UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENERÍA ELÉCTRICA
DEPARTAMENTO DE COMPUTACIÓN

"IMPLEMENTACIÓN DE TECNOLOGÍAS DE DETECCIÓN DE INTRUSOS PARA LA RED DE LA S.S.A."

T E S I S
QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

DIANA GARCÍA MARTÍNEZ
MARIO AXEL LUNA SOLÍS

ASESOR DE TESIS:
ING. FILIBERTO MANZO GONZÁLEZ

M É X I C O , D . F . **M A R Z O , 2 0 0 9**

Universidad Nacional Autónoma de México

Facultad de Ingeniería
División de Ingeniería Eléctrica
Departamento de Computación

"Implementación de tecnologías de detección de intrusos para la red de la S.S.A."

México Distrito Federal, Marzo 2009

Índice General

Índice de Figuras

Introducción, Objetivo e Historia

INTRODUCCIÓN

La alta conectividad de las redes de telecomunicaciones en el mundo y en nuestro país hace difícil mantener niveles óptimos de seguridad. En el 2008, el país contaba con 17.85 millones de computadoras, 11.1 millones de ellas con acceso a Internet (Ref. 1).

El diseño de aplicaciones introduce nuevas vulnerabilidades en los sistemas informáticos, habitualmente relacionadas con el tratamiento de parámetros de entrada (valores de formularios, datos de sesión) que permitirán en función de la aplicación ejecución de código arbitrario dentro del sistema operativo o en la aplicación mediante la ejecución de sentencias embebidas en el lenguaje de desarrollo. De igual manera, un tipo determinado de ataques basados en la sobrecarga de búfer pueden ser en algunos casos detectados por sensores de red.

Las vulnerabilidades en aplicaciones producto de un elevado número de ataques, traen consigo el desarrollo de sistemas creados para mantener la integridad de los sistemas informáticos, utilizando mecanismos de alerta con el objetivo de mantener seguras las redes.

En este contexto, la Facultad de Ingeniería de la UNAM implementa mecanismos de seguridad. Su red se encuentra dividida en dominios, entre ellos: minería, posgrado, fi-c, fi-b y fi-a. El dominio fi-a lberga entre otros, la red de la Secretaría de Servicios Académicos (SSA), administrada por la Unidad de Servicios de Cómputo Administrativos (USECAD).

La red de la SSA presta diversos servicios a la comunidad de la Facultad de Ingeniería: aloja servidores de correo, web y bases de datos, además de manejar información estadística de procesos académicos.

Debido a su importancia se cuenta con diversos controles de acceso, entre ellos firewalls. Éstos evitan conexiones no autorizadas hacia un punto específico de la red interna. Resulta congruente para los administradores de la red, adicionar el uso de tecnologías que ayuden a mantener niveles de seguridad aceptables. Un gran avance en la seguridad de redes son los Sistemas Detectores de Intrusos (Intrusion Detection System IDS).

Este documento muestra las bases teóricas necesarias para comprender las comunicaciones y sus necesidades, la seguridad informática y la implementación óptima de un Sistema Detector de Intrusos, así como la administración que permitirá aprovechar al máximo este tipo de recurso.

Para entender un IDS primero debemos saber qué es una intrusión. El Insitito Nacional de Estándares y Tecnologías (NIST, National Institute of Standards and Technology) define la detección de intrusos como el proceso de monitorear eventos y ocurrencias en sistemas de IT (Information Technologies) y analizar sus características. Estas intrusiones son el resultado de ataques en los controles de acceso al sistema provenientes de la red, usuarios no autorizados quienes intentan obtener privilegios adicionales (Ref. 2).

Las intrusiones se definen en relación a una política de seguridad establecida. A menos que se conozca qué está permitido en un sistema y qué no, es innecesario hablar de Detección de Intrusos.

En síntesis, una intrusión es cualquier conjunto de acciones que puede comprometer la integridad, confidencialidad o disponibilidad de información o un recurso informático.

Uno de los puntos que debemos plantearnos a la hora de hablar de IDS es si realmente necesitamos uno de ellos en nuestro entorno de trabajo; a fin de cuentas, debemos tener ya un sistema de protección perimetral basado en firewalls, y por éste fallara, cada sistema habría de estar configurado de una manera correcta, de forma que incluso sin firewall cualquier máquina pudiera seguirse considerando relativamente segura.

En el momento que alguien consiga romper la seguridad de nuestro entorno informático, hemos de ser capaces de detectar ese problema tan pronto como sea posible. Ningún sistema informático puede considerarse completamente seguro, incluso aunque nadie consiga violar nuestras políticas de seguridad.

Existen dos grandes enfoques a la hora de clasificar a los sistemas de detección de intrusos: en función de qué sistemas vigilan, o bien de cómo lo hacen (Ref. 3).

Si elegimos la primera aproximación tenemos dos grupos de sistemas de detección de intrusos: los que analizan actividades de un solo host en busca de posibles ataques, y los que lo hacen de una red (generalmente, de un mismo dominio de colisión).

La detección de intrusos basada en red (NIDS), consiste en colocar un dispositivo que reciba una copia de todo el tráfico de entrada y salida, lo haga pasar por un motor de detección y dependiendo de los resultados de este procesamiento informe de aquel trafico que sea anormal de acuerdo con los criterios especificados en el motor de análisis.

OBJETIVO

El objetivo del presente documento es llevar a cabo el análisis y despliegue de un NIDS basado en Snort (Ref. 4) para fortalecer a la institución en relación de auditoria de controles de acceso y políticas, ayudando a diagnosticar anomalías en el tráfico de la red.

Existen dos factores de éxito para este proyecto: tener clara la misión de la SSA y entender la tecnología de detección de intrusos. El primero consiste en conocer cuales son los procesos dentro de la Secretaría de Servicios Académicos, y qué servicios de seguridad son necesarios. La forma de lograr este objetivo es conociendo las políticas de cómputo, y analizar los posibles ataques en este contexto. El segundo factor, es el relacionado con los Sistemas Detectores de Intrusos basados en red, la forma en la que operan, sus requerimientos, ventajas, desventajas.

HISTORIA

Con la evolución del cómputo y su presencia en procesos corporativos, la seguridad de la información claramente se ha convertido en una preocupación dentro del mundo de la informática; sin embargo no resulta sorprendente que las primeras preocupaciones en este tema surgieran en el ambiente militar. En 1972 la fuerza aérea de los Estados Unidos, publico un trabajo en donde señalaba que la USAF (US Air

Force), estaba consciente de los problemas de seguridad informática debido a que estos aplicaban a casi todos sus aspectos de operación y administración.

En 1980 James P. Andersson (quien también trabajó en la USAF) publico un estudio llamado “How to use accounting audit files to detect unauthorized access”, presentando técnicas para mejorar la auditoria y vigilancia de sistemas. La idea de hacer de la detección una tarea automatizada normalmente es acreditada a James P. Andersson por este trabajo. En éste se describían una serie de pasos que comenzaban con definir las amenazas que existían, después comprender cuales amenazas tenían un mayor grado de probabilidad de ocurrir y como reconocer esos ataques en archivos de auditoría.

Entre 1984 y 1986 Dorothy Denning y Peter Neumann desarrollaron el primer modelo de sistema detector de intrusos en tiempo real. A este prototipo se le llamó IDES (Intrusion Detection Expert System), este trabajaba con un conjunto de reglas orientadas a detectar ataques conocidos.

Los dos trabajos mencionados el de James P. Andersson y el IDES, marcaron el inicio en la investigación en sistemas detectores de intrusos entre los años 1980 y 1990. Durante este periodo uno de los principales interesados en dicha tecnología fue el gobierno de los Estados Unidos.

Fue a mediados de los años 90’s que la tecnología de detección de intrusos fue accesible para las masas, sobresaliendo dos aplicaciones: el NetRanger diseñado por Wheelgroup y RealSecure de Internet Security Systems.

La compañía WheelGroup formada en 1995 para comercializar NetRanger que fue originalmente diseñado para la fuerza aérea de los Estados Unidos y en 1998 adquirida por Cisco. De esta forma NetRanger pasó a formar parte de la arquitectura de seguridad de dicha empresa.

Ambos sistemas detectores de intrusos basaban su funcionamiento en reglas y requerían actualizar las firmas en su base de datos para detectar nuevos ataques.

En diciembre de 1998, Martin Roesch lanzó la primera versión del que se convertiría en el NIDS bajo licencia GPL (Ref. 5) más popular, Snort. Dicho sistema se encontraba disponible solo para sistemas UNIX y tenía capacidades limitadas; sin embargo fue hasta 1999 que Snort despegó como IDS durante el

2000. La versión 1.5 lanzada en diciembre de1999, tenia la capacidad de realizar análisis de paquetes en tiempo real y crear logs. Michael Davis lanzo en junio del 2000 la primera versión de Snort para Windows. En el 2006 Checkpoint compró a Sourcefire, empresa que implementó Snort en sus sistemas; sin embargo, conserva su licencia GPL y se lanzan versiones mejoradas constantemente, siendo la última versión estable, hasta la elaboración de este trabajo, la de abril 11 de 2008.

Se pretende que el desarrollo de la tesis sea en 7 capítulos. En “Introducción” se describe la necesidad de contar con comunicaciones seguras y la evolución del cómputo, la historia y evolución de los IDS`s y los objetivos.

En ”Conceptos y Definiciones” se manejan dos temas fundamentales en este trabajo: Redes de Datos y Seguridad Informática. La primer parte describe la teoría de redes, el modelo de comunicación OSI y algunos de los protocolos principales de la suite TCP/IP. Se definen conceptos de seguridad y servicios de seguridad, la evolución de los ataques y algunos vectores de ataque como gusanos, virus y malware.

En el capítulo titulado “Técnicas de mitigación y defensa en profundidad” se estudian los diferentes controles y técnicas utilizadas para implementar los servicios de seguridad. Se presenta el ciclo coordinado llamado defensa en profundidad, sus beneficios y etapas.

“Sistemas Detectores de Intrusos” muestra las clasificaciones, características, estrategias de detección, capacidades/limitaciones de los IDS. Al final nos enfocamos a la detección de intrusos basada en red y la arquitectura comúnmente empleada para su implementación.

“Implementación del IDS” documenta la metodología empleada. Fue en este punto en donde se determinó en donde colocar el NIDS y se trazó el plan para lograrlo.

Los últimos dos capítulos presentan el “Análisis de Resultados” y “Conclusiones” del presente trabajo. Con esto se finaliza la presentación de esta implementación en beneficio de la Facultad de Ingeniería, en especial de la red de servidores de USECAD.

Capítulo 1

Conceptos y Definiciones

1. CONCEPTOS BÁSICOS DE REDES (Ref. 6)

Llamaremos “redes de computadoras” a un conjunto de computadoras autónomas interconectadas, es decir, que pueden intercambiar información y servicios. En cuanto a la clasificación de las redes no existe alguna que ajuste todas las redes, pero destacan de manera importante la tecnología de transmisión y la escala.

Hay dos tipos de tecnología de transmisión que se utilizan de manera externa: enlaces de difusión y enlaces de punto a punto. Las redes de difusión (broadcast) tienen un solo canal de comunicación, por éste todas las máquinas de la red comparten medio. Los sistemas de difusión permiten el direccionamiento de un paquete a todos los destinos utilizando un código en el campo de dirección. Cuando se transmite un paquete todas las máquinas de la red lo reciben y procesan, este modo de operación se conoce como difusión (broadcasting). Algunos sistemas de difusión también soportan la transmisión a un subconjunto de máquinas, algo conocido como multidifusión (multicasting). En contraste, las redes punto a punto constan de muchas conexiones entre pares individuales de máquinas. Para ir de un origen al destino, un paquete en este tipo de red podría tener que visitar primero una o más máquinas intermedias. La transmisión de punto a punto con un emisor y un receptor se conoce como unidifusión (unicasting).

Un criterio alternativo para la clasificación de las redes es su escala:

- Redes de área personal
- Redes de área local
- Redes de área metropolitana
- Redes de área amplia
- Interred

Las **redes de área personal** (Personal Area Network, PAN) es una red de comunicación entre distintos dispositivos cercanos al punto de acceso (computadoras, teléfonos celulares, PDA, dispositivos de audio, impresoras). Estas redes son de poco alcance.

Las **redes de área local** (Local Area Network, LAN) son redes limitadas a un entorno físico reducido. Su aplicación es la interconexión de computadoras personales y estaciones de trabajo para compartir recursos e intercambiar datos y aplicaciones. Aunque se han producido avances tecnológicos que mejoran la velocidad de las comunicaciones, tales como la Ethernet de 10 Gigabits, de 1 Gigabit y Metro Optical, la distancia sigue siendo un problema.

Una **red de área metropolitana** (Metropolitan Area Network, MAN) representa una evolución del concepto de red de área local a un área más amplia, cubriendo distancias mayores que en algunos casos no se limitan a un entorno metropolitano sino que pueden llegar a una cobertura regional e incluso nacional mediante la interconexión de diferentes redes de área metropolitana.

Una **red de área amplia** (WAN) abarca una gran área geográfica con frecuencia un país o un continente. Contiene un conjunto de máquinas diseñado para programas de usuario. Cuenta con una infraestructura basada en poderosos nodos de conmutación que llevan a cabo la interconexión de dichos elementos, la velocidad a la que circulan los datos por las redes WAN suele ser menor que la que se puede alcanzar en las redes LAN.

Una **interred** se forma cuando se interconectan redes diferentes. Muchas LAN conectadas entre sí permiten que funcione Internet. Pero las LAN tienen sus limitaciones de tamaño.

Con el modelo OSI a modo de ejemplo, el objetivo consiste en construir la funcionalidad de la red en módulos independientes. Esto permite que una variedad de tecnologías LAN existan en las Capas 1 y 2 y una variedad de aplicaciones funcionen en las Capas 5, 6 y 7.

El modelo OSI proporciona un mecanismo en el cual se separan los detalles de las capas inferior y superior. Esto permite que los dispositivos intermedios de networking "retransmitan" el tráfico.

Esto nos lleva al concepto de internetworking o la construcción de redes de redes. Una red de redes recibe el nombre de internet, que se escribe con "i" minúscula. Cuando se hace referencia a las redes desarrolladas por el DoD (Departament of Defense) en las que corre la red mundial (Worldwide Web www), se utiliza la letra "I" mayúscula y recibe el nombre de Internet. Internetworking debe ser escalable respecto del número de redes y computadoras conectadas, capaz de manejar el transporte de datos a lo largo de grandes distancias, tiene que ser flexible para admitir la constante innovación tecnológica y capaz de ajustarse a las condiciones dinámicas de la red. Las internetworks deben estar diseñadas para permitir que en cualquier momento, en cualquier lugar, cualquier persona reciba la comunicación de datos.

1.1 TOPOLOGÍAS DE RED

En las redes de computadoras existen dos tipos de topologías, que juntas hacen posible la comunicación de datos

1.2.1 Topología física

Esta topología se refiere a la disposición de las conexiones entre equipos de la red, es fácil visualizar esta característica en las redes con medios guiados (cable), ya que siguen patrones geométricos, la topología física es relevante en la elección de los medio a utilizar.

Algunas de las topologías físicas existentes son:

- **Bus.** En un bus todas las computadoras y dispositivos de red están conectados a una misma línea. Las implementaciones antiguas utilizaban cable coaxial, en la actualidad el bus se establece en un dispositivo (ej. hub) y se interconectan los nodos con cable UTP (Unshielded Twisted Pair).

  Debido a la dependencia hacia el bus, esta topología presenta baja tolerancia a fallos y falta de confidencialidad, ya que cualquier dispositivo puede monitorear la actividad de los demás dispositivos en la red.

- **Anillo**. Los dispositivos están conectados juntos cada sistema tiene dos conexiones en la red, una para transmisión y otra para recepción. La información fluye por el aro, en un circuito cerrado dado que el ultimo dispositivo esta conectado al primero, las interconexiones pueden realizarse con cable coaxial o fibra. Esta topología comparte las mismas fallas en confidencialidad y resistencia a fallos que la topología de bus.

- **Estrella**. Es la topología mas utilizada en la actualidad. Consiste en la conexión entre dispositivos y computadoras por medio de una entidad central, el uso de esta topología en redes switcheadas resulta en una implementación que provee una gran confidencialidad y resistencia a fallos.

1.2.2 Topología lógica (Acceso al Medio)

La topología lógica es la manera en la que las señales viajan de un origen a un destino. Las reglas que siguen dichas señales se especifican en los protocolos de acceso al medio, algunos ejemplos de topologías lógicas son:

- **Ethernet**. Es la topología lógica o protocolo de capa 2 mas utilizado, realiza transmisión en banda base (medio compartido) y utiliza el protocolo CSMA/CD (Carrier Sentido Múltiple Access / Detección de colisión) para evitar colisiones,

- **Token Ring y FDDI** (Fiber Distributed Data Interface). Token Ring utiliza el modelo de time slicing para transmitir datos, de esta forma cada dispositivo tiene la oportunidad de transmitir en un periodo de tiempo dedicado. La comunicación se lleva a cabo por medio de un token. El equipo receptor verifica si está dirigido a él, en caso contrario lo transmite al siguiente equipo. FDDI agrega un anillo mas para aumentar las expectativas de disponibilidad (redundancia).

1.3 MODELO OSI

El modelo OSI (Open System Interconnection) está basado en una propuesta desarrollada por la ISO (Organización Internacional de Estándares) como un primer paso hacia la estandarización internacional de los protocolos utilizados en varias capas o niveles. Tiene que ver con la conexión de sistemas abiertos a la comunicación con otros sistemas.

Este modelo no es en sí una arquitectura de red, debido a que no especifica los servicios y protocolos exactos que se utilizarán en cada capa. Sólo indica lo que debe hacer cada nivel; sin embargo ISO también ha producido estándares para todas las capas, aunque éstos no son parte del modelo de referencia mismo.

Para lograr comunicación en dos sistemas, ambos deben contar con el mismo modelo de capas. La capa más alta del sistema emisor se comunica con la capa más alta del sistema receptor, pero esta comunicación se realiza vía capas inferiores de cada sistema. La única comunicación directa entre capas de ambos sistemas es en la capa inferior (capa física).

El modelo OSI (Fig. 1) considera 7 capas:

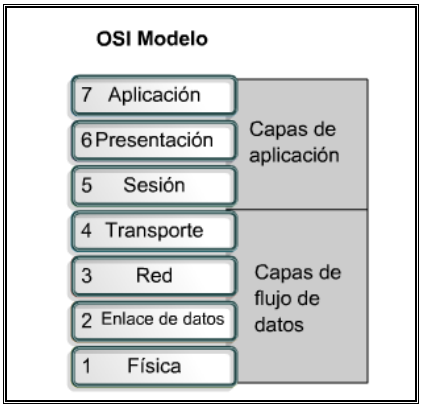


Fig. 1.1 Modelo OSI

1.3.1 Capa 7. Aplicación

Provee servicios y procedimientos para las aplicaciones del usuario. Usualmente el usuario no interactúa directamente con el nivel de aplicación, ocultando la complejidad subyacente. Así por ejemplo, un usuario no manda una petición "HTTP/1.0 GET index.html" para conseguir una página en html, ni lee directamente el código xml.

Entre los protocolos más conocidos se destacan: HTTP (HyperText Transfer Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol), SSH (Secure Shell) y SNMP (Simple Network Management Protocol).

La capa de aplicación realiza las siguientes funciones:

- Identificación del corresponsal mediante la dirección.
- Determinación de la disponibilidad y establecimiento de la autorización.
- Determinación de la metodología de costos de la comunicación y calidad de servicio (errores y costo).

1.3.2 Capa 6. Presentación

Esta capa permite la presentación de la información que las entidades de aplicación mencionan en su comunicación. Se ocupa de la sintaxis (reglas gramaticales para representación de los datos, secuencia y ortografía de los comandos) y no de la semántica (función que cumple cada parte del mensaje, objetivo de la capa 7). Las funciones de esta capa son:

- Transformación y selección de la sintaxis para la capa 7.
- Transferencia de datos, negociación y renegociación de la sintaxis.
- Establecimiento del formato de datos (compresión de código).

1.3.3 Capa 5. Sesión

Establece, gestiona y finaliza las conexiones entre usuarios (procesos o aplicaciones) finales. Ofrece servicios cruciales para la comunicación, como:

- Control de la sesión entre el emisor y el receptor.
- Control de la concurrencia (que dos comunicaciones a la misma operación crítica no se efectúen al mismo tiempo).
- Mantener los puntos de verificación (checkpoints). Esto con el objetivo de que ante una interrupción de transmisión, la misma se pueda reanudar desde el último punto de verificación en lugar de repetirla desde el principio.

Por lo tanto, el servicio provisto por esta capa es la capacidad de asegurar que dada una sesión establecida entre dos máquinas, la misma se pueda efectuar para las operaciones definidas de principio a fin, reanudándolas en caso de interrupción. En muchos casos, los servicios de la capa de sesión son parcial o totalmente prescindibles.

1.3.4 Capa 4. Transporte.

Esta capa optimiza el uso del servicio de la red disponible para ofrecer la calidad de funcionamiento que requiere la capa 5 a un mínimo costo. Son ejemplos los protocolos TCP, UDP, SPX (NetWare). Las funciones son:

- Direccionamiento de la transmisión de datos mediante el concepto de port.
- Multiplexación y división de conexiones (optimiza los costos).
- Detección de errores y comprobación de calidad de servicio. Eventualmente provee la retransmisión.
- Segmentación y concatenación de extremo a extremo.

1.3.5 Capa 3. Red

Asegura la independencia de la capa 4 respecto del encaminamiento en la conexión de red. Son ejemplo el protocolo IP e IPX (Netware). Las funciones son:

- Direccionamiento y conexión en la red de datos.
- Es responsable del ensamble de datos en el servicio no orientado a conexión.
- Obtención de los parámetros de calidad del servicio y notificación de errores.
- Reiniciación, liberación y acuse de recibo de los datos.
- Los protocolos ICMP y ARP se encuentran en esta capa.

Los dispositivos de capa 3 tienen más funciones que sólo las de dividir los dominios de colisión, filtran paquetes basados en la dirección IP (Ref. 7) destino. La única forma en que un paquete se enviará es si su dirección IP destino se encuentra fuera del dominio broadcast y si el router tiene una ubicación identificada para enviar el paquete. Un dispositivo de Capa 3 (router) crea varios dominios de colisión y broadcast (Fig. 2).

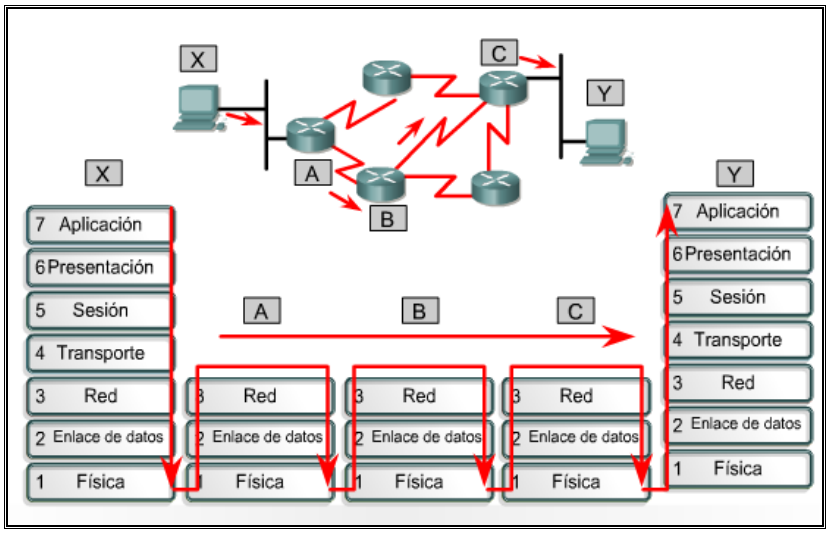


Fig. 1.2 Transmisión de datos entre dos host

- **1.3.5.1 Router**

Los Routers son dispositivos para interconexión de redes que opera en la capa tres (nivel de red). Toman las decisiones complejas permitiendo asegurar el enrutamiento de paquetes entre redes o determinando la ruta que debe tomar el paquete de datos. No todas las redes están conectadas directamente a otra. El router debe contar con alguna metodología para manejar esta situación.

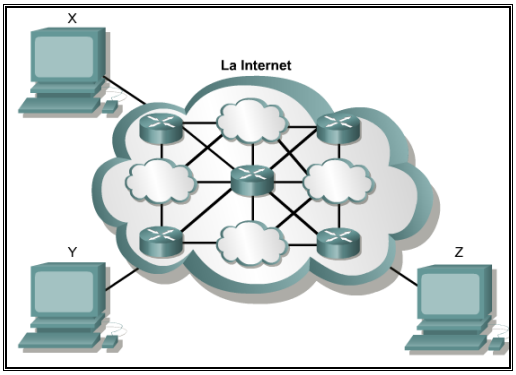


Fig. 1.3 Detalles que se ocultan a los usuarios

1.3.6 Capa 2. Enlace de datos

Esta capa proporciona los medios para establecer, mantener y liberar las conexiones entre los niveles 3 de cada extremo. Las funciones que se pueden identificar son:

- Conexión de enlace de datos con sincronismo de trama.
- Identificar los puntos extremos y control del flujo de datos.
- Notificar errores y los parámetros de calidad del servicio.
- Dispositivos tales como switches, bridges, y NIC forman parte de esta capa.

Los dispositivos de Capa 2 filtran tramas de datos basados en la dirección MAC destino. La trama se envía y dirige a un destino desconocido fuera del dominio de colisión (8i se trata de un broadcast, multicast o unicast). La única vez en que la trama no se envía es cuando el dispositivo de Capa 2 encuentra que el host emisor y el receptor se encuentran en el mismo dominio de colisión.

Los dispositivos de Capa 2 hacen un seguimiento de las direcciones MAC y el segmento en el que se encuentran. Al hacer esto, estos dispositivos pueden controlar el flujo de tráfico en el nivel de Capa 2. Esta función hace que las redes sean más eficientes, al permitir que los datos se transmitan por diferentes segmentos de la LAN al mismo tiempo sin que las tramas colisionen. Al usar puentes y switches, el dominio de colisión se divide efectivamente en partes más pequeñas que se transforman cada una a su vez en un dominio de colisión.

Estos dominios de colisión más pequeños tendrán menos hosts y menos tráfico que el dominio original. Cuanto menor sea la cantidad de hosts en un dominio de colisión, mayores son las probabilidades de que el medio se encuentre disponible. Siempre y cuando el tráfico entre los segmentos puenteados no sea demasiado pesado, una red puenteada funciona bien. De lo contrario, el dispositivo de Capa 2 puede desacelerar las comunicaciones y convertirse en un cuello de botella en sí mismo.

- **1.3.6.1 Bridge**

Es un dispositivo de capa 2 a nivel de software diseñado para conectar dos segmentos LAN. El propósito de un bridge (puente) es filtrar el tráfico de una LAN para que el tráfico local siga siendo local, pero permitiendo la conectividad a otros segmentos para enviar el tráfico dirigido a esos otros segmentos. Se podría pensar en construir una LAN grande en vez de conectar varias LAN mediante puentes, pero:

- Cuando hay una sola LAN, un error en una zona, bloquearía toda la LAN. Cuando se conectan varias LAN con puentes, el error en una LAN no implica el daño en otra.
- Varias LAN pequeñas tienen mayores prestaciones que una grande, sobre todo porque las longitudes de cableado son menores.
- El establecer varias LAN en vez de una sola. Mejora las condiciones de seguridad, ya que hay áreas que deben ser más seguras y así se implementan con una LAN conectada con las otras LAN

- **1.3.6.2 Swtich**

Un switch es un dispositivo de capa 2. La diferencia entre el hub y el switch es que el último toma decisiones basándose en las direcciones MAC y los hubs no toman ninguna decisión. Un switch es la idea aterrizada de un bridge a nivel hardware. Como los switches son capaces de tomar decisiones, hacen que la LAN sea mucho más eficiente. Los switches hacen esto enviando los datos sólo hacia el puerto al que está conectado el host destino apropiado. Por el contrario, el hub envía datos desde todos los puertos, de modo que todos los hosts deban ver y procesar (aceptar o rechazar) todos los datos.

1.3.7 Capa 1. Física.

En esta capa se proporcionan los vínculos necesarios para la conexión al medio de enlace. Las funciones son:

- Conexión física al medio de transmisión.
- Definición de las características mecánicas, eléctricas, funcionales y de procedimiento.
- Identificación del enlace de datos y notificación de condiciones de falla.

Los dispositivos de Capa 1 (hubs) no funcionan como filtros, entonces todo lo que reciben se transmite al segmento siguiente. La trama simplemente se regenera y retemporiza y así vuelve a su calidad de transmisión original. Cualquier segmento conectado por dispositivos de Capa 1 forma parte del mismo dominio, tanto de colisión como de broadcast. Algunos elementos de esta capa son:

- **1.3.7.1 Hub**

El término hub se refiere tradicionalmente a un dispositivo con un solo puerto de "entrada" y un solo puerto de "salida", actualmente el término repetidor multipuerto se utiliza también con frecuencia. En el modelo OSI, los repetidores se clasifican como dispositivos de Capa 1, dado que actúan sólo a nivel de los bits y no tienen en cuenta ningún otro tipo de información.

El propósito de un repetidor es regenerar y retemporizar las señales de red a nivel de los bits para permitir que los bits viajen a mayor distancia a través de los medios.

- **1.3.7.2 Tarjeta de interfaz de red interna (NIC)**

Una NIC (Network Interface Card) o adaptador LAN, provee capacidades de comunicación en red desde y hacia un PC. La NIC se comunica con la red a través de una conexión serial y con la computadora a través de una conexión paralela. Utiliza una petición de interrupción (Interruption Request, IRQ), una dirección de E/S y espacio de memoria superior para funcionar con el sistema operativo. Un valor IRQ es número asignado por medio del cual una computadora puede esperar que un dispositivo específico lo interrumpa cuando dicho dispositivo envía a la computadora señales acerca de su operación.

1.4 MEDIOS DE TRANSMISIÓN

1.4.1 Medios Guiados

El cable es uno de los principales en distancias cortas. Mediante éste los datos se transportan de un dispositivo de red a otro. La elección del tipo de medio resulta importante en el diseño e implementación de las redes y dicha elección dependerá de la topología, tamaño y velocidad.

- **1.4.1.1. UTP**

El cable UTP (Fig. 4 Unshielded Twisted Pair o cable trenzado sin apantallar) es ampliamente utilizado para la conexión de teléfonos y conexión de computadoras. Consiste en al menos un par de hilos de cobre cada uno con una cubierta de plástico y enrollados bajo cierto factor de torsión con el fin de reducir la interferencia electromagnética.

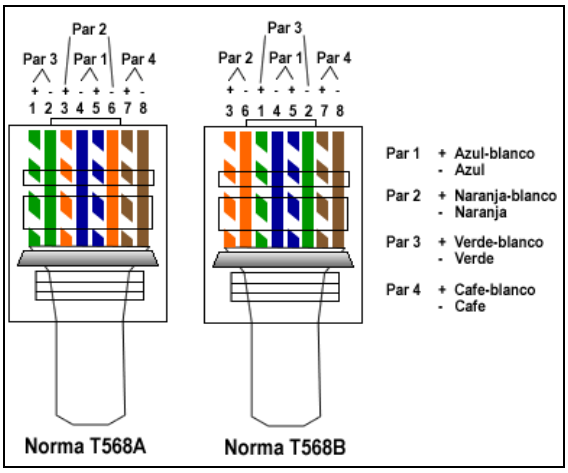


Fig. 1.4 Norma T568A y T568B

El cable UTP esta dividido en categorías de acuerdo con el ancho de banda que puede soportar. A mayor torsión mayor es el efecto de aislamiento soportando un ancho de banda superior. UTP utiliza como interfaz física en sus extremos al conector RJ-45 (Registered Jack).

La tabla 5 muestra un cuadro de categorías UTP, sus principales aplicaciones y la velocidad de transmisión.

Categoría	**Uso**
Categoría 1 y 2	Voz
Categoría 3	Voz/Datos -10Mbps
Categoría 4	Voz/Datos -16Mbps
Categoría 5 y 5e	Voz/Datos -100 Mbps
Categoría 6	Estándar para Gigabit Ethernet

Tabla 1 Categorías UTP y usos principales

- **1.4.1.2 STP**

Shielded Twisted Pair (Par Trenzado Apantallado) tiene la diferencia que cada par tiene una pantalla protectora. Además de tener una lamina externa de aluminio o de cobre trenzado alrededor del conjunto de pares diseñada para reducir y absorber el ruido eléctrico (Fig. 6).

Además de su resistencia al ruido el STP también evita la emanación de señales del cable hacia el exterior, resultando esta característica importante en ambientes en donde se desean niveles altos de seguridad.

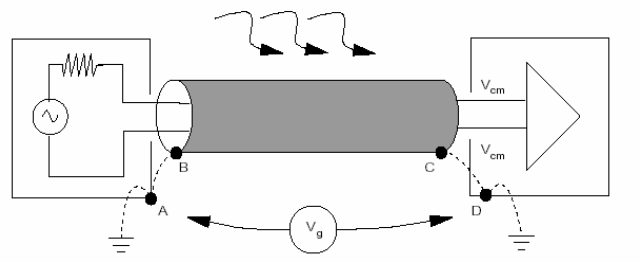


Fig. 1.5 Características cable STP

- **1.4.1.3 FIBRA ÓPTICA**

Una fibra óptica (Fig. 7) es una guía de onda dieléctrica. Un tubo de vidrio muy pequeño en dos capas integrado por un núcleo y un revestimiento. Su principio de funcionamiento se basa en los fenómenos de reflexión y refracción de la luz.

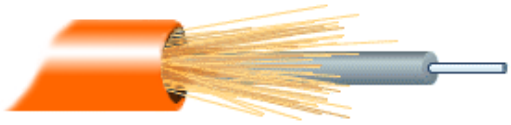


Fig. 1.6 Estructura interna de fibra óptica

Cuenta con muchas ventajas entre las que destacan inmunidad a RFI/EMI, seguridad, es ligera, ocupa poco espacio y tiene mayor capacidad de transmisión de datos. Dependiendo de las trayectorias del haz de luz dentro del cable de fibra óptica, se puede clasificar como multimodo o monomodo.

La fibra multimodo, puede propagar más de un modo de luz, se usan en distancias cortas y normalmente utilizan diodos láser de baja intensidad.

En una fibra monomodo, el núcleo ha sido reducido para permitir un solo modo de propagación. La trayectoria del haz de luz es paralela al eje de la fibra y permite alcanzar mayores distancias y transmisión de datos a mayor velocidad que la fibra multimodo.

1.5 Ethernet y Medios Físicos

La mayor parte del tráfico en Internet se origina y termina en conexiones de Ethernet. Esta tecnología ha evolucionado para satisfacer la creciente demanda LAN de alta velocidad. En el momento en que aparece un nuevo medio, como la fibra óptica, se adapta para sacar ventaja de un ancho de banda superior y de un menor índice de errores que la fibra ofrece. Ahora, el mismo protocolo que transportaba datos a 3 Mbps en 1973 transporta datos a 10 Gbps.

El éxito de Ethernet se debe factores como: sencillez y facilidad de mantenimiento, capacidad para incorporar nuevas tecnologías, confiabilidad y bajo costo de instalación y de actualización.

Con la llegada de Gigabit Ethernet, lo que comenzó como una tecnología LAN ahora se extiende a distancias que hacen de Ethernet un estándar de red de área metropolitana (MAN) y red de área amplia (WAN). La idea original de Ethernet nació del problema de permitir que dos o más host utilizaran el mismo medio y evitar que las señales interfirieran entre sí. Ethernet no es una tecnología para networking, sino una familia de tecnologías para que incluye Legacy, Fast Ethernet y Gigabit Ethernet. Las velocidades de Ethernet pueden ser de 10, 100, 1000 ó 10000 Mbps.

Para permitir el envío local de las tramas en Ethernet, se debe contar con un sistema de direccionamiento, una forma de identificar los computadores y las interfaces de manera exclusiva. Ethernet utiliza direcciones MAC que tienen 48 bits de largo y se expresan como doce dígitos hexadecimales.

Los primeros seis dígitos hexadecimales, que IEEE (Institute of Electrical and Electronics Engineers) administra, identifican al fabricante o al vendedor. Esta porción de la dirección de MAC se conoce como Identificador Exclusivo Organizacional (OUI). Los seis dígitos hexadecimales restantes representan el número de serie de la interfaz u otro valor administrado por el proveedor mismo del equipo. Las direcciones MAC a veces se denominan direcciones grabadas (BIA) ya que estas direcciones se graban en la memoria de sólo lectura (ROM) y se copian en la memoria de acceso aleatorio (RAM) cuando se inicializa la NIC.

En la capa MAC de enlace de datos se agregan encabezados e información final a los datos de la capa superior. El encabezado y la información final contienen información de control destinada a la capa de enlace de datos en el sistema destino. Los datos de las entidades de las capas superiores se encapsulan dentro de la trama de la capa de enlace, entre el encabezado y el cierre, para luego ser enviada sobre la red.

La NIC utiliza la dirección MAC para evaluar si el mensaje se debe pasar o no a las capas superiores del modelo OSI. La NIC realiza esta evaluación sin utilizar tiempo de procesamiento de la CPU permitiendo mejores tiempos de comunicación en una red Ethernet.

En una red Ethernet cuando un dispositivo envía datos, puede abrir una ruta de comunicación hacia el otro dispositivo utilizando la dirección MAC destino.

El dispositivo origen adjunta un encabezado con la dirección MAC del destino y envía los datos a la red. A medida que estos datos viajan a través de los medios de red, la NIC de cada dispositivo de la red verifica si su dirección MAC coincide con la dirección destino física que transporta la trama de datos. Si no hay concordancia la NIC descarta la trama de datos. Cuando los datos llegan al nodo destino la NIC

hace una copia y pasa la trama hacia las capas superiores del modelo OSI. En una red Ethernet todos los nodos deben examinar el encabezado MAC aunque los nodos que están comunicando estén lado a lado.

- **1.5.1 Estructura de la trama Ethernet**

La trama (Fig. 8) es casi idéntica para todas las velocidades de Ethernet desde 10 Mbps hasta 10000 Mbps. Sin embargo, en la capa física casi todas las versiones de Ethernet son diferentes las unas de las otras, teniendo cada velocidad un juego distinto de reglas de diseño.

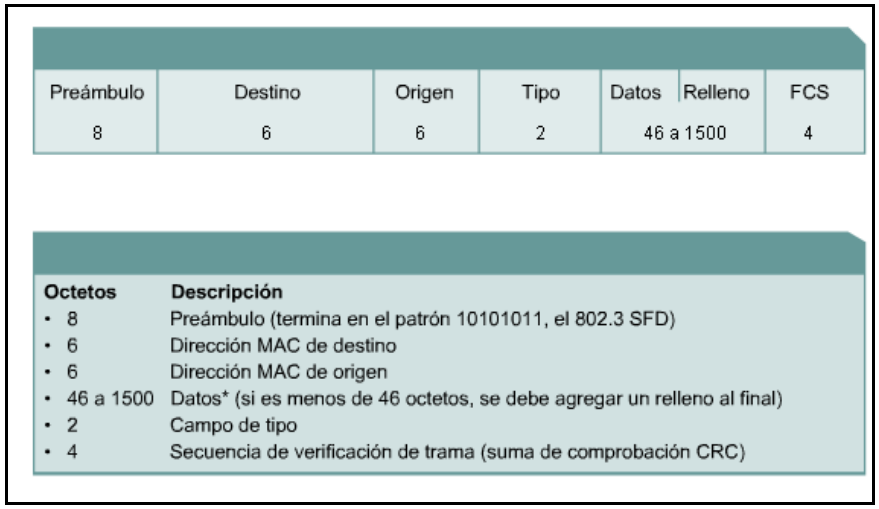


Fig. 1.7 Trama Ethernet

Algunos de los campos que se permiten o requieren en la trama 802.3 de Ethernet son: preámbulo, delimitador de inicio de trama, dirección destino, dirección origen, longitud/tipo, datos y relleno, FCS y extensión

- El *preámbulo* es un patrón alternado de unos y ceros que se utiliza para la sincronización de los tiempos en implementaciones de 10 Mbps y menores de Ethernet. Las versiones más veloces de Ethernet son síncronas y esta información de temporización es redundante pero se retiene por cuestiones de compatibilidad.

- Un d*elimitador de Inicio de trama* es un campo de un octeto que marca el final de la información de temporización y contiene la secuencia de bits 10101011.

- El campo de *dirección destino* contiene la dirección destino MAC. La dirección destino puede ser unicast, multicast o de broadcast.

- El campo de *dirección de origen* contiene la dirección MAC de origen. La dirección origen generalmente es la dirección unicast del nodo de transmisión de Ethernet. Sin embargo, existe un número creciente de protocolos virtuales en uso que utilizan y a veces comparten una dirección MAC origen específica para identificar la entidad virtual.

- El campo *longitud/tipo* admite dos usos diferentes. Si el valor es menor a 1536 decimal 0x600 (hexadecimal), entonces el valor indica la longitud. La interpretación de la longitud se utiliza cuando la capa LLC (Logical Link Control) proporciona la identificación del protocolo. El valor del tipo especifica el protocolo de capa superior que recibe los datos una vez que se ha completado el procesamiento de Ethernet. La longitud indica la cantidad de bytes de datos que sigue este campo.

- Los campos de *datos* y de *relleno* de ser necesario, pueden tener cualquier longitud mientras que la trama no exceda el tamaño máximo permitido. La unidad máxima de transmisión (MTU) para Ethernet es de 1500 octetos, de modo que los datos no deben superar dicho tamaño.

  Se inserta un relleno no especificado inmediatamente después de los datos del usuario cuando no hay suficiente longitud para que la trama cumpla con la extensión mínima especificada. Ethernet requiere que cada trama tenga entre 64 y 1518 octetos de longitud.

- Una *FCS* contiene un valor de verificación CRC (Control de Redundancia Cíclica) de 4 bytes creado por el dispositivo emisor y recalculado por el dispositivo receptor para verificar la existencia de tramas dañadas. Ya que la corrupción de un solo bit en cualquier punto desde el inicio de la dirección destino hasta el extremo del campo de FCS hará que la checksum (suma de verificación) sea diferente, la cobertura de la FCS se auto-incluye.

Ethernet ha evolucionado a las Tecnologías Fast, Gigabit y MultiGigabit que actualmente dominan las nuevas instalaciones de LAN. A tal punto que ha llegado a ser el estándar para las conexiones horizontales, verticales y entre edificios. Las versiones de Ethernet actualmente en desarrollo están borrando la diferencia entre las redes LAN, MAN y WAN.

Las tecnologías de Ethernet de alta velocidad y full-duplex que dominan el mercado están resultando ser suficientes a la hora de admitir aplicaciones intensivas de QoS. Esto hace que las potenciales aplicaciones de Ethernet sean aún más amplias (Tabla 9).

ETHERNET	TASA DE TRANSFERENCIA	TOPOLOGÍA FÍSICA	TIPO DE CABLE
10base2	10Mbps	Bus	Coaxial
10base5	10Mbps	Bus	Coaxial
100baseTX	100Mbps	Estrella	2 pares de cat5 UTP
100baseT4	100Mbps	Estrella	4 pares de cat3 UTP
1000baseSx	1000Mbps	Estrella	1 par de fibra multimodo
1000baseLX	1000Mbps	Estrella	1 par fibra monomodo
1000baseT	1000Mbps	Estrella	4 pares cat5 o superior UTP

Tabla 2 Tecnologías Ethernet

1.6 MODELO TCP/IP

Descripción y configuración

El Protocolo de Control de Transporte/Protocolo Internet (TCP/IP) es un conjunto de protocolos o reglas desarrollados para permitir que las computadoras que operan entre sí puedan compartir recursos a través de una red. El diseño de TCP/IP es ideal para la poderosa y descentralizada red que es Internet. Muchos de los protocolos utilizados hoy en día se diseñaron utilizando el modelo TCP/IP de cuatro capas.

Resulta útil conocer los modelos de networking OSI y TCP/IP. Cada modelo ofrece su propia estructura para explicar cómo funciona una red, pero los dos comparten muchas características.

Todo dispositivo conectado a Internet que necesite comunicarse con otros dispositivos en línea debe tener un identificador exclusivo; el identificador se denomina dirección IP. IPv4, la versión en mayor uso de IP, se diseñó antes de que se produjera una gran demanda de direcciones. El crecimiento explosivo de Internet ha amenazado con agotar el suministro de direcciones IP. La división en subredes, la traducción de direcciones en red (NAT, Network Address Translation) y el direccionamiento privado se utilizan para extender el direccionamiento IP sin agotar el suministro. Otra versión de IP conocida como IPv6 mejora la versión actual proporcionando un espacio de direccionamiento mucho mayor, integrando o eliminando los métodos utilizados para trabajar con los puntos débiles del IPv4.

Además de la dirección física MAC, cada computadora necesita de una dirección IP exclusiva a veces llamada dirección lógica, para formar parte de la Internet. Varios son los métodos para la asignación de una dirección IP a un dispositivo.

Algunos dispositivos siempre cuentan con una dirección estática mientras que otros cuentan con una dirección temporal que se les asigna cada vez que se conectan a la red, el dispositivo puede obtenerla de varias formas.

1.6.1 Direcciones IP y máscaras de red

Las direcciones binarias de 32 bits (IPv4) que se usan en Internet se denominan direcciones de Protocolo Internet (IP). Cuando se asignan direcciones IP, algunos de los bits del lado izquierdo representan una red. La cantidad de bits designados depende de la clase de dirección. Los bits restantes en la dirección IP de 32 bits identifican una computadora de la red en particular (Fig. 10).

La computadora se denomina host. La dirección IP está formada por una parte de red y otra de host que representa a una computadora en particular de una red específica. Para informarle al host cómo se ha dividido la dirección IP de 32 bits, se usa un segundo número de 32 bits denominado máscara de subred. Esta máscara es una guía que indica cómo se debe interpretar la dirección IP al identificar cuántos de los bits se utilizan para identificar la red del computador. La máscara de subred completa los unos desde la parte izquierda de la máscara de forma secuencial. Una máscara de subred siempre estará formada por unos hasta que se identifique la dirección de red y luego estará formada por ceros desde ese punto hasta el extremo derecho de la máscara. Los bits de la máscara de subred que son ceros identifican la computadora o host en esa red.

Clase de dirección	Bits de mayor peso	Intervalo de dirección del primer octeto	Número de bits en la dirección de red	Número de redes	Número de hosts por red
Clase A	0	0-127	8	126	16,777,216
Clase B	10	128-191	16	16,384	65,536
Clase C	110	192-223	24	2,097,152	254
Clase D	1110	224-239	28	No es aplicable	No es aplicable

Tabla 3. Clasificación de Direcciones IP

1.6.2 Historia y futuro de TCP/IP

El Departamento de Defensa de EE.UU. creó el modelo de referencia TCP/IP (Fig. 11) porque necesitaba una red que pudiera sobrevivir ante cualquier circunstancia; requería una transmisión de datos confiable hacia cualquier destino de la red, en cualquier circunstancia.

La creación del modelo TCP/IP ayudó a solucionar este difícil problema de diseño. Desde entonces TCP/IP se ha convertido en el estándar en el que se basa la Internet.

El modelo TCP/IP tiene cuatro capas: la capa de aplicación, la capa de transporte, la capa de Internet y la capa de acceso de red (enlace). Es importante observar que algunas de las capas del modelo TCP/IP poseen el mismo nombre que las capas del modelo OSI. Resulta fundamental no confundir las funciones de las capas de los dos modelos ya que estas desempeñan diferentes funciones en cada modelo.

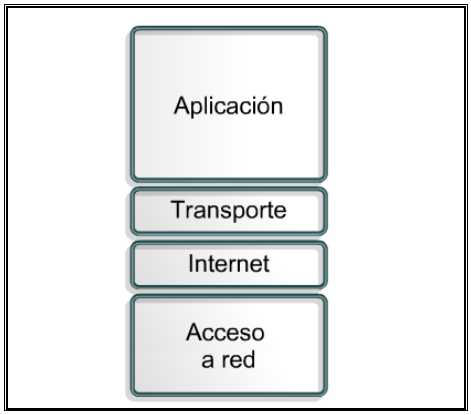


Fig. 1.8 Modelo TCP/IP

- **1.6.3 Capa de aplicación**

La capa de aplicación del modelo TCP/IP maneja protocolos de alto nivel, aspectos de representación, codificación y control de diálogo. El modelo TCP/IP combina todos los aspectos relacionados con las aplicaciones en una sola capa y asegura que estos datos estén correctamente empaquetados antes de que pasen a la capa siguiente. TCP/IP incluye no sólo las especificaciones de Internet y de la capa de

transporte, tales como IP y TCP, sino también las especificaciones para aplicaciones comunes. TCP/IP tiene protocolos que soportan la transferencia de archivos, e-mail, y conexión remota, entre otros tenemos:

- *Protocolo de Transferencia de Archivos* (FTP, File Transfer Protocol)

Es un servicio confiable orientado a conexión que utiliza TCP para transferir archivos entre sistemas que admiten la transferencia FTP. Permite las transferencias bidireccionales de archivos binarios y archivos ASCII.

- *Protocolo Trivial de Transferencia de Archivos* (TFTP, Trivial File Transfer Protocol)

Es un servicio no orientado a conexión que utiliza el Protocolo de datagrama de usuario (UDP). Los Routers utilizan el TFTP para transferir los archivos de configuración e imágenes IOS de Cisco y para transferir archivos entre los sistemas que admiten TFTP. Es útil en algunas LAN porque opera más rápidamente que FTP en un entorno estable.

- *Sistema de Archivos de Red* (NFS, Network File System)

Es un conjunto de protocolos para un sistema de archivos distribuido, desarrollado por Sun Microsystems que permite acceso a los archivos de un dispositivo de almacenamiento remoto, por ejemplo, un disco rígido a través de una red.

- *Protocolo Simple de Transferencia de Correo* (SMTP, Simple Mail Transfer Protocol)

Administra la transmisión de correo electrónico a través de las redes informáticas. No admite la transmisión de datos que no sea en forma de texto simple.

- *Emulación de Terminal* (Telnet, Telecommunication network)

Telnet tiene la capacidad de acceder de forma remota a otra computadora. Permite que el usuario se conecte a un host de Internet y ejecute comandos. El cliente de Telnet recibe el nombre de host local. El servidor de Telnet recibe el nombre de host remoto.

- *Protocolo Simple de Administración de Red* (SNMP, Simple Network Management Protocol)

Es un protocolo que provee una manera de monitorear y controlar los dispositivos de red y de administrar las configuraciones, la recolección de estadísticas, el desempeño y la seguridad.

- *Sistema de Nombres de Dominio* (DNS, Domain Name System)

Es un sistema que se utiliza en Internet para convertir los nombres de los dominios y de sus nodos de red publicados abiertamente en direcciones IP.

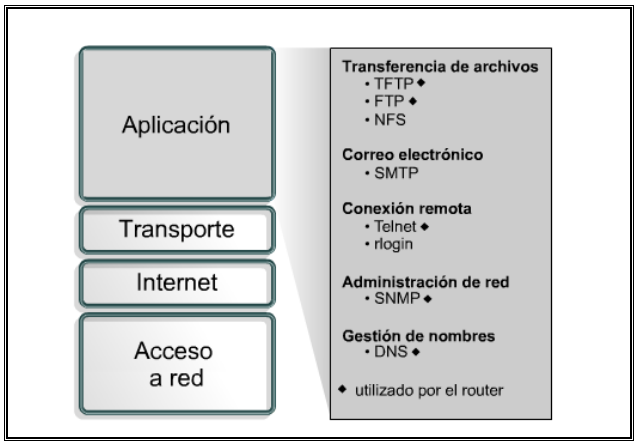


Fig. 1.9 Protocolos de la capa de aplicación

▪ 1.6.4 Capa de transporte

Comprende el trayecto desde el host origen hacia el host destino. Esta capa forma una conexión lógica entre los puntos finales de la red, el host transmisor y el host receptor. Los protocolos de transporte (TCP y UDP) segmentan y reensamblan los datos mandados por las capas superiores en el mismo flujo de datos, o conexión lógica entre los extremos. La corriente de datos de la capa de transporte brinda transporte de extremo a extremo.

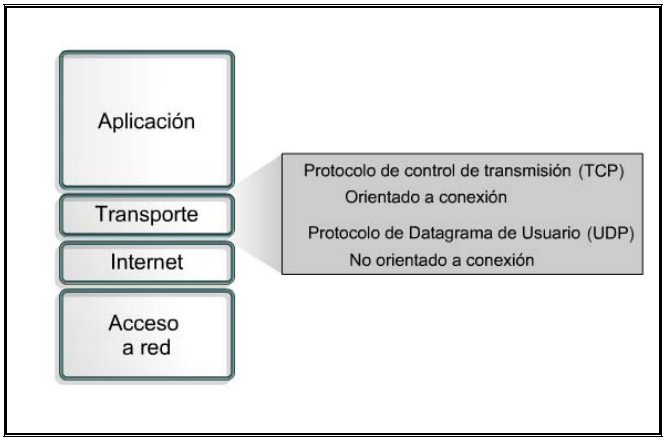


Fig. 1.10 Funciones de la capa de transporte

1.6.4.1 PROTOCOLO TCP (*Transmission Control Protocol, Protocolo de Control de Transmisión*)

TCP se diseñó específicamente para proporcionar un flujo de bytes confiable de extremo a extremo a través de una interred no confiable. Considerando que una interred presenta diferentes topologías, anchos de banda, retardos, tamaños de paquete y otros parámetros. TCP tiene un diseño que se adapta de manera dinámica a las propiedades de la interred. Se definió formalmente en el RFC 793, siendo en el RFC 1122 donde se detallan más especificaciones y algunos arreglos de errores.

El servicio TCP se obtiene al hacer que el servidor y el cliente creen puntos terminales llamados sockets. Cada socket tiene un número (dirección) que consiste en la dirección IP del host y un número de 16 bits llamado puerto. Los números de puerto menores a 1024 se llaman puertos bien conocidos y se reservan para servicios estándar (Ref. 8). Todas las conexiones TCP son dúplex total y de punto a punto, esto significa que el tráfico puede ir en ambas direcciones al mismo tiempo y que cada conexión tiene exactamente dos puntos finales.

Una característica clave de TCP es que cada byte de una conexión tiene su propio número de secuencia de 32 bits que se utiliza para confirmaciones de recepción y para el mecanismo de ventana. El protocolo básico usado por las entidades TCP es el protocolo de ventanas deslizantes. Cuando un transmisor envía un segmento también inicia un temporizador. Cuando llega el segmento al destino, la entidad TCP receptora devuelve un segmento que contiene un número de confirmación de recepción igual al siguiente número de secuencia que espera recibir. Si el temporizador del emisor expira antes de la recepción de la confirmación el emisor envía de nuevo el segmento.

1.6.4.1.2 Encabezado TCP

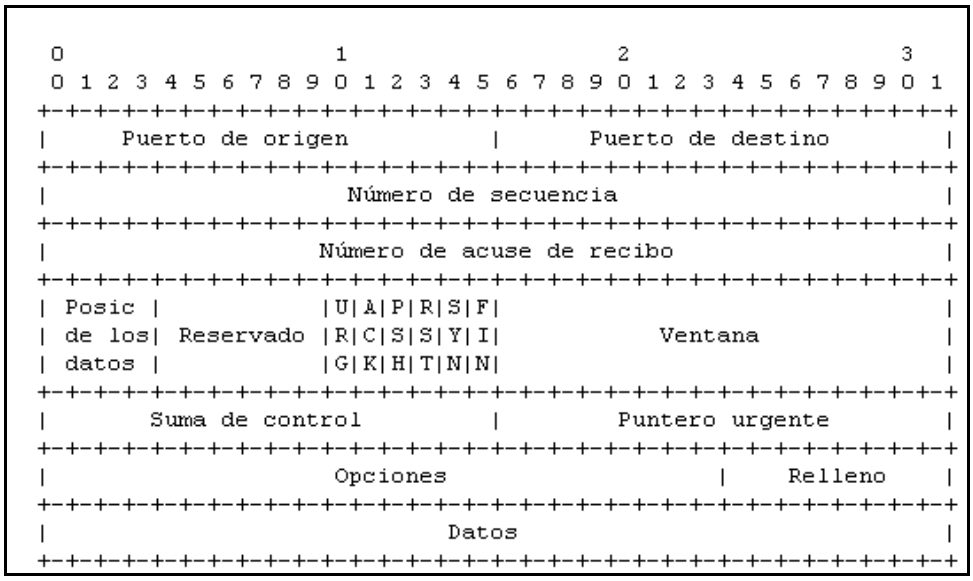


Fig. 1.11 Encabezado TCP

Los campos de *puerto de origen* y *puerto destino* identifican los puntos terminales locales de la conexión. Los puertos bien conocidos se especifican en el apéndice. La dirección de un puerto más la dirección IP de su host forman un punto terminal único de 48 bits. Los puntos terminales de origen y destino en conjunto identifican la conexión.

Los campos *número de secuencia* y *número de confirmación de recepción* desempeñan sus funciones normales siendo este último el que especifica el siguiente byte esperado, no el último byte correctamente recibido. Ambos tienen 32 bits de longitud porque cada byte de datos está numerado en un flujo TCP.

La *longitud* del encabezado TCP indica la cantidad de palabras de 32 bits contenidas en el encabezado TCP. Técnicamente este campo indica el comienzo de los datos en el segmento.

A continuación viene un campo de 6 bits que no se usa; protocolos inferiores lo habrían utilizado para corregir errores de diseño original.

Continúan seis indicadores de 1 bit que se activan en 1 para señalarse activo. *URG* el apuntador urgente sirve para indicar un desplazamiento en bytes a partir del momento actual de secuencia en el que se encuentran datos urgentes. *ACK* indica que el número de confirmación de recepción es válido, si ACK es 0 el segmento no contiene una confirmación de recepción, por lo que se ignora el campo de número de confirmación de recepción. *PSH* indica datos que se deben transmitir de inmediato, por este medio se solicita al receptor que los datos se entreguen a la aplicación a su llegada y no los almacene en búfer hasta la recepción de un búfer completo. *RST* se usa para restablecer una conexión debido a una caída de host u otra razón, también sirve para rechazar un segmento no válido o un intento de abrir una conexión.

El bit *SYN* se usa para establecer conexiones. En esencia *SYN* se usa para denotar CONNECTION REQUEST y CONNECTION ACCEPTED, el bit ACK sirve para distinguir entre ambas posibilidades.

El bit *FIN* se usa para liberar una conexión. Especifica que el emisor no tiene más datos que transmitir.; sin embargo, tras cerrar una conexión, un proceso puede continuar recibiendo datos indefinidamente. Los segmentos SYN y FIN tienen números de secuencia y por tanto la garantía de procesarse en el orden correcto.

El control de flujo en TCP se maneja usando ventanas deslizantes de tamaño variable. El campo *tamaño de ventana* indica la cantidad de bytes que pueden enviarse comenzando por el byte cuya recepción se ha confirmado.

También se proporciona una *suma de verificación* para agregar confiabilidad. Es una suma del encabezado, los datos y el pseudoencabezado. Al realizar este cálculo, se establece el campo de suma de verificación en cero y se rellena el campo de datos con un byte cero adicional si la longitud es un número impar. El pseudoencabezado contiene las direcciones IP de 32 bits de las máquinas de origen y destino, el número de protocolo de TCP y la cuenta de bytes del segmento TCP. La inclusión del pseudoencabezado en el cálculo de la suma de verificación ayuda a detectar paquetes mal entregados.

El campo *opciones* ofrece una forma de agregar características extra no cubiertas por el encabezado normal.

- **1.6.4.1.3 Establecimiento de una conexión TCP**

En TCP las conexiones se establecen mediante el *three-way handshake* (acuerdo de tres vías Fig. 15). Para establecer una conexión el servidor espera pasivamente una solicitud entrante ejecutando las primitivas *LISTEN* y *ACCEPT*; del otro lado el cliente ejecuta una primitiva *CONNECT* especificando la dirección y el puerto IP con el que se desea conectar y el tamaño máximo de segmento TCP que está dispuesto a aceptar. La primitiva *CONNECT* envía un segmento TCP con el bit *SYN* encendido y el bit *ACK* apagado y espera una respuesta.

Al llegar el segmento al destino, la entidad TCP notifica si existe algún proceso que haya ejecutado un *LISTEN* en el puerto indicado. Si no lo hay envía una respuesta con el bit *RST* encendido para rechazar la conexión.

Para liberar una conexión, cualquiera de las partes puede enviar un segmento TCP con el bit *FIN* establecido, lo que significa que no tiene más datos por transmitir. Normalmente se requieren cuatro segmentos TCP para liberar una conexión, un *FIN* y un *ACK* para cada sentido.

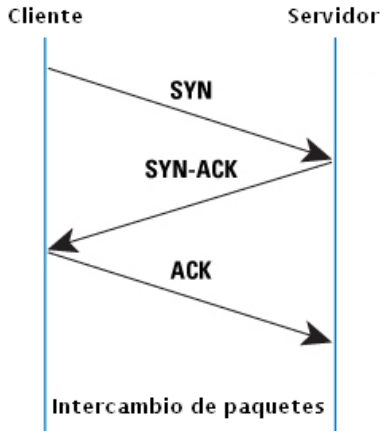


Fig. 1.12 Tree-way handshake

- **1.6.4.1.4 Administración de conexiones TCP**

Los pasos necesarios para establecer y liberar conexiones pueden representarse en una máquina de estados finitos como se observa en la figura. En cada estado son legales ciertos eventos. Al ocurrir un evento, debe iniciarse alguna acción; de lo contrario, si suceden otros eventos, se informa de un error.

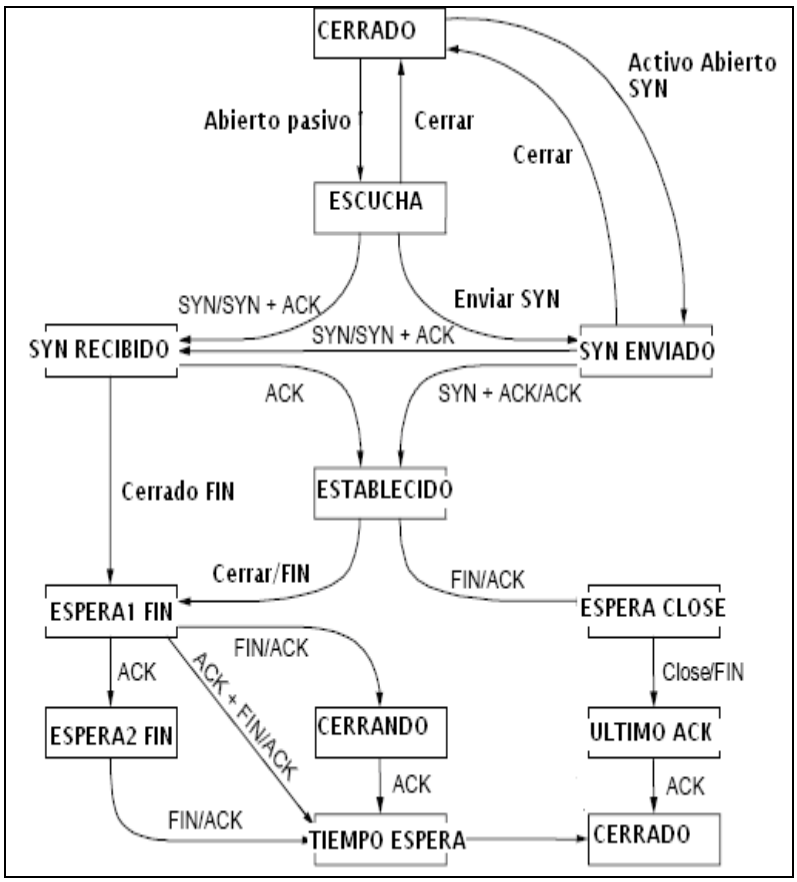


Fig. 1.13 Diagrama de Estados de una conexión TCP

Cada conexión comienza en el estado *CLOSED* (cerrado) y deja ese estado cuando hace alguna apertura pasiva (*LISTEN*), o una apertura activa (*CONNECT*). Si el otro extremo realiza la acción opuesta, se establece una conexión y el estado se vuelve *ESTABLISHED* (establecido). La liberación de la conexión puede iniciarse desde cualquiera de los dos lados. Al completarse, el estado regresa a *CLOSED* (cerrado).

1.6.4.2 PROTOCOLO UDP (User Datagram Protocol, Protocolo de Datagramas de usuario)

El conjunto de protocolos de Internet soporta un protocolo de transporte no orientado a conexión, UDP. Este protocolo proporciona una forma para que las aplicaciones envíen datagramas IP encapsulados sin tener que establecer una conexión. Sse describe en el RFC 768.

UDP transmite segmentos que consiste en un encabezado de 8 bytes seguido por la carga útil. Los dos puertos sirven para identificar los puntos terminales dentro de las máquinas de origen y destino. Cuando llega un paquete UDP, su carga útil se entrega al proceso que está enlazado al puerto de destino mediante la primitiva BIND. El valor de contar con UDP en lugar de simplemente utilizar IP es la adición de los puertos de origen y destino.

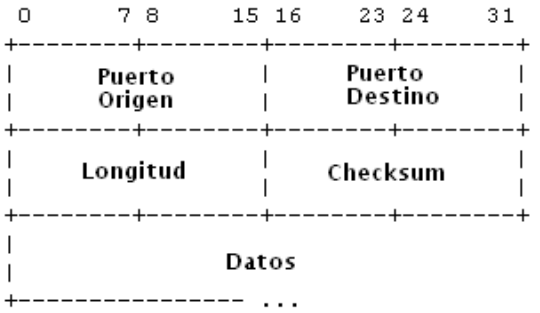


Fig. 1.14 Formato cabecera UDP

UDP es usado normalmente para aplicaciones de streaming (audio, video, etc) donde la llegada a tiempo de los paquetes es más importante que la fiabilidad, o para aplicaciones simples de tipo petición/respuesta como el servicio DNS donde la sobrecarga de las cabeceras que aportan la fiabilidad es desproporcionada para el tamaño de los paquetes.

El puerto de *origen* se necesita principalmente cuando debe enviarse una respuesta al origen para especificar cuál proceso de la máquina emisora va a obtenerlo.

El campo *longitud* incluye el encabezado de 8 bytes y los datos. El campo *suma de verificación* es opcional y se almacena como 0 si no se calcula.

UDP no realiza control de flujo, control de errores o retransmisión cuando se recibe un segmento erróneo. Lo que sí realiza es proporcionar una interfaz al protocolo IP con la característica agregada de desmultiplexar varios procesos utilizando los puertos. Una aplicación que utiliza UDP es DNS (Sistema de Nombres de Dominio)

TCP y UDP: son usados para dar servicio a una serie de aplicaciones de alto nivel. Las aplicaciones con una dirección de red dada son distinguibles entre sí por su número de puerto TCP o UDP. Por convención, los puertos bien conocidos (well-known ports) son asociados con aplicaciones específicas.

- **1.6.5 Capa de Internet**

El propósito de la capa de Internet es seleccionar la mejor ruta para enviar paquetes por la red. El protocolo principal que funciona en esta capa es el Protocolo de Internet (IP). La determinación de la mejor ruta y la conmutación de los paquetes ocurren en esta capa.

Los siguientes protocolos operan en la capa de Internet TCP/IP:

- IP proporciona un enrutamiento de paquetes no orientado a conexión de máximo esfuerzo, no se ve afectado por el contenido de los paquetes, sino que busca una ruta de hacia el destino.
- El Protocolo de mensajes de control en Internet (ICMP) suministra capacidades de control y envío de mensajes.
- El Protocolo de resolución de direcciones (ARP) determina la dirección de la capa de enlace de datos, la dirección MAC, para las direcciones IP conocidas.
- El Protocolo de resolución inversa de direcciones (RARP) determina las direcciones IP cuando se conoce la dirección MAC.

1.6.5.1 PROTOCOLO IP

El Protocolo de Internet (IP, Internet Protocol Fig. 18) es un protocolo no orientado a conexión utilizado para la comunicación de datos a través de una red de paquetes conmutados. Los datos en una red basada en IP son enviados en bloques conocidos como paquetes o datagramas. Un datagrama IP consiste en una parte de encabezado y una parte de texto. El encabezado tiene una parte fija de 20 bytes y una parte opcional de longitud variable. El formato del encabezado se muestra a continuación.

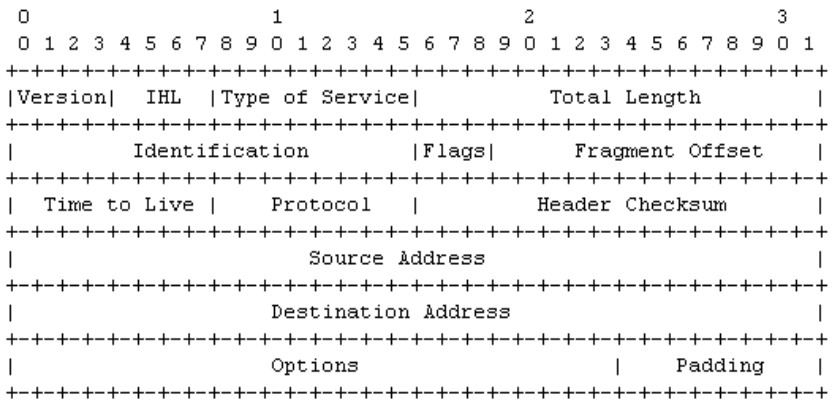


Fig. 1.15 Cabecera IP

Se transmite en orden de izquierda a derecha comenzando por el bit de mayor orden del campo de *versión* que lleva el registro del protocolo al que pertenece el datagrama. Dado que la longitud del encabezado no es constante se incluye un campo en el encabezado, *IHL* para indicar la longitud en palabras de 32 bits.

El campo de *tipo de servicio* tiene como propósito distinguir las diferentes clases de servicios. Son posibles varias combinaciones de confiabilidad y velocidad.

Originalmente el campo de 6 bits contenía un campo de precedencia de tres bits y tres banderas *D, T* y *R*. Los tres bits permiten al host especificar que es lo que le interesa del grupo {retardo (delay), velocidad real de transporte (throughput), confiabilidad (reliability)}.

La *longitud total* incluye todo el datagrama, tanto el encabezado como los datos. La longitud máxima es de 65,535 bytes.

El campo de *identificación* es necesario para que el host de destino determine a qué datagrama pertenece un fragmento recién llegado. Todos los fragmentos de un datagrama contienen el mismo valor de identificación. Le sigue un bit sin uso y luego dos campos de un bit. DF (Don't Fragment) significa no fragmentar, y MF significa más fragmentos. Todos los fragmentos excepto el último tienen establecido este bit, que es necesario para saber cuándo han llegado todos los fragmentos de un datagrama. El desplazamiento del fragmento indica en qué parte del datagrama actual va este fragmento.

El campo de *tiempo de vida* es un contador que sirve par limitar la vida de un paquete. Se supone que cuenta el tiempo en segundos, permitiendo una vida máxima de 255 segundos. Cuando el contador llega a cero, el paquete se descarta y se envía de regreso un paquete de aviso al host origen. Esta característica evita que los datagramas vaguen eternamente, algo que de otra manera podría ocurrir si se llegan a corromper las tablas de enrutamiento.

El campo *protocolo* indica el protocolo de las capas superiores al que debe entregarse el paquete (TCP, UDP).

La *suma de verificación* del encabezado verifica solamente el encabezado. Es útil para la detección de errores generados por palabras de memoria erróneas en un enrutador.

La *dirección de origen y destino* indican el número de red y el número de host. El campo de *opciones* se diseñó para proporcionar un recurso que permitiera que las versiones subsiguientes del protocolo incluyeran información no presente en el diseño original, es decir, permite la innovación. Originalmente se definieron cinco opciones, pero se han agregado otras más. La lista completa se mantiene en línea (Ref. 9).

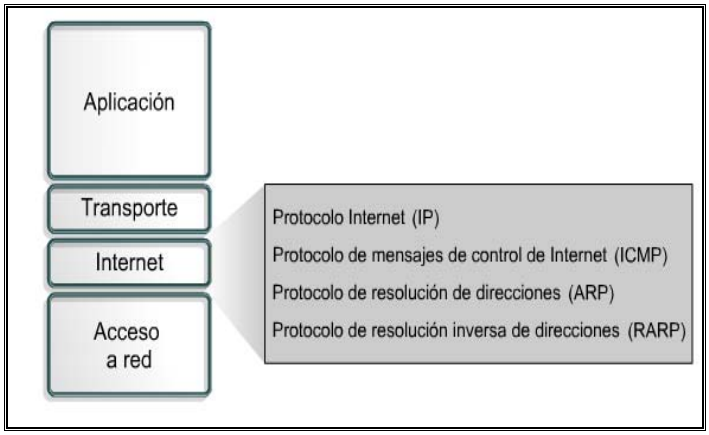


Fig. **1.16** Protocolos de la capa de Internet

1.6.6 Capa de acceso de red

La capa de acceso de red también se denomina capa de host a red. La capa de acceso de red es la capa que maneja todos los aspectos que un paquete IP requiere para efectuar un enlace físico real con los medios de la red. Esta capa incluye los detalles de la tecnología LAN y WAN y todos los detalles de las capas físicas y de enlace de datos del modelo OSI.

Los controladores para las aplicaciones de software, las tarjetas de módem y otros dispositivos operan en la capa de acceso de red. ´

La capa de acceso de red define los procedimientos para realizar la interfaz con el hardware de red y para tener acceso al medio de transmisión. Los estándares del protocolo de módems tales como el Protocolo Internet de enlace serial (SLIP) y el Protocolo de punta a punta (PPP) brindan acceso a la red. Debido a un intrincado juego entre las especificaciones del hardware, el software y los medios de transmisión, existen muchos protocolos que operan en esta capa. La mayoría de los protocolos reconocibles operan en las capas de transporte y de Internet del modelo TCP/IP.

Las funciones de la capa de acceso de red incluyen la asignación de direcciones IP a las direcciones físicas y el encapsulamiento de los paquetes IP en tramas. Basándose en el tipo de hardware y la interfaz de la red, la capa de acceso de red definirá la conexión con los medios físicos de la misma.

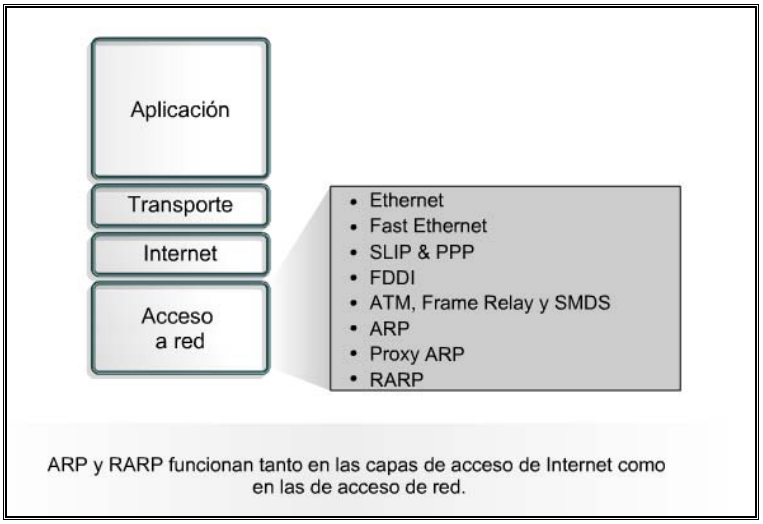


Fig. 1.17 Protocolos de acceso de Red

1.7 Comparación entre OSI y TCP/IP

La siguiente es una comparación de los modelos OSI y TCP/IP analizando sus correspondientes similitudes y diferencias

Similitudes entre los modelos OSI y TCP/IP:

- Ambos se dividen en capas.
- Ambos tienen capas de aplicación, aunque incluyen servicios muy distintos.
- Ambos tienen capas de transporte y de red similares.
- Se supone que la tecnología es de conmutación por paquetes y no de conmutación por circuito.
- Los profesionales de networking deben conocer ambos modelos.

Diferencias entre los modelos OSI y TCP/IP:

- TCP/IP combina las capas de presentación y de sesión en una capa de aplicación
- TCP/IP combina la capas de enlace de datos y la capa física del modelo OSI en una sola capa
- TCP/IP parece ser más simple porque tiene menos capas
- La capa de transporte TCP/IP que utiliza UDP no siempre garantiza la entrega confiable de los paquetes mientras que la capa de transporte del modelo OSI sí.

Internet se desarrolla de acuerdo con los estándares de los protocolos TCP/IP. El modelo TCP/IP gana credibilidad gracias a sus protocolos. En general, las redes no se construyen en base del modelo OSI. El modelo OSI se utiliza como guía para comprender el proceso de comunicación.

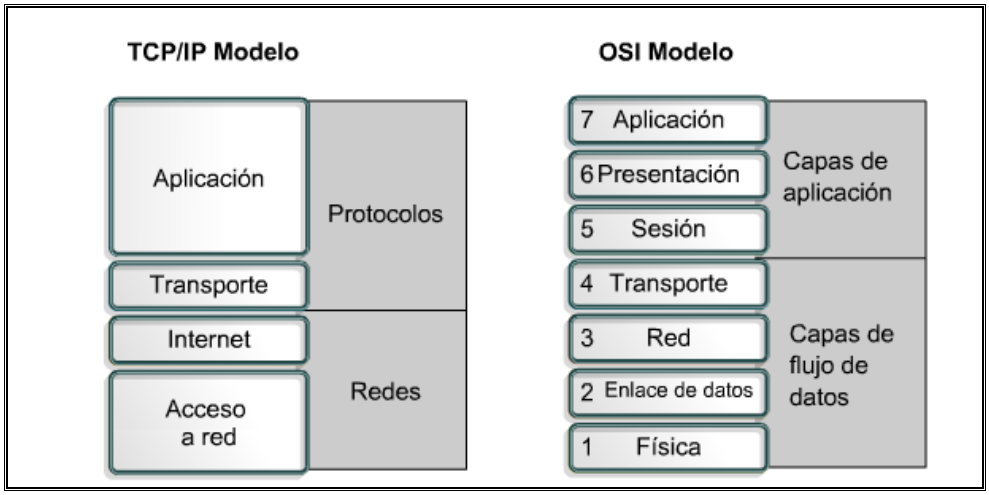


Fig. 1.18 Comparación entre TCP/IP y OSI

1.8 SEGURIDAD DE LA INFORMACIÓN

La Organización Internacional de Estándares publicó en el año 1995 el estándar ISO 17799. Básicamente es un conjunto de controles que incluyen las mejores prácticas en seguridad de la información. Es un estándar genérico de seguridad reconocido internacionalmente. Su intención es servir como punto de referencia para identificar los controles necesarios en la mayoría de las situaciones en que los sistemas de información se ven involucrados; en él se define la seguridad de la información de la forma:

> *"La seguridad de la información se define aquí como la preservación de las siguientes características:*
> *a) Confidencialidad: se garantiza que la información sea accesible sólo a aquellas personas autorizadas a tener acceso a ella.*
> *b) Integridad: se salvaguarda la exactitud y totalidad de la información y los métodos de procesamiento.*
> *c) Disponibilidad: se garantiza que los usuarios autorizados tengan acceso a la información y a los recursos relacionados con ella toda vez que se requiera (Ref. 10)".*

Otros estándares como el 7498-2 y el ITU-T X.800 hacen uso del concepto de servicio de seguridad para describir la arquitectura de seguridad OSI. Un servicio de seguridad es una característica que debe tener un sistema para satisfacer una política de seguridad (Ref.11).

La arquitectura de seguridad OSI identifica cinco clases de servicios de seguridad: confidencialidad, autenticación, integridad, control de acceso y no repudio.

Resulta claro que la definición de seguridad de la información gira en torno a la idea de un servicio o característica que la información debe tener de acuerdo a su contexto.

En 2005 ISO publicó como estándar el ISO 27001, al tiempo que se revisó y actualizó ISO17799. Esta última norma se renombra como ISO 27002:2005 el 1 de Julio de 2007, manteniendo el contenido así como el año de publicación formal de la revisión.

La serie de normas ISO/IEC 27000 (Fig. 22, Ref. 12) son estándares de seguridad publicados por la Organización Internacional para la Estandarización (ISO) y la Comisión Electrotécnica Internacional (IEC). Contiene las mejores prácticas recomendadas en Seguridad de la información para desarrollar, implementar y mantener Especificaciones para los Sistemas de Gestión de la Seguridad de la Información (SGSI)

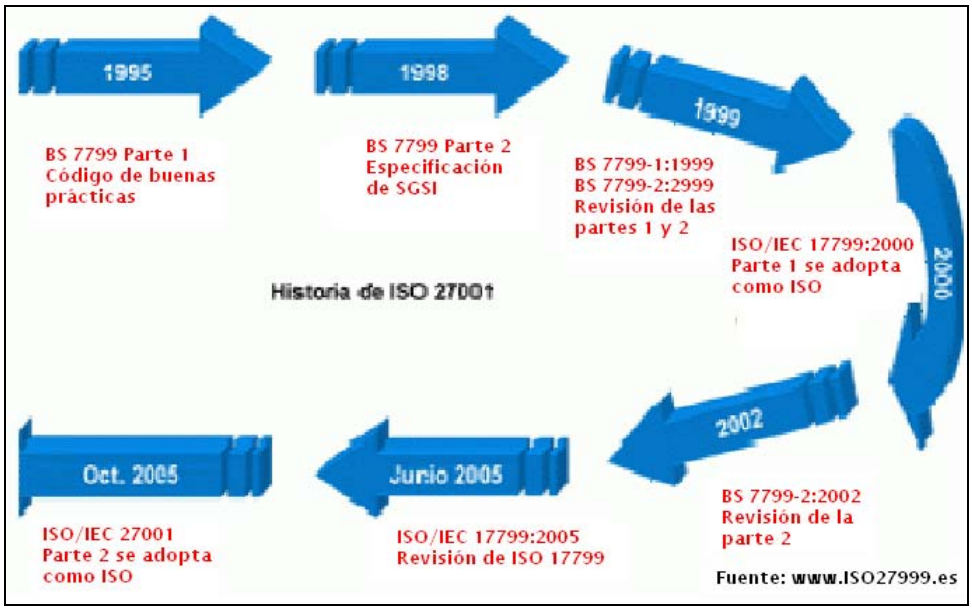


Fig. 1.19 Evolución del estándar 17799

En Marzo de 2006, posteriormente a la publicación de la ISO27001:2005, BSI publicó la BS7799-3:2006, centrada en la gestión del riesgo de los sistemas de información

A continuación se describen brevemente los servicios de seguridad establecidos previamente.

1.8.1 Confidencialidad.

Consiste en garantizar que la información solo pueda ser accedida por las partes autorizadas para ello, por nadie más. Éste servicio y su implementación constituye uno de los principales objetivos de la seguridad informática. Actualmente, una de las técnicas más importantes para implementar este servicio es la criptografía.

Integridad.

Este servicio protege a los activos del sistema contra modificaciones, alteraciones, borrado, inserción y, en general, contra todo tipo de acción que atente contra los activos. Estrictamente hablando, la integridad como tal no se puede garantizar; lo que se puede garantizar es que si la información sufre alteración, ésta pueda ser detectada.

Autenticación.

Este servicio consiste en garantizar que las entidades participantes en una comunicación sean las que dicen ser. Es el proceso de identificación de una parte ante las demás.

Control de acceso

Protege a los activos del sistema contra accesos y uso no autorizados. Para su implementación existe un gran número de técnicas y control de acceso; ya que solo usuarios autorizados pueden tener acceso a la información y a los recursos relacionados cada vez que se requieran.

No repudio.

Proporciona protección contra la posibilidad de que alguna de las partes involucradas en una comunicación niegue haber enviado o recibido un mensaje o acción.

1.8.2 Conceptos de Seguridad

Para comprender mejor la seguridad de la información es necesario definir otros conceptos. En primer lugar se entenderá como ***sistema de cómputo*** al conjunto formado por la colección de equipos, programas, medios de almacenamiento, datos o información. De la misma forma entenderemos como ***compromiso de seguridad*** a cualquier forma posible de pérdida o daño en un sistema de cómputo. Por lo tanto, comprometer la seguridad de un sistema equivale a la posibilidad de provocar pérdida o daño al sistema.

Una ***vulnerabilidad*** consiste en cualquier debilidad que pueda explotarse para causar pérdida o daño a un sistema.

Una ***amenaza*** es cualquier circunstancia con el potencial de causar pérdida o daño a un sistema existiendo 4 tipos principales de amenazas: interrupción, intercepción, modificación y fabricación. En el caso de una ***interrupción***, un activo del sistema se pierde, se hace no disponible o inutilizable. Una ***intercepción*** significa que alguna parte no autorizada logre acceso a un activo del sistema. Cuando una parte no autorizada logra acceso al sistema y puede manipular este activo, se trata de una ***modificación.*** Por último, una parte no autorizada puede ***fabricar*** objetos falsos en un sistema.

Un ***ataque*** se define como cualquier acción que explota una vulnerabilidad, se reconocen dos categorías fundamentales ataques pasivos y activos. Un ataque pasivo consiste en sólo observar comportamiento o leer información sin alterar la información ni el estado del sistema. Un ataque activo, por el contrario, tiene la capacidad de modificar o afectar la información, el estado del sistema o ambos.

A la entidad que realiza un ataque se le conoce como ***atacante o intruso***. Dicha entidad no necesariamente tiene que ser una persona, puede ser cualquier proceso, computadora, dispositivo, etc.

En todos los casos, el objetivo de un atacante siempre es aprovechar en su favor la información, medio o recursos de cómputo.

La información es definida por la RAE (Real Academia Española) como “Conocimientos así comunicados o adquiridos (Ref.13)”. La característica de intangibilidad de la información la hace un activo difícil de evaluar, sin embargo en un contexto informático es claro que es la información la que de acuerdo con su etimología, da forma a los procesos y debido a ello es que debe protegerse

1.8.3 Panorama de la Evolución de la Seguridad

Al principio cuando las computadoras empezaron a utilizarse, estas consistían en grandes mainframes administrados por gobiernos o grandes universidades. La seguridad no era un gran problema dado que los participantes en el uso de tecnologías de cómputo eran pocos y estaban perfectamente identificados.

Las técnicas empleadas en esos tiempos consistían principalmente en seguridad física y su uso era la investigación. Resulta lógico pensar que los diseños de sistemas operativos y protocolos de comunicación no tenían como su objetivo principal la seguridad sino la funcionalidad.

En 1981 IBM introduce la primera computadora personal y en poco tiempo la tecnología de procesamiento de datos es accesible para un mayor número de personas y para tareas diversas. El diseño de estos primeros dispositivos tampoco tenia a la seguridad como prioridad, ejemplo claro fue el año de 1988 cuando Robert Morris Jr. (Ref. 14), escribió el primer programa auto replicable que se propago rápidamente en Internet.

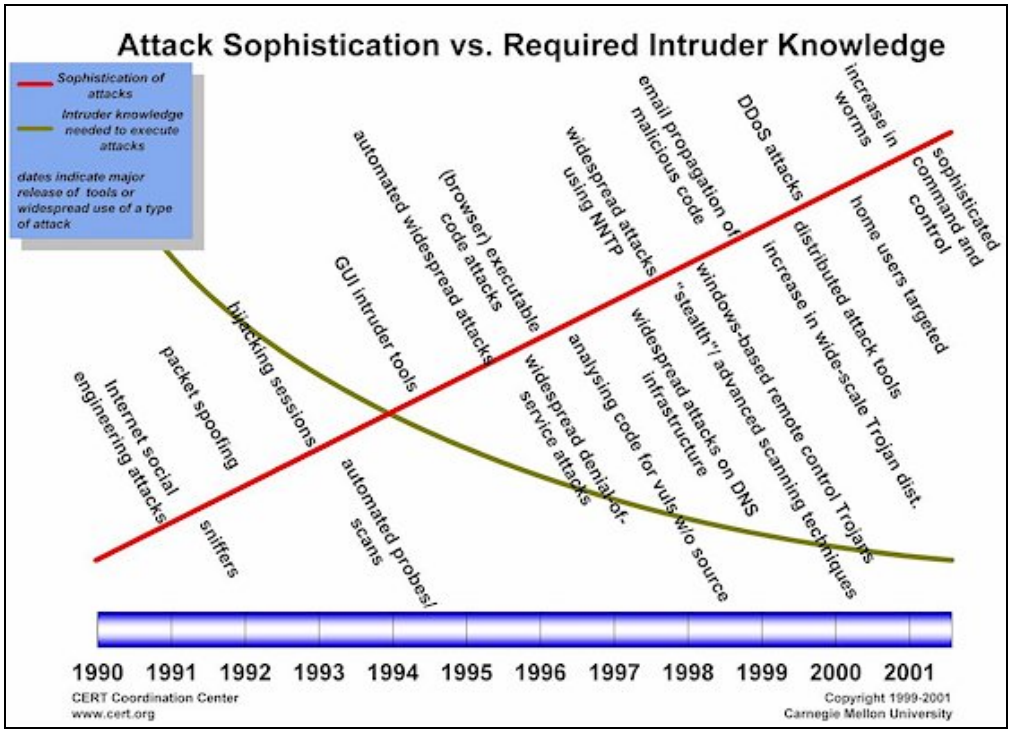


Fig. 1.20 Evolución de los ataques
(http://www.cert.org/congressional_testimony/cert-internet-trends2c.jpg)

Internet proporciona la infraestructura para la comunicación y el intercambio de información. Utiliza la suite de protocolos TCP/IP cuya filosofía es el procesamiento distribuido y la comunicación abierta, hecho del cual se derivan muchas de sus vulnerabilidades ya que el canal es promiscuo y los mensajes pueden ser interceptados por un tercero y la mayoría del tráfico viaja en claro.

Internet en los últimos años ha crecido a un ritmo acelerado (Fig. 23), debido a esto el acceso es prácticamente anónimo. Una etapa que representó una revolución y que ha cambiando nuestra relación con el cómputo, es la aparición de tecnologías inalámbricas y cómputo móvil. El número de dispositivos y técnicas para irrumpir en un sistema tiene la tendencia a aumentar, mientras la dificultad y conocimientos necesarios para llevar a cabo un ataque disminuye.

Mientras que en un principio las intrusiones a los sistemas tenían como motivación la curiosidad o el reto intelectual, conforme Internet se ha convertido en un vehículo económico importante (comercio electrónico, banca en línea) se ha observado que las motivaciones han cambiado.

Como cualquier actividad humana, los ataques a sistemas se profesionalizan, pasan de ser un esfuerzo puntual a una técnica organizada que persigue desde el fraude a individuos a la desestabilización de un gobierno.

1.8.4 AMENAZAS

Código Malicioso

Las primeras computadoras como la ENIAC (Electronic Numeral Integrator and Computer) se programaban por medio de la interconexión de cables a secciones de control. Cuando se iniciaba la ejecución de un programa, este continuaba su ejecución ininterrumpida de principio a fin y en un instante determinado solo había un programa en ejecución. Para que las computadoras pudieran convertirse en dispositivos de manejo de información de propósitos generales, fue necesario cambiar este paradigma. En primer lugar los recursos como procesador y memoria empezaron a compartirse, permitiendo que, mientras un programa espera la salida de un proceso de entrada/salida, otro programa utilizara por ese espacio de tiempo al microprocesador. Este hecho necesario en el paradigma de computación actual presenta implicaciones de seguridad al correr aplicaciones, “simultáneamente” un programa malicioso podría tener acceso a los datos del otro programa.

Otra característica fundamental que presenta implicaciones de seguridad es la aplicación de las ideas de John Von Neumann sobre programas almacenados. En donde para construir una máquina de propósito general era necesario almacenar no solo los datos y cálculos intermedios arrojados por un programa, sino también las instrucciones que definen el procedimiento para procesar los datos.

En una maquina de propósito especifico el procedimiento puede ser parte de ella, sin embargo en una maquina de propósito general cambiar las instrucciones debe ser tan fácil domo cambiar los datos sobre los que actúan.

La solución, codificar las instrucciones de forma numérica y guardarlas junto con los datos en la misma memoria. A este concepto se le conoce como programa almacenado, idea alrededor de la cual gira la llamada arquitectura Von Neumann (Ref. 15). El concepto de programa almacenado, parte fundamental de la computación como la conocemos hoy en día, tiene complicaciones ya que si el programa es tratado y almacenado de la misma forma en la que los demás datos, otro programa podría acceder a él y modificar el curso de ejecución o concatenar sus propias instrucciones

1.8.5 VIRUS

En 1984 Fred Cohen, en su tesis de doctorado presento la que es aceptada como la definición formal de virus:

> *Considere un set de programas que producen una o mas salidas. Para cada par de programas p y q, p eventualmente produce q y si y sólo si p produce q. Un virus es el set máximo de programas V para cada par de programas p y q en V, p eventualmente produce q, y q eventualmente produce p.*

Fig. 1.21 Virus, definición formal

La idea fundamental de la definición de Cohen, es que un virus informático es un conjunto de caracteres (instrucciones) que interpretadas en un ambiente (sistema operativo) tienen la capacidad de replicarse en otra parte del ambiente (sistema operativo).

Normalmente la palabra virus se utiliza para referirse a muchas formas de malware de manera genérica, lo cual es incorrecto.

El virus es solo la porción de código que se replica en otros programas mediante diversas técnicas . Partiendo de la definición de Cohen el único servicio de seguridad en contra del cual atenta un virus es la integridad. Si el programa llamado genéricamente virus, además de replicarse lleva a cabo otra acción como recolectar información del sistema, esta funcionalidad entra en una categoría distinta de malware.

Un programa ejecutable consta de dos partes: un encabezado y una imagen que se carga en memoria para su ejecución. El encabezado contiene información entre la cual se encuentra el tamaño y un apuntador hacia la primera instrucción a ejecutar; este apuntador tiene dos valores: el segmento de código (CS) y el apuntador a la instrucción (IP)

En un programa ejecutable infectado, el encabezado es alterado, modificando el apuntador hacia la primera instrucción y el tamaño del programa. De esta forma son ejecutadas primero las instrucciones del virus para después ejecutar el programa normalmente (Ref. 16).

1.8.6 GUSANOS

Los gusanos son programas con la capacidad de replicarse sin necesidad de modificar el código existente en el sistema. Delega a otro programa su propia ejecución, normalmente en la forma de alguna vulnerabilidad. En comparación con los virus, los gusanos requieren una menor interacción entre el usuario y el programa, es esta característica la que les permite propagarse a una mayor velocidad.

Se reconocen cuatro etapas de un gusano (Fig. 25): la fase inactiva, la de propagación, activación y ejecución. Resultando la fase de propagación característica de los gusanos ya que en esta etapa se llevan a cabo las siguientes acciones:

- Buscar más sistemas para infectar.
- Establecer una conexión con el sistema remoto
- Copiarse en el sistema y hacer que esa copia se active

En esta etapa también podrían verificar si una de sus copias ya esta activa en el sistema. En los sistemas de multiprogramación un gusano podría intentar disfrazar su presencia nombrándose como un proceso del sistema.

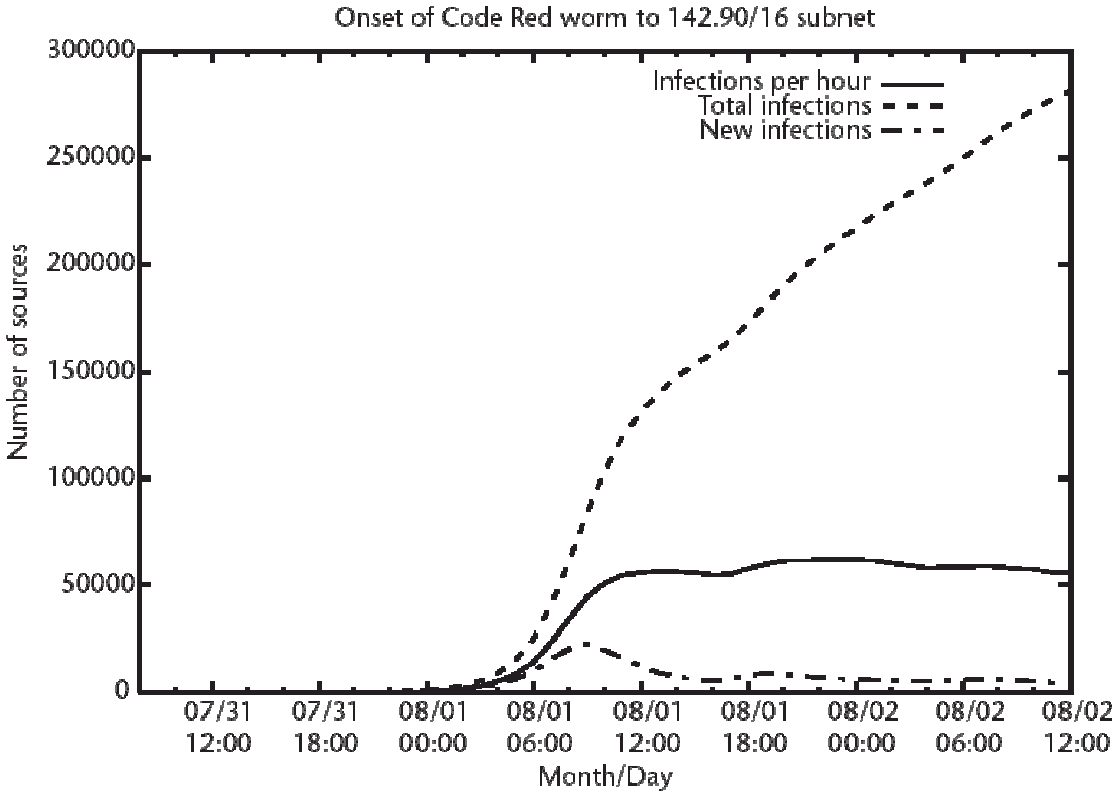


Fig. 1.22 Evolución de los gusanos
(http://vx.netlux.org/lib/img/anj01/figure32.gif)

1.8.7 CABALLOS DE TROYA

Es un programa o procedimiento de comandos útil (o aparentemente útil) que contiene código oculto que al invocarse lleva a cabo funciones no deseadas o perjudiciales. Los caballos de troya pueden llevar a cabo diferentes acciones:

- Acción directa. Los efectos dañinos del programa oculto se revelan inmediatamente después de ser ejecutados
- Puerta trasera. El programa crea un proceso que escucha probablemente en un puerto TCP/IP y que permite al atacante conectarse a ese puerto.
- Rootkit. El programa reemplaza ejecutables con sus propias versiones con el objetivo de hacer más difícil su detección u obtener información del sistema.

1.8.5 HÍBRIDOS

En los inicios del malware la clasificación entre virus, gusanos o caballo de troya era simple pero con el tiempo clasificar al malware se ha vuelto una tarea mas complicada ya que los programas maliciosos ocupan diferentes funcionalidades. Normalmente los proveedores de programas antivirus, clasifican a estos programas de acuerdo con la funcionalidad mas representativa del malware, identificar estas características ayuda a su eliminación del sistema.

Capítulo 2

Técnicas de Mitigación y Defensa en Profundidad

2.1 FIREWALLS

Son por mucho los dispositivos con mayor presencia en el panorama de la seguridad en redes. Su principio de funcionamiento se asemeja a la estrategia medieval que consistía en excavar un foso profundo alrededor de los castillos, esto obligaba a que todos los que entraban o salían del castillo pasaran a través de un puente levadizo en donde funcionarios podían inspeccionarlos. El propósito principal de un firewall es evitar los accesos no autorizados entre redes y proporcionar un único punto de defensa con acceso controlado y auditado a los servicios, dentro y fuera de la red de una organización. Un firewall basa su funcionamiento examinando los paquetes IP que viajan entre el servidor y el cliente.

Las capacidades que brinda un firewall (Ref. 17) son:

- Define un único punto de entrada y salida que deja fuera de la red protegida a usuarios no autorizados, prohíbe que los servicios potencialmente vulnerables entren o salgan de la red. El usar este único punto simplifica la administración de seguridad debido a que las capacidades de seguridad se consolidan sobre un único sistema o conjunto de sistemas.

- Proporciona una ubicación para monitorear los eventos relacionados a seguridad. Auditorías y alarmas se pueden implementar en el firewall

Entre los servicios que un firewall proporciona son: protección a los servicios vulnerables, acceso controlado a los sistemas, seguridad concentrada, registro de apertura de sesiones, ejecución de políticas de seguridad, además de que debe ser un sistema inmune a intrusiones

De acuerdo con la técnica utilizada, los firewalls se pueden clasificar en: filtrado de paquetes, proxy de aplicación y proxy de circuito.

2.1.1 Filtrado de paquetes

Este tipo de firewall examina los encabezados IP y la capa de transporte (TCP o UDP) y puede o no construir una tabla de estado de conexiones. Con base en estos criterios un paquete IP es analizado,

si existe coincidencia con alguna de las reglas se toma la decisión de reenviar o desechar, si no existe coincidencia se toma una acción por defecto que puede ser permisiva o prohibitiva.

La principal ventaja de este tipo de firewall es que analiza los paquetes muy rápidamente, de esta forma no se ve afectado el desempeño de la red. Presenta como desventaja el hecho de que la inspección solo se lleva a cabo a nivel de los encabezados, el contenido de los paquetes no es analizado además de que existen técnicas especificas como falsificación de direcciones IP o fragmentación que logran falsos negativos en las reglas.

El filtrado de paquetes se configura como una lista de reglas basadas en coincidencias en los campos de los encabezados IP o TCP. Si hay una coincidencia con una de las reglas se invoca esa regla para determinar si el paquete se reenvía o se descarta, de lo contrario se toma una acción por omisión.

Si un firewall puede bloquear conexiones TCP o UDP hacia o desde puertos específicos, entonces se pueden implementar políticas que permitan que ciertos tipos de conexiones puedan realizarse a ciertos anfitriones específicos, pero no a otros anfitriones.

- *Ventajas del firewall filtrado de paquetes.* Es el firewall más común, simple y fácil de emplear. Típicamente son transparentes para los usuarios y muy rápidos. Los sistemas del sitio usualmente tienen acceso directo a Internet, mientras que todos o la mayor parte de los accesos a sistemas del sitio, desde Internet se bloquean. Usualmente servicios inherentemente peligrosos tales como NIS, NFS y X Windows se bloquean.

- **2.1.2 Proxy de aplicación**

En este tipo de firewall la comunicación entre el cliente y el servidor es realizada mediante una aplicación. Cuando un cliente intenta una conexión como FTP o TELNET, el proxy solicita los parámetros requeridos y una vez que estos son proporcionados y validados se encarga de transmitir el tráfico hacia el servidor correspondiente. Este tipo de firewall tiende a ser más seguros que el de filtrado de paquetes ya que el proxy tendrá claramente definidos los protocolos y aplicaciones soportados, además es fácil registrar y auditar todo el tráfico entrante en el nivel de aplicación.

La principal desventaja del proxy de aplicación es que afecta el rendimiento de la red y esta sujeto a la disponibilidad de la existencia del proxy adecuado para cada protocolo.

- **2.1.3 Proxy de circuito**

Otro tipo de firewall es el proxy de circuito. Es un sistema que no permite una conexión TCP de punta a punta, su característica es que define una conexión entre él mismo y el usuario externo y él mismo y el usuario interno, una vez establecidas estas conexiones el sistema envía paquetes entre ambas. La mejora en la seguridad consiste en determinar que conexiones son permitidas.

2.2 SISTEMAS DETECTORES DE INTRUSOS (IDS, Ref. 18)

La detección de intrusos es el proceso mediante el cual se monitorea la actividad en la red o en un host con el fin de identificar problemas de seguridad, es decir, violaciones a políticas de seguridad de la institución. Un IDS monitorea la actividad que sabe maliciosa, alerta y activa alarmas para que el personal correspondiente responda al incidente de seguridad y pueda tomar alguna acción. En función del sistema y la configuración de la red, un IDS puede buscar ataques que provienen desde afuera de la red, o también monitorear las actividades de la red interna.

La tecnología de detección de intrusos se ha usado por muchos años y se considera madura. Numerosas organizaciones utilizan la detección de intrusos no solo para identificar ataques, sino también para obtener otras estadísticas que ayuden en la administración de redes y equipos.

La detección de intrusos requiere sobre todo un plan de administración y su implementación es el paso siguiente al establecimiento de políticas y de medidas de protección

2.3 SISTEMAS PREVENTORES DE INTRUSOS (IPS)

La tecnología de prevención de intrusos añade otra capa a la protección de recursos. A diferencia de un IDS que solo detecta ataques y genera alertas, un IPS detendrá la acción que considera maliciosa antes de que esta se lleve a cabo con éxito. Al igual que los IDS, los IPS se clasifican de acuerdo al nivel en el

cual trabajan. Existen, el IPS basado en red que busca patrones de ataque en el trafico de una red, y el IPS basado en host que detiene ataques a nivel sistema operativo o aplicación. En general los sistemas de prevención de intrusos consisten en un control de seguridad convencional que además tiene la capacidad de detener la acción antes de que esta termine. Ejemplo de esto son firewalls o NIDS con la capacidad de desechar paquetes, appliances o sistemas antivirus capaces de monitorear y detener llamadas a interrupciones del sistema operativo.

2.4 HONEYPOTS, HONEYNETS, HONEYTOKEN

La tecnología honey se refiere al uso de recursos informáticos (hardware o software) con el objetivo de identificar y recopilar trafico malicioso, técnicas de intrusión y motivaciones de atacantes. Esto se logra colocando dispositivos ampliamente monitoreados sin ninguna función especifica para la organización y ningún uso autorizado, de esta forma se tiene la capacidad de analizar cualquier evento ya que la naturaleza de la actividad registrada será accidental o maliciosa.

Existen diversas clasificaciones para esta tecnología una de las más comunes es de acuerdo con su alcance: honeynets, honeypots y honeytoken (Ref. 19).

- Honeypot (Ref. 20)

  Es un recurso de red destinado a ser atacado o comprometido. De esta forma, un honeypot será examinado atacado y probablemente comprometido por cualquier atacante. Los honeypot no tienen en ningún caso la finalidad de resolver o arreglar fallos de seguridad en nuestra red. Son los encargados de proporcionar información valiosa sobre los posibles atacantes en potencia a nuestra red antes de que comprometan sistemas reales

- Honeynets (Ref. 21)

  Una honeynet es una red de computadoras denominadas comúnmente honeypots, cuyo único propósito reside en ser comprometidas por un intruso. Una honeynet es monitoreada de cerca en donde todo el tráfico de red dirigido hacia los honeypots es sospechoso por

naturaleza, esta actividad puede ser clasificada como una prueba, un escaneo o un ataque en curso.

Una de las ventajas del uso de honeynets para el monitoreo pasivo de red es que los honeypots pueden ser implantados en computadoras de uso o de bajo costo, no se requiere contar con equipos nuevos o con altos recursos de cómputo, debido a que estos equipos no serán empleados en ambientes de producción. Sin embargo, una de las desventajas del uso de honeynets es que aun cuando el tráfico es sospechoso por naturaleza, dicha actividad no representará una muestra significativa de toda la actividad maliciosa existente en la red. Otra desventaja es la posibilidad de que el intruso identifique que ha comprometido a un honeypot y elimine, o peor aún, modifique los rastros de la intrusión.

- Honeytoken (Ref. 22)

  Es un recurso digital de cualquier tipo (documento, archivo de música, número de tarjeta de crédito, etc.) destinado únicamente a interaccionar con posibles atacantes. Este tipo de recursos deben cumplir con dos requisitos fundamentales: no deben tener valor real y deben ser similares a un recurso real, ya que el objetivo es que el atacante confíe en su autenticidad.

2.5 DEFENSA EN PROFUNDIDAD

Dicho concepto surge en el ámbito militar. La aparición de armamento sofisticado capaz de dañar severamente las edificaciones, provocó un cambio en la arquitectura y construcción de los inmuebles militares. Se comenzaron a construir edificios que utilizaban la profundidad del terreno para desplegar capas de defensa en torno al área crítica (Ref. 23). De esta forma un ataque debería soportar la resistencia de varias medidas para tener éxito. Esta estrategia es benéfica ya que la naturaleza de cada control trabaja coordinadamente para lograr su objetivo.

Los controles de acuerdo con sus funciones individuales pueden ser: disuasivos, correctivos o de detección. Esta diversidad trabajando para lograr un fin en común es la esencia de la defensa en profundidad.

Fig. 2.1 El Pentágono, Google Maps

Los conceptos fundamentales de defensa en profundidad que se han trasladado al contexto informático son:

- Los elementos a proteger están rodeados por varias líneas de defensa.
- Cada una de las líneas de defensa participa en la defensa global.
- Cada línea de defensa es independiente. La pérdida de la capa de protección anterior tiene un efecto global pero no debilita a la siguiente capa.

2. 6 CRIPTOGRAFÍA

La utilización de la criptografía aparece desde la época antigua en Mesopotamia (1500 a.C.), pero no fue hasta aproximadamente el año de 1949 cuando científicos británicos y polacos (entre ellos Alan Turing) utilizaron las matemáticas para romper códigos empleados en la segunda guerra mundial. Fué entonces cuando la criptografía perdió la categoría de arte y comenzó su era científica.

Los procesos de cifrado y descifrado son las operaciones fundamentales en la criptografía la cual es entendida como un conjunto de técnicas que operan sobre los mensajes para convertirlos en representaciones que carecen de sentido para cualquiera que no deba recibirlos. Estas operaciones integran lo que se conoce como algoritmo criptográfico o algoritmo de cifrado E y que, junto con el elemento único de la transformación conocido como clave, el mensaje M a cifrar o mensaje en claro y el mensaje cifrado C conforman, los elementos del sistema de cifrado o criptosistema (Figura 27).

Es precisamente, de acuerdo el número de llaves empleado para cifrar y descifrar la información, su clasificación más común: simétrica, asimétrica o hash

En la criptografía simétrica, existe una sola clave para el cifrado y el descifrado. Tiene la característica de ser veloz con los recursos computacionales accesibles actualmente. Implementa el servicio de seguridad de confidencialidad.

La criptografía asimétrica surgió en los años 70’s y requiere una pareja de claves para cifrar y descifrar. Una clave pública accesible y una privada que sólo es conocida por el propietario. Debido a las características de este tipo de cifrado, se implementa autenticación y no repudio. Los algoritmos de firma digital están basados en este tipo de criptografía, también varios protocolos de intercambio de llaves tiene la característica de ser computacionalmente más costosos que la criptografía simétrica.

La técnica de hash utiliza una función irreversible que da como resultado un bloque de tamaño fijo. Dada esa característica es ampliamente utilizada en protocolos que pretenden mantener la integridad de la información. Se utilizan para ``comprimir" un mensaje de longitud variable tomado como entrada a uno de tamaño fijo (valor hash) producido como salida, reduciendo el tiempo de generación de firmas por algoritmos de firmas digitales

En la defensa en profundidad, la criptografía es un potente mecanismo ya que es útil para diversos fines como intercambio de información sensible o almacenamiento de información confidencial.

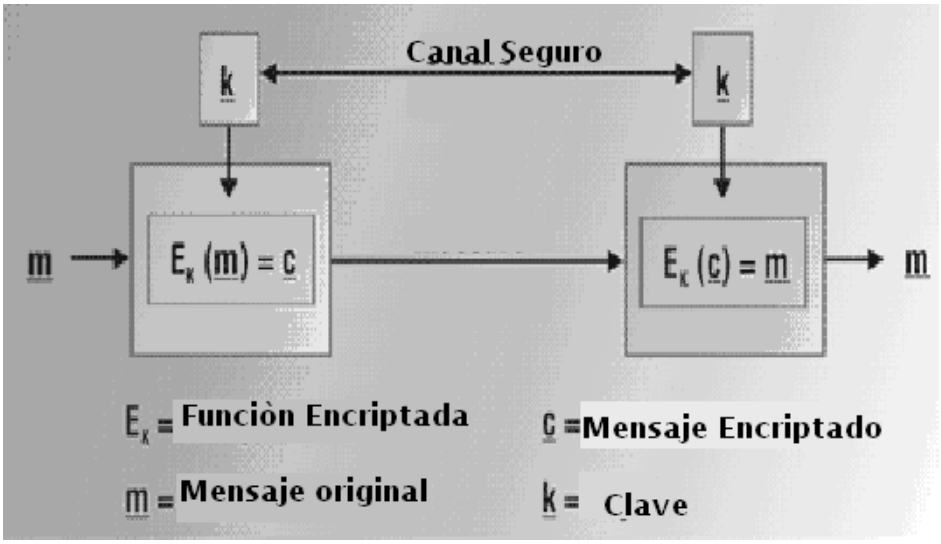


Fig. 2.2 Criptosistema
http://www.industrial-embedded.com/articles/nueckel/images/1.png

2.7 EL PROCESO DE SEGURIDAD

La importancia de la seguridad informática y su constante evolución es un proceso que requiere varias etapas (Fig. 28). Analizar los riesgos a los cuales está expuesta la organización representa un primer paso en la planeación de una estrategia de seguridad. De éste análisis se desprenden políticas, métricas de monitoreo y de auditoría, y un plan en caso de incidentes.

Es durante la etapa de análisis en donde se clasifican los activos, sus vulnerabilidades y riesgos, y su impacto en la organización en caso de pérdida total o parcial. El éxito de este análisis dependerá de la interacción de los elementos anteriormente mencionados. Difícilmente se logrará asegurar un activo en su totalidad, en lugar de ello se buscará comprender cual es el nivel aceptable de riesgo. Se debe verificar que las medidas tomadas para su protección están siendo efectivas, como reaccionar en caso de un incidente y lograr que su costo de protección no exceda el costo de pérdida.

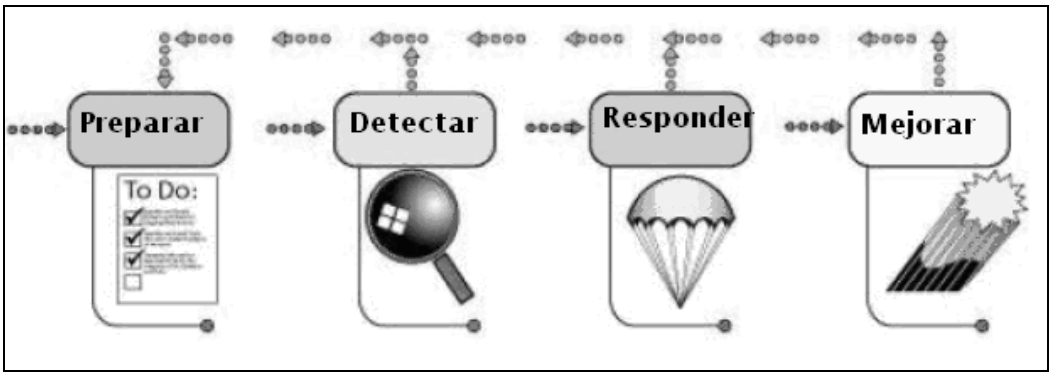


Fig. 2.3 Proceso de seguridad
http://www.sei.cmu.edu/news-at-sei/columns/security_matters/2001/2q01/security-2q01.htm

- **2.7.1 Políticas**

Las políticas juegan un papel fundamental en este proceso de seguridad ya que es en este documento donde se establece la postura de la institución, se fomenta la productividad y se logra un mejor ambiente de trabajo. Además permite elaborar normas operativas que deben entenderse como reglas que hay que seguir obligatoriamente (Ref. 24).

Una política de seguridad tiene que cumplir muchos objetivos. Deberá proteger a la gente y la información; especificar las reglas para el comportamiento esperado por usuarios, administradores de sistema, dirección, y personal de seguridad. Autorizar al personal de seguridad para supervisar, monitorear, e investigar los eventos necesarios; además de definir y autorizar la respuesta a incidentes, esto ayudará a reducir al mínimo el riesgo impulsando el cumplimiento de regulación y legislación correspondientes (Ref. 25).

Las políticas además de perseguir el cumplimiento de los puntos citados también deben poseer ciertas características: ser específicas, evaluables, alcanzables y realistas. Además de los citados anteriormente, la OCDE (Ref. 26) considera como elementos de las políticas:

i. Concientización. Los participantes saber de la necesidad de contar con sistemas y redes de información seguros.

ii. Responsabilidad. Todos los participantes son responsables de la seguridad de los sistemas y redes de información.

iii. Respuesta. Los participantes deben actuar de manera adecuada y conjunta para prevenir detectar y responder a incidentes que afecten la seguridad.

iv. Ética. Los participantes deben respetar los interese legítimos de terceros

v. Evaluación del riesgo. Los participantes deben llevar a cabo evaluaciones de riesgo.

vi. Diseño y realización de la seguridad. Los participantes deben incorporar la seguridad como un elemento esencial de los sistemas y redes de información.

vii. Gestión de la seguridad. Los participantes deben adoptar una visión integral de la administración de la seguridad

viii. Reevaluación. Los participantes deben realizar modificaciones pertinentes sobre sus políticas, prácticas, medidas y procedimientos de seguridad de manera constante.

- **2.7.2 Monitoreo y respuesta a incidentes**

Después de haber establecido las políticas correspondientes se debe continuar con la detección y mitigación de incidentes de seguridad. La detección implica el conocimiento de condiciones estables de la red y los sistemas, así como de la disposición de instrumentos para evaluar los parámetros correspondientes. .

En una red de computadoras existen cientos de dispositivos, cada uno registrando miles de transacciones en periodos mínimos de tiempo. Revisar logs e identificar actividades que pongan en peligro los recursos informáticos son tareas que requieren no solo conocimiento de la organización que y de lo que se quiere proteger, sino también conocimientos acerca de las técnicas y patrones que indican acciones maliciosas. Es entonces cuando resulta útil definir los conceptos de incidente y evento.

Un evento es una acción que se observa directamente o bien que se puede demostrar que ocurrió. La secuencia de inicio de un sistema, o un paquete a través de la red, son ejemplos de eventos.

Un incidente, esta conformado por eventos, y su característica fundamental es que un incidente representa daño a la infraestructura informática, humana, o bien la intención de daño (Ref. 27).

La respuesta a incidentes consiste en contar con un plan de acción en caso de que ocurra una situación que ponga en peligro la seguridad de una organización (Fig. 29). El objetivo que debe perseguir un equipo de respuesta a incidentes consisten en minimizar los daños que un ataque podría provocar, actuando en tiempo real, esto por medio de:

- *Preparación*. Contar con el hardware, software, e información como procedimientos y roles de cada miembro del equipo.
- *Identificación*. Definir patrones que correspondan a un incidente.
- *Contención*. Procedimientos para aislar un incidente. Realizar respaldos y reunir información necesaria para identificar entre otras cosas, detalles del incidente, origen y consecuencias.
- *Erradicación*. Solucionar el problema. Esto mediante la identificación de las causas del incidente, asegurar equipos y probar que la posible vulnerabilidad ha sido corregida.
- *Recuperación*. Restaurar las operaciones y monitorear de cerca el comportamiento de los sistemas.
- *Mejora*. Obtener conclusiones con respecto al incidente, con el objetivo de mejorar el proceso de seguridad.

Fig. 2.4 Proceso de Respuesta a Incidentes
http://www.brighthub.com/computing/enterprise-security/articles/3098.aspx

La respuesta a incidentes es una actividad que puede resultar difícil; pero en caso de que ocurra un ataque hacia la organización, el hecho de contar con un plan, aunque sea mínimo, disminuye la posibilidad de cometer errores debidos al tiempo de respuesta tan corto que se tiene.

- **3.6.3 Auditoría**

El término auditoria se emplea para referirse a actividades de recopilación y análisis de información. Su objetivo es evaluar un proceso y compararlo con un estándar, o bien con un proceso previamente planeado y documentado.

En seguridad informática, ejemplos de actividades comunes en un proceso de auditoria son los siguientes:

- Verificación con el cumplimiento de las políticas
- Escaneo de vulnerabilidades y pruebas de penetración
- Creación de configuraciones base de sistemas y control de cambios

La auditoria de seguridad informática, examina los controles dentro de una infraestructura de tecnologías de la información. La evaluación, resultado de dicho análisis, indica a que nivel se esta cumpliendo con estándares o buenas practicas; sin embargo no se puede perder de vista que el cumplimiento de estándares, políticas y buenas practicas de la industria no garantiza que la organización este segura.

Una buena evaluación, resultado de una auditoria nos indica que no se esta descuidando el manejo de la información.

Capítulo 3

Sistemas Detectores de Intrusos

Se han descrito términos relacionados con redes, protocolos, conceptos de seguridad y mecanismos de mitigación a incidentes; tales como firewalls, IDS`s y tecnologías honey.

La esencia de esta tesis está fundamentada en la detección de intrusos basada en red NIDS (Network Intrusión Detection System). Es por ello que dicho tema es digno de discutirse en un capítulo completo (Ref. 28).

Como mencionamos anteriormente, la detección de intrusos es el proceso de vigilar los eventos que ocurren en un sistema de cómputo o red; y analizarlos para buscar signos que indiquen problemas de seguridad, es decir, violaciones a las políticas de seguridad.

La función básica de un sistema de detección de intrusos IDS (Intrusión Detection System), es registrar los indicadores de actividad intrusiva y activar las alertas correspondientes. En función del sistema y la configuración de la red, un IDS puede buscar ataques que provienen desde afuera de la red, o también puede monitorear las actividades desde la red interna.

3.1 FUNCIONAMIENTO

En términos simples los sistemas de detección de intrusos están formados por tres componentes funcionales: una fuente de información que proporciona el flujo de registros de eventos; un motor de análisis que encuentra indicadores de intrusión; y un componente de respuestas que genera reacciones basadas en el resultado arrojado por el motor de análisis.

3.1.1 Fuente de datos o generador de eventos.

Las fuentes de datos pueden clasificarse en cuatro categorías: host, red, aplicación y objetivo. El término monitoreo se utiliza para describir la acción de recolectar información de una fuente de datos y pasarla a un motor de análisis.

- Los monitores basados en host recolectan los datos desde fuentes internas a una computadora usualmente del nivel de sistema operativo. Estas fuentes pueden incluir los registros de auditoria del sistema operativo y las bitácoras del mismo

- Los monitores basados en red recolectan los paquetes que pasan por la red. Frecuentemente esto se hace con el uso de dispositivos de red que se configuran en modo promiscuo.

- Los monitores basados en aplicaciones obtienen información de las aplicaciones que se están ejecutando. Las fuentes de información son las bitácoras de las aplicaciones y otros registros internos de las aplicaciones.

- Los monitores basados en un objetivo funcionan un poco diferente al resto de los mencionados que generan sus propios datos. Usan funciones criptográficas de hash para detectar alteraciones a objetos del sistema y comparan dichas alteraciones con la política

3.1.2 Motor de análisis de datos

Una vez definidas las fuentes de información se debe determinar el motor de búsqueda. Este componente del sistema toma la información de las fuentes de datos y la examina para detectar indicios de un ataque o violaciones a la política de seguridad.

En la mayoría de los casos se recurre a tres tipos de análisis: detección de anomalías, detección de abusos o alguna mezcla de las dos. Los detalles de dichos anàlisis son:

- Detección de usos indebidos (“misuse”). Se busca la ocurrencia de algo que se ha definido como “malo”. Para tal efecto se filtran los eventos buscando patrones de actividad que puedan coincidir con un ataque u otra violación a una política de seguridad. La detección de abusos utiliza técnicas de coincidencia de patrones. Generalmente los sistemas comerciales utilizan este tipo de técnica.

- Detección de anomalías. Se busca algo raro o inusual. Se analizan los eventos del sistema utilizando técnicas estadísticas para encontrar patrones de actividad que aparentan ser normales.
- Mezcla de ambos. Existen grandes ventajas al combinar los dos tipos de análisis. La detección de anomalías permite al sistema identificar ataques nuevos o desconocidos, mientras que la detección de abusos protege contra ataques conocidos.

3.1.3 Mecanismo de respuesta

Una vez que se identifica la ocurrencia de un evento de seguridad, el sistema debe contar con un componente que determina la acción a ejecutar en tal caso. La respuesta no se limita a tomar una acción en contra del sospechoso, sino en disparar alarmas de diferentes tipos. Los tipos de respuesta son:

- Sistemas Detectores de Intrusos de respuesta activa.

  Afectan al progreso del ataque. La respuesta puede ser llevada a cabo de forma automática por el sistema o mediante intervención humana. Estas acciones pueden ser de diversa naturaleza: ejecutar acciones contra el intruso, corregir el entorno o recopilar más información.

- Sistemas Detectores de Intrusos de respuesta pasiva.

  Son aquellos IDS que notifican a la autoridad competente o administrador de la red mediante sistemas de alerta. No actúa sobre el ataque o atacante.

3.2 CARACTERÍSTICAS

De acuerdo con los expertos en seguridad existen una serie de requerimientos que deben ser cubiertos por un sistema de detección de intrusos para que éste sea considerado como un buen producto. Éstas características son:

- Efectividad. Es el requerimiento más importante de todos, pues los IDS deben ser capaces de detectar de una forma exacta y consistente los ataques u otros patrones definidos.
- Facilidad de uso. Contar con expertos en seguridad resulta difícil y muy caro, por lo que es necesario que la herramienta pueda ser manejada por una persona que no sea necesariamente experta en seguridad.
- Adaptabilidad. El sistema debe ser adaptable a diferentes plataformas, ambientes y políticas. La gran mayoría de ambientes de cómputo no son homogéneos, por lo que el IDS debe ser capaz de entender las entradas provenientes de otros sistemas
- Robustez. El sistema debe ser lo suficientemente confiable y contar con mecanismos de redundancia u otras características que le permitan seguir operando en caso de falla..
- Rapidez. Capaz de ejecutar una vigilancia y reportar los eventos en el momento en que ocurren.
- Eficiencia. Debe hacer uso óptimo de los recursos de cómputo, almacenamiento y ancho de banda, de forma que afecte en lo mínimo el desempeño del ambiente que vigila
- Seguridad. Contar con características que eviten que éste sea utilizado por personal no autorizado.

Además de las ya mencionadas, existe un segundo nivel de requerimientos para un IDS ideal tal como:

- Escalabilidad. En un sistema escalable los componentes deben tener interfaces estándar bien documentadas. Estas interfaces deben soportar los mecanismos de autenticación apropiados.
- Equilibrio. El sistema debe permitir a los usuarios mantener un balance entre las necesidades de administración y las necesidades de seguridad.

3.3 CAPACIDADES Y LIMITACIONES

Ventajas:

• Monitorización. Cuentan con métodos para monitorizar y analizar tanto los eventos de sistema como el comportamiento de los usuarios.

• Emisión de informes de auditoría. Extraen los datos más relevantes de entre grandes cantidades, lo que facilita el trabajo del auditor de seguridad.

• Correlación. Aunque de forma limitada, pueden establecer patrones de relación entre ataques o comportamientos similares.

• Análisis en tiempo real. Utilizan mecanismos de análisis y registro que permiten tal modalidad.

• Alarmas. Comunican alarmas a los responsables cuando se produce una situación anormal, como una intrusión.

• Facilidad de uso. Cuentan con características de detección automática así como una interfaz fácil de uso.

Inconvenientes:

• Solución definitiva. No existe ninguna solución única que resuelva todos los problemas de seguridad.

• Falsos positivos. Uno de los inconvenientes más populares es el de las falsas alarmas.

• Defensa ante nuevos ataques. No pueden detectar ataques de reciente aparición o variantes de ataques existentes.

• Calidad de los datos. No pueden compensar errores producidos por el uso de datos de mala calidad.

• Conocimiento de cada situación. Estos sistemas no conocen de antemano las particularidades de cada entorno en que son implementados.

• Encriptación. El uso de comunicaciones cifradas (como SSH) puede inhabilitar la utilidad de un detector de intrusiones basado en red.

• Calidad de los protocolos. No compensan las debilidades asociadas al diseño de un protocolo.

3.4 ESTRATEGIAS

La primera línea de defensa de los sistemas de información consiste en los controles de acceso físico y lógico, sin embargo estas medidas no aseguran totalmente no tener intrusos. Debido a esto fue necesario desarrollar estrategias que faciliten la tarea de identificación de sistemas comprometidos. Existen muchas razones por las cuales esto es deseable, mientras mas rápido se descubra que existe un intruso su expulsión del sistema se realizara de manera mas oportuna. El uso de una estrategia adecuada de detección de intrusos funciona como elemento disuasivo, finalmente la detección de intrusos sirve para recopilar información acerca de las técnicas empleadas por el atacante, esto debe ser incluido en la protección futura de los sistemas de la organización.

Las estrategias más comúnmente empleadas en la detección de intrusos son:

- Detección de anomalías
- Firmas de ataques conocidos
- Correlación de eventos

Cada una de estas estrategias emplea más de un método para cumplir con su tarea, y dependiendo de las necesidades del contexto en el cual se desplieguen, existirán diferencias en su efectividad.

3.4.1 Detección de anomalías.

Consiste en un análisis diferencial que puede partir de criterios muy simples hasta criterios altamente sofisticados. Esta estrategia se basa en la idea de conocer el estado normal de un sistema y

con base en ello monitorear su actividad, en caso de que alguno de los parámetros evaluados salga del patrón de conducta se obtiene un indicio de la presencia de un intruso.

La detección de anomalías se auxilia en la estadística, implica la recolección de datos de usuarios legítimos durante un periodo de tiempo. Esta estrategia a su vez tiene dos enfoques:

- *Detección de umbrales*. En este enfoque se definen los umbrales, independientemente del usuario, para la frecuencia en que se producen distintos acontecimientos.

- *Basado en perfiles.* Se desarrolla un perfil de actividad de cada usuario y se utiliza para detectar cambios en el comportamiento de cuentas individuales.

Los dispositivos que utilizan la detección de anomalías tradicional, usualmente lo hacen solo para aplicaciones y protocolos específicos. Este análisis normalmente se basa en el tráfico que no es considerado como normal generado por la aplicación, para señalar en que condiciones requieren generar alertas. Esto ha demostrado ser susceptible a falsos positivos y otra de sus características importantes es que solo el comportamiento identificado a priori será el que genere alertas

El otro enfoque de la detección de intrusos basado en anomalías. Es el análisis de protocolos y aplicaciones, aquí el IDS posee el conocimiento del protocolo y comportamiento de una aplicación determinada y cuando encuentra tráfico que incurre en alguna violación, genera alertas. De esta forma se pueden detectar ataques conocidos y desconocidos; sin embargo, el IDS será incapaz de detectar ataques que operen dentro de las condiciones consideradas normales.

Otras desventajas importantes están relacionadas con las características de los protocolos y aplicaciones (Ref. 29):

- Es necesario destinar una buena cantidad de tiempo y esfuerzo, analizando y comprendiendo funcionamientos normales.

- Cada proveedor sigue sus propias reglas cuando diseña su aplicación lo que obliga al análisis individual por aplicación.
- Los protocolos a lo largo del tiempo evolucionan, se les añaden características, forzando a llevar a cabo el análisis de comportamiento constantemente.

Mientras la habilidad de detectar ataques no conocidos es atractiva, la complejidad de la detección de intrusos basada en el análisis de protocoles es un factor importante a considerar cuando se planea desplegar esta estrategia.

3.4.2 Firmas de ataques conocidos

Este método es por mucho el mas utilizado en la detección de intrusos. Consiste en identificar alguna característica distintiva para algún evento de interés en particular, y construir con ello una regla que genere una alerta cada que encuentre dicha característica en el trafico de la red.

```
alert TCP $EXTERNAL any -> $INTERNAL 21 (msg: "IDS327/ftp_ftp-user-warez";
flags: A+; content: "user warez"; nocase; classtype: syst
em-attempt; reference: arachnids,327;)
```

En la práctica el proceso fundamental de esta técnica consiste en llevar a cabo el análisis que de cómo resultado la identificación de la característica distintiva del evento que deseamos genere alertas, un análisis incorrecto deriva en el incremente de falsos positivos y falsos negativos.

Estos detectores de intrusos basados en firmas funcionan con un conjunto de reglas en donde cada regla contiene el criterio que identifica un evento de interés. Los criterios los determina el IDS, y puede ser desde búsqueda de valores específicos en los encabezados, cadenas de texto en el campo de datos o la combinación de características múltiples.

Cuando el IDS recibe tráfico, busca concordancias con las reglas programadas y genera alertas según sea el caso. Este proceso es transparente para el encargado de monitorear la red (Fig. 31) siendo su

tarea fundamental, interpretar los resultados arrojados así como generar reglas que ayuden a reforzar la postura de tráfico permitido en la organización.

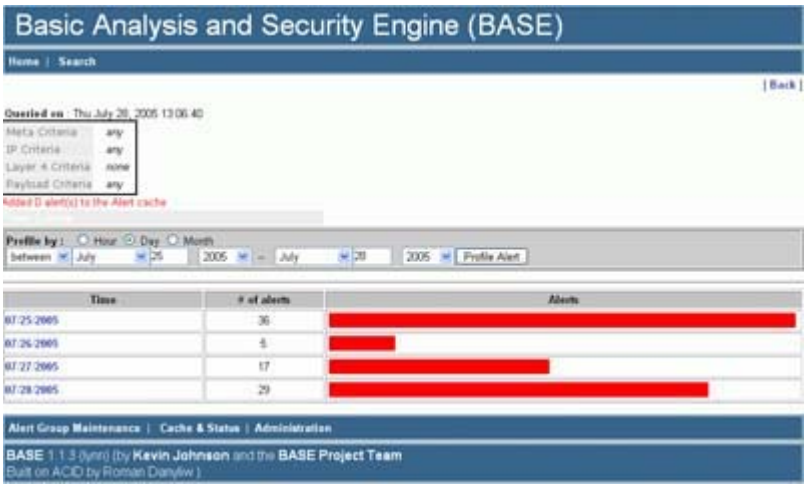


Fig. 3.1 Pantalla de la interfaz gráfica BASE

Un lenguaje flexible para escribir reglas es una característica deseable en un IDS. Permite a la organización crear reglas propias de acuerdo con su contexto y también aumenta la granularidad del monitoreo. La complejidad del lenguaje varía de acuerdo con los criterios de búsqueda que el IDS soporte; por ejemplo, el lenguaje de la herramienta Snort. Como la mayoría de los IDS permite examinar tráfico en busca de:

- Protocolos, comúnmente de capa 3 o capa 4.
- Direcciones IP origen o destino.
- Puerto TCP/UDP origen o destino..
- Características de la información en los encabezados del protocolo. Por ejemplo, combinaciones de banderas de TCP sin uso dentro del protocolo

3.4.3 Correlación de eventos.

Los entornos de comunicaciones constan de varios dispositivos entre ellos routers, switches, firewalls, sistemas detectores de intrusos, equipos finales, software antivirus etc., todos ellos

generando registros de eventos y bitácoras. Es una tarea compleja para un analista revisar cada uno del los registros generados.

La correlación de eventos pretende centralizar el manejo y análisis de dicha información, comúnmente por medio de un sistema experto que pueda ser programado para interpretar adecuadamente la información proveniente de diversas fuentes (Ref. 30)

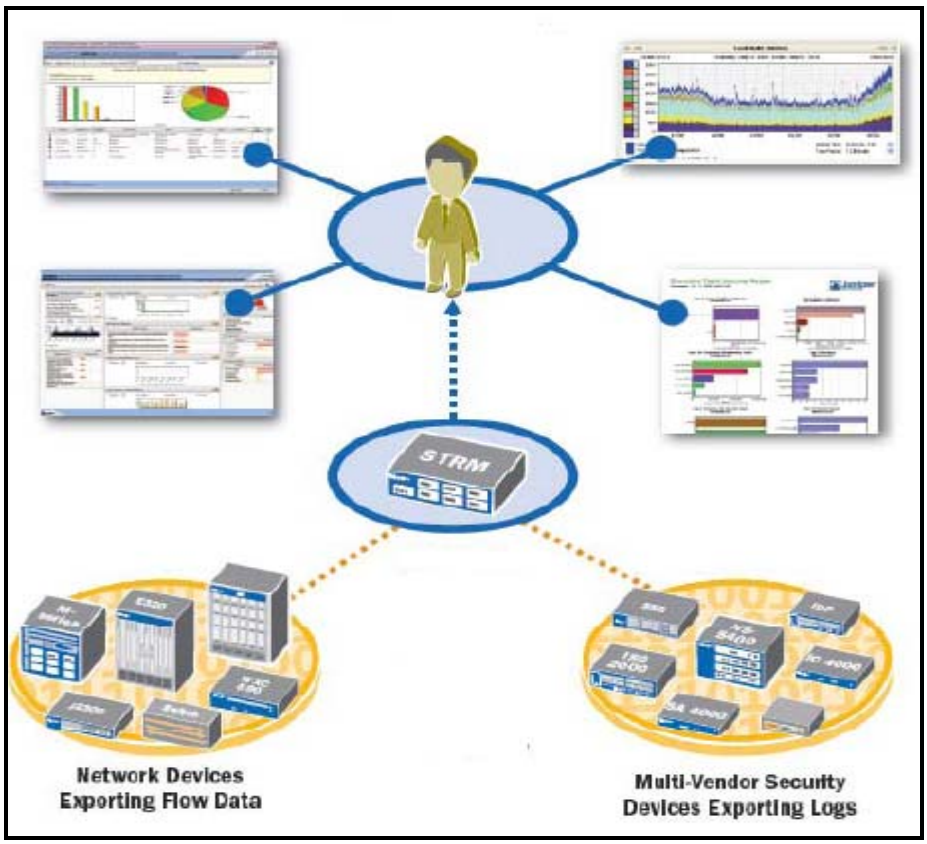


Fig. 3.2 Dispositivo de correlación de eventos
http://www.itstrap.net/descargas/boletines/Bolet%C3%ADnITStrap_2_3_23mar08.pdf

Algunos tipos de correlación en función de los datos utilizados son:

- Correlación entre varios puntos de la infraestructura. Esta permite dar seguimiento a los puntos por donde se han generado comportamientos anómalos, de esta manera se obtiene el beneficio de evitar recibir alertas diversas para el mismo evento.

- Correlación entre eventos. Existen eventos que por si mismos no tienen significado en especial, pero cuando se presentan en presencia de otros eventos identifican ataques específicos. Un ejemplo de esta situación lo constituye un escaneo de puertos, la conexión a un puerto por si misma no representa un evento de interés pero intentos de conexión a los 65535 puertos en un periódo corto de tiempo es una situación que un administrador debe poder detectar.

- Correlación de datos capturados con datos conocidos de sistemas objetivo. El evento producido se compara con información del sistema al cual está dirigido y en función de ello se le asignara un nivel de criticidad. Por ejemplo, un exploit para el servidor web IIS no funcionara en servidores Apache. Para un administrador resulta importante conocer este intento de explotación; sin embargo el manejo del incidente será diferente a como se daría si el exploit tuviera mayor posibilidad de tener éxito.

3.5 SISTEMAS DETECTORES DE INTRUSOS BASADOS EN RED (NIDS)

3.5.1 Arquitectura de un NIDS

Los NIDS pueden abstraerse en una arquitectura dividida en tres partes:

- Captura y decodificación. Es el encargado de capturar y generar descriptores validos para el motor de análisis. Tomando Snort como ejemplo, el encargado de llevar a cabo esta tarea es un sniffer de red que puede procesar tanto protocolos de capa de enlace de datos (PPP, SLIP, ATM) como varios protocolos de la suite TCP/IP, dicho sniffer utiliza las librerías winpcap o libpcap para cumplir con su objetivo.
- Motor de análisis (Fig. 33). Es el conjunto de rutinas que toman los descriptores enviados por el modulo de captura y decodificación y los comparan con los criterios programados que poseen acciones indexadas a ellos. Snort crea una lista enlazada bidimensional (Ref. 31), la lista base se denomina CHAIN HEADER y la que se deriva de ella CHANGE OPTION

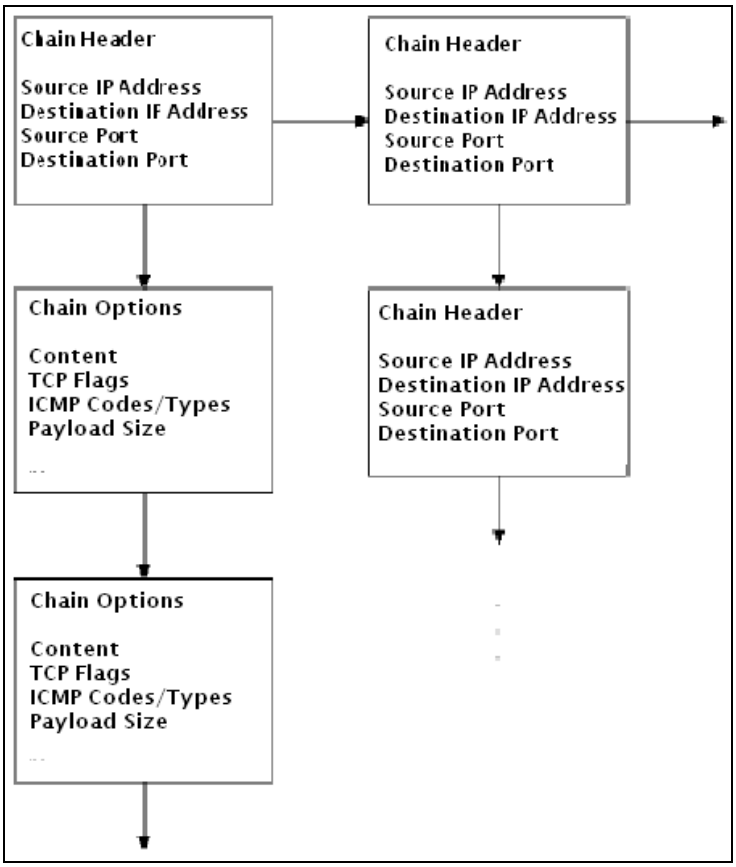


Fig. 3.3 Motor de Análisis de un NIDS
http://www.usenix.org/events/lisa99/full_papers/roesch/roesch_html/fig3.eps.gif

Cuando llega un paquete, se visitan los criterios de manera recursiva hasta la primera coincidencia. Otra característica importante de este modulo de Snort es la capacidad de agregar módulos llamados preprocesadores, que tienen funcionalidades desde mejorar el desempeño hasta llevar a cabo análisis en paquetes fragmentados (Ref. 32).

- Componente de respuesta. Programas encargados del almacenamiento, formato de las alertas y destino. La herramienta BASE (Basic Analysis and Security Engine) es comúnmente utilizada en implementaciones con Snort (Ref. 33) ya que provee de una interfaz web, que permite realizar consultas.

3.5.2 NIDS ventajas y desventajas

En cualquier organización, la seguridad informática es solo una pequeña parte de un proceso que abarca áreas diversas. Cada entidad, independientemente de su naturaleza posee una misión y sus esfuerzos se distribuyen para cumplir dicho cometido.

La seguridad informática debe contribuir con el objetivo global y es debido a este que la colocación de cualquier control debe ser precedido de un análisis de impacto en la misión de la institución.

Ventajas de la detección de intrusos basada en red

La detección de intrusos basada en red es una tecnología utilizada por muchos años, madura y ampliamente difundida. A pesar de que actualmente su evolución de respuesta inmediata ya está disponible (IPS), dadas sus características continúa figurando en la arquitectura de seguridad de diversas organizaciones. Algunas de las ventajas de los sistemas detectores de intrusos basados en red son:

- Dado que su objetivo es identificar tráfico nocivo que circula por la red (escaneo de puertos, desbordamientos de pila, gusanos de internet, conexiones y protocolos no autorizados, exploits, herramientas de hacking), representa el motor de la respuesta a incidentes.
- Es un punto de retroalimentación, ya que informa cuando algún dispositivo o política no esta funcionando correctamente (permitiendo tráfico nocivo hacia la red) facilitando que esta situación se corrija.
- Un NIDS añade una capa más a las estrategias de seguridad (segmentación, firewall, cifrado) es altamente recomendable por que indica que la institución adquiere una postura de defensa en profundidad de sus recursos.

Desventajas de un NIDS

El éxito de cualquier control o estrategia de seguridad, depende en gran medida de tener claro el objetivo de seguridad que se pretende lograr.

- EL NIDS no está diseñado (no es su objetivo) para detener el tráfico que detecta como nocivo, así que dejara pasar el tráfico siendo la única consecuencia de una detección de tráfico nocivo, la generación de alertas y/o logs.
- El uso de un NIDS implica que el tráfico será monitoreado por un sniffer lo que representa un posible problema hacia la confidencialidad de los usuarios, siendo necesario notificar explícitamente que el trafico que sale y entra de dichos equipos será monitoreado.
- El NIDS no puede analizar el tráfico cifrado en una instalación convencional debido a que no cuenta con las claves de cifrado. Por lo tanto no es una medida efectiva para los protocolos que implementen SSL (HTTPS, SSH) o cualquiera que transporte datos cifrados.
- El NIDS requiere a un administrador para las siguientes tareas:
    - Revisar de preferencia en tiempo real las alertas que se generan y logs.
    - Mantener las firmas actualizadas y generar firmas propias.
    - Verificar el funcionamiento del sistema en general.
    - Auditar el funcionamiento del NIDS.

Los IDS basados en red trabajan escuchando a la red y examinando los paquetes conforme pasan. Para este tipo de sistemas existen tres obstáculos principales:

- La velocidad de la línea, ya que los productos simplemente no pueden hacer frente a todo el volumen de datos que fluye en la red.
- En ambientes que trabajan con circuitos virtuales (switches) el IDS debe ser colocado cuidadosamente, de modo que la carga pase a través de un puerto escucha.

Como la mayoría de los controles de seguridad, el uso de un detector de intrusos basado en red no representa solo un costo inicial en hardware y posiblemente licencias de software.

3.5.3 CAMPOS INDICATIVOS DE ATAQUE (Ref. 34)

Para lograr su objetivo, al menos una de las interfaces de red de la máquina sensor trabaja en modo promiscuo, capturando y analizando todas las tramas en busca de patrones indicativos de un ataque. Los patrones que sirven para identificar un ataque se pueden observar en los diferentes campos de una trama de red TCP/IP (Ref. 35). Los casos más habituales incluyen:

- ***Campos de fragmentación.***

  Una cabecera IP contiene dieciséis bits reservados a información sobre el nivel de fragmentación del datagrama, de ellos uno no se utiliza y trece indican el desplazamiento del fragmento que transportan. Los otros dos bits indican o bien que el paquete no ha de ser fragmentado por un router intermedio (DF, Don´t Fragment) o bien que el paquete ha sido fragmentado y no es el último que se va a recibir (MF, More Fragments).

  Valores incorrectos de parámetros de fragmentación de los datagramas se han venido utilizando típicamente para causar importantes negaciones de servicio a los sistemas, incluso para obtener la versión del sistema operativo que se ejecuta en un determinado host (Ref. 36). Ciertas combinaciones de bits relacionados con la fragmentación son indicativos para sospechar un ataque.

- ***Dirección origen y destino.***

  Las direcciones origen y destino también son campos interesantes al momento de detectar intrusiones en nuestros sistemas o red. No tenemos más que pensar el tráfico entrante a nuestra DMZ (Ref. 37) que tenga como destino un servidor de bases de datos, o conexiones hacia puertos no autorizados presentando un origen desconocido.

  Otro ejemplo clásico son las peticiones originadas desde Internet y que tienen como destino máquinas de nuestra organización que no están ofreciendo servicios directos al exterior, como un servidor cuyo acceso está restringido a uso interno de la red.

- ***Puerto origen y destino.***

Los puertos origen y destino (especialmente el último) son un excelente indicativo de actividades sospechosas en la red. Aparte de los intentos de acceso no autorizado a servicios de nuestros sistemas, se pueden detectar actividades que también supondrían a priori violaciones de nuestras políticas de seguridad como la existencia de troyanos, ciertos tipos de barridos de puertos, o la presencia de servidores no autorizados dentro de nuestra red.

- ***Flags TCP***

La cabecera TCP contiene un campo de seis bits con las banderas URG, ACK, PSH, RST, SYN y FIN. Cada una de ellas con una finalidad diferente. Evidentemente el valor de estos bits será 0 o 1, lo cual de forma aislada no representa alarma alguna.

No obstante, ciertas combinaciones de valores suelen ser sospechosas. Por ejemplo, una trama con los dos bits SYN y FIN activados simultáneamente sería indicio de una conexión que trata de abrirse y cerrarse al mismo tiempo.

- ***Campo de datos.***

El campo de datos de un paquete es donde más probabilidades tenemos de localizar un ataque contra nuestros sistemas. Con toda probabilidad el firewall detendrá tramas cuya cabecera sea "sospechosa". Por ejemplo, aquellas cuyo origen no esté autorizado a alcanzar su destino.

Acabamos de ver sólo algunos ejemplos de campos de una trama TCP/IP que, al presentar determinados valores pueden ser indicativos de un ataque. Sin embargo no todo es tan sencillo como comprobar ciertos parámetros de cada paquete que circula por uno de nuestros segmentos. También es necesario que un detector de intrusos basado en red sea capaz de notificar otros ataques que no se pueden apreciar en una única trama. Uno de estos ataques es la presencia de peticiones que aunque por sí mismas no sean sospechosas, por su repetición en un intervalo de tiempo más o menos pequeño pueda ser indicativo de un ataque (barrido de puertos).

Otros ataques difíciles de detectar analizando tramas de forma independiente son las negaciones de servicio distribuidas (DDoS, Distributed Denial of Service), justamente por el gran número de orígenes que el ataque tiene por definición.

Los sistemas de detección de intrusos basados en red, de los que hemos hablado a lo largo de este capítulo son una herramienta muy utilizada. No obstante, como casi cualquier herramienta relacionada con la seguridad estos sistemas no son ninguna solución definitiva, y su implantación ha de verse complementada con una correcta configuración de elementos como firewalls o software especializado.

3.5.4 EJEMPLOS DE NIDS (Ref. 38)

Actualmente existen productos IDS comerciales, freeware y shareware. A continuación se presenta una lista parcial de los IDS basados en red más ampliamente utilizados.

Nombre	Compañía	Modelo de Costo	Descripción
Snort 	Sourcefire	Open Source	Snort es un ligero sistema de detección de intrusos de red, capaz de realizar en tiempo real el análisis de tráfico de paquetes sobre redes IP. Realizar análisis de protocolo, el contenido de búsqueda puede ser modificado al entorno. Suele ser utilizado para detectar una variedad de ataques tales como desbordamientos de búfer, escaneo de puertos, ataques CGI, y mucho más. Snort utiliza un lenguaje flexible de reglas para describir el tráfico, así como un motor de detección que utiliza una arquitectura modular plugin. Snort la capacidad de alerta en tiempo real así como, la incorporación de mecanismos de alerta para syslog. Se puede utilizar directamente como un analizador de paquetes como tcpdump, o como un detector de intrusos de red.
Bro 	Vern Paxson	Freeware	Es un hermano de código abierto basado en Unix. Utiliza la detección de intrusos basada en red, que supervisa pasivamente el tráfico de la red y busca de actividad sospechosa. Analiza el tráfico de la red para extraer su aplicación a nivel de la semántica y, a continuación, la ejecución de eventos orientados a los analizadores para comparar la actividad con los patrones problemáticos. Continúa

Continuación Cisco Secure IDS (formalmente NetRanger) 	Cisco Systems, Inc.	Commercial	El Sistema de detección de intrusos Cisco, está diseñado para proteger eficazmente los datos y la infraestructura de la información. Garantiza la protección y continuidad de las actividades y minimiza los efectos de intrusiones costosas.
SecureNet IDS/IPS 	Intrusion inc	Commercial	El Sistema de intrusos SecureNet ofrece críticas profundas de paquetes de análisis. Puede ser desplegado pasivamente para detección de intrusos (IDS) o activamente para la prevención de intrusiones (IPS). En ambos escenarios, el Sistema de SecureNet le da información sobre el tráfico en su red. El sistema SecureNet puede ser desplegado con la más amplia gama de configuraciones de red. Pasivo para la detección de intrusos o despliegues de prevención de intrusos pueden ser configurados para bloquear o permitir el tráfico de la red en caso de fallo.
Manhunt 	Symantec Corporation.	Commercial	Symantec Manhunt ofrece alta velocidad, detección de intrusos de red en tiempo real y análisis de correlación proactivo de prevención y respuesta. Protege redes de empresas internas y externas de intrusiones y denegaciones de servicio. La capacidad de detectar amenazas desconocidas, usando la detección de anomalías ayuda a eliminar la exposición a vulnerabilidades.

Tabla 4. Cuadro comparativo de NIDS freeware y shareware

Capítulo 4

Implementación del Sistema Detector de Intrusos

Una parte fundamental, para determinar el éxito en cualquier proyecto es la definición de sus alcances y objetivos. En el presente capítulo presentamos nuestras metas, y justificamos las decisiones que se tomaron al implementar la solución de detección de intrusos.

Como se ha mencionado, la detección de intrusos basada en red es una técnica ampliamente usada por muchas organizaciones, considerada con gran desarrollo y con múltiples configuraciones en su implementación.

Analizamos trabajos que profundizan ampliamente en el tema de IDS's. Las implementaciones van desde el uso clásico (un dispositivo recibiendo y analizando tráfico de la red), hasta soluciones que combinan sensores con diferentes sistemas operativos reportando a una consola central.

Como se observa en el cuadro comparativo del capítulo IV, todas las soluciones son efectivas dependiendo del escenario que se presente. En nuestro contexto, la solución que mejor se adapta es Snort, en las siguientes páginas se argumenta dicha herramienta.

4.1 MÉTRICA DEL PROYECTO

La presente tesis pretende ser un paso inicial en el uso de la detección de intrusos basada en red. Tenemos la convicción de que introducir elementos demasiado complejos para la administración va en perjuicio de la metodología que actualmente se esta usando en el departamento. Será tarea de los administradores en turno una vez que estén familiarizados con el entorno, llevar a cabo cambios o introducir otros componentes para mejorar los resultados acerca de la detección de intrusos.

Dicho lo anterior, resumimos nuestros objetivos entorno a este proyecto como:

- Sumar la NIDS a la estrategia global de seguridad. Utilizar esta técnica y sus beneficios en la DMZ (Demilitarized Zone) de USECAD. Conjuntamente al firewall, esta herramienta registrará el evento brindando un conocimiento mayor acerca de lo que ocurre en la red.

- Verificar las reglas del firewall y el cumplimiento de políticas. Colocar reglas que respalden las existentes en el firewall con la finalidad de verificar su cumplimiento y generar su registro.
- Sugerir técnicas de administración. Como hemos mencionado, cualquier control de seguridad requiere un cierto tiempo para perfeccionar. En primer lugar, los administradores del IDS deben familiarizarse con la tecnología y sus principios fundamentales.

Como resultado, obtendremos un equipo que cuente con un Sistema de Detección de Intrusiones basado en red, con una configuración inicial que alerte ataques conocidos e identifique tráfico no permitido por las políticas del departamento.

El equipo que contenga el NIDS habrá de ser sometido a pruebas de hardening y demostrar un nivel aceptable de seguridad. Además, en etapas posteriores deberá contar con una configuración exclusiva para alertar ataques dirigidos a los sistemas de la DMZ antes mencionada.

4.2 PLANEACIÓN

Una vez que tenemos definidos los objetivos consideramos de vital importancia elaborar un plan de trabajo, es decir, dividir el esfuerzo total en etapas y abordar cada una de éstas de manera progresiva.

Dichas fases se encuentran seguidas sub-objetivos señalados a continuación:

- ⇨ Análisis preliminar
  - Identificar la distribución (lógica y física) de la red en la que se va a colocar el NIDS.
  - Requerimientos de un NIDS.
  - Leer las reglas del firewall con el fin de que el NIDS tenga como uno de sus objetivos reforzar dichas reglas.
  - Identificar políticas que se puedan reforzar con un NIDS.

- ⇨ Acuerdo de Tecnología
  - Identificar el NIDS que se ajusta a las necesidades del departamento.
  - Enlistar requerimientos específicos basados en la tecnología elegida.
- ⇨ Implementación
  - Realizar la instalación del sistema operativo y paquetes necesarios.
  - Escribir reglas para el tráfico local.
- ⇨ Integración
  - Definir y escribir reglas relacionadas con las políticas locales.
  - Llevar a cabo una revisión preliminar del funcionamiento de las reglas.
  - Pruebas.
  - Recomendaciones.

Cada uno de los puntos se encuentra ampliamente detallado en páginas posteriores. Cabe aclarar que la información relacionada con la red no se encuentra detallada y ha sido modificada, ya que este documento es público y los datos podrían ser utilizados para fines indebidos.

4.3 ANÁLISIS PRELIMINAR

El primer paso en la instalación del NIDS consiste en identificar el lugar donde se pretende monitorear el tráfico. Esto requiere el análisis de la topología de la red además de contar con algunos manuales s de los dispositivos de networking de interés. Se obtuvo el diagrama que se muestra a continuación del cual pudimos resaltar conclusiones importantes.

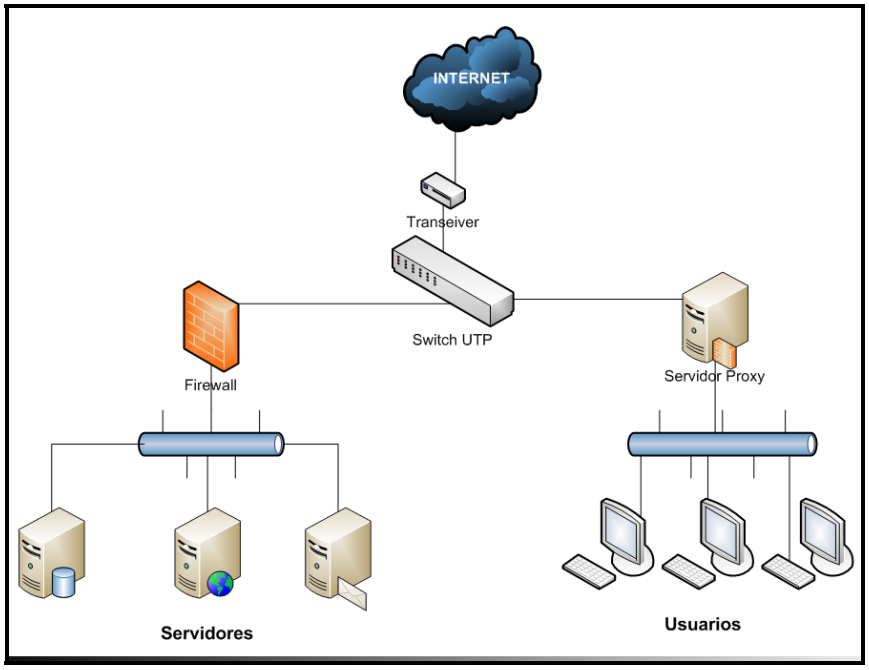


Fig. 4.1 Topología general de la red de USECAD

En primer lugar confirmamos que la red para la conectividad únicamente switches. Esto es relevante debido a que la segmentación hace más complicado el monitoreo, pero ya que nos enfocamos a la DMZ de servidores no representa problema alguno.

Otro punto importante es la identificación del lugar para la instalación del NIDS. Después de reconocer el trafico existente en cada punto elaboramos un diagrama detallado mediante la inspección física de los dispositivos, anotando detalles como tipo de equipo (switch, servidor, PC) marca y modelo, número de puertos ocupados/libres y capacidad de monitoreo.

Se continuó con la exploración colocando un analizador de protocolos y configurando al switch para retransmitir tráfico de otros puertos (configuración de puerto espejo, port mirroring). En esta etapa recibimos tráfico y aplicamos filtros para identificar las subredes cuyo flujo pasa por un puerto determinado, para verificar que realmente estamos analizando el tráfico correcto.

Configuración del puerto espejo

Mediante la configuración de esta opción en un dispositivo de red, se hace una copia del tráfico que se recibe en un puerto hacia otro donde estará conectado el equipo que realizará la captura. Detallamos la configuración ya que sin ella no se puede llevar a cabo el análisis correspondiente.

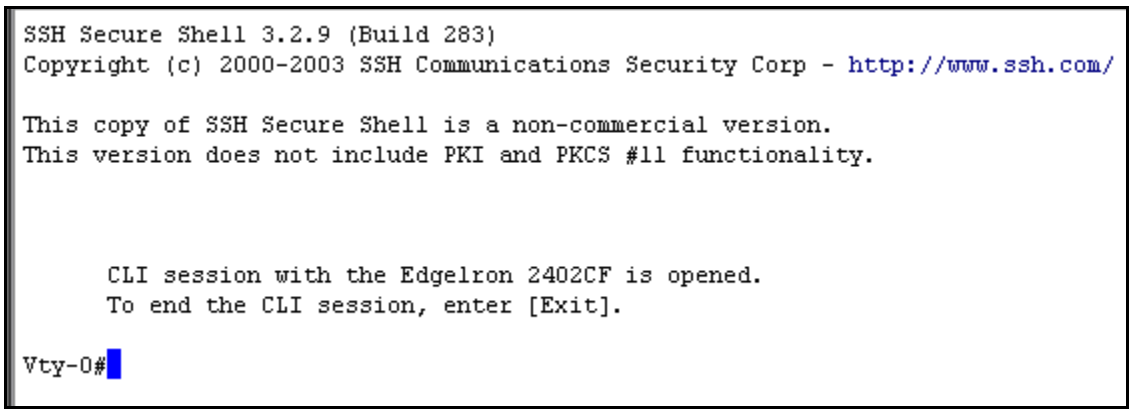


Fig. 4.2 Autenticación para el puerto espejo

Una vez dentro del switch, ingresamos al modo de configuración global

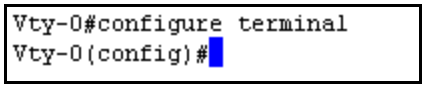


Fig. 4.3 Modo configuración global en switch

Navegamos al modo de configuración de interfaz Aquí se especifican el tipo y número de interfaz a la cual queremos aplicar la configuración.

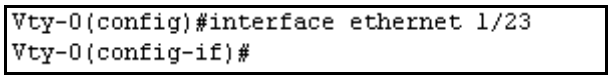


Fig. 4.4 Configuración de la interfaz

Indicamos el o los puertos de los cuales se va a recibir tráfico.

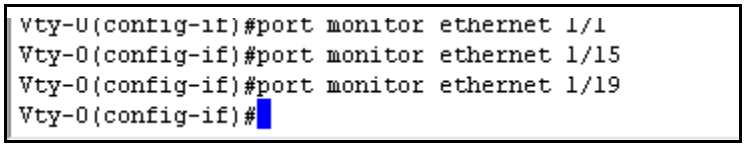


Fig. 4.5 Especificando puertos a monitorear

Con el comando show port monitor desde el prompt de configuración se comprueba la configuración realizada.

```
Vty-0(config-if)#end
Vty-0#show port monitor
Port Mirroring
-------------------------------------
 Destination port (listen port): Eth1/23
 Source port (monitored port)  : Eth1/ 1
 Mode                          :RX/TX

 Destination port (listen port): Eth1/23
 Source port (monitored port)  : Eth1/15
 Mode                          :RX/TX

 Destination port (listen port): Eth1/23
 Source port (monitored port)  : Eth1/19
 Mode                          :RX/TX

Vty-0#
```

Fig. 4.6 Resultado del comando showport

Como resultado en esta primer etapa identificamos puertos del switch, que en nuestra implementación inicial servirán en el monitoreo.

- **4.3.1 POSICIÓN DEL IDS**

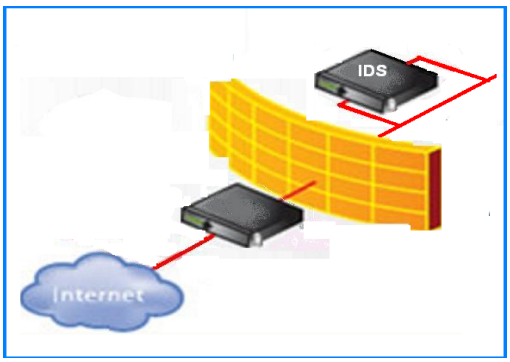


Fig. 4.7 Ubicación en red recomendada para un NIDS

Si colocamos el IDS antes del firewall capturaremos todo el tráfico de entrada y salida de la DMZ. La posibilidad de falsas alarmas es elevada. La colocación detrás del firewall monitorizará todo el tráfico que no sea detectado y detenido por el firewall, por lo que será considerado como malicioso en un alto porcentaje de los casos. La posibilidad de falsas alarmas muy inferior.

- **4.3.2 ACUERDO DE LA TECNOLOGÍA**

Utilizamos una distribución de Linux basada en Slax llamada Backtrack. La presentación viene en un live CD basado a su vez en Slackware. Cuenta con una gran variedad de herramientas orientadas a auditoria de seguridad, su instalación es parecida a la de cualquier otra distribución de Linux. Elegimos este sistema ya que cuenta con herramientas de seguridad preconfiguradas (Nessus, Ntop), que de requerirse pueden ser ejecutadas en forma rápida y proporcionar información valiosa al administrador.

4.3.2.1 SNORT

Snort es un IDS o Sistema de Detección de Intrusiones basado en Red. 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, intentos de aprovechamiento de vulnerabilidades, análisis de protocolos, escaneo de puertos, exploits, web-cgi, virus, SQL injections, etc., todo esto en tiempo real.

Se encuentra disponible bajo licencia GPL, es decir, posibilita la modificación y redistribución del software, pero únicamente bajo esa misma licencia. Es gratuito y funciona bajo plataformas Windows y UNIX/Linux.

Con millones de descargas hasta la fecha, Snort es posiblemente la tecnología en sistemas de detección de intrusos (IDS) más conocida. Su sistema en la detección y prevención de ataques se ha convertido en el estándar de facto para instituciones que utilizan software libre.

La colocación de Snort en la red puede realizarse según el tráfico que se quiere vigilar: paquetes entrantes, salientes, dentro del firewall, fuera de él, etc. Snort es un programa capaz de cumplir las siguientes funciones (Ref. 39):

- Modo sniffer: Es decir, atendiendo tanto a los paquetes dirigidos a dicha interfaz como a los que no le correspondería escuchar y mostrar dicho tráfico. Este modo de operación se asemeja al programa "tcpdump" y es de hecho muy similar a él, hasta el punto de que comparte el sistema de registro en el disco.
- Modo packet logger (registro de paquetes): Almacena en un sistema de logs toda la actividad de la red en que se encuentre instalado.
- Modo IDS: En el que se motoriza por pantalla o en un sistema basado en logs, toda la actividad de la red a través de un archivo de configuración en el que se especifican las reglas y patrones a filtrar para estudiar los posibles ataques.

Snort no se limita exclusivamente al protocolo IP, TCP, UDP o ICMP aunque sobresalga su especialización en ellos. También es capaz de escuchar otros protocolos como Ethernet, FDDI, ARP, entre otros.

Para ser capaz de filtrar y analizar el tráfico lo mejor posible, Snort lleva una serie de módulos que le permiten:

- Recomponer el tráfico fragmentado
- Entender el tráfico http, como p.ej. la expansión de caracteres " " = "%20"
- Detectar escaneos remotos en busca de puertos abiertos

Además está diseñado de modo que sea fácil incorporarle nuevos módulos para expandir su funcionalidad.

4.3.2.2 Funcionamiento del motor de Snort.

Se divide en los siguientes componentes:

- Decodificador del paquete.
- Preprocesadores.
- Motor de detección (Comparación contra firmas).
- Loggin y sistema de alerta.
- Plugins de salida.

El *decodificador de paquete* toma los paquetes de diferentes tipos de interfaces de red y prepara el paquete para ser preprocesado o enviado al motor de detección.

Los *preprocesadores* son componentes o plugins que pueden ser usados con Snort para arreglar, rearmar o modificar datos antes que el motor de detección haga alguna operación para encontrar si el paquete esta siendo enviado por un intruso. Algunos preprocesadores realizan detección buscando anomalías en las cabeceras de los paquetes y generando alertas. Son muy importantes porque preparan los datos para ser analizados en base a las reglas en el motor de detección.

El *motor de detección* es el responsable de detectar si alguna actividad de intrusión existe en un paquete. El motor utiliza las reglas que han sido definidas para este propósito. Las reglas (o cadenas) son comparadas contra todos los paquetes. Si un paquete concuerda con una regla, la acción configurada en la misma es ejecutada.

Dependiendo que detecte el motor dentro de un paquete el *logging y sistema de alerta*, se encarga de generar una alerta. Los logs son almacenados en archivos de texto, archivos con formato tcpdump u otro formato.

Los *plugins de salida* toman la salida del sistema de alerta y permiten almacenarlas en distintos formatos o reaccionar antes el mismo (syslog).

4.3.2.3 Reglas Snort.

Una característica que consideramos importante en Snort es la capacidad de crear reglas que se ocupen de objetivos particulares de la organización. El lenguaje de las reglas cumple con este fin, al ser simple y flexible. Snort utiliza direcciones IP, puertos, datos del encabezado y contenido del paquete para comparar con sus reglas.

En el modo de operación de IDS (no en línea), existen 5 tipos de reglas.

- Reglas de alerta. En caso de coincidencia, el evento se registrara y generara una alerta, en el archivo de alertas.
- Reglas de paso. No se toma en cuenta el tráfico que coincida con los criterios de este tipo de regla.
- Reglas de registro. El evento se registra, pero no se genera una alerta.
- Reglas dinámicas. Son reglas que se activan, en función de otras reglas (de activación) y una vez activas, funcionan como reglas de registro.
- Reglas de activación. Cuando un paquete cumple con el criterio, este tipo de regla crea una alerta y activa una regla dinámica.

El orden de comparación predeterminado en el que Snort evalúa las reglas es: reglas de alerta, reglas de paso y al final reglas de registro.

Sintaxis.

Las reglas de Snort están formadas por dos partes: el encabezado y las opciones. El encabezado contiene la siguiente infamación:

- Tipo de regla (alert, log, pass).
- Protocolo (IP, UDP, ICMP o TCP).
- Operador de dirección (-> ,<>)

- Direcciones IP origen y destino. Estas se pueden especificar mediante un numero de subred y la longitud del prefijo o bien individualmente con su representación de 32 bits, e inclusive para mas de una dirección se puede escribir una después de la otra separadas por comas, adicionalmente se puede utilizar el operador ”!” como negación.
- Puertos. El puerto origen o destino de la comunicación, parra definir un rango se utiliza ":".

La parte de opciones de la regla contiene, el mensaje asociado al evento o el contenido a comparar con el paquete. Las opciones están siempre entre paréntesis y siguen la sintaxis (<palabra clave> <valor>). Algunas de las palabras clave son:

- Msg. Se utiliza, para especificar un mensaje en las alertas, o registros.
- Pcre. Aplica una expresión regular, compatible con perl al paquete.
- Flags. Compara con las banderas de TCP.
- Logto. Los paquetes que cumplen con esta regla, se envían a un archivo de bitácora distinto (especificado en la misma regla) al predeterminado.

<tipo> <protocolo> <IP y puerto origen> <dirección> <IP y puerto destino> (<opciones>)

4.3.2.4 REQUISITOS DEL NIDS

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. Para ello, el sistema debe cubrir características especiales de hardware que permita el procesamiento de los paquetes de forma rápida y óptima. En este trabajo, utilizamos un equipo con las siguientes características:

> Procesador Pentium 4

> Disco Duro de 80 GB

> Memoria RAM de 1000 MB

> Tarjetas de red Gigabit Ethernet 10/100/1000

> Cable UTP cat 6e

4.4 IMPLEMENTACIÓN

La instalación en el disco duro de esta distribución se lleva a cabo por medio de una interfaz grafica. Para particionar el disco se utilizó fdisk, se inició el equipo con el live CD y se ejecutó el siguiente comando #fdisk /dev/hda

Se requieren tres particiones. Una para alojar la carpeta boot, otra para el área de intercambio de memoria a disco (swap) y la tercera para el sistema.

```
Command (m for help): p

Disk /dev/hda: 6442 MB, 6442450944 bytes
255 heads, 63 sectors/track, 783 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1               1          62      497983+  83  Linux
/dev/hda2              63         575     4120672+  82  Linux swap
/dev/hda3             576         783     1670760   83  Linux
```

Fig. 4.8 Sistema de particiones

Con las particiones creadas se inicia el instalador de Backtrack, y se llena con los parámetros correspondientes.

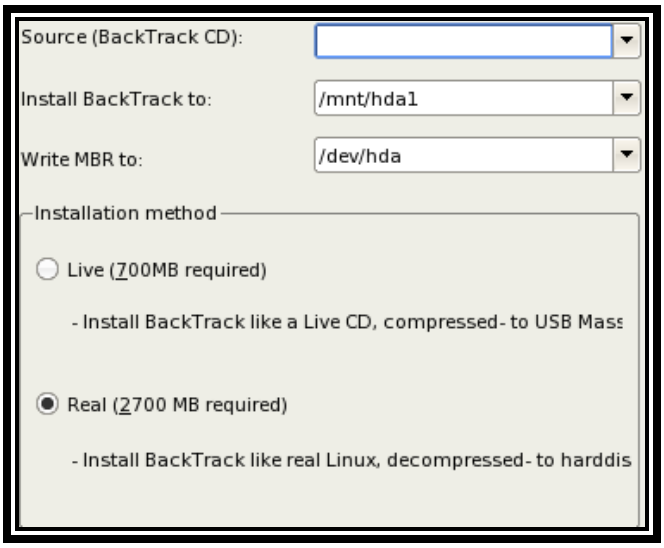


Fig. 4.9 Instalador de BT

Una vez terminada la instalación del sistema operativo, se realizaron configuraciones adicionales para evitar introducir vulnerabilidades a la red.

Para determinar como realizar el hardening adecuado al equipo fue necesario analizar las posibles amenazas. En primer lugar el acceso a los equipos críticos como switches, servidores, firewall y el mismo NIDS se ubican en un cuarto con acceso restringido. Como se abordó anteriormente en este trabajo la seguridad física es la primer barrera de defensa, cumpliendo con la política actual el equipo posee su contraseña correspondiente.

Se verificaron los servicios ejecuta en el sistema. Los procesos que no son necesarios y que poseen un puerto UDP o TCP accesible desde la red fueron eliminados. En sistemas Linux, el kernel se encarga de iniciar los procesos que controlan el hardware y después las aplicaciones en sus diferentes niveles de ejecución (0-6).

El proceso de arranque de los sistemas operativos Linux cambia ligeramente de distribución en distribución, la característica principal es que esta secuencia esta controlada por scripts, si se eliminan o se comentan las líneas que controlan a un servicio en particular, en el siguiente inicio del sistema dicho servicio no iniciará.

Una vez instalado el sistema operativo y realizado el aseguramiento correspondiente se ejecutan los scripts de instalación. Backtrack (BT) cuenta con scripts que permiten la instalación automatizada de Snort, base de datos (MySQL), y entorno web para visualizar alertas (Basic Analysis and Security Engine, BASE). Dichas instrucciones son: /usr/bin/setup-snort

```
************************************************
* Snort / MySQL / Base Setup and Initialization
* muts@offensive-security.com
* Please Read Instructions Carefully
************************************************\n
Please enter desired MySQL root password:
root
Please enter desired MySQL snort user password:
toor
Setting up Snort...Please be patient.
Installing all prepared tables
Fill help tables

To start mysqld at boot time you have to copy support-files/mysql.server
to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
To do so, start the server, then issue the following commands:
/usr/bin/mysqladmin -u root password 'new-password'
/usr/bin/mysqladmin -u root -h bt password 'new-password'
See the manual for more instructions.

You can start the MySQL daemon with:
cd /usr ; /usr/bin/mysqld_safe &

You can test the MySQL daemon with the benchmarks in the 'sql-bench' directory:
cd sql-bench ; perl run-all-tests

Please report any problems with the /usr/bin/mysqlbug script!

The latest information about MySQL is available on the web at
http://www.mysql.com
Support MySQL by buying support/licenses at http://shop.mysql.com
* Setting up permissions on MySQL.
* Linking
* Starting MySQL server.
* Setting a Mysql root password.
* Creating a MySQL Snort User.
* Importing Snort Database into MySQL.
* Starting Apache Web Server.
/usr/local/apache/bin/apachectl start: httpd started
* Setting up snort.conf
* Starting Snort.
Done! - Please read the instructions to come...
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

The BASE web-frontend has been setup and is now running.

Please visit: http://192.168.1.67/base/base_db_setup.php
to complete the configuration.

1) Click Create BASE AG on the far right side
2) Click Main Page link above Alert Group Maintenance
```

Fig. 4.10 Script para la instalación de Snort

Para activar Snort y todos los programas que necesita, se ejecuta el script /usr/bin/start-snort-all

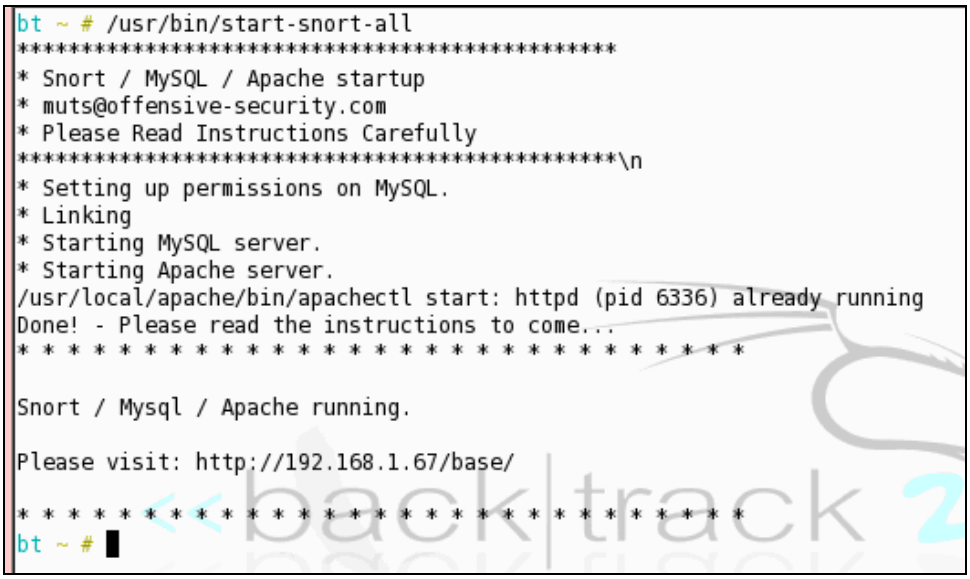


Fig. 4.11 Script para iniciar dependencia de Snort

Debido a que el equipo se encuentra conectado en un punto crítico de la red, representa un punto de interés para cualquier atacante. Es por ello que la configuración original sufrió modificaciones adicionales. Sin pretender ser exhaustivos sino efectivos en el fortalecimiento de la seguridad del equipo, se detallan algunas medidas tomadas para cumplir con el nivel aceptable de seguridad.

Como primer paso, se realizó un escaneo de puertos hacia el equipo recién instalado con Snort funcionando.

Fig. 4.12 Comando nmap sobre el equipo

Como se muestra los procesos que utilizan puertos TCP son el servidor web y de bases de datos, así como el sistema X11 y el servicio IPP que lleva a cabo funciones de impresión en los sistemas Linux.

Éste último no sólo es innecesario, sino que tiene una larga historia de vulnerabilidades asociadas. Este proceso es iniciado cuando el sistema se enciende, por lo tanto fueron modificados los scripts de inicio del sistema.

Se llevó a cabo un escaneo de vulnerabilidades utilizando la herramienta Nessus.

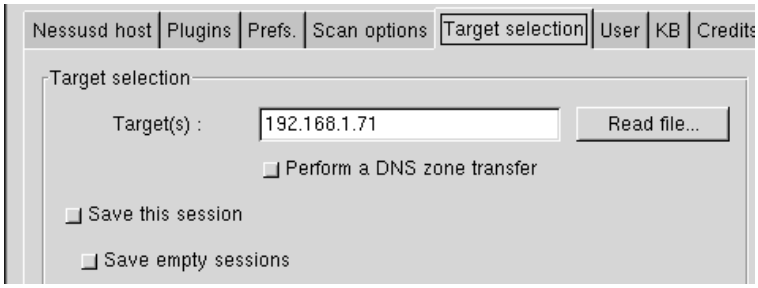


Fig. 4.13 Ejecución de Nessus en el equipo

Ejecutamos Nessus con todos los plug-ins, actualizados. Como se muestra en la figura con esta configuración obtenemos solo advertencias, ninguna vulnerabilidad crítica.

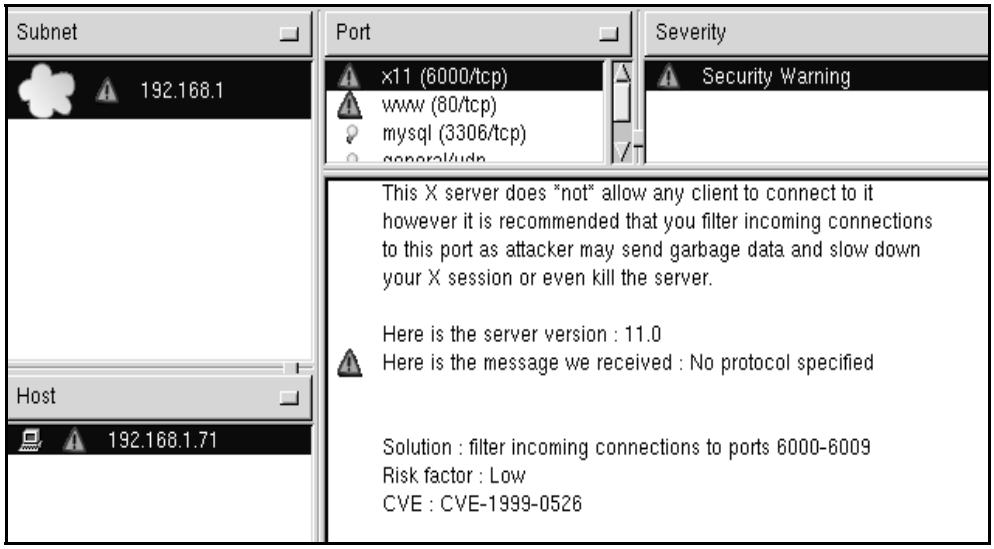


Fig. 4.14 Vulnerabilidades encontradas por Nessus

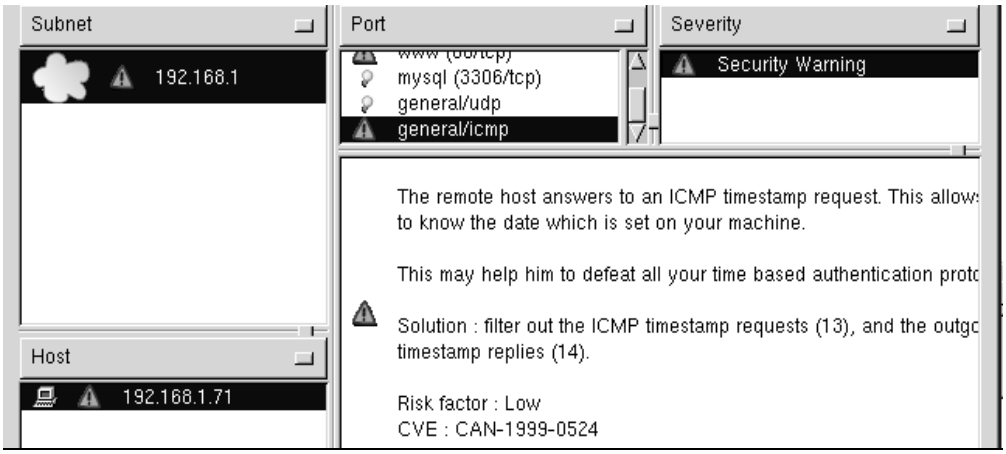


Fig. 4.15 Recomendaciones en el filtrado de mensajes ICMP

Después de revisar los puertos que esperan conexiones, determinar cuáles son necesarios para la aplicación y verificar las vulnerabilidades y posibles medidas para corregirlas, podemos eliminar algunos servicios y filtrar otros, revisamos el archivo inittab.

```
bt / # cat /etc/inittab | more
#
# inittab       This file describes how the INIT process should set up
#               the system in a certain run-level.
#
# Version:      @(#)inittab             2.04    17/05/93        MvS
#                                       2.10    02/10/95        PV
#                                       3.00    02/06/1999      PV
#                                       4.00    04/10/2002      PV
#
# Author:       Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
# Modified by:  Patrick J. Volkerding, <volkerdi@slackware.com>
#

# These are the default runlevels in Slackware:
#   0 = halt
#   1 = single user mode
#   2 = unused (but configured the same as runlevel 3)
#   3 = multiuser mode (default Slackware runlevel)
#   4 = X11 with KDM/GDM/XDM (session managers)
#   5 = unused (but configured the same as runlevel 3)
#   6 = reboot

# Default runlevel. (Do not set to 0 or 6)
id:3:initdefault:
```

Fig. 4.16 Contenido del archivo inittab

Este archivo indica los niveles de ejecución que los procesos siguen cuando se inicia el equipo (del nivel 1 al 4) siendo de importancia para la configuración del equipo el nivel 3 multiusuario.

```
# Script to run when going single user (runlevel 1).
su:1S:wait:/etc/rc.d/rc.K

# Script to run when going multi user.
rc:2345:wait:/etc/rc.d/rc.M
```

Fig. 4.17 Niveles de ejecución

Para detener un proceso una vez iniciado el sistema se debe modificar para esta distribución, el archivo /etc/rc.d/rc.M que corresponde con el nivel de ejecución multiusuario. En este archivo se encuentran las instrucciones para iniciar diferentes procesos entre ellos algunos para iniciar controladores de dispositivos. Para evitar el inicio del proceso IPP basta con ubicar las líneas encargadas de su ejecución y comentarlas.

```
# Start the print spooling system.  This will usually be LPRng (lpd) or CUPS.
if [ -x /etc/rc.d/rc.cups ]; then
  # Start CUPS:
  /etc/rc.d/rc.cups start
elif [ -x /etc/rc.d/rc.lprng ]; then
  # Start LPRng (lpd):
  . /etc/rc.d/rc.lprng start
fi
```

Fig. 4.18 Contenido del archivo /etc/rc.d/rc.M

Después de reiniciar el equipo se crearon las siguientes reglas en el firewall personal con la herramienta IPTABLES .para permitir el acceso restringuido al equipo, es decir, sólo la IP proporcionada puede controlar la aplicación.

```
bt ~ # iptables -A INPUT -s 192.168.1.70 -p tcp -d 192.168.1.71 --dport 80 -j ACCEPT
bt ~ # iptables -A INPUT -s 0.0.0.0/0 -d 192.168.1.71 -j DROP
```

Fig. 4.19 Creación de reglas con IPTables

Un nuevo escaneo de puertos nos devuelve la siguiente salida.

```
bt / # nmap -sT -T 5 192.168.1.71

Starting Nmap 4.20 ( http://insecure.org ) at 2008-10-19 17:23 GMT
All 1697 scanned ports on 192.168.1.71 are filtered

Nmap finished: 1 IP address (1 host up) scanned in 109.751 seconds
bt / #
```

Fig. 4.20 Escaneo de puertos con nmap

Nessus tampoco puede encontrar al equipo, ya que con IPTABLES se prohíbe todo el tráfico de cualquier protocolo hacia el equipo. Si bien estas medidas no detendrán del todo a un atacante, harán más difícil su acción, obligándolo a utilizar más herramientas y llevar a cabo más acciones, incrementando las probabilidades de ser detectado.

Snort cuenta con un conjunto de reglas que alertan sobre tráfico malicioso y que es recomendable mantener actualizadas. Las reglas están organizadas en archivos de texto, con nombres descriptivos, éstos indican que son reglas asociadas a un protocolo, aplicación o tecnología, esto facilita su organización y análisis

Fig. 4.21 Estructura de las reglas en Snort

Dentro de este esquema el archivo \etc\snort\rules\local.rules, está destinado a alojar las reglas exclusivamente creadas a nuestro contexto. En esta etapa para actualizar las reglas dado que no hemos escrito aún las propias, basta con obtener la última versión a la que tenemos acceso. De acuerdo con las políticas de snort.org, se descargan y descomprimen, para posteriormente copiarlas en la carpeta correspondiente /etc/snort/rules.

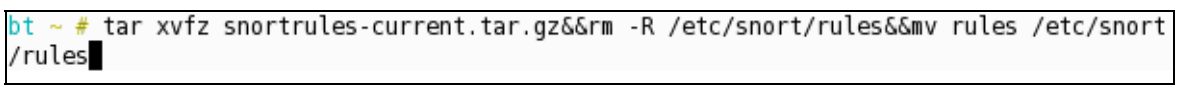


Fig. 4.22 Actualización de reglas

4.5 INTEGRACIÓN

La Unidad de Servicios de Cómputo Administrativos (USECAD) tiene como objetivos específicos: desarrollar los sistemas de cómputo que soporten las actividades académico-administrativas que generan los órganos de la Facultad, contar con sistemas de información flexibles en su operación y de fácil mantenimiento, diseñar e incorporar nuevos productos y servicios que permitan el cumplimiento de los objetivos planteados, y garantizar la operación normal de los equipos de cómputo de la Secretaría de Servicios Académicos (Ref. 40).

En este contexto, se cuenta con mecanismos de seguridad para conseguir que las actividades de la S.S.A resulten productivas y alcancen sus objetivos. Entre otros, el proceso de inscripciones es llevado a cabo por la USECAD, el NIDS fue puesto en marcha durante el proceso correspondiente al semestre 2009-1.

El NIDS busca integrar un control más a este esquema y servir como una fuente de eventos para la identificación de incidentes de seguridad.

La configuración del firewall, fue nuestra guía para la creación de reglas orientadas a vigilar el tráfico que alcanza la subred de servidores de la S.S.A. La estrategia del firewall consiste en prohibir cualquier conexión y solo permitir aquello que se indique explícitamente, sin duda esta filosofía facilita la revisión de su funcionamiento.

Por una parte tenemos reglas de ataques conocidos, y por otra escribimos reglas análogas a la configuración del firewall. De esta manera si el equipo encargado de filtrar tuviése algún fallo, el IDS que es un sistema independiente se encargará de notificarnos dicho evento.

La aplicación que se utiliza para filtrar paquetes en el firewall, es Packet Filter de BSD. Entre sus características se encuentran: capacidades de traducción de direcciones de red (NAT), priorización de paquetes y filtrado de paquetes con tabla de estado. Las decisiones se toman con base en información de capa 3 (direcciones IP origen-destino, protocolos embebidos) y capa 4 (puertos TCP/UDP origen-destino).

La forma en la que se procedió para lograr una configuración propicia fue analizar el tráfico permitido y crear reglas para detectar la condición contraria a la correspondiente prohibición del firewall. Una vez que el IDS está ejecutando el juego de reglas recién definido utilizamos la herramienta hping3 (Ref. 41). Esta permite crear paquetes TCP, UDP o ICMP de manera prácticamente arbitraria, para comprobar su eficacia.

Para ilustrar el procedimiento, presentamos algunas de las reglas del firewall e IDS, y su correspondiente prueba con hping3 para verificar el funcionamiento de la configuración. Esto se verificó para cada regla del firewall. En este trabajo solo se documentan algunas reglas.

El grupo de administradores y los autores de este trabajo decidimos no presentar toda la configuración tanto del IDS como del firewall, por cuestiones de seguridad. Sólo se tomaron algunos de los casos representativos utilizando protocolos TCP, UDP e ICMP.

- **4.5.1 Plan de Pruebas**

En la instalación de controles de seguridad una etapa sumamente importante consiste en la certificación del funcionamiento de la medida que se esta implementando. Inducir una condición (normalmente en conflicto con una política) y analizar la forma en la que se comporta el dispositivo o medida en cuestión para evaluar su cumplimiento con nuestras expectativas. Este subtema documenta el método empleado dejando el análisis de los resultados para las secciones siguientes.

Previo a esta etapa de pruebas se realizó el análisis de las reglas del firewall. Se creó una regla o conjunto de reglas para cada una de las políticas especificadas para el NIDS, ya con las opciones de la herramienta de distribución libre Hping (que permite crear paquetes tcp, udp o icmp con valores arbitrarios en las cabeceras) se formaron los comandos necesarios para crear paquetes con valores catalogados como trafico no autorizado, que después serán enviados por la red.

Para esta etapa se planteo el siguiente escenario: se activo el monitoreo de puertos en el switch, se puso en funcionamiento el NIDS con las reglas definidas para detectar trafico no autorizado por el firewall, en el puerto encargado de monitorear y en una de los equipos conectados al mismo switch instalamos la herramienta, Hping.

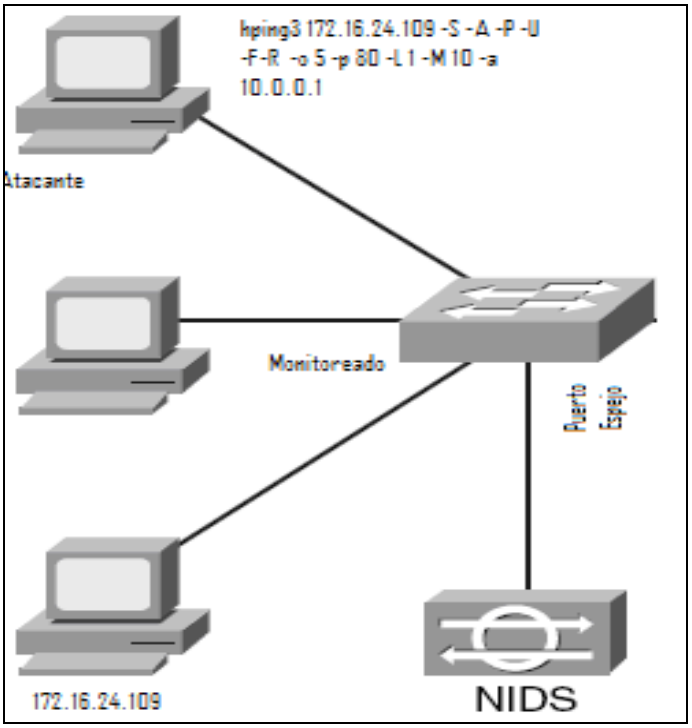


Fig. 4.23 Diagrama de Pruebas

Esta configuración nos permite simular, la situación en la que el firewall falla y permite pasar todo el tráfico a la red de servidores. Desde el equipo etiquetado como atacante creamos paquetes que intencionalmente rompen las políticas de tráfico permitido (puertos y subredes no autorizados etc) y los enviamos por la red. El NIDS detecta que el tráfico coincida con una violación a la política autorizada y genera una alerta, esto se realizó para cada una de las reglas.

- ***4.5.2 Análisis de Reglas***

pass in log quick on $ext_if proto tcp from { 172.16.24.105, 172.16.24.106, 172.16.24.108, 172.16.24.102, 172.16.24.104 } to 172.16.24.109 port { 22, 80, 443, 4000, 8080, 8089 } keep state

La regla define un conjunto de direcciones (172.16.24.105, 172.16.24.106, 172.16.24.108, 172.16.24.102, 172.16.24.104) autorizadas para comunicarse con los puertos 22, 80, 443 de la dirección 172.16.24.109.

En el archivo */etc/snort/rules/snort.conf*, se declararon las siguientes variables para facilitar la redacción y mantenimiento de las reglas.

#Direcciones autorizadas
var R_1 [172.16.24.105, 172.16.24.106, 172.16.24.108, 172.16.24.102, 172.16.24.104]

#Direcciones con perfil de administración
var R_1b [172.16.24.105, 172.16.24.108]

#Direcciones con perfil de desarrollo de aplicaciones
var R_1c [172.16.24.102, 172.16.24.104]

El archivo */etc/snort/rules/local.rules*, contiene la regla que hace uso de las variables definidas.

alert tcp !$R_1 any -> 172.16.24.109 any (msg:"Acceso no autorizado a Base de datos"; sid:100001;rev:1;))

Esta regla inspecciona paquetes con dirección IP origen distinta a las autorizadas y dirección IP destino 172.16.24.109. Esta condición es contraria a la existente en el firewall, sin embargo la condición debe ser complementada con el caso en el que el tráfico proviene de direcciones autorizadas hacia puertos no autorizados.

Con el fin de crear reglas precisas se crean condiciones que evalúen, que aun proviniendo de direcciones autorizadas se dirijan al puerto adecuado. Esto se logra con el siguiente bloque de reglas:

#Bloque de excepciones (comunicación permitida)

pass tcp $R_1b any ->172.16.24.109 22

pass tcp $R_1b any -> 172.16.24.109 80

pass tcp $R_1b any -> 172.16.24.109 443

pass tcp 172.16.24.106 any -> 172.16.24.109 80

pas tcp $R_1c any -> 172.16.24.109 22

pass tcp $R_1c any -> 172.16.24.109 80

#Prohibición de cualquier otra combinación de IP autorizada

alert tcp $R_1 any -> 172.16.24.109 any (msg:"Acceso Interno no autorizado a Base de datos";sid:100002;rev:1;)

Una de las condiciones de falla se puede generar lanzando paquetes desde un equipo que no sea filtrado por el firewall con la herramienta hping3. Para verificar que no ocurriera tal evento creamos un paquete con las siguientes características: dirección IP origen (-a) 10.0.0.1 (no autorizada), IP destino 172.16.24.109 puerto destino (-p) 80 y banderas de tcp activas Syn, Ack, Push, Fin, Reset, Urgent, la instrucción es:

hping3 172.16.24.109 -S -A -P -U -F -R -o 5 -p 80 -L 1 -M 10 -a 10.0.0.1

La figura muestra la pantalla que se genera cuando el IDS encuentra la coincidencia con la regla.

ID #	Time	Triggered Signature
1 - 1209	2008-12-01 14:05:59	[local] [snort] Acceso no autorizado a Base de datos

Sensor	Sensor Address	Interface	Filter
	192.168.1.66	eth0	*none*

Alert Group	*none*

Source Address	Dest. Address	Ver	Hdr Len	TOS	length	ID	DF	MF	offset	TTL	chksum
10.0.0.1	172.16.24.109	4	20	5	40	16691			0	64	27418

Options	*none*

Source Port	Dest Port	R1	R0	URG	ACK	PSH	RST	SYN	FIN	seq #	ack	offset	res	window	urp	chksum
3626 [sans] [portsdb] [tantalo] [sstats]	80 [sans] [portsdb] [tantalo] [sstats]			X	X	X	X	X	X	10	1	5	0	1	0	53921

Options	*none*

Fig. 4.24 Vista del reporte generado

Se observa el mensaje que describe el evento definido en el cuerpo de la regla.

Triggered Signature
[local] [snort] Acceso no autorizado a Basede datos

Fig. 4.25 Mensaje del evento generado

El encapsulado de IP, muestra que el paquete viene de una dirección 10.0.0.1, y se dirige a 172.16.24.109. Dado que la dirección origen no está dentro de la lista de direcciones autorizadas se genera la alerta.

Source Address	Dest. Address	Ver	Hdr Len	TOS	length	ID	DF	MF	offset	TTL	chksum
10.0.0.1	172.16.24.109	4	20	5	40	16691			0	64	27418

Fig. 4.26 Desglose de cabecera IP

La cabecera de TCP indica además de los puertos origen y destino las banderas de TCP que se activaron durante la construcción del paquete.

Source Port	Dest Port	R 1	R 0	U R G	A C K	P S H	R S T	S Y N	F I N	seq #	ack	offset	res	window	urp	chksum
3626 [sans] [portsdb] [tantalo] [sstats]	80 [sans] [portsdb] [tantalo] [sstats]			X	X	X	X	X	X	10	1	5	0	1	0	53921

Options *none*

Fig. 4.27 Desglose de cabecera TCP

Como observa, se analizó una alerta TCP. Continuamos con una regla UDP que permite a la dirección IP 172.16.24.122, y que tiene por objetivo realizar consultas de resolución de nombres.

pass out log quick on $ext_if proto udp from 172.16.24.122 to any port { 53 } keep state

La correspondiente regla de Snort es:

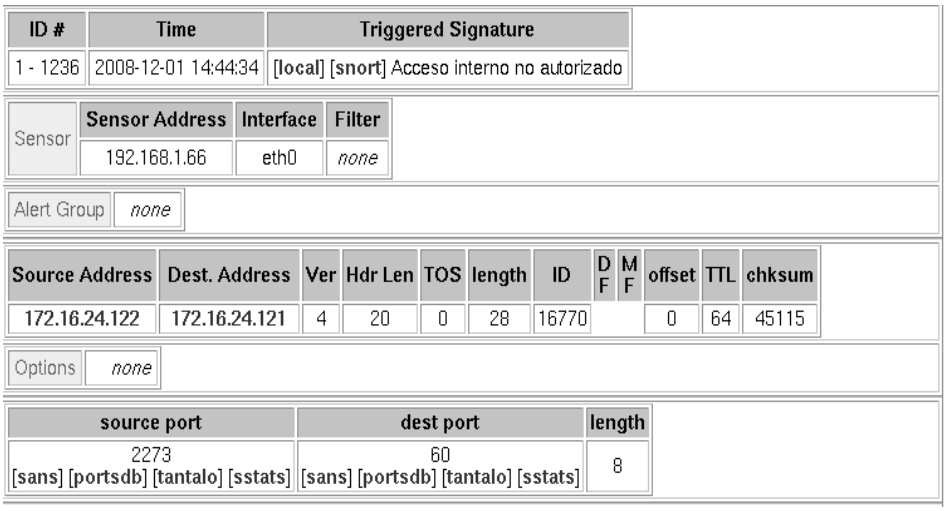
alert udp 172.16.24.122 any -> any !53 (msg:"Acceso interno no autorizado"; sid:200001;rev:1;))

La regla indica que un paquete con dirección origen 172.16.24.122, protocolo UDP y puerto destino diferente al 53, generaran alertas La condición de error se puede crear con hping3 de la siguiente forma:

hping3 172.16.24.121 -2 -a 172.16.24.122 --destport 60

Este paquete tendrá como dirección destino 172.16.24.121, utilizará como protocolo de transporte UDP (-2), dirección origen (-a) 172.16.24.122 y puerto destino (--destport) 60. Esto significaría que el servidor está enviando información a un puerto no autorizado.

La figura muestra la alerta que se genera cuando el paquete es detectado por el IDS.



ID #	Time	Triggered Signature
1 - 1236	2008-12-01 14:44:34	[local] [snort] Acceso interno no autorizado

Sensor	Sensor Address	Interface	Filter
	192.168.1.66	eth0	*none*

Alert Group	*none*

Source Address	Dest. Address	Ver	Hdr Len	TOS	length	ID	DF	MF	offset	TTL	chksum
172.16.24.122	172.16.24.121	4	20	0	28	16770			0	64	45115

Options	*none*

source port	dest port	length
2273 [sans] [portsdb] [tantalo] [sstats]	60 [sans] [portsdb] [tantalo] [sstats]	8

Fig. 4.28 Vista de la alerta generada

La alerta incluye una marca de tiempo y el mensaje definido en el cuerpo de la regla.

ID #	Time	Triggered Signature
1 - 1236	2008-12-01 14:44:34	[local] [snort] Acceso interno no autorizado

Fig. 4.29 Alerta del mensaje generado

La cabecera de IP muestra las direcciones origen y destino. Hasta este punto, no existe coincidencia con la regla del IDS.

Source Address	Dest. Address	Ver	Hdr Len	TOS	length	ID	DF	MF	offset	TTL	chksum
172.16.24.122	172.16.24.121	4	20	0	28	16770			0	64	45115

Options	*none*

Fig. 4.30 Desglose de cabecera IP

La cabecera UDP nos muestra que el puerto destino es inaccesible ya que la regla indica que solo se permite como puerto destino el 53. Cualquier otro genera una alerta para el IDS y un evento para los administradores.

source port	dest port	length
2273 [sans] [portsdb] [tantalo] [sstats]	60 [sans] [portsdb] [tantalo] [sstats]	8

Fig. 4.31 Desglose de cabecera UDP

Con el objetivo de identificar las alertas, se destaca en el cuerpo de la misma un identificador único (SID), y un número de revisión (REV) que sirve para depurar la regla y notificar cambios en la implementación original.

Capítulo 5

Resultados Obtenidos

La instalación de un NIDS en USECAD es otra medida de seguridad dado que se cuenta herramientas preventivas, políticas claras respecto al tráfico autorizado y un equipo de administradores proponiendo y mejorando constantemente las estrategias de seguridad.

El esfuerzo invertido en la instalación e implementación de reglas, análisis de la red y necesidades, además de la configuración de los equipos de interconexión que nos permiten monitorear tráfico, sólo representa un esfuerzo inicial en la detección de intrusos. A pesar de que esta técnica es una de las que mayor auge ha tenido en el ámbito de la seguridad, su integración en cualquier escenario no resulta una tarea libre de trabajo continuo.

El paso siguiente para la USECAD, es evaluar periódicamente el funcionamiento de las reglas bajo las diferentes condiciones de tráfico, propias de las necesidades de comunicación y procesamiento. Ajustar el **N**IDS a los cambios que requiera la red al paso del tiempo y entrenar al grupo encargado de monitorear las alertas para interpretar adecuadamente cuando éstas representan un evento o un incidente.

A pesar del trabajo que representa la administración de un IDS, los beneficios potenciales son muchos. El NIDS representa una fuente de conocimiento del tráfico de la red, una herramienta para verificar el funcionamiento correcto de otros dispositivos así como para realizar análisis forense. El uso de herramientas de detección de intrusos no presenta resultados inmediatos a la dependencia, es una inversión (en tiempo y esfuerzo) a mediano y largo plazo.

Obteniendo como beneficios no sólo la capacidad de reacción oportuna por parte de un administrador, sino también la de familiarizarse con las problemáticas de tráfico para que en un futuro, si la institución lo considera pertinente, despliegue un control de tipo preventivo automatizado (Sistema de Prevención de Intrusos, IPS).

El NIST (National Institute of Standars and Technology) agencia de gobierno de los E.U.A., encargada de establecer y proveer estándares de diversos tipos ha documentado el ciclo de respuesta a incidentes (abordado en el Capítulo III), un plan dividido en etapas ampliamente aceptado en la industria de la seguridad informática. Éstas consisten en:

- Preparación.
- Identificación.
- Contención.
- Erradicación.
- Recuperación
- Mejora

Es importante ubicar nuestra implementación dentro de este marco de trabajo, para resaltar los beneficios obtenidos. Durante la elaboración de este proyecto, y debido a la naturaleza de un IDS, las primeras etapas del ciclo de respuesta a incidentes, son los puntos en los que hemos trabajado en gran medida.

Preparación, analiza lo que el departamento considera como trafico permitido. Este punto fue fundamental para enfocar nuestro trabajo y plantear un objetivo para el IDS. La etapa de preparación también contempla la asignación de la tarea o creación de un puesto dentro de la USECAD con la finalidad de revisar periódicamente las alertas generadas, así como el mantenimiento del IDS.

Dentro del ciclo de respuesta a incidentes, es la parte de detección fundamental para este capítulo. La introducción de este nuevo control de seguridad a la estrategia de administración de la red proporcionada por USECAD, monitorea a nivel de capa tres las actividades hacia la red de servidores.

Esto arroja en una serie de eventos para que un analista declare un incidente o realice un ajuste en el funcionamiento en la red. Si bien, estas primeras etapas son las más relevantes para el IDS, también existen ventajas en las etapas subsecuentes.

Recuperación consiste en restaurar las operaciones normales de un sistema, se requiere un monitoreo exhaustivo para verificar que se ha eliminado cualquier rastro de intrusión.

Colocar reglas para supervisar la actividad de ciertos equipos es una actividad propia de un IDS. En la última etapa, en donde se llega a la aprobación de la existencia de un incidente, se requiere conocer qué fue lo que ocurrió y de qué manera se llevó a cabo. Los datos existentes en los logs de los servidores y registros del IDS, resultan un cúmulo de información valiosa para conocer más sobre un incidente, e integrar las estrategias apropiadas para evitar que se repita.

El detector de intrusos fue instalado en el proceso de inscripciones 2009-1, durante el cual no hubo contratiempos causado por usuarios no autorizados (intrusiones). Sobresale una falla detectada en DGSCA, durante la cual hubo conexión intermitente, ya que ésta última provee a USECAD de una acometida de fibra óptica.

El firewall está diseñado para bloquear conexiones que no se hayan establecido, por lo tanto no podemos saber qué tipo de tráfico estamos recibiendo. Se colocó un sensor del IDS antes del firewall con la finalidad de conocer más las conexiones dirigidas a la DMZ, esto proporciona a los administradores la capacidad de modificar su entorno de seguridad dependiendo de los resultados observados.

Un segundo sensor colocado después del firewall y en el cual se enfocó nuestro análisis fue el encargado de alertarnos acerca de posibles fallas en el firewall, refiriéndonos a fallas los posibles contratiempos que se producen en ocasiones al pfctl de OpenBSD. Comprobando con este último sensor la seguridad proporcionada por el firewall.

La configuración utilizada durante este trabajo nos permite tener pre-instaladas varias herramientas de monitoreo de redes, una de ellas Ntop. Con dicha utilidad podemos obtener información valiosa acerca de la utilización del ancho de banda por los usuarios, aplicaciones, etc. El administrador en turno puede hacer uso de éstos beneficios sin tener que hacer una complicada configuración.

Capítulo 6

Conclusiones

Esta tesis surgió dentro de la Unidad de Servicios de Computo Académico de la Facultad de Ingeniería (USECAD) con el objetivo de contribuir a la dinámica emprendedora de los administradores del departamento, dado que día a día participan activamente proponiendo mejoras y nuevas técnicas en materia de administración de redes y seguridad informática.

En este trabajo se distinguen dos secciones, en los primeros capítulos se tratan temas de redes y conceptos de seguridad a nivel teórico. Siendo congruentes con el espíritu y misión de la UNAM y de la Facultad de Ingeniería que a lo largo de nuestra formación fomenta principios básicos de Ingeniería y Ciencias de la Computación de manera independiente a las tecnologías existentes en el mercado. La segunda parte corresponde a la solución, una implementación de detección de intrusiones en donde se plasman las políticas del departamento y de donde se desprenden los beneficios tangibles para la S.S.A.

El primer objetivo, que era implementar un sistema detector de intrusos en la red de servidores de la SSA durante el proceso de inscripciones fue cumplido con éxito. El sistema puede ser utilizado por el personal autorizado de la USECAD, ya sea para mantenimiento y actualización o para modificaciones en la configuración. Se cuenta con documentos que pueden ser consultados por el Administrador en turno donde se describen el esquema de funcionamiento.

Una ventaja obtenida que no era parte de nuestros objetivos es un nuevo hardening a equipos de red (servidores y switches) que fortalece el entorno de la DMZ.

Se logró un mejor conocimiento de la estructura y funcionamiento de la red, ya que todo sistema debe analizar su entorno de aplicación antes de ser implementado. El descubrimiento de la conexión actual de los servidores es de gran ayuda al administrador, ya que evita pérdida de tiempo localizar un nodo.

Un factor decisivo en la realización del presente documento son las facilidades brindadas por el administrador de la red en cuanto a información, permisos para acceder a las instalaciones y al equipo, el habernos proporcionado todos los elementos necesarios para que pudiésemos cumplir con nuestro proyecto nos llevaron a mostrar éstos resultados.

La aportación de este trabajo a nuestra vida profesional es inmensa. El manejo de dispositivos de comunicaciones, herramientas de monitoreo, interpretación de información y la consulta con personal conocedor del tema, hicieron de esta tesis una grata experiencia que puede ser aplicada en otras ámbitos con enfoques diferentes, pero teniendo como base la capacidad de diseño y análisis que hemos logrado a lo largo de este contenido.

Apéndice A

Licencia GPL "GNU Public License" (GPL)

Licencia GPL "GNU Public License" (GPL)

Versión 2 , Junio 1991

Preámbulo

Las licencias de la mayoría del software están diseñadas para eliminar su libertad de compartir y modificar dicho software. Por contra, la GNU General Public License (GPL) está diseñada para garantizar su libertad de compartir y modificar el software. Software libre para garantizar la libertad de sus usuarios. Esta licencia GNU General Public License (GPL) se aplica en la mayoría de los programas realizado por la Free Software Foundation (FSF, Fundación del Software Libre) y en cualquier otro programa en los que los autores quieran aplicarla. También, muchos otros programas de la Free Software Foundation están cubiertos por la GNU Lesser General Public License (LGPL) e igualmente puede usarla para cubrir sus programas.

Cuando hablamos de Software Libre, hablamos de libertad, no de precio. Nuestra licencia General Public License (GPL) está diseñada para asegurarle las libertades de distribuir cópias de Software Libre (y cobrar por ese servicio si quiere), asegurarle que recibirá el código fuente del programa o bien podrá conseguirlo si quiere, asegurarle que puede modificar el programa o modificar algunas de sus piezas para un nuevo programa y para garantizarle que puede hacer todas estas cosas.

Para proteger sus derechos, necesitamos realizar restricciones que prohíben a cualquiera denegar estos derechos o pedirle que reniegue de sus derechos. Estas restricciones se traducen en ciertas obligaciones por su parte si ustéd piensa distribuir copias del programa o tiene intención de modificarlo.

Por último, cualquier programa está amenazado constantemente por las patentes de software. Desearíamos evitar el riesgo que distribuidores de programas libres adquieran individualmente patentes, transformando de facto el software libre en privativo. Para evitar este problema, dejamos claro que cualquier patente deberá ser licenciada para permitir el uso libre de cualquier persona o no ser patentada.

Los términos y condiciones para la copia, distribución y modificación del software se especifican en el siguiente punto.

SIN GARANTÍA

11. DADO QUE ESTE PROGRAMA ESTÁ LICENCIADO LIBRE DE COSTE, NO EXISTE GARANTÍA PARA EL PROGRAMA, EN TODA LA EXTENSIÓN PERMITIDA POR LA LEY APLICABLE. EXCEPTO CUANDO SE INDIQUE POR ESCRITO, LOS DUEÑOS DEL COPYRIGHT Y/O OTRAS PARTES PROVEEDORAS FACILITAN EL PROGRAMA "TAL CUAL" SI GARANTÍA DE NINGUNA CLASE, NI EXPLÍCITA NI IMPLÍCITA, INCLUYENDO, PERO NO LIMITANDO, LAS GARANTÍAS APLICABLES MERCANTILES O DE APLICABILIDAD PARA UN PROPÓSITO PARTICULAR. USTED ASUME CUALQUIER RIESGO SOBRE LA CALIDAD O LAS PRESTACIONES DEL PROGRAMA. SI EL PROGRAMA TIENE UN ERROR, USTED ASUME EL COSTE DE TODOS LOS SERVICIOS NECESARIOS PARA REPARARLO O CORREGIRLO.

Versión recortada

Apéndice B

Bibliografía y Mesografía

Bibliografía y Mesografía

1. http://www.amipci.org.mx/temp/EstudioAMIPCInuevastecnologiasdeInternetenMexico2008RESUMENEJECUTIVO-01700120012109469550B.pdf
2. SANS Institute 2008, “How to Choose Intrusion Detection Solution”, Biiju Shah, Versión 1.2.e.
3. http://www.wikilearning.com/tutorial/seguridad_en_unix_y_redesclasificacion_de_los_idses/9777-99
4. En páginas posteriores se detallará más del tema.
5. La Licencia Pública General de GNU, llamada comúnmente GNU GPL, la usan la mayoría de los programas de GNU y más de la mitad de las aplicaciones de software libre.
6. La mayor parte de este capítulo fue tomada del libro:

   Tanenbaum Andrew S., “Redes de Computadoras “, ed. Pearson, Cuarta edición, México 2003.
7. Más detalles en el subtema "Direcciones IP y Máscaras de red”.
8. http://www.iana.org
9. http://www.iana.org/assignments/ip-parameters
10. ISO/IEC 17799:2000 - Information technology. Code of practice for information security management.
11. ISO 7498 – Information processing systems – Open systems interconnection
12. http://www.iso27000.es/download/doc_iso27000_all.pdf
13. http://www.rae.es/rae.html
14. William Stallings, et al., “Fundamentos de seguridad en redes, aplicación y estándares”, ed Pearson, Segunda Edición, México 2004.
15. http://es.wikipedia.org/wiki/John_Von_Neumann

16. http://vx.netlux.org/lib/afc11.html
17. La mayor parte de este capítulo fue tomada del libro: William Stallings, et al., “Fundamentos de seguridad en redes, aplicación y estándares”, ed Pearson, Segunda Edición, México 2004
18. El tema IDS se profundiza en el capítulo IV
19. http://www.honeynet.unam.mx/docs/Cecec.pdf
20. L. Spitzner, “Honeypots: Tracking Hackers”. Addislon-Wesley, ISBN from-321-10895-7, 2002.
21. http://www.enterate.unam.mx/Articulos/2007/febrero/honeynet.htm
22. L. Spitzner, “Honeytokens: The Other Honeypot”, 2003. Disponible en línea en: www.securityfocus.com/infocus/1713
23. http://www.ssi.gouv.fr/es/confianza/documents/methods/mementodep-V1.1_es.pdf
24. Glueck, W. y Lawrence, J., 1984; Rue, L. y Lloyd L.B., 1983
25. http://www.sans.org/reading_room/whitepapers/policyissues/1331.php
26. Organisation for Economic Co-operation and Development, Paris y Ministerio de Administraciones Públicas, España
27. http://www.sans.org/reading_room/whitepapers/incident/2068.php
28. La información presentada pertenece al libro: Daltabuit Enrique, et al., “La Seguridad de la Información”, ed. Limusa, México 2007.
29. http://www.sans.org/resources/idfaq/network_based.php
30. www.itstrap.net/descargas/boletines/BoletínITStrap_2_3_23mar08.pdf
31. http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/urbina_p_j/capitulo4.pdf
32. http://www.sans.org/resources/idfaq/fragroute.php
33. http://www.sun.com/bigadmin/features/articles/snort_base.html

34. http://www.rediris.es/cert/doc/unixsec/node26.html

35. W. Richard Stevens. "*TCP/IP Illustrated*, volume 1", Addison Wesley, 1994. Andrew Tanenbaum. "Computer Networks.", Prentice Hall, 1996.

36. http://www.insecure.org/nmap/nmap-fingerprinting-article.html.

37. www.cudi.edu.mx/primavera_2006/presentaciones/wireless02_mario_farias.pdf

38. http://www.networkintrusion.co.uk/index.php/products/IDS-and-IPS/Network-IDS.html

39. http://euitio178.ccu.uniovi.es/wiki/index.php/Snort

40. http://servacad.fi-a.unam.mx/usecad/_info/

41. http://www.hping.org/