Cuando se utilizan de manera excesiva los recursos de un sistema. Se detectan normalmente por usos anormales de los recursos de E/S.
- Rechazo de servicio: Detectados por uso atípico de los recursos del sistema.
- Uso malicioso: Detectado normalmente por modelos de comportamiento atípico, violaciones de las restricciones de seguridad, o uso de privilegios especiales."[1]

1.3. Quién es un intruso y tipos.

En el caso de ataques semidirigidos o dirigidos vamos a tener un intruso que desea hacerse con el control de nuestro ordenador con fines diversos. Los intrusos suelen clasificarse según su intencionalidad en:

- Curioso: Solamente busca curiosear en la información o datos personales que hay en el Sistema Informático. Generalmente no causa daños de consideración.
- Búsqueda de renombre: Intruso que busca prestigio dentro de su comunidad y que busca entrar en sistemas "difíciles" o con cierto renombre. Los daños que pueda causar se derivan de la prueba que decida colocar para demostrar que ha entrado en el Sistema.
- Ocupas: Entran en el sistema para aprovechar su capacidad de cálculo o instalar algún servidor web o ftp para intercambiar programas dentro de su comunicad. Resultan molestos, pero no buscan causar daños en el sistema.
- De paso: Utilizan el sistema en el que entran como puente para acceder al sistema que realmente constituye su objetivo. No suelen causar daños de consideración, de hecho procuran permanecer ocultos.

[1] http://penta.ufrgs.br/gereseg/node50.htm 18/01/06

- Malicioso: Su objetivo es introducirse en el sistema y causar el mayor daño posible. Suelen moverse por motivos personales y se centran en la destrucción o alteración de la información, así como desestabilización del sistema operativo.
- Competencia: Se trata de intrusos provenientes de la competencia directa en el mercado de nuestra empresa y tienen como objetivo robar secretos industriales, o producir sabotajes que empeoren la capacidad competitiva.

El que estos intrusos puedan acceder a nuestro sistema depende del nivel de seguridad que tengamos implantado y de su nivel de cualificación. Según su nivel de cualificación los intrusos pueden clasificarse en:

- Aficionados: Son aficionados a los ordenadores, hábiles en la navegación por internet y que, sin tener demasiados conocimientos de informática, se hacen con “exploits” que circulan en las páginas de “hackers” y “crackers”. Son los más numerosos, pero sus limitaciones en informática les impiden ir más allá y no suelen pasar de curiosear. Su acción puede evitarse manteniendo actualizado el sistema operativo e instalando todos los parches de seguridad.
- Intrusos con conocimientos medios de informática: Estos intrusos tienen ciertos conocimientos de informática, y saben con cierta exactitud que es lo que están haciendo. Pueden modificar los exploits existentes y tiene capacidad no solo para entrar sino para compilar programas en la máquina atacada, hacerse con el control de la misma y comprometer por completo la seguridad y la información almacenada. Es imperativo mantener el sistema y los parches de seguridad completamente actualizados, utilizar firewalls y realizar cierto seguimiento del estado del sistema y de los ficheros de registro para mantener unos niveles aceptables de seguridad frente a estos intrusos.
- Crackers (no Hackers): Los crackers son informáticos avanzados, con amplios conocimientos y que son capaces de crear “exploits” a medida para el sistema en el que desean infiltrarse. Es casi imposible estar completamente inmunizados ante ellos, pero pueden ponérseles las cosas difíciles si se establece una política minuciosa de seguridad, realizando una configuración “a medida” del sistema operativo y servicios, utilizando firewalls, y por supuesto manteniendo una actualización rigurosa del sistema y aplicaciones

críticas. Debe asimismo realizarse una vigilancia exhaustiva de la red con IDS. Son poco numerosos, y si nuestro sistema no contiene información muy importante es raro que seamos víctima de un ataque tan sofisticado, pero nunca se está a salvo de los motivos personales y las amistades casuales.

1.4. Cómo intentan entrar los intrusos en los sistemas.

Un intruso suele seguir unos pasos para entrar en el sistema.

Primero recopila información general de fallos de seguridad (bugs) y de mensajes oficiales que muestran los pasos que hay que dar para aprovechar un determinado fallo de seguridad, incluyendo los programas necesarios (exploits)

Dichos fallos se aprovechan para conseguir introducirse en el sistema y están basados casi siempre en los protocolos TCP/IP, en servicios de red como NFS o NIS, o en los comandos remotos UNIX.

Los protocolos basados en TCP/IP que se suelen aprovechar son TELNET, FTP, TFTP, SMTP, HTTP, etc. Cada uno de ellos tiene sus propios agujeros de seguridad que se van parcheando con nuevas versiones, aunque siempre aparecen nuevos bugs.

Toda esa información está en Internet y sólo es necesario saber buscarla. Por lo tanto, el proceso de hacking sigue las siguientes etapas:

- Obtención de la información del equipo a atacar.
- Entrada en el equipo.
- Obtención de la cuenta de root.
- Mantener los privilegios de root.
- Borrar las huellas.

Generalmente la información que se recopila del equipo a atacar será:

El tipo de sistema operativo a atacar.

La versión de Sendmail usada, información que se consigue tecleando telnet <equipo> 25. El número 25 es el número de puerto que utiliza normalmente dicho demonio. Una vez conectados para salir, basta utilizar QUIT o, para la obtención de ayuda, HELP. Para evitar esto, basta configurar el enrutador de manera que todas las

conexiones procedentes de fuera pasen a un equipo central y que sea desde éste desde donde se distribuya el correo internamente.

Qué servicios RPC tiene, para lo que basta con escribir rpcinfo –p <equipo>.

Información de todo el dominio, es decir, de los equipos que lo integran. Normalmente se usa WHOIS para descubrir cual es el dominio.

Login de los usuarios que tienen acceso al equipo. Muchas veces esto se obtiene a través del servicio FINGER si el host atacado tiene este servicio disponible. Otra manera es encontrar direcciones de correo electrónico que apunten a esa máquina o usar mecanismos de ingeniería social.

En cuanto a la penetración en el sistema podemos diferenciar dos formas básicas de introducirse:

Entrar directamente, sin necesidad de poseer una cuenta en el sistema. Una opción es hacerlo como se detallaba al principio, con los comandos remotos.

Conseguir el fichero de contraseñas del equipo y crackearlo. Para crackearlo existen varios programas, tanto para UNIX como para Windows.

Una vez introducidos en el equipo, los hackers intentarán obtener privilegios de root y para ello explotarán los bugs encontrados para el sistema en el primer paso. Lo que también hacen es intentar explotar bugs que afecten a sistemas UNIX en general. Si siguen sin funcionar, explotarán el sistema (hasta donde le permitan sus privilegios) para tener una visión general de cómo está protegido, por ejemplo, viendo si los usuarios tienen ficheros .rhosts, si determinados ficheros tienen permisos SUID qué usuario tiene determinados ficheros, etc. Y a partir de ahí existirán dos opciones principalmente: la primera es que se olviden durante unos días del equipo para poder recopilar más información sobre bugs actualizados y la segunda es la de hackear otra máquina del mismo dominio, que sea algo más insegura.

Una vez hackeado el equipo inseguro, colocarán un sniffer para conseguir una cuenta para el otro equipo.

Un sniffer no es más que un programa que captura todo lo que pasa por la red, poniendo al equipo en modo promiscuo. La obtención de un sniffer es tan sencilla como navegar por Internet, pero incluso programas como Etherfind, Tcpdump o Ethereal pueden ser usados para este fin, aunque no hayan sido concebidos para ello.

La manera de comprobar si un sistema está en modo promiscuo es tecleando ifconfig -a. Una manera de evitar los sniffers es mediante switches en la redes de acceso general del resto de la red.

Una vez que los intrusos consiguen privilegios de root deben conseguir mantenerlos. Existen diversas formas de conseguirlo, es decir, asegurar que la próxima vez que los hackers entren en el sistema con la cuenta de un usuario que posee privilegios normales, puedan conseguir los privilegios de root de forma más fácil y sin complicaciones. Para ello, la forma más empleada es el sushi (set-uidshell), más conocida como huevo. El sushi consiste en copiar un shell a un directorio público, en el que un usuario normal pueda ejecutar los ficheros, y cambiar el nombre al que ellos quieran. Hay que asegurarse de que el shell copiado tenga como propietario al root y, posteriormente cambiar los permisos del fichero con las cifras 4755. El 4 significa que cualquier usuario que ejecute dicho fichero lo estará ejecutando con los privilegios del propietario. Como en este caso el propietario es root y el fichero en cuestión es un shell, el sistema les abrirá a los hackers un shell con privilegios de root. Con esta operación, la próxima vez que accedan al sistema con la cuenta de un usuario normal, sólo tendrán que ejecutar el shell antes mencionado y se convertirán en root.

Por último, un intruso con unos conocimientos mínimos siempre intentará eliminar sus huellas. El sistema operativo guarda varios registros de las conexiones de los usuarios al equipo, por lo que los hackers intentarán eliminarlos. Existen varios modos de borrar sus huellas en estos ficheros. La primera es que, como la mayoría no son ficheros de texto, no podrán editarlo con un editor de texto, pero si existen programas conocidos con el nombre de zappers (los más habituales son los siguientes: marry.c, zap.c, zap2.c, remove.c, cloak.c, ...), que pueden borrar los datos relativos a un usuario en particular dejando el resto de la información intacta. La segunda manera es mucho más radical, que consiste en dejar el fichero con cero bytes o incluso borrarlo. Esta manera sólo se utiliza como último recurso, ya que suscita muchas sospechas por parte de los administradores.

1.5. Estadísticas de intrusiones.

Desde que se establecieron las redes y su crecimiento se han detectado problemas con respecto a las intrusiones presentada, para demostrar ello el Centro de Coordinación CERT de la Universidad Carnegie Mellon, presenta datos estadísticos del número de intrusiones detectadas y reportadas al www.CERT.org desde 1988 hasta el 2003, las cuales nos indican claramente el incremento significativo a lo largo de los años (ver Anexo 1), teniendo en cuenta este tipo de información vemos la necesidad de cuidar los aspectos de seguridad informática de nuestros sistemas.

1.6. Conclusiones.

Los ataques de intrusiones generados desde el Internet o desde el interior de las redes nos hacen pensar que los sistemas de seguridad son un factor decisivo para el bienestar de una empresa ya que el resguardo de la información es primordial, razón por la cual se deben tomar las medidas necesarias para proteger las vulnerabilidades de nuestros sistemas ya que cada vez los atacantes desarrollan técnicas nuevas y más sofisticadas para lograr su objetivo, esto nos lleva a pensar que en cualquier momento podríamos estar dentro de las estadísticas de sistemas que han sido victimas de ataques e intrusiones, para evitar aquello no debemos descuidar las seguridades ya que es la única forma de mantener nuestras redes saludables y libres de riesgos innecesarios.

CAPITULO 2

DESCUBRIENDO AL INTRUSO

Introducción.-

Ante la sospecha de que nuestro sistema haya sido objeto de un ataque, se ha de determinar lo siguiente:

- Si realmente el sistema ha sido atacado.
- Si el ataque ha tenido éxito
- En qué grado se ha comprometido nuestro sistema en caso de que haya sido atacado.

La tarea de detectar posibles intrusos será más o menos fácil en función del sistema operativo del que dispongamos puesto que algunos sistemas operativos modernos son complejos y poseen numerosos “sitios” en los cuales los intrusos pueden ocultar sus actividades. La mayor parte de los intrusos dejan señales de sus actividades en el sistema.

2.1. Detección en sistemas UNIX/Linux.

En general puede ser conveniente espiar un poco al intruso para obtener más pruebas y después desconectar el interfaz de red si es posible. Si no fuera posible desconectar el interfaz, deberíamos usar algún filtro para las conexiones procedentes de la dirección del atacante. Programas como ipchains (o ipfwadm en su caso) pueden realizar esta labor. Si desconectamos el interfaz o denegamos (no rechazar) los paquetes procedentes de esa dirección el intruso lo podría interpretar como un error de red, más que una detección del ataque. Si no se pudiera limitar el acceso a las direcciones que usa el intruso, intente cerrar la cuenta del usuario. Observe que cerrar una cuenta no es una cosa simple. Tiene que tener en cuenta los ficheros *.rhosts*, el acceso FTP y otras posibles puertas traseras.

En general no es aconsejable apagar el sistema. Por supuesto, nunca apagarlo en caliente; esto podría hacernos perder la información que tenemos en memoria. En Linux podemos ver la lista de procesos que hay en ejecución y matar aquellos que puedan estar dañando al sistema.

Somos el destino del ataque o somos un punto intermedio.

Se puede dar la situación que nuestra máquina no sea el destino final del ataque. Puede que el intruso la haya utilizado como punto intermedio para atacar a otros sistemas e intentar dificultar el seguimiento de las pistas. En este caso, además de limitar las acciones del atacante deberíamos notificarlo al administrador del destino del ataque y conservar todas las pruebas existentes por si se pudieran reclamar judicialmente.

Es habitual que durante los próximos minutos el atacante vuelva a intentar continuar con sus acciones, tal vez usando una cuenta diferente y/o una dirección de red distinta.

Si cree que ha sido objeto de un ataque que no está documentado, debería notificarlo a alguna organización de seguridad como CERT o similar para que se pueda solucionar lo antes posible y evitar que otros sistemas lo puedan padecer.

2.2. Cómo saber si hay un intruso “actualmente” en el sistema.

Cuando sospechamos que un intruso puede que se encuentre actualmente en el sistema debemos realizar dos pasos fundamentales:
Comprobar si los usuarios que se encuentran actualmente en el sistema son sospechosos.
Comprobar que procesos se están ejecutando y quién los ejecuta.
Las sospechas de que un intruso se encuentra en nuestro sistema pueden venir fundamentadas porque en el intento de comprobar si dicho intruso ha atacado el sistema nos damos cuenta, por ejemplo en las fechas de los log’s o en las fechas de procesos (o ficheros), que existe una gran posibilidad que se encuentre en él en ese mismo instante. Por ello a continuación se va a explicar los dos pasos fundamentales comentados anteriormente.

Si creemos que hay intrusos en nuestro sistema, lo primero a determinar es dónde están y qué están haciendo. Existen diversos comandos que permiten conocer los usuarios que están actualmente en el sistema:

- Comando Finger

El comando finger despliega información acerca de los usuarios que están trabajando en algún sistema.

El formato de este comando es el siguiente:

finger [nombre del sistema o login del usuario]

Por default, el comando finger despliega información acerca de los usuarios que se encuentran conectados al mismo sistema que nosotros. La información que se incluye de cada usuario es: login, fullname, nombre de la terminal, tiempo que lleva el usuario sin trabajar en el sistema (idle time), localización, y nombre de los host desde los cuales se conectan los usuarios (si es que el sistema los reconoce).

El idle time se mide en minutos enteros, es decir, no muestra fracciones menores a un minuto. El tiempo se mide en horas y minutos si es que aparece el caracter ':', también se puede medir en días y horas, esto ocurre cuando el caracter 'd' está presente.

Cuando uno o más argumentos fueron dados a la hora de ejecutar el comando, se despliega información más detallada sobre el usuario que le dimos como parámetro, esto ocurre aún si la persona no está conectada al sistema en ese momento. La información que le dimos como parámetro puede ser el apellido o el nombre del usuario.

La información aparece desplegada en un formato de multilínea, que incluye información como la siguiente:

Localización del directorio de trabajo del usuario y el shell que dicho usuario tiene.

Tiempo que lleva el usuario conectado al sistema, si es que dicho usuario está trabajando aún, sino, despliega la fecha y la hora en que ese usuario entró al sistema y desde qué máquina estableció la conexión.

La última fecha en que dicho usuario recibió y leyó correo electrónico.

Despliega el plan del usuario. Es decir, el contenido del archivo .plan que se encuentra en el directorio de trabajo de dicho usuario.

Describe los proyectos que el usuario tiene pendientes, esta información se encuentra guardada en el archivo .project de cada usuario.

Si el nombre que se da como argumento a este comando comienza con '@', entonces dicho nombre no se tomará como el login-name de un usuario, sino como el nombre de alguna máquina y el comando hará un finger remoto a dicho sistema.

Opciones.-

-l

Forza a que la información aparezca en un formato largo.

Ejemplos.-

Queremos conocer la lista de usuarios que se encuentran trabajando en la máquina goya: (ver Anexo 2).

goya% finger

Queremos conocer la lista de usuarios que se encuentran trabajando en la máquina mexplaza: (ver Anexo 3).

goya% finger @mexplaza.staff.udg.mx

- Comando who

El comando who despliega información acerca de los usuarios que están trabajando en algún sistema.

El formato de este comando es el siguiente:

who [am i]

Si este comando lo utilizamos sin argumentos, nos muestra una lista de los usuarios que están trabajando en el sistema, despliega el login, el nombre de la terminal, y el tiempo que lleva cada usuario conectado al sistema.

Si escribimos completo el comando con su argumento, es decir: "who am i", el comando indica el login-name con el que estamos trabajando. Y nos despliega el nombre del host, el nombre de la terminal y el tiempo que llevamos conectados al sistema.

Ejemplo.-

Queremos la lista de todos los usuarios que se encuentran en la máquina mexplaza: (ver Anexo 4).

goya% who

- Comando w

El comando w despliega información acerca de los usuarios que están trabajando en algún sistema y qué es lo que cada uno de ellos está haciendo.

El formato de este comando es el siguiente:

w [user]

El comando w despliega un resumen de las actividades que actualmente está desarrollando el sistema. Incluye qué es lo que cada uno de los usuarios está haciendo. La línea de encabezado muestra la hora y la fecha actual, desde cuando el sistema está arriba, el número de usuarios que se encuentran conectados, el promedio de trabajos que ha realizado el servidor hace 1, 5 y 15 minutos.

Los campos de información que se desplegan son: el login de cada uno de los usuarios, el nombre del tty o terminal que se está utilizando, la hora y fecha en que cada usuario se conectó al sistema, el tiempo que el usuario lleva sin teclear nada, el tiempo de CPU utilizado por todos los procesos (procesos padres e hijos) en una terminal, el tiempo de CPU que está utilizando actualmente el proceso activo, y los nombres y argumentos de los procesos de los actuales.

Si el nombre del usuario es incluído, la información de salida se restringe únicamente a dicho usuario.

Ejemplo.-

Queremos saber qué es lo que están haciendo los usuarios de la máquina unicornio en estos momentos: (ver Anexo 5).

unicornio% w -s

- Comando users

El comando users despliega una lista de los login de todos los usuarios que se encuentran conectados al sistema en ese momento, de una manera compacta, y en una sola línea.

Ejemplo.-

Queremos conocer login de los usuarios de la máquina unicornio que están en estos momentos. (ver Anexo 6).

unicornio% users

- Comando rusers

El comando rusers permite saber quién está dentro de toda la red.

El formato de este comando es el siguiente:

rusers [host...]

El comando rusers produce una salida similar a la que produce el comando users y el comando who, pero únicamente para máquinas remotas. Se hace una llamada broadcast a la red y se presenta la respuesta que hubo de cada una de las máquinas. Normalmente, la información se muestra conforme se va recibiendo, pero el orden se puede cambiar, de acuerdo a las opciones que le demos como argumentos.

Por default se imprimen una lista al estilo de la que produce el comando users, pero tiene una línea por máquina.

Ejemplo.-

Queremos conocer los usuarios que están conectados a la máquina unicornio: (ver Anexo 7).

unicornio% rusers

2.3. Cómo detectar que “ya ha ocurrido” una intrusión.

La utilización de los comandos y consejos a los que se hace referencia a continuación es aconsejable ante la sospecha de que un intruso haya estado en nuestro sistema pero que sabemos que ya lo ha abandonado.

Ante dicha sospecha debemos buscar una serie de señales que nos permitan encontrar huellas de que el intruso haya dejado tras de sí en el sistema. Estas señales se pueden enumerar en una serie de pasos como:

- Examinar los archivos log.
- Buscar archivos setuid y setgid.
- Chequear los archivos binarios del sistema.
- Comprobar puertos abiertos.
- Chequear si hay sniffers.
- Examinar archivos que estén ejecutándose como 'cron' y 'at'.

- Chequear si hay servicios no autorizados.
- Examinar el archivo /etc/passwd.
- Chequear la configuración del sistema y la red.
- Buscar todos lados archivos escondidos o inusuales.
- Examinar todas las máquinas en la red local.

2.3.1. Examinar los archivos log.

Lo primero que se debe de hacer siempre que se tenga la sospecha de que el sistema ha sido atacado (y lo más importante) es examinar los archivos log a conexiones de lugares inusuales u otra actividad inusual.

Por ejemplo, se debe buscar el último acceso al sistema de un usuario, el conteo de procesos, todos los accesos generados por syslog y otros accesos de seguridad. Hay que tener en cuenta que esto no es infalible ya que muchos intrusos modifican los archivos log para esconder su actividad.

A continuación se hará un listado de los principales log's que se deben revisar, de herramientas que muestran algunos logs e incluso de cómo un intruso podría modificarlos para borrar sus huellas:

- xferlog

Si el sistema comprometido tiene servicio FTP, este fichero contiene el loggeo de todos los procesos del FTP y su localización suele ser el directorio /var/adm/. Podemos examinar que tipo de herramientas a subido el intruso y que ficheros ha bajado de nuestro servidor. Suele ser bastante interesante revisar este log ya que un intruso puede usar carpetas ocultas del directorio del FTP para guardar la información y aplicaciones que necesite para atacar el sistema.

La información que almacena este log suele ser la siguiente:

- La hora y la fecha a la que se transfiere.
- Nombre del host remoto que inicia la transferencia.
- Tamaño de fichero transferido.
- Nombre del fichero transferido.

- Modo en que el archivo fue transferido (ASCII o binary).
- Flags especiales (C para comprimidos, U para descomprimidos, T para un archivo tar).
- Dirección de transferencia.
- El tipo de usuario que entró en el servicio (a para un usuario anónimo, para un invitado y r para un usuario local).

- secure

Algunos sistemas Unix loggean mensajes al fichero secure, ya que utilizan algún software de seguridad para ello, como el TCP Wrapper.

En todo momento una conexión establecida con uno de los servicios que se están ejecutando bajo inetd (ahora, xinetd) y que usan TCP Wrappers, un mensaje de logeo es añadido a al fichero "secure" que se suele encontrar en "/var/secure". Cuando examinemos el fichero log, debemos buscar anomalías tales como servicios a los que se accedió por un método no habitual y desde host desconocidos.

- wtmp

Guarda un log cada vez que un usuario se introduce en el equipo, sale de él o la máquina resetea. Dicho fichero se ubica normalmente en /etc/wtmp, /var/log/wtmp ó /var/adm/wtmp y contiene la información en formato usuario con la hora de conexión, IP origen del usuario, ... por lo que podemos averiguar de donde provino el intruso.

- lastlog

En él se encuentra el momento exacto en que entró el usuario en el equipo por última vez. En algunas versiones de Unix también almacena el último acceso fallido en la cuenta de un usuario. Se ubica en /var/log/lastlog o en /var/adm/lastlog y su contenido suele ser visualizado cada vez que se entra en el sistema:

login: jb

password: jb

Last login: Tue May 12 07:49:59 on tty01

- systlog

Esto no es un log sino una aplicación que viene con el sistema operativo UNIX. Dicha aplicación genera mensajes que son enviados a determinados ficheros donde quedan registrados. Estos mensajes son generados cuando se dan unas determinadas condiciones relativas a seguridad, información, etc. Los mensajes de errores típicos están ubicados en /var/log/messages, /usr/adm/messages, /var/adm/messages o incluso /var/syslog.

2.3.2. Buscar archivos setuid y setgid.

Los sistemas Unix permiten a los usuarios elevar temporalmente sus privilegios a través de un mecanismo llamado setuid. Cuando un archivo con el atributo setuid es ejecutado por un usuario, el programa se va a ejecutar con los permisos del propietario del mismo. Por ejemplo, el programa “login” es un programa con el atributo setuid y propiedad del root. Cuando un usuario lo invoca se habilita el acceso al sistema con privilegios de “súper usuario” en lugar de los del propio usuario.

2.3.3. Comprobar puertos abiertos.

Un intruso que ha atacado nuestro sistema pudo haber dejado puertos o conexiones abiertas de procesos. Para poder comprobar esto se puede usar el comando “netstat”, que principalmente nos da información de las conexiones abiertas. Lo que se debería hacer es comparar la salida de este comando con la de “last -n” para poder comprobar si existe relación entre los usuarios que se conectaron al sistema y las conexiones abiertas.

2.4. Chequear la configuración del sistema y la red.

Otro paso es examinar las entradas no autorizadas en los archivos de configuración de nuestro sistema y de nuestra red. En particular hay que buscar entradas con signo '+' y nombres de host no locales inapropiados en /etc/hosts.equiv, /etc/hosts.lpd y en todos los archivos .rhosts (especialmente root, uucp, ftp, ...) del sistema. Estos ficheros no deberían tener atributo de escritura para todo el mundo.

Específicamente, el fichero .rhosts es empleado para permitir el acceso remoto a un sistema y en algunas ocasiones es usado por los intrusos como puertas traseras.

Si el fichero fue modificado recientemente puede que se haya usado para sabotear el sistema.

Inicialmente y periódicamente debemos verificar que el host remoto y el nombre de los usuarios en dichos ficheros son consistentes.

2.5. Conclusiones.

Como conclusión podemos anotar que lo fundamental es descubrir si realmente ha entrado un intruso, ya que en muchas ocasiones pensamos que ha entrado alguien pero no es cierto. Por eso, ante todo calma, esto es lo más importante para un buen administrador

En principio, si estamos al día en materia de seguridad, así como de fallos que van surgiendo, no tendremos problemas de que un intruso entre en nuestro sistema. Realmente con un poco de esfuerzo podemos tener un servidor altamente seguro que nos evitara alrededor del 85% de los intentos de acceso no autorizados a nuestro sistema, pero en muchas ocasiones el peligro viene de los propios usuarios internos del sistema, los cuales presentan un gran riesgo debido a que ya tienen acceso al sistema, pero como siempre existen métodos de seguridad para controlar a los usuarios legítimos.

Una vez detectado que existe un intruso tenemos múltiples herramientas para detectar la intrusión y verificar los cambios realizados en el sistema así como rastrear la pista y sellar los agujeros de seguridad hay que tomar en cuenta que es importante no apagar el equipo ni reiniciarlo en lo posible ya que al hacerlo borramos logs que nos ayudarían a dar con el infiltrado.

.

PARTE II:

DETECCIÓN DE INTRUSOS UTILIZANDO UN IDS

(SISTEMA DE DETECCIÓN DE INTRUSOS)

CAPITULO 3

SEGURIDAD EN SISTEMAS DE INFORMACIÓN DETECCIÓN DE INTRUSOS.

Introducción.-

El área de Detección de Intrusos (ID, por sus siglas en inglés) es una de las más recientes dentro del vasto campo de la seguridad informática. Su objetivo consiste en identificar automáticamente un incidente que está ocurriendo, que puede ocurrir o que ha ocurrido dentro de un sistema a fin de conocer sus causas y limitar sus efectos.

La complejidad de esta disciplina radica en tres aspectos fundamentales. En primer lugar, la cantidad de información que debe procesarse es sumamente grande. Ya sea mediante el uso de datos contenidos en archivos de bitácoras, monitoreando un programa en tiempo de ejecución, o filtrando tráfico de red, es necesario identificar signos de intrusión que permitan detectar incidentes en tiempo real. Estas fuentes de datos pueden contener una cantidad de datos tal que impida su procesamiento eficiente y veloz a través de los algoritmos de detección. En la medida en la que se tienen respuestas más rápidas por parte de un sistema de ID, puede limitarse el peligro de un incidente de manera más efectiva. En segundo lugar está la creciente complejidad y diversidad de los ataques informáticos. La sofisticación del software es cada vez mayor y las probabilidades de fallas en configuración y codificación se incrementan proporcionalmente a esta complejidad. El resultado es una serie de ataques difíciles de entender y detectar de manera automática, es esa la razón fundamental que existen aplicaciones que son conocidas como IDS que no son mas que Sistemas de Detección de Intrusos que monitorean el trafico de las redes y generan alarmas mediante reglas establecidas que detectan un trafico anormal en la red siendo esta una posible intrusión, alertando de esta manera al administrador de red.

3.1. Descripción de un Sistema de Detección de Intrusos (IDS).

El Sistema de Detección de Intrusiones (IDS) es un complemento dentro de las normativas de seguridad que intenta detectar o monitorizar los eventos ocurridos en un determinado sistema informático o red informática en busca de intentos de comprometer la seguridad de dicho sistema.

Buscan patrones previamente definidos que impliquen cualquier tipo de actividad sospechosa o maliciosa sobre nuestra red o host.

Aportan a nuestra seguridad una capacidad de prevención y de alerta anticipada ante cualquier actividad sospechosa. No están diseñados para detener un ataque, aunque sí pueden generar ciertos tipos de respuesta ante éstos.

Aumentan la seguridad de nuestro sistema, vigilan el tráfico de nuestra red, examinan los paquetes analizándolos en busca de datos sospechosos y detectan las primeras fases de cualquier ataque como pueden ser el análisis de nuestra red, barrido de puertos, etc.

3.1.2. Por qué utilizar un IDS.

La detección de intrusiones permite a las organizaciones proteger sus sistemas de las amenazas que aparecen al incrementar la conectividad en red y la dependencia que tenemos hacia los sistemas de información.

Los IDS han ganado aceptación como una pieza fundamental en la infraestructura de seguridad de la organización. Hay varias razones para adquirir y usar un IDS:

Prevenir problemas al disuadir a individuos hostiles.

Al incrementar la posibilidad de descubrir y castigar a los atacantes, el comportamiento de algunos cambiará de forma que muchos ataques no llegarán a producirse. Esto también puede jugar en nuestra contra, puesto que la presencia de un sistema de seguridad sofisticado puede hacer crecer la curiosidad del atacante.

Detectar ataques y otras violaciones de la seguridad que no son prevenidas por otras medidas de protección.

Los atacantes, usando técnicas ampliamente conocidas, pueden conseguir accesos no autorizados a muchos sistemas, especialmente a aquellos conectados a redes

públicas. Esto a menudo ocurre cuando vulnerabilidades conocidas no son corregidas.

Aunque los vendedores y administradores procuran dar a conocer y corregir estas vulnerabilidades, hay situaciones en las que esto no es posible:

- En algunos sistemas heredados, los sistemas operativos no pueden ser parcheados o actualizados. Incluso en los sistemas en los que podemos aplicar parches, los administradores a veces no tienen el suficiente tiempo y recursos para seguir e instalar las últimas actualizaciones necesarias. Esto es un problema común, sobre todo en entornos que incluyen un gran número de hosts con sistemas operativos y hardware variado.
- Un sistema de detección de intrusos puede ser una excelente herramienta de protección de sistemas.
- Un IDS puede detectar cuando un atacante ha intentado penetrar en un sistema explotando un fallo no corregido. De esta forma, podríamos avisar al administrador para que llevara a cabo un backup del sistema inmediatamente, evitando así que se pierda información valiosa y entre sus funciones podemos encontrar las siguientes:
- Detectar preámbulos de ataques (normalmente pruebas de red y otras actividades).

Cuando un individuo ataca un sistema, lo hace típicamente en fases predecibles. En la primera fase, el atacante hace pruebas y examina el sistema o red en busca de un punto de entrada óptimo. En sistemas o redes que no disponen de un IDS, el atacante es libre de examinar el sistema con un riesgo mínimo de ser detectado. Esto le facilita la búsqueda de un punto débil en nuestra red.

La misma red con un IDS monitorizando sus operaciones le presenta una mayor dificultad. Aunque el atacante puede examinar la red, el IDS observará estas pruebas, las identificará como sospechosas, podrá activamente bloquear el acceso del atacante al sistema objetivo y avisará al personal de seguridad de lo ocurrido para que tome las acciones pertinentes.

Cuando se hace un plan para la gestión de seguridad de la red o se desea redactar la política de seguridad de la organización, es necesario conocer cual es el riesgo de la

organización a posibles amenazas, la probabilidad de ser atacada o si incluso ya está siendo atacada.

Un IDS nos puede ayudar a conocer la amenaza existente fuera y dentro de la organización, ayudándonos a tomar decisiones acerca de los recurso de seguridad que deberemos emplear en nuestra red y del grado de cautela que deberemos adoptar al redactar la política de seguridad.

Proveer información útil sobre las intrusiones que se están produciendo.

Incluso cuando los IDSs no son capaces de bloquear ataques, pueden recoger información relevante sobre éstos. Esta información puede, bajo ciertas circunstancias, ser utilizada como prueba en actuaciones legales.

También se puede usar esta información para corregir fallos en la configuración de seguridad de los equipos o en la política de seguridad de la organización.

3.1.3. Qué hace un IDS.

- Monitorea diversas fuentes de información de los sistemas analizando de varias maneras esta información
- Compara el tráfico con patrones de ataques
- Identifica problemas relacionados con el abuso de privilegios
- Realiza análisis estadístico en busca de patrones de actividad anormal

3.1.4. Para qué un IDS si ya tenemos Firewall.

- La mayoría de Firewalls funcionan como guardias frontales únicamente
- Los IDS son el equivalente a los sistemas de alarma con multiples sensores y monitoreo por circuito cerrado de video
- Muchas veces el enemigo ya está dentro
- Algunos productos de Firewall han incluido IDS pero se siguen llamando Firewalls.

3.1.5. Qué puede ser detectado por un IDS y por un Firewall no.

- Ataques por entunelamiento de tráfico
- Ataques a través de vulnerabilidades en aplicaciones
- Ataques que se originan desde la porción segura de la red

3.1.6. Qué se puede lograr con IDS.

- Un mayor grado de seguridad al resto de la infraestructura de seguridad
- Hacer uso de información muchas veces ignorada, para ver que está pasando en realidad
- Apoyar el rastreo de actividades intrusas desde el punto de entrada al de salida o impacto
- Reconocer alteraciones en sistemas de archivos
- Reconocer ataques en tiempo real
- Automatizar la búsqueda de trazas de ataques en Internet

3.1.7. Ventajas de los IDS

- Una subred completa puede ser cubierta por un IDS
- Teóricamente indetectables
- Mínimo impacto a la red
- Permiten detectar ataques DOS
- Independencia del ambiente operativo
- Livianos y Fáciles de implementar

3.1.8. Desventajas de los IDS

- Generación de falsos positivos
- No pueden analizar tráfico cifrado
- Son tan efectivos como la última actualización de patrones

- Alta latencia entre el ataque y la notificación
- Dificultad para realizar análisis en redes congestionadas
- No indican si un ataque ha sido exitoso o no

3.2. Tipos de IDS.

3.2.1. Según sus características.

- HIDS (Host IDS)

"Protege contra un único Servidor, PC o host. Monitorizan gran cantidad de eventos, analizando actividades con una gran precisión, determinando de esta manera qué procesos y usuarios se involucran en una determinada acción.

Recaban información del sistema como ficheros, logs, recursos, etc, para su posterior análisis en busca de posibles incidencias. Todo ello en modo local, dentro del propio sistema. Fueron los primeros IDS en desarrollar por la industria de la seguridad informática. (ver Anexo 8).

- NIDS (Net IDS).

Protege un sistema basado en red. Actúan sobre una red capturando y analizando paquetes de red, es decir, son sniffers del tráfico de red. Luego analizan los paquetes capturados, buscando patrones que supongan algún tipo de ataque.

Pueden analizar grandes redes y su impacto en el tráfico suele ser pequeño. Actúan mediante la utilización de un dispositivo de red configurado en modo promiscuo (analizan,"ven" todos los paquetes que circulan por un segmento de red aunque estos nos vayan dirigidos a un determinado equipo).

Analizan el tráfico de red, normalmente, en tiempo real. No sólo trabajan a nivel TCP/IP, también lo pueden hacer a nivel de aplicación."[2]

Las capas del TCP/IP y las del modelo OSI, y su correspondencia: (ver Anexo 9).

Nota: A este tipo de IDS pertenece snort (NIDS) (ver Anexo 10).

[2] http://www.maestrosdelweb.com/editorial/snort/ 24/01/06

3.2.2. Por el tipo de respuesta.

- Pasivos: Son aquellos IDS que notifican a la autoridad competente o administrador de la red mediante el sistema que sea, alerta, etc. Pero no actúa sobre el ataque o atacante.
- Activos: Generan algún tipo de respuesta sobre el sistema atacante o enviar algún tipo de respuesta predefinida en nuestra configuración.

Snort puede funcionar de las dos maneras.

3.3. Arquitectura de un IDS.

Normalmente la arquitectura de un IDS, a grandes rasgos, está formada:

1. La fuente de recogida de datos. Estas fuentes pueden ser un log, dispositivo de red, o como en el caso de los IDS basados en host, el propio sistema.
2. Reglas que contienen los datos y patrones para detectar anomalías de seguridad en el sistema.
3. Filtros que comparan los datos snifados de la red o de logs con los patrones almacenados en las reglas.
4. Detectores de eventos anormales en el tráfico de red.
5. Dispositivo generador de informes y alarmas. En algunos casos con la sofisticación suficiente como para enviar alertas via mail, o SMS.

Esto es a modo general. Ya veremos que cada IDS implementa la arquitectura de manera diferente.

Snort, por ejemplo, tiene una arquitectura dividida en tres subsistemas:

- Decodificador de paquetes
- Motor de detección
- Loggins y alertas

Evidentemente, son parte de la arquitectura global de un IDS que hemos comentado líneas más arriba.

3.4. Topología del IDS.

Posición de IDS (ver Anexo 11).-

Si colocamos el IDS antes del cortafuegos capturaremos todo el tráfico de entrada y salida de nuestra red. La posibilidad de falsas alarmas es grande.

La colocación detrás del cortafuegos monitorizará todo el tráfico que no sea detectado y parado por el firewall o cortafuegos, por lo que será considerado como malicioso en un alto porcentaje de los casos . La posibilidad de falsas alarmas muy inferior.

Algunos administradores de sistemas colocan dos IDS, uno delante y otro detrás del cortafuegos para obtener información exacta de los tipos de ataques que recibe nuestra red ya que si el cortafuegos está bien configurado puede parar o filtras muchos ataques.

En ambientes domésticos, que es el propósito de esta monografía sobre IDS y Snort, podemos colocar el IDS en la misma máquina que el cortafuegos. En este caso actúan en paralelo, es decir, el firewall detecta los paquetes y el IDS los analizaría.

3.5. Qué hacer cuando se detecta un intruso.

En caso de que exista la suficiente certeza de la detección de un incidente, el SDI tiene como función principal alertar al administrador o personal de seguridad, para que tome acciones al respecto.

Los IDS pueden clasificarse en base a varios aspectos: método de detección, tipo de monitoreo y forma de recolección y análisis de información. Según el método de detección, los hay de detección de mal uso y detección de anomalías.

3.6. Software requerido.

Puesta en Marcha de un IDS (Sistema de Detección de Intrusos de Red Activo)

Un IDS tiene varios aplicaciones o componentes que interactúan entre si para complementar la solución completa, para la implementación de un IDS pasivo hemos utilizado el siguiente software.-

- FEDORA CORE 3 (Sistema operativo)
- APACHE (Servidor web para acceder al sistema desde la web)

- PCRE (Expresiones regulares compatibles con perl)
- SNORT (Sistema de detección de intrusos)
- MYSQL (Base de datos para almacenar los logs de las intrusiones)
- PHP (Lenguaje para la interfase grafica)
- BASE (Interfaz gráfica para el manejo del NIDS)
- ADODB (Conjunto de librerías de bases de datos)
- JPGraph(Librería de clases para PHP para creación dinámica de imágenes)

3.7. Conclusiones.-

El propósito de un sistema de detección de intrusos (IDS -Intrusion Detection System) es identificar los accesos no autorizados o el uso incorrecto de un sistema de computación. Estos sistemas son similares a las alarmas antirrobo. Hacen sonar una alarma y algunas veces toman acciones correctivas cuando un intruso es detectado. Estos, generalmente se dividen en dos categorías: identificación de anormalidades en el sistema o uso incorrecto de los mismos.

Los detectores de anormalidades vigilan cualquier comportamiento que se desvíe del uso normal de los sistemas, mientras que los detectores de usos incorrectos hacen lo propio con cualquier comportamiento que coincida con un conocido escenario de ataque.

Como ejemplo de ello, algunos sistemas actúan como un programa de captura de paquetes de redes, interpretando actividad hostil reconociendo los patrones de tráfico en las redes que indiquen que se está produciendo un ataque. Una vez que la vulnerabilidad es identificada, el administrador es informado vía correo electrónico y una alarma es desplegada en la consola de administración. Adicionalmente, el ataque puede ser determinado automáticamente, al ser introducido a una base de datos o grabado para su posterior revisión.

CAPITULO 4

IMPLEMENTACIÓN DE DE LAS APLICACIONES PARA UN (IDS)

Introducción.-

Los IDS examinan, registran, o actúan sobre el tráfico de red. Prepararse para implementar un sistema de detección de intrusos, o IDS, como parte de su estrategia integral de seguridad, significa conocer a fondo la arquitectura de su red, de manera que pueda emplazar los sensores IDS de forma efectiva. En efecto, la ubicación es un factor crítico para que su IDS pueda protegerle adecuadamente de invasiones externas. El tráfico de Internet penetrará en su red, casi con toda seguridad, por un router. Como mucho, el router aplicará un juego inicial de filtros antes de dejar pasar dicho tráfico. Aunque el funcionamiento ordinario de un router no requiere configurar dichos filtros, los expertos en seguridad lo recomiendan. Este filtrado actúa como primera capa de defensa, manteniendo las conexiones peligrosas, como las de ICMP broadcast o protocolo de mensajes de control de Internet, siendo el ICMP la base de los ataques del tipo “Smurf”, que utilizan mensajes ping y la dirección broadcast de la IP objetivo para saturar un enlace a Internet.

4.1. Introducción al SNORT.

Snort es un IDS o Sistema de detección de intrusiones basado en red (NIDS).

Implementa un motor de detección de ataques y barrido de puertos que permite registrar, alertar y responder ante cualquier anomalía previamente definida como patrones que corresponden a ataques, barridos, intentos aprovechar alguna vulnerabilidad, análisis de protocolos, etc conocidos. Todo esto en tiempo real.

Snort (www.snort.org) está disponible bajo licencia GPL, gratuito y funciona bajo plataformas Windows y UNIX/Linux. Es uno de los más usados y dispone de una gran cantidad de filtros o patrones ya predefinidos, así como actualizaciones constantes ante casos de ataques, barridos o vulnerabilidades que vayan siendo detectadas a través de los distintos boletines de seguridad, este sistema puede ser configurado de modo pasivo o activo, para el desarrollo de este proyecto se manejara de modo pasivo.

Este IDS implementa un lenguaje de creación de reglas, el cual es flexible, potente y sencillo.

Durante su instalación ya nos provee de cientos de filtros o reglas para backdoor, ddos, finger, ftp, ataques web, CGI, escaneos Nmap....

Puede funcionar como sniffer (se puede ver qué ocurre en la red, todo el tráfico en tiempo real), registro de paquetes (permite guardar en un archivo los logs para su posterior análisis, un análisis offline) o como un IDS.

La colocación de snort en la red se realizará según el tráfico que se desee vigilar: paquetes que entran, paquetes salientes, dentro del firewall, fuera del firewall y en realidad prácticamente donde queramos.

4.1.2. Reglas snort.

Las reglas snort se dividen en dos secciones: cabecera de la regla y opciones:

La cabecera contiene la acción de la regla en sí, protocolo, IPs, máscaras de red, puertos origen y destino y destino del paquete o dirección de la operación.

La sección opciones contiene los mensajes y la información necesaria para la decisión a tomar por parte de la alerta en forma de opciones.

Las reglas de snort las dividiremos de la siguiente manera:

- Cabecera
- Acción
- Protocolos involucrados
- Direcciones IP
- Números de puerto
- Dirección de la operación
- Opciones
- Mensaje
- Opciones de decisión

Ejemplo.-

Veamos ahora un ejemplo de regla snort para alertar de un escaneo nmap del tipo TCP ping:

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"Escaneo ping con nmap";flags:A;ack:0; reference:arachnids,28;classtype:attempted-recon; sid:628; rev:1;)

Analicemos esta alerta:

- Cabecera

Acción de la regla: alert

Protocolo: tcp

Direccion IP origen: $EXTERNAL_NET (toda la red)

Puerto IP origen: any (cualquiera)

Direccion IP destino: $HOME_NET (toda nuestra red)

Puerto IP origen: any (cualquiera)

Dirección de la operación: -> (puede ser ->, <-, <>)

- Opciones

Mensaje: msg

Opciones: flags:A;ack:0; reference:arachnids..(1)

Conceptos.-

- flags:A Establece el contenido de los flags o banderas TCP, en este caso ACK (puede tener varios valores y operadores que veremos más adelante).
- ack:0 Caso particular para valor ACK=0, es el valor que pone nmap para TCP ping scan.
- reference:arachnids,28 Referencia un a un Advisory, alerta tipo Bugtrac, etc.
- classtype:attempted-recon Categoría de la alerta según unos niveles predefinidos y prioridades (veremos más adelante las categorías).
- sid:628 Identificación única para esta regla snort según unos tramos determinados.

- rev:1 Identificación de la revisión o versión de la regla.

- Instalación de las reglas creadas:

Las reglas snort de ubican en ficheros .rules (snort rules). Aquí vemos parte del contenido de uno de estos ficheros:

(C)Copyright 2001,2002, Martin Roesch, Brian Caswell, et al.
All rights reserved.
$Id: virus.rules,v 1.16 2002/08/18 20:28:43 cazz Exp $
#

#------------
VIRUS RULES
#-----------

#

NOTE: These rules are NOT being actively maintained.
#

#
If you would like to MAINTAIN these rules, e-mail
snort-sigs@lists.sourceforge.net

alert tcp any 110 -> any any (msg:"Virus - SnowWhite Trojan Incoming"; content:"Suddlently"; sid:720; classtype:misc activity; rev:3;)

alert tcp any 110 -> any any (msg:"Virus - Possible pif Worm"; content: ".pif"; nocase; sid:721; classtype:misc-activity; rev:3;)

alert tcp any 110 -> any any (msg:"Virus - Possible NAVIDAD Worm"; content: "NAVIDAD.EXE"; nocase; sid:722; classtype:misc-activity; rev:3;)

alert tcp any 110 -> any any (msg:"Virus - Possible MyRomeo Worm"; content: "myromeo.exe"; nocase; sid:723; classtype:misc-activity; rev:3;)

alert tcp any 110 -> any any (msg:"Virus - Possible MyRomeo Worm"; content: "myjuliet.chm"; nocase; sid:724; classtype:misc-activity; rev:3;)

Muestra de las reglas dentro del IDS (ver Anexo 12).

4.1.3. Reglas a implementar en el ISP de ETAPATELECOM S.A.

A continuación vamos a ver algunas de las reglas para el IDS a ser implementado en ETAPATELECOM S.A. para proteger los sistemas de intrusiones no autorizadas.

Los posibles ataques se describen a continuación:

- Ataques contra el servidor Web a través del puerto 80.-

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"INTENTO DE ACCESO AL SERVIDOR WEB";flags: A+; content:"/phf";flags: A+; nocase; reference:arachnids,128; reference:cve,CVE-1999-0067;)

- Scaneos nmap.-

“icmp.rules:alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"INTENTO DE SCANEO ICMP Nmap2.36BETA or HPING2 Echo ";itype:8;dsize:0; reference:arachnids,162;)

icmp.rules:alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:" INTENTO DE SCANEO ICMP PING NMAP"; dsize: 0; itype: 8; reference:arachnids,162;)

scan.rules:alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:" INTENTO DE SCANEO nmap fingerprint attempt";flags:SFPU; reference:arachnids,05;)

scan.rules:alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:" INTENTO DE SCANEO nmap TCP";flags:A;ack:0; reference:arachnids,28;)”[3]

- Acceso telnet con login fallido.-

alert tcp $TELNET_SERVERS 23 -> $EXTERNAL_NET any (msg:"TELNET login incorrect"; content:"Login incorrect"; flow:from_server,established; reference:arachnids,127; classtype:bad-unknown; sid: 718; rev:6;)

- Acceso a la red de seguridad DMZ

alert tcp $EXTERNAL any -> $DMZ 80 (msg:"INTENTO DE ACCESO A LA RED DMZ"; flow:to_server; uricontent:"%3F";)

- Acceso a al servidor de mail.-

[3] http://www.wikilearning.com/reglas_snort_para_casos_varios-wkccp-4735-16.htm 26/01/06

alert tcp $EXTERNAL_NET any -> 192.168.2.2 25 (msg:”ENCONTRADO INTENTO DE ACCESO AL E-MAIL”; content:”HELO”;)

- Acceso a las paginas internas de aplicaciones de la empresa.-

alert tcp any any <> $WEB_SERVER 80 (content: "mrtg.etapaonline.net.ec";

msg: "Intento de acceso a paginas internas"; react: block, msg;)

4.2. Consola basic analysis and security engine (BASE) y B.D. MYSQL.

Nuestro objetivo va a ser configurar snort para que logee en la base de datos MYSQL, para después instalar BASE, una aplicación Web escrita en PHP que nos permitirá acceder a toda la información que proporciona snort de manera ordenada y sencilla. BASE nos permitirá realizar búsquedas de todo tipo en la base de datos, estas búsquedas pueden ser por ip fuente/destino, por fecha, por ataque, por protocolo, realizar informes, graficas, etc.,

4.3. Puesta en marcha de un IDS con snort.

Para la puesta en marcha del IDS vamos a tomar en cuenta que vamos a implantar un IDS pasivo lo que significa que la función es configurar reglas para detectar alertas que le lleven al administrador a estar pendiente de las intrusiones o problemas que se presenten en sus redes, nuestro IDS va a estar conformado por la aplicación snort con base de datos en MYSQL y para la generación de alertas y reportes vamos a utilizar la consola BASE, también instalaremos APACHE para la administración vía web, la ubicación del IDS va a estar antes del Firewall por lo que estará de cara hacia el Internet y su función será de registrar el trafico entrante hacia nuestra red.

4.3.1. Prerrequisitos

Sin importar qué sistemas vigile o su forma de trabajar, cualquier sistema de detección de intrusos ha de cumplir algunas propiedades para poder desarrollar su trabajo correctamente. En primer lugar, y quizás como característica más importante, el IDS ha de ejecutarse continuamente sin nadie que esté obligado a supervisarlo; independientemente de que al detectar un problema se informe a un operador o se

lance una respuesta automática, el funcionamiento habitual no debe implicar interacción con un humano. Podemos fijarnos en que esto parece algo evidente:
muy pocas empresas estarían dispuestas a contratar a una o varias personas simplemente para analizar logs o controlar los patrones del tráfico de una red. Sin entrar a juzgar la superioridad de los humanos frente a las máquinas (¿Puede un algoritmo determinar perfectamente si un uso del sistema está correctamente autorizado?) o viceversa (¿Sería capaz una persona de analizar en tiempo real todo el tráfico que llega a un servidor web mediano?), hemos de tener presente que los sistemas de detección son mecanismos automatizados que se instalan y configuran de forma que su trabajo habitual sea transparente a los operadores del entorno informático.
Otra propiedad, y también como una característica a tener siempre en cuenta, es la aceptabilidad o grado de aceptación del IDS; al igual que sucedía con cualquier modelo de autenticación, los mecanismos de detección de intrusos han de ser aceptables para las personas que trabajan habitualmente en el entorno.
Una tercera característica a evaluar a la hora de hablar de sistemas de detección de intrusos es la adaptabilidad del mismo a cambios en el entorno de trabajo. Como todos sabemos, ningún sistema informático puede considerarse estático: desde la aplicación más pequeña hasta el propio kernel de Unix.
Pasando por supuesto por la forma de trabajar de los usuarios.
Todo IDS debe además presentar cierta tolerancia a fallos o capacidad de respuesta ante situaciones inesperadas; insistiendo en lo que comentábamos antes sobre el carácter altamente dinámico de un entorno informático, algunos o muchos de los cambios que se pueden producir en dicho entorno no son graduales sino bruscos, y un IDS ha de ser capaz de responder siempre adecuadamente ante los mismos.

4.3.2. Compilar snort

Para la compilación e instalación de snort debemos seguir los siguientes pasos, es aconsejable leer los archivos de readme e install que vienen con el paquete, los cuales son de gran ayuda para la instalación del mismo, los pasos a seguir son sencillos y son los siguientes:

- Descomprimos el paquete de instalación de snort:

tar -xvzf snort-2.3.0.tar.gz

- Entramos al directorio de snort:

cd snort-2.3.0

- En este paso especificamos que queremos instalar con soporte a mysql:

./configure --with-mysql

- Ejecutamos la instalación:

make

make install

- Agregamos un grupo y un usuario llamado snort:

groupadd snort

useradd -g snort snort

- Creamos un directorio llamado snort:

mkdir /etc/snort

- Creamos un directorio llamado rules es donde van a estar las reglas del snort:

mkdir /etc/snort/rules

- Creamos un directorio donde se van a guardos los logs de snort:

mkdir /var/log/snort

Instalando las reglas.-

cd rules

cp * /etc/snort/rules

cd ../etc

cp * /etc/snort

4.3.3. Configuración MYSQL

Para que Snort deje sus logs en la base de datos, primero que crear una nueva base de datos con sus tablas correspondientes, asi como un usuario que tenga acceso a esa base de datos.

En el punto de los pre-requisitos hemos dejado la MYSQL recién instalada y funcionando, ahora hay que añadirle usuarios. El usuario con más permisos es el “root”, aunque es importante entender que los usuarios del sistema no son los mismos usuarios que los de la MYSQL.

De modo que desde la cuenta de cualquier usuario del sistema se puede acceder a cualquier cuenta de MYSQL.

Si tecleamos para empezar el siguiente comando, entraremos directamente en la MYSQL y seguimos con los pasos a continuación:

MYSQL

MYSQL> SET PASSWORD FOR root@localhost=PASSWORD('password');

>Query OK, 0 rows affected (0.25 sec)

MYSQL> create database snort;

>Query OK, 1 row affected (0.01 sec)

MYSQL> grant INSERT,SELECT on root.* to snort@localhost;

>Query OK, 0 rows affected (0.02 sec)

MYSQL>set password for snort@localhost=PASSWORD('password_from_snort.conf');

>Query OK, 0 rows affected (0.25 sec)

MYSQL> grant create, insert, select, delete, update on snort.* to snort@localhost;

>Query OK, 0 rows affected (0.02 sec)

MYSQL> grant create, insert, select, delete, update on snort.* to snort;

>Query OK, 0 rows affected (0.02 sec)

MYSQL> exit

>Bye

Después de haber seguido los pasos anteriores ejecutamos el siguiente comando para crear las tablas:

MYSQL -u root -p < ~/snortinstall/snort-2.3.0/schemas/create_mysql snort

Enter password: (Ingresamos la clave de root de MYSQL)

Ahora debemos verificar y asegurarnos que la base de datos snort ha sido creada correctamente, para esto realizamos los siguientes pasos:

MYSQL –p

>Enter password:

MYSQL> show databases;

(Usted debería ver lo siguiente (ver Anexo 13).

MYSQL> use snort

>Database changed

MYSQL> show tables;

(ver Anexo 14).

exit;

4.3.4. Configurar snort

Debemos editar el archivo de configuración que hemos copiado antes en /etc. Cuando editemos el /etc/snort.conf veremos que está muy comentado, y antes de cada opción hay una explicación de varias líneas.

Primero definimos el rango de direcciones de nuestra red interna.

var HOME_NET [192.168.1.0/24,192.168.10.0/24,172.26.0.0/24]

Definimos todo lo que no sea la red interna en este caso definimos la red externa:

var EXTERNAL_NET !$HOME_NET

Cambiamos la línea que dice, aquí le decimos al snort la dirección en donde están las reglas:

var RULE_PATH ../rules" to "var RULE_PATH /etc/snort/rules

Ahora tenemos que decir al snort que se loguee con MYSQL, para eso nos vamos a la parte abajo y agregamos la línea que dice:

output database: log, MYSQL, user=snort password=snort dbname=snort host=localhost

Ahora debemos asegurarnos que snort inicie con el sistema:

Adicionamos esta línea a /etc/rc.local:

/usr/local/bin/snort -c /etc/snort/snort.conf -i eth0 -g snort -D

4.3.5. Configuración APACHE

Con respecto a la configuración del APACHE esto se lo realizó en el momento de la instalación del sistema operativo Linux, ahora para verificar si esta funcionando bien realizamos lo siguiente:

Primero alzamos el servicio del APACHE:

service httpd start

Para comprobar que el APACHE y el PHP están funcionando correctamente, creamos un archivo llamado test.php en el directorio /var/www/html y colocamos esta línea el archivo:

<?php phpinfo(); ?>

Ahora utilizamos el navegador para comprobar que esta funcionando correctamente y colocamos la siguiente dirección (http://Direccion_IP/test.php)

Debería dar información de tu sistema, así como también de APACHE y PHP.

4.3.6. Instalando y configurando BASE

Para la instalación de BASE realizamos los siguientes pasos:

Copiamos el instalador de BASE en el directorio /var/www/html con el siguiente comando:

cp base-1.0.1.tar.gz /var/www/html/

Entramos al directorio:

cd /var/www/html

Procedemos a descomprimir el paquete de instalación de BASE:

tar –xvzf base-1.0.1.tar.gz

Borramos el paquete de instalación del directorio mencionado anteriormente:

rm –rf base-1.0.1.tar.gz

Ingresamos al directorio de BASE y cambiamos de nombre al archivo de configuración que viene por defecto:

cd /var/www/html/base/

cp base_conf.php.dist base_conf.php

Editamos el archive “base_conf.php” e ingresamos los siguientes parámetros:

$BASE_urlpath = "/base";

$DBlib_path = "/var/www/html/adodb";

$DBtype = "mysql";

$alert_dbname = "snort";

$alert_host = "localhost";

$alert_port = "";

$alert_user = "snort";

$alert_password = "password_del_archivo_snort_conf";

$archive_dbname = "snort";

$archive_host = "localhost";

$archive_port = "";

$archive_user = "snort";

$archive_password = " password_del_archivo_snort_conf ";

$ChartLib_path = "/var/www/html/jpgraph-1.16/src";

Ahora abrimos una ventana del navegador, si tu navegador está en la computadora local digitamos localhost/base. Si tu navegador se encuentra en una maquina

diferente colocamos la dirección IP de la computadora de la siguiente manera http://Dirección_IP/base.

Damos un clic en la opción que dice “Setup Page”, en la pagina resultante debemos dar clic en el botón “setup AG” que nos entregara la siguiente pantalla (ver Anexo 15).

Damos clic en Home y veremos la página principal de BASE (ver Anexo 16).

4.3.7. Autentificación para acceder al BASE

Para que BASE nos pida autentificación (login y password) al entrar, podemos hacerlo de varios modos, pero nosotros lo recomendamos hacer con la autentificación del propio servidor Web.

Y para esto debemos seguir los siguientes pasos:

Creamos el directorio passwords:

mkdir /var/www/passwords

Ejecutamos el siguiente comando:

/usr/bin/htpasswd -c /var/www/passwords/passwords base

Editamos el archivo httpd.conf que se encuentra en (/etc/httpd/conf) y colocamos lo siguiente después de la línea que dice:

<Directory />

Options FollowSymLinks

AllowOverride None

</Directory>

Estas son las líneas que debemos agregar para que BASE nos pida un password.

<Directory "/var/www/html/base">

AuthType Basic

AuthName "SnortIDS"

AuthUserFile /var/www/passwords/passwords

Require user base

</Directory>

4.4. Herramientas a utilizar para detector intrusos.

En Linux las herramientas para detectar intrusiones son gratuitas y se encuentran fácilmente disponibles. La primera línea de defensa debería ser un cortafuegos robusto, seguido de filtros de paquetes en todas las máquinas accesibles desde Internet, uso liberal de TCP-WRAPPERS, logs y lo más importante, software automatizado para que examine los logs en tu lugar (hoy en día para un administrador es impracticable que pueda leer los ficheros de log).

4.4.1. Herramientas para logs.

- Psionic PortSentry

"El tercer componente de la suite Abacus, detecta y guarda un log de los escaneos de puertos, incluyendo escaneos clandestinos (stealth) (básicamente debería ser capaz de detectar cualquier cosa que sea posible hacer con Nmap). Se puede configurar el Psionic Portsentry para que bloquee la máquina atacante (en mi opinión es una mala idea, pues se podría utilizar para generar un ataque de denegación de servicio en hosts legítimos), haciendo difícil el completar un escaneo de puertos. Puesto que esta herramienta está en fase beta, no recomendaría su uso, sin embargo, con el tiempo debería madurar hasta convertirse en una herramienta sólida y útil."[4]

4.4.2. Detección de ataques basada en Host.

- Cortafuegos

La mayoría de los cortafuegos soportan guardar logs de los datos, y el ipfwadm/iptables no son una excepción, utilizando la opción –l se debería generar una entrada en el syslog para cada paquete, utilizando filtros automatizados (para esto es bueno el Perl) se pueden detectar tendencias/ataques hostiles, etcétera. Puesto que la mayoría de los cortafuegos guardan un log vía syslog, se puede centralizar todo el log de paquetes del cortafuegos en un único host .

[4] http://www.psionic.com/abacus/portsentry/ 28/01/06

- Tcp-wrappers

"El tcp-wrappers de wietse te permite restringir las conexiones a varios servicios basándose en direcciones IP, pero incluso más importante es el hecho de que te permite configurar una respuesta, se puede hacer que te envíe un correo, haga finger a la máquina atacante, etcétera (sin embargo, hay que utilizarlo con cuidado)."[5]

- Klaxon

"Si bien ha quedado obsoleto por el TCP_WRAPPERS y los logs de los cortafuegos, Klaxon todavía puede ser útil para detectar escaneos de puertos, si no se quiere bloquear por completo la máquina."[6]

- Psionic HostSentry

"Aunque este software todavía no está listo para el consumo en masa, he pensado que lo mencionaría de todas formas, pues forma parte de un proyecto más grande (el proyecto Abacus, http://www.psionic.com/abacus/). En resumen, el Psionic HostSentry construye un perfil de accesos del usuario y después lo compara con la actividad actual, para resaltar cualquier actividad sospechosa."[7]

4.4.3. Detección de ataques basada en red.

- NFR

"El NFR (Registro de Vuelo de Red, Network Flight Recorder) es mucho más que un sniffer de paquetes, en realidad guarda un log de los datos y detecta en tiempo real los ataques, escaneos, etcétera. Es una herramienta muy potente y para ejecutarse requiere una significativa inversión de tiempo, energías y potencia de máquina, pero está en la cima de la cadena alimenticia en cuanto a detección."[8]

4.5. Herramientas de análisis.

---

[5] eftp://ftp.porcupine.org/pub/security/ 28/01/06

[6] ftp://ftp.eng.auburn.edu/pub/doug 28/01/06

[7] http://www.psionic.com/abacus/hostsentry/ 28/01/06

[8] http://www.nfr.com/ 28/01/06

En el mercado existen diferentes herramientas para analizar vulnerabilidades de un red. Estas herramientas son muy útiles para los administradores de red preocupados por al seguridad e integridad de su red y la información que en ella manejan. Entre los principales analizadores se puede encontrar NESSUS y SATAN, los cuales ofrecen una amplia gama de reglas para evaluar las vulnerabilidades y, además, permiten la incorporación de nuevas reglas para hacer más riguroso y específico el análisis. Sin embargo, estas herramientas se convierten en armas de doble filo, pues pueden ser usadas con el objetivo de mejorar la seguridad de la red o pueden ser usadas por hackers con el objetivo de detectar vulnerabilidades y realizar ataques.

4.6. Conclusiones.-

Como se ha visto en este capítulo mediante la configuración y análisis de las herramientas necesarias para un sistema de detección de intrusos podemos anotar que la aplicación snort es una muy buena alternativa a la hora de implementar un IDS siendo esta la más usada a nivel mundial y estando al alcance de todos ya que es una aplicación de software libre, existiendo la suficiente documentación dentro de la web para la ejecución y análisis de la misma.

CAPITULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones.-

Al terminar la monografía se han obtenido las siguientes conclusiones:

- Existen varios tipos de intrusos que pueden convertirse en una amenaza en nuestros sistemas.
- Los ataques no solo son desde el exterior de nuestras redes (Internet) sino que en gran parte provienen de nuestras redes internas, por gente que conoce nuestros sistemas.
- Antes de las intrusiones los piratas hacen escaneos de nuestras vulnerabilidades.
- Los ataques son una realidad, ya que las estadísticas demuestran que se incrementan notablemente en los últimos tiempos, siendo posibles victimas sino ponemos las debidas seguridades en nuestros sistemas.
- Vemos que en el sistema operativo Linux existen múltiples herramientas para localizar al intruso cuando a comprometido nuestro sistemas y podemos seguirle la pista para ver los daños causados en el sistema, pudiendo solucionarlo sin formatear el equipo.
- No debemos apagar el equipo después de una intrusión ya que borraríamos los logs y le perderíamos la pista al intruso.
- Antes de realizar correcciones debemos estar seguros de que hemos sido victimas de una intrusión.
- Debemos realizar chequeos de nuestras vulnerabilidades constantemente.
- La implementación de un IDS es un factor importante, ya que sin el nuestros sistemas de seguridad están a ciegas del trafico en nuestra red.
- Existen dos tipos de IDS como son: los pasivos, que verifican las intrusiones y los activos, que además coordinan con el Firewall para bloquear al intruso.

- Un IDS no es un Firewall, pero si un complemento que trabaja independientemente y nos ayuda a contrarrestar ataques.
- Existen múltiples herramientas y aplicaciones que nos ayudan a implementar un IDS.
- No necesitamos gastar dinero en aplicaciones costosas para la implementación de los IDS, ya que las más utilizadas y mejores son gratuitas, siendo código GNU.
- La herramienta mas utilizada es el snort y que es una aplicación que nos ofrece múltiples ventajas para controlar las intrusiones y de fácil manejo.
- El guardar la información de los intentos de intrusiones es importante para realizar estadísticas y así evaluar nuestras seguridades.
- Si somos victimas de una intrusión debemos notificarlo al organismo de control CERT.org.

5.2. Recomendaciones.-

- No debemos pensar que porque no hemos sido victimas de ataques no lo seremos, debemos implementar nuestras seguridades en nuestras redes, tanto para protegernos de extraños como de nuestros empleados.
- Tener siempre presente las herramientas necesarias o comandos que necesitaremos para verificar intrusiones y rastrear a los intrusos dentro de nuestros sistemas.
- No asustarnos si vemos anomalías en los sistemas antes de estar seguros que somos victimas de una intrusión o un ataque.
- Analizar detenidamente en base al origen de nuestra empresa en los posibles agresores de los sistemas, ya que la mejor manera de protegerse es conociendo a nuestro enemigo.
- Verificar si nosotros somos el destino del ataque o nos están utilizando para atacar otras redes.

- Es importante estar empapados de las configuraciones de nuestros sistemas y tener fuentes de respaldo de las mismas, para reemplazarlas en caso de modificaciones sufridas por algún ataque.
- Dejar abiertos solos los puertos necesarios en los sistemas.
- Restringir el acceso a personas o redes ajenas a nuestro trabajo.
- Instalar sistemas de detección de intrusos, tanto para monitorear las redes externas como las internas y guardar la información de los logs en bases de datos, para realizar reportes y tener una idea clara de lo que sucede en nuestras redes.
- Implementar un pequeño sistema que genere un aviso o envié un mensaje al celular avisando de esta manera al administrador.

CAPITULO 7

BIBLIOGRAFÍA

Para el desarrollo del presente proyecto de monografía hemos recurrido a diferentes medios para recavar la información necesaria para el desarrollo de la misma, los medios utilizados han sido el Internet y libros como podemos citar los siguientes:

http://www.fentlinux.com/listing/manuales/inst-fc3.pdf
Notas de Instalación Fedora Core 3
Fecha de ingreso: 17/01/06

http://eiee.univalle.edu.co/~telecomunicaciones/tesis/Manual/Manual_vulnerabilidades.pdf
MANUAL DE DETECCIÓN DE VULNERABILIDADES
DE SISTEMAS OPERATIVOS LINUX Y UNIX EN REDES TCP/IP
Fecha de Ingreso: 17/01/2006

http://andercheran.upv.es/~toni/personal/transpas-ids.pdf
Antonio Villalón Huerta
Fecha de Ingreso: 17/01/2006

http://penta.ufrgs.br/gereseg/node50.htm
Fecha de Ingreso: 18/01/2006

http://club.telepolis.com/websecure/tutoriales/tutorial_snort.pdf
Armando Mira
Fecha de Ingreso: 18/01/2006

http://www-ma2.upc.es/~cripto/Q1-03-04/presentacionIDS.pdf
Sacha Fuentes
Fecha de Ingreso: 18/01/2006

http://redes-linux.all-inone.net/manuales/seguridad/snort_Mysql_acid.pdfAlianza WiFi
LINUCA
Fecha de Ingreso: 20/01/2006

http://www.sun.com/bigadmin/features/articles/snort_base.html#intro#intro
BigAdmin System Administration Portal
Feature Articles
Fecha de Ingreso: 21/01/2006

http://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/es/security-guide/ch-intro.html
Red Hat Enterprise Linux 4: Manual de seguridad
Fecha de Ingreso: 22/01/2006

http://www.net-security.org/dl/articles/snort_enterprise.pdf
Snort Enterprise Implementation
Fecha de Ingreso: 24/01/2006

http://www.maestrosdelweb.com/editorial/snort/
Fecha de Ingreso: 24/01/2006

http://www.snort.org/docs/
Snort Documents
Fecha de Ingreso: 24/01/2006

http://www.Snort.org/docs/writing_rules/
Snort
Fecha de Ingreso: 24/01/2006

http://www.mysql.com
Mysql
Fecha de Ingreso: 25/01/2006

http://www.php.net
PHP
Fecha de Ingreso: 25/01/2006

http://fedora.redhat.com
FEDORA TM
Fecha de ingreso: 25/01/2006

http://www.wikilearning.com/reglas_snort_para_casos_varios-wkccp-4735-16.htm
Fecha de Ingreso: 26/01/2006

eftp://ftp.porcupine.org/pub/security/
Fecha de Ingreso: 28/01/2006

ftp://ftp.eng.auburn.edu/pub/doug
Fecha de Ingreso 28/01/2006

http://www.psionic.com/abacus/hostsentry/
Fecha de Ingreso: 28/01/2006

http://www.nfr.com/
Fecha de Ingreso: 28/01/06

http://www.nessus.org
Nessus Vulnerability Scanner
Fecha de Ingreso: 28/01/2006

http://www.ldc.usb.ve/~miguel/portknocking_and_snort.pdf
PORTKNOCKING & SNORT
Fecha de Ingreso: 30/01/2006

http://www.wikilearning.com/creacion_de_reglas_con_snort-wkccp-4735-14.htm
Creación de reglas con snort
Fecha de Ingreso: 10/02/06

http://www.unap.cl/davidcontreras/Ejecucion4/apuntes/linux/paraver.htm
UNIX BASICO
Fecha de Ingreso: 15/02/06

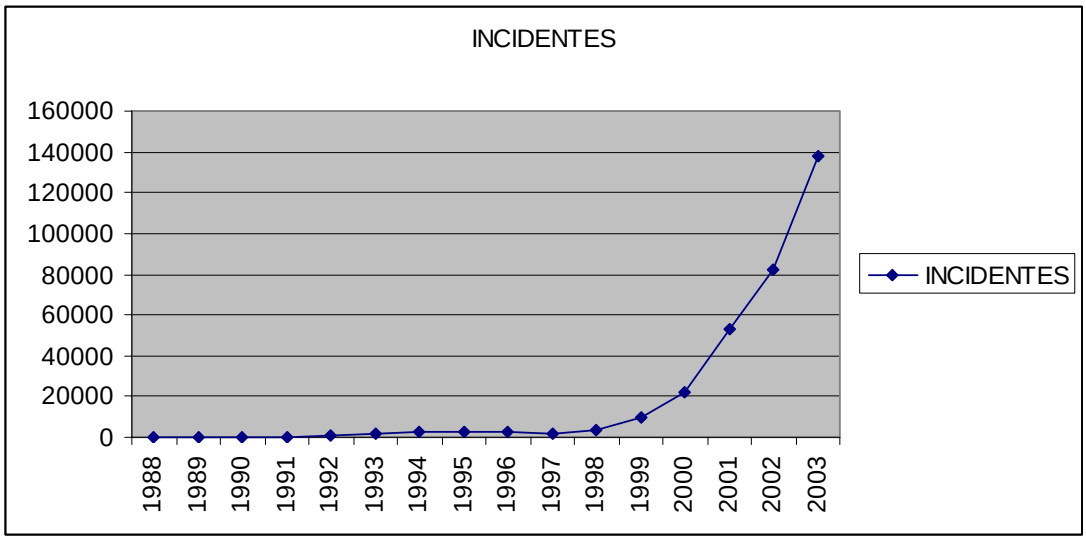
Hackers 3
Secretos y soluciones para la seguridad de redes
MCCLURE Stuart
SCAMBRAY Joel
KURTZ George

Linux para usuarios de Windows
MILLER Michael

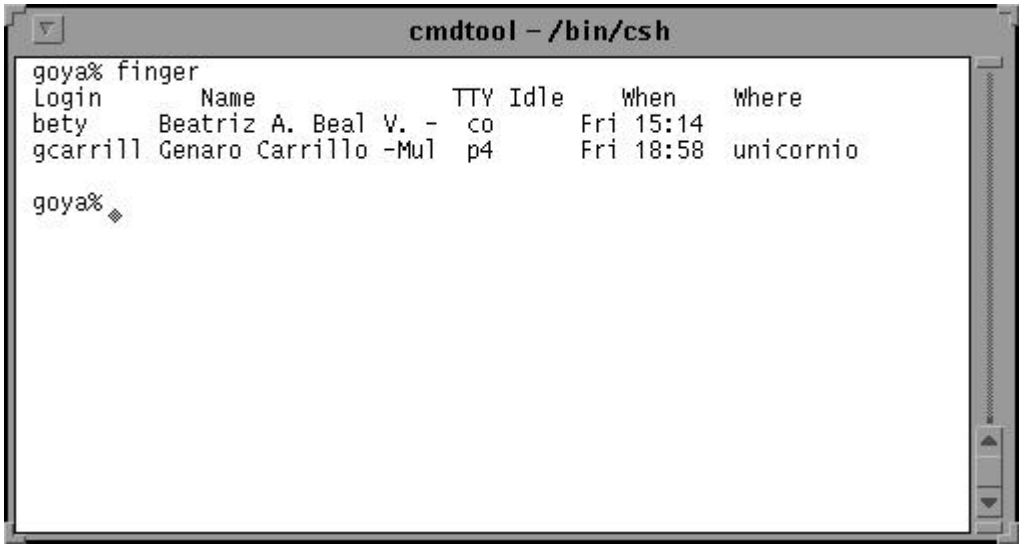
Linux 6ª Edición
BANDEL David
NAPIER Robert

ANEXOS

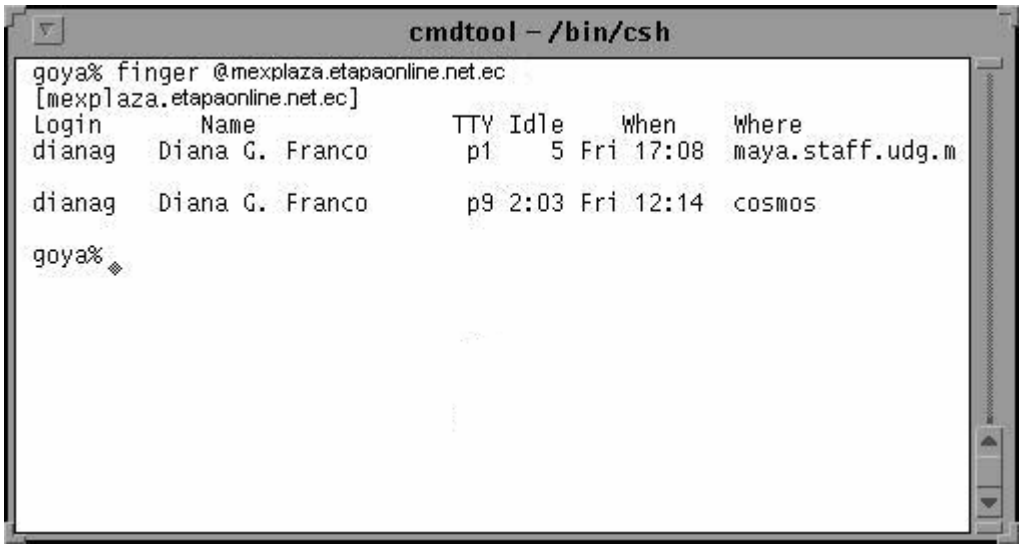
Anexo 1.- Estadísticas de intrusiones registradas en el CERT.



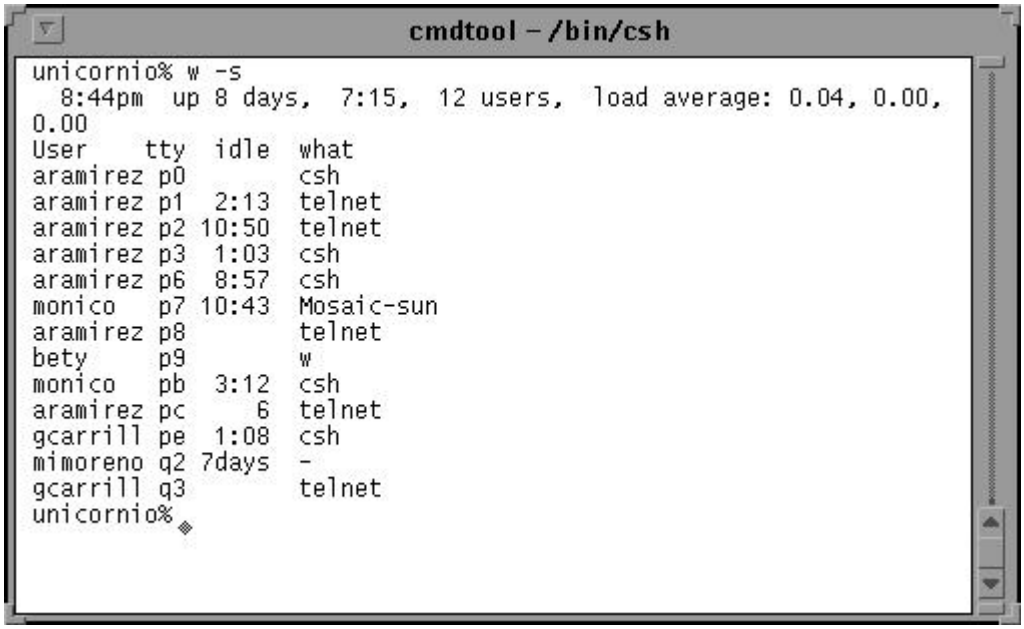
Anexo 2.- Aplicación del comando finger.



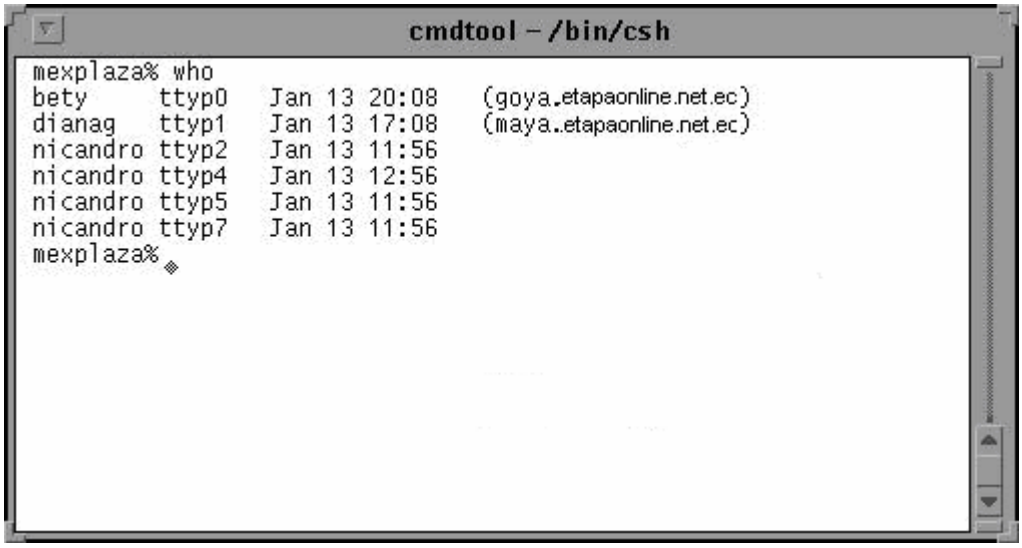
Anexo 3.- Aplicación del comando finger verificando otro PC.



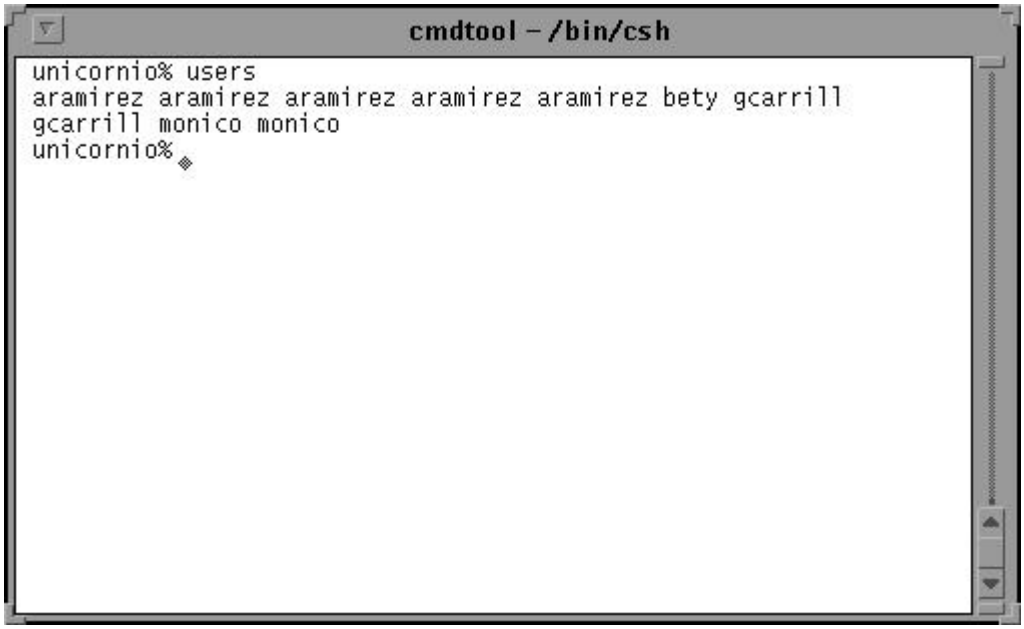
Anexo 4.- Aplicación del comando w.



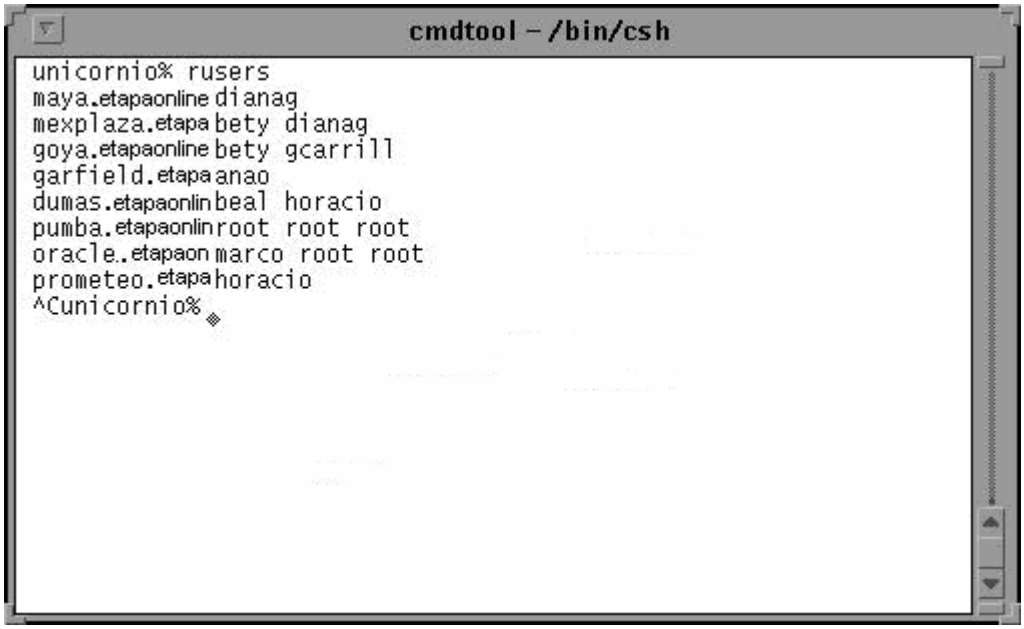
Anexo 5.- Aplicación del comando who.



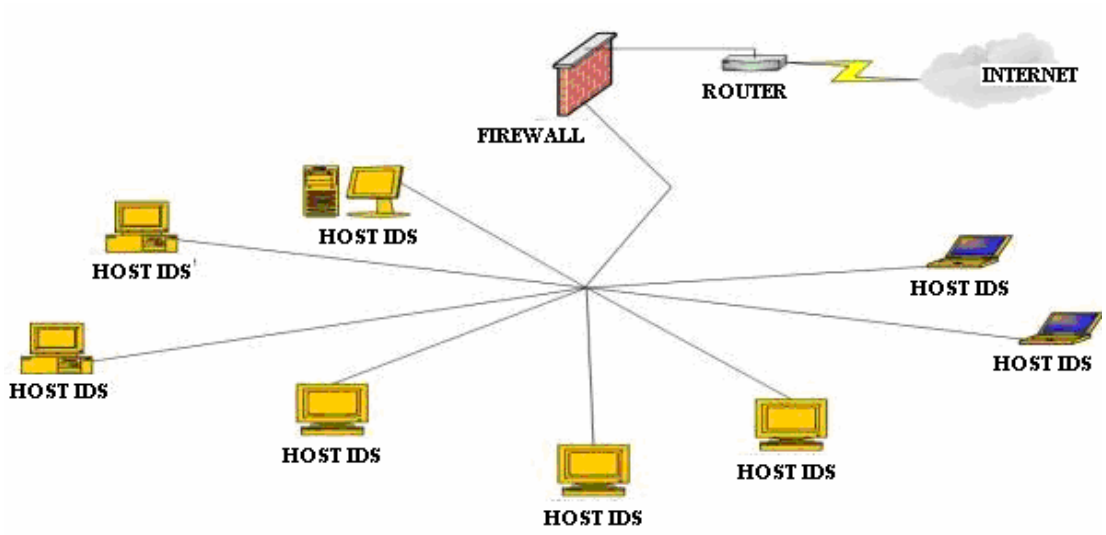
Anexo 6.- Aplicación del comando users.



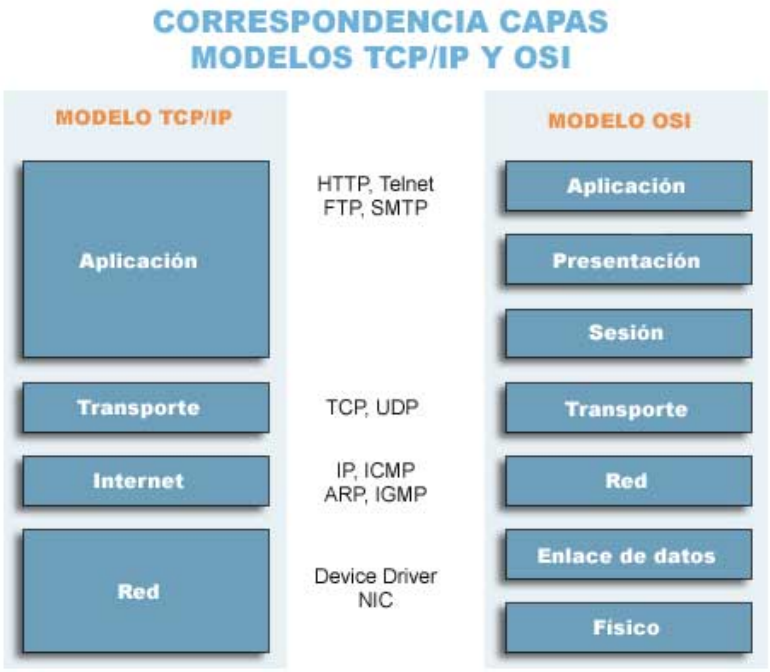
Anexo 7.- Aplicación del comando rusers.



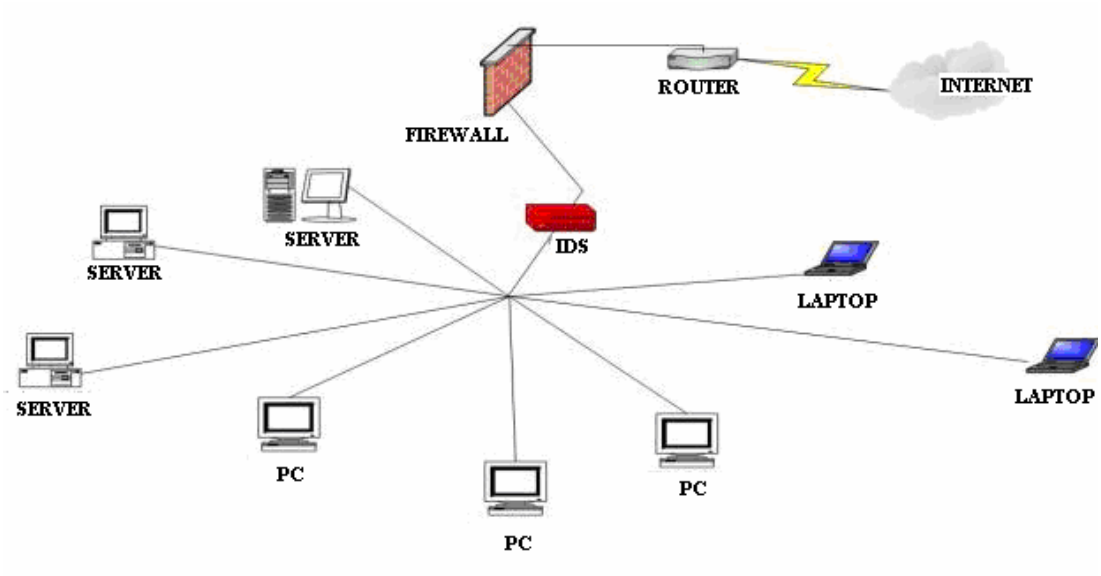
Anexo 8.- Estructura de un Host IDS.



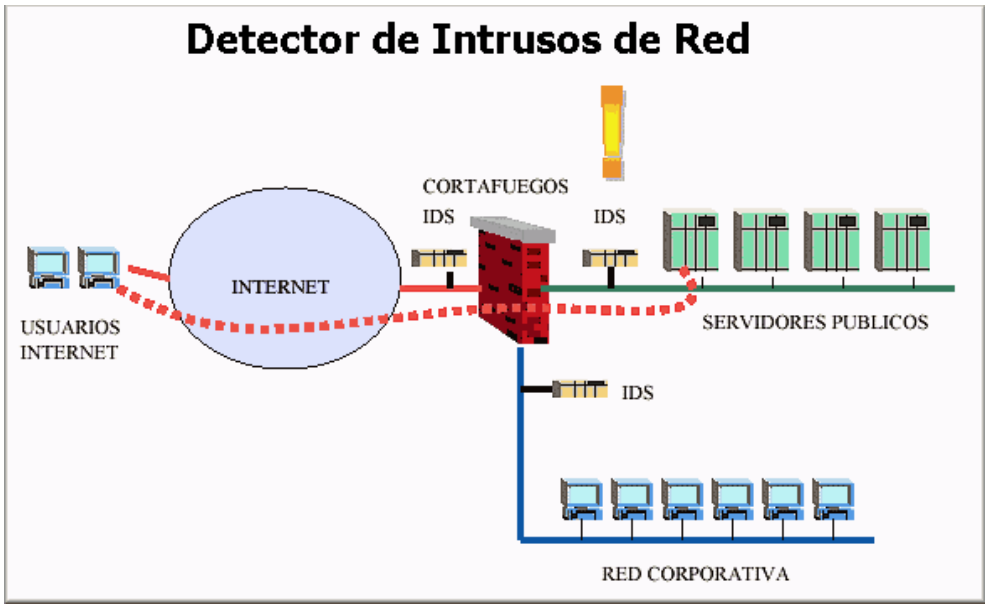
Anexo 9.- Capas del TCP/IP y las del modelo OSI, y su correspondencia.



Anexo 10.- Estructura de un Network IDS.



Anexo 11.- Ubicación de un IDS en una red.



Anexo 12.- Reglas del IDS.

```
[root@localhost rules]# ls
attack-responses.rules  experimental.rules  info.rules       policy.rules      sid-msg.map      VRT-License.txt
backdoor.rules          exploit.rules       local.rules      pop2.rules        smtp.rules       web-attacks.rules
bad-traffic.rules       finger.rules        misc.rules       pop3.rules        snmp.rules       web-cgi.rules
cgi-bin.list            ftp.rules           multimedia.rules  porn.rules        snort.conf       web-client.rules
chat.rules              generators          mysql.rules      reference.config  sql.rules        web-coldfusion.rules
classification.config   gen-msg.map         netbios.rules    rpc.rules         telnet.rules     web-frontpage.rules
ddos.rules              icmp-info.rules     nntp.rules       rservices.rules   tftp.rules       web-iis.rules
deleted.rules           icmp.rules          oracle.rules     scan.rules        threshold.conf   web-misc.rules
dns.rules               icmp.rules.bak      other-ids.rules  shellcode.rules   unicode.map      web-php.rules
dos.rules               imap.rules          p2p.rules        sid               virus.rules      x11.rules
```

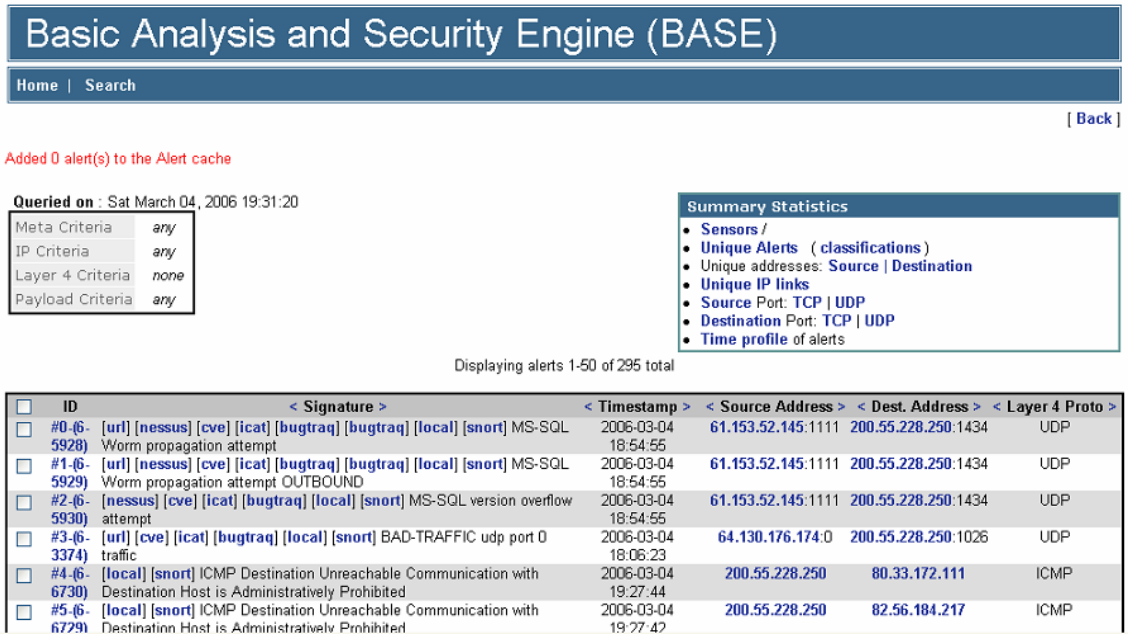
Anexo 13.- Vista de las bases de datos en mysql.

```
+-----------+
| Database
+-----------+
| mysql
| Snort
| test
+-----------+
```

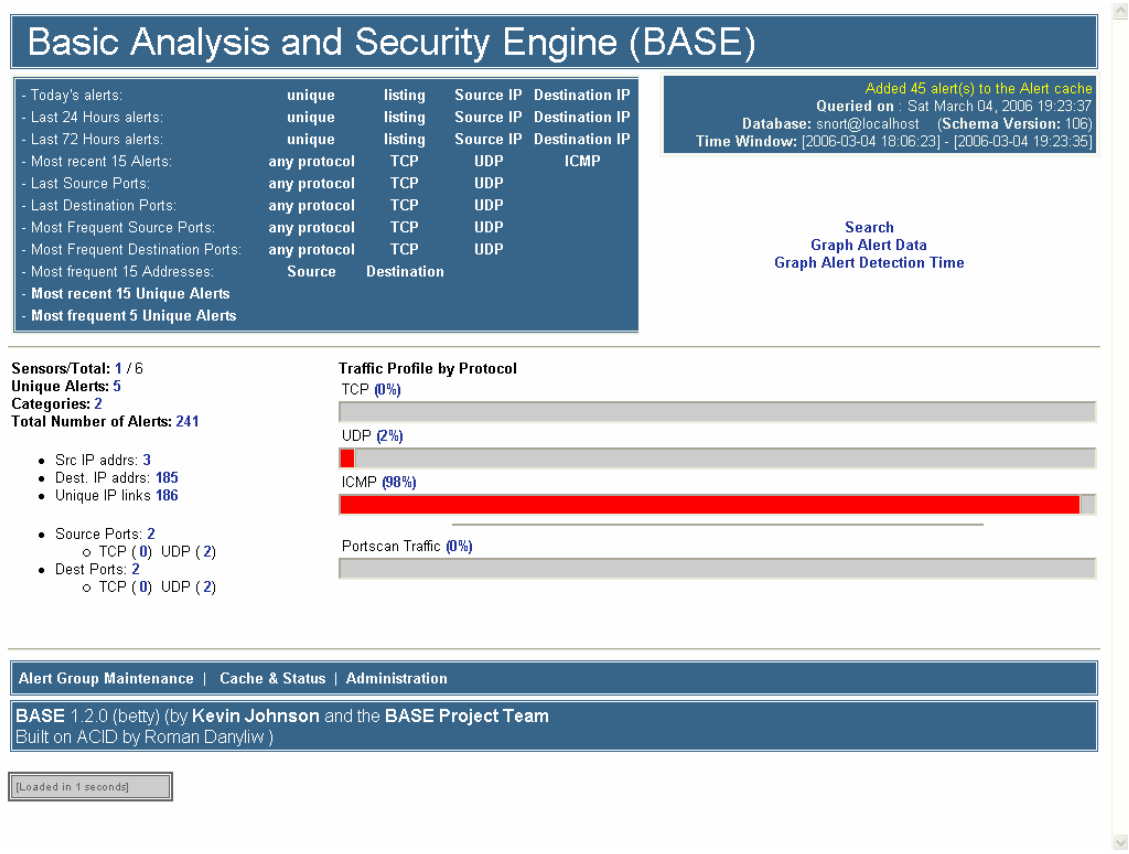
Anexo 14.- Vista de las tablas de datos en mysql.

```
+-----------------+
| Tables_in_snort
+-----------------+
| data
+-----------------+
| Tables_in_snort
+-----------------+
| data
| detail
| encoding
| event
| icmphdr
| iphdr
| opt
| reference
| reference_system
| schema
| sensor
| sig_class
| sig_reference
| signature
| tcphdr
| udphdr
+-----------------+
```

Anexo 15.- Pantalla de la aplicación BASE.



Anexo 16.- Pantalla del home de BASE.



Anexo 17.- Diseño de la monografía.