

CAPÍTULO 8

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar



“Esto es lo que llamamos Criptograma, en el cual el sentido está oculto bajo letras embarulladas a propósito y que, convenientemente dispuestas, formarían una frase inteligible —dijo el profesor—

Viaje al centro de la tierra. Julio Verne

PROTECCIÓN

Una vez conocidas las vulnerabilidades y ataques a las que está expuesto un sistema es necesario conocer los recursos disponibles para protegerlo. Mientras algunas técnicas son evidentes (seguridad física por ejemplo) otras pautas no lo son tanto e incluso algunas pueden ocasionar una sensación de falsa seguridad.

Muchas de las vulnerabilidades estudiadas son el resultado de implementación incorrecta de tecnologías, otras son consecuencias de la falta de planeamiento de las mismas pero, como ya se ha mencionado, la mayoría de los agujeros de seguridad son ocasionados por los usuarios de dichos sistemas y es responsabilidad del administrador detectarlos y encontrar la mejor manera de cerrarlos.

En el presente capítulo, después de lo expuesto y vistas la gran cantidad de herramientas con las que cuenta el intruso, es el turno de estudiar implementaciones en la búsqueda de mantener el sistema seguro.

Siendo reiterativo, ninguna de las técnicas expuestas a continuación representarán el 100% de la seguridad deseado, aunque muchas parezcan la panacea, será la suma de algunas de ellas las que convertirán un sistema interconectado en confiable.

8.1 VULNERAR PARA PROTEGER

Los intrusos utilizan diversas técnicas para quebrar los sistemas de seguridad de una red. Básicamente buscan los puntos débiles del sistema para poder colarse en ella. El trabajo de los Administradores y Testers no difiere mucho de esto. En lo que sí se diferencia, y por completo, es en los objetivos: mientras que un intruso penetra en las redes para distintos fines (investigación, daño, robo, etc.) un administrador lo hace para poder mejorar los sistemas de seguridad.

En palabras de Julio C. Ardita¹: “(...) los intrusos cuentan con grandes herramientas como los Scanners, los cracking de passwords, software de análisis de vulnerabilidades y los exploits(...) un administrador cuenta con todas ellas empleadas para bien, los Logs, los sistemas de detección de intrusos y los sistemas de rastreo de intrusiones”.

Al conjunto de técnicas que se utilizan para evaluar y probar la seguridad de una red se lo conoce como Penetration Testing, uno de los recursos más poderosos con los que se cuenta hoy para generar barreras cada vez más eficaces.

Un test está totalmente relacionado con el tipo de información que se maneja en cada organización. Por consiguiente, según la información que deba ser protegida, se determinan la estructura y las herramientas de seguridad; no a la inversa

El software y el Hardware utilizados son una parte importante, pero no la única. A ella se agrega lo que se denomina “políticas de seguridad internas” que cada organización (y usuario) debe generar e implementar.

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

8.1.1 ADMINISTRACIÓN DE LA SEGURIDAD

Es posible dividir las tareas de administración de seguridad en tres grandes grupos:

- **Autenticación:** se refiere a establecer las entidades que pueden tener acceso al universo de recursos de cómputo que cierto medio ambiente puede ofrecer.
- **Autorización:** es el hecho de que las entidades autorizadas a tener acceso a los recursos de cómputo, tengan acceso únicamente a las áreas de trabajo sobre las cuales ellas deben tener dominio.
- **Auditoría:** se refiere a la continua vigilancia de los servicios en producción. Entra dentro de este grupo el mantener estadísticas de acceso, estadísticas de uso y políticas de acceso físico a los recursos.

Por regla general, las políticas son el primer paso que dispone a una organización para entrar en un ambiente de seguridad, puesto que reflejan su “voluntad de hacer algo” que

¹ ARDITA, Julio César. Director de Cybsec S.A. Security System y ex-Hacker. Entrevista personal realizada el día 15 de enero de 2001 en instalaciones de Cybsec S.A. <http://www.cybsec.com>

permita detener un posible ataque antes de que éste suceda (proactividad). A continuación se citan algunos de los métodos de protección más comúnmente empleados.

1. **Sistemas de detección de intrusos:** son sistemas que permiten analizar las bitácoras de los sistemas en busca de patrones de comportamiento o eventos que puedan considerarse sospechosos, sobre la base de la información con la que han sido previamente alimentados. Pueden considerarse como monitores.
2. **Sistemas orientados a conexión de red:** monitorizan las conexiones que se intentan establecer en una red o equipo en particular, siendo capaces de efectuar una acción sobre la base de métricas como: origen y destino de la conexión, servicio solicitado, permisos, etc. Las acciones que pueden emprender suelen ir desde el rechazo de la conexión hasta alerta al administrador. En esta categoría están los cortafuegos (Firewalls) y los Wrappers.
3. **Sistemas de análisis de vulnerabilidades:** analizan sistemas en busca de vulnerabilidades conocidas anticipadamente. La “desventaja” de estos sistemas es que pueden ser utilizados tanto por personas autorizadas como por personas que buscan acceso no autorizado al sistema.
4. **Sistemas de protección a la integridad de información:** sistemas que mediante criptografía o sumas de verificación tratan de asegurar que no ha habido alteraciones indeseadas en la información que se intenta proteger. Algunos ejemplos son los programas que implementan algoritmos como Message Digest (MD5) o Secure Hash Algorithm (SHA), o bien sistemas que utilizan varios de ellos como PGP, Tripwire y DozeCrypt.
5. **Sistemas de protección a la privacidad de la información:** herramientas que utilizan criptografía para asegurar que la información sólo sea visible para quien tiene autorización. Su aplicación se realiza principalmente en las comunicaciones entre dos entidades. Dentro de este tipo de herramientas se pueden citar a Pretty Good Privacy (PGP), Secure Sockets Layer (SSL) y los Certificados Digitales.

Resumiendo, un modelo de seguridad debe estar formado por múltiples componentes o capas que pueden ser incorporadas de manera progresiva al modelo global de seguridad en la organización, logrando así el método más efectivo para disuadir el uso no autorizado de sistemas y servicios de red. Podemos considerar que estas capas son:

1. Política de seguridad de la organización.
2. Auditoría.
3. Sistemas de seguridad a nivel de Router–Firewall.
4. Sistemas de detección de intrusos.
5. Plan de respuesta a incidentes.
6. Penetration Test.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

8.1.2 PENETRATION TEST, ETHICAL HACKING O PRUEBA DE VULNERABILIDAD

“El Penetration Test es un conjunto de metodologías y técnicas, para realizar una evaluación integral de las debilidades de los sistemas informáticos. Consiste en un modelo que

reproduce intentos de acceso, a cualquier entorno informático, de un intruso potencial desde los diferentes puntos de entrada que existan, tanto internos como remotos.”²

El objetivo general del Penetration Test es acceder a los equipos informáticos de la organización tratada e intentar obtener los privilegios del administrador del sistema, logrando así realizar cualquier tarea sobre dichos equipos. También se podrá definir otros objetivos secundarios que permitan realizar pruebas puntuales sobre algunos ámbitos particulares de la empresa.

El Penetration Test se compone de dos grandes fases de testeo:

1. **Penetration Test Externo:** el objetivo es acceder en forma remota a los equipos de la organización y posicionarse como administrador del sistema. Se realizan desde fuera del Firewall y consisten en penetrar la Zona Desmilitarizada para luego acceder a la red interna. Se compone de un elevado número de pruebas, entre las que se puede nombrar:
 - Pruebas de usuarios y la “fuerza” de sus passwords.
 - Captura de tráfico.
 - Detección de conexiones externas y sus rangos de direcciones.
 - Detección de protocolos utilizados.
 - Scanning de puertos TCP, UDP e ICMP.
 - Intentos de acceso vía accesos remotos, módems, Internet, etc.
 - Análisis de la seguridad de las conexiones con proveedores, trabajadores remotos o entidades externas a la organización .
 - Pruebas de vulnerabilidades existentes y conocidas en el momento de realización del Test.
 - Prueba de ataques de Denegación de Servicio.
2. **Penetration Test Interno:** este tipo de testeo trata de demostrar cual es el nivel de seguridad interno. Se deberá establecer que puede hacer un Insider y hasta donde será capaz de penetrar en el sistema siendo un usuario con privilegios bajos. Este Test también se compone de numerosas pruebas:
 - Análisis de protocolos internos y sus vulnerabilidades.
 - Autenticación de usuarios.
 - Verificación de permisos y recursos compartidos.
 - Test de los servidores principales (WWW, DNS, FTP, SMTP, etc.).
 - Test de vulnerabilidad sobre las aplicaciones propietarias.
 - Nivel de detección de la intrusión de los sistemas.
 - Análisis de la seguridad de las estaciones de trabajo.
 - Seguridad de la red.
 - Verificación de reglas de acceso.
 - Ataques de Denegación de Servicio

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

² ARDITA, Julio Cesar. "Prueba de Vulnerabilidad". ©1996–2001 CYBSEC S.A.
<http://www.cybsec.com/0302.htm>

8.1.3 HONEYPOTS–HONEYNETS

Estas “Trampas de Red” son sistemas que se activan con la finalidad específica de que los expertos en seguridad puedan observar en secreto la actividad de los Hackers/Crackers en su hábitat natural.

Actualmente un equipo de Honeynet Project³ trabaja en el desarrollo de un documento sobre la investigación y resultados de su trampa, la cual fue penetrada a la semana de ser activada (sin publicidad).

“Consiste en activar un servidor y llenarlo de archivos tentadores, hacer que sea difícil, pero no imposible penetrarlo y sentarse a esperar que aparezcan los intrusos. Los Honeynets dan a los crackers un gran espacio para recorrer. Presentan obstáculos que poseen el nivel de complejidad suficiente para atraerlos, pero sin irse al extremo para no desalentarlos (...). Ellos juegan con los archivos y conversan animadamente entre ellos sobre todos los ‘fascinantes programas’ que encuentran, mientras el personal de seguridad observa con deleite cada movimiento que hacen”, dijo Dan Adams. “Francamente, siento una combinación de sentimientos con respecto a espiar a la gente, aunque no sean buenas personas”⁴.

Esta última frase se está presentando a menudo en el tema de la investigación (y vigilancia) electrónica. Este es el caso del ex–director del proyecto Honeynet J. D. Glaser, quien renunció a su puesto después de aclarar que está convencido “que la vigilancia electrónica no es correcta, aunque se utilice en aras de la investigación (...). Ampliar un Honeynet es parecido a entrapar los derechos de otros, aunque sean los derechos de un delincuente.”

Con respecto a algunos de los resultados obtenidos por el grupo de investigación puede observarse el siguiente ejemplo:

A un intruso le tomo menos de un minuto irrumpir en la computadora de su universidad a través de Internet, estuvo dentro menos de media hora y a los investigadores le tomo 34 horas descubrir todo lo que hizo.
Se estima que esas 34 horas de limpieza pueden costar U\$2.000 a una organización y U\$22.000 si se debiera tratar con un consultor especializado.

8.2 FIREWALLS

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar□
www.cfbsoft.com.ar□

Quizás uno de los elementos más publicitados a la hora de establecer seguridad, sean estos elementos. Aunque deben ser uno de los sistemas a los que más se debe prestar atención, distan mucho de ser la solución final a los problemas de seguridad.

De hecho, los Firewalls no tienen nada que hacer contra técnicas como la Ingeniería Social y el ataque de Insiders.

Un **Firewall** es un sistema (o conjunto de ellos) ubicado entre dos redes y que ejerce la una política de seguridad establecida. Es el mecanismo encargado de proteger una red confiable de una que no lo es (por ejemplo Internet).

Puede consistir en distintos dispositivos, tendientes a los siguientes objetivos:

³ Honeynet Project: <http://project.honeynet.org>

⁴ ADAMS, Dan. Administrador de los sistemas London SecTech, quien sigue de cerca el proyecto Honeynet.

1. Todo el tráfico desde dentro hacia fuera, y viceversa, debe pasar a través de él.
2. Sólo el tráfico autorizado, definido por la política local de seguridad, es permitido.

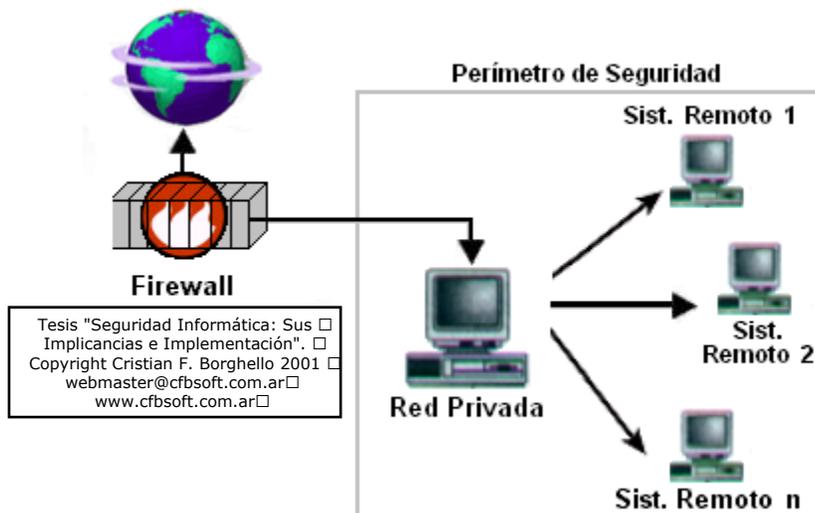


Gráfico 8.1 – Firewall

Como puede observarse, el Muro Cortafuegos, sólo sirven de defensa perimetral de las redes, no defienden de ataques o errores provenientes del interior, como tampoco puede ofrecer protección una vez que el intruso lo traspasa.

Algunos Firewalls aprovechan esta capacidad de que toda la información entrante y saliente debe pasar a través de ellos para proveer servicios de seguridad adicionales como la encriptación del tráfico de la red. Se entiende que si dos Firewalls están conectados, ambos deben “hablar” el mismo método de encriptación–desencriptación para entablar la comunicación.

8.2.1 ROUTERS Y BRIDGES

Cuando los paquetes de información viajan entre su destino y origen, vía TCP/IP, estos pasan por diferentes Routers (enrutadores a nivel de Red).

Los Routers son dispositivos electrónicos encargados de establecer comunicaciones externas y de convertir los protocolos utilizados en las LAN en protocolos de WAN y viceversa.

En cambio, si se conectan dos redes del tipo LAN se utilizan Bridges, los cuales son puentes que operan a nivel de Enlace

La evolución tecnológica les ha permitido transformarse en computadoras muy especializadas capaz de determinar, si el paquete tiene un destino externo y el camino más corto y más descongestionado hacia el Router de la red destino. En caso de que el paquete provenga de afuera, determina el destino en la red interna y lo deriva a la máquina correspondiente o devuelve el paquete a su origen en caso de que él no sea el destinatario del mismo.

Los Routers “toman decisiones” en base a un conjunto de datos, regla, filtros y excepciones que le indican que rutas son las más apropiadas para enviar los paquetes.

8.2.2 TIPOS DE FIREWALL

8.2.2.1 FILTRADO DE PAQUETES

Se utilizan Routers con filtros y reglas basadas en políticas de control de acceso. El Router es el encargado de filtrar los paquetes (un Choke) basados en cualquiera de los siguientes criterios:

1. Protocolos utilizados.
2. Dirección IP de origen y de destino.
3. Puerto TCP-UDP de origen y de destino.

Estos criterios permiten gran flexibilidad en el tratamiento del tráfico. Restringiendo las comunicaciones entre dos computadoras (mediante las direcciones IP) se permite determinar entre cuales máquinas la comunicación está permitida.

El filtrado de paquetes mediante puertos y protocolos permite establecer que servicios estarán disponibles al usuario y por cuales puertos. Se puede permitir navegar en la WWW (puerto 80 abierto) pero no acceder a la transferencia de archivos vía FTP (puerto 21 cerrado).

Debido a su funcionamiento y estructura basada en el filtrado de direcciones y puertos este tipo de Firewalls trabajan en los niveles de Transporte y de Red del Modelo OSI y están conectados a ambos perímetros (interior y exterior) de la red.

Tienen la ventaja de ser económicos, tienen un alto nivel de desempeño y son transparentes para los usuarios conectados a la red. Sin embargo presenta debilidades como:

1. No protege las capas superiores a nivel OSI.
2. Las necesidades aplicativas son difíciles de traducir como filtros de protocolos y puertos.
3. No son capaces de esconder la topología de redes privadas, por lo que exponen la red al mundo exterior.
4. Sus capacidades de auditoría suelen ser limitadas, al igual que su capacidad de registro de actividades.
5. No soportan políticas de seguridad complejas como autenticación de usuarios y control de accesos con horarios prefijados.

8.2.2.2 PROXY-GATEWAYS DE APLICACIONES

Para evitar las debilidades asociadas al filtrado de paquetes, los desarrolladores crearon software de aplicación encargados de filtrar las conexiones. Estas aplicaciones son conocidas como Servidores Proxy y la máquina donde se ejecuta recibe el nombre de Gateway de Aplicación o Bastion Host.

El Proxy, instalado sobre el Nodo Bastión, actúa de intermediario entre el cliente y el servidor real de la aplicación, siendo transparente a ambas partes.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

Cuando un usuario desea un servicio, lo hace a través del Proxy. Este, realiza el pedido al servidor real devuelve los resultados al cliente. Su función fue la de analizar el tráfico de red en busca de contenido que viole la seguridad de la misma. Gráficamente:



Gráfico 8.2 – Bastión Host

8.2.2.3 DUAL-HOMED HOST

Son dispositivos que están conectados a ambos perímetros (interior y exterior) y no dejan pasar paquetes IP (como sucede en el caso del Filtrado de Paquetes), por lo que se dice que actúan con el “IP-Forwarding desactivado”.

Un usuario interior que desee hacer uso de un servicio exterior, deberá conectarse primero al Firewall, donde el Proxy atenderá su petición, y en función de la configuración impuesta en dicho Firewall, se conectará al servicio exterior solicitado y hará de puente entre este y el usuario interior.

Es decir que se utilizan dos conexiones. Uno desde la máquina interior hasta el Firewall y el otro desde este hasta la máquina que albergue el servicio exterior.

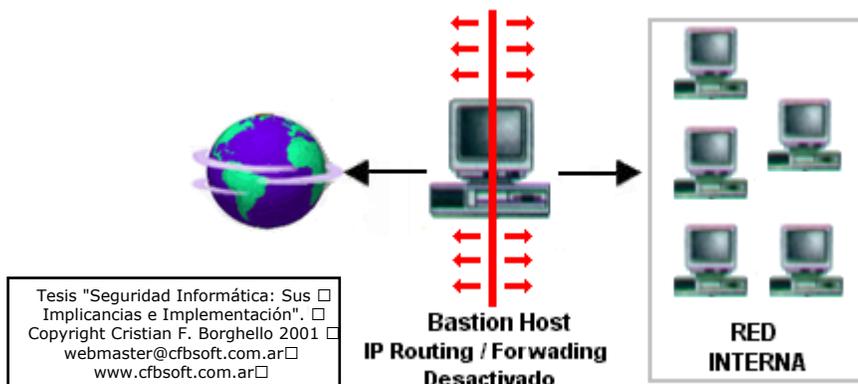


Gráfico 8.3 – Dual-Homed Host

8.2.2.4 SCREENED HOST

En este caso se combina un Router con un host bastión y el principal nivel de seguridad proviene del filtrado de paquetes. En el bastión, el único sistema accesible desde el exterior, se ejecuta el Proxy de aplicaciones y en el Choke se filtran los paquetes considerados peligrosos y sólo se permiten un número reducido de servicios.



Gráfico 8.4 – Screened Host

8.2.2.5 SCREENED SUBNET

En este diseño se intenta aislar la máquina más atacada y vulnerable del Firewall, el Nodo Bastión. Para ello se establece una Zona Desmilitarizada (DMZ) de forma tal que sin un intruso accede a esta máquina no consiga el acceso total a la subred protegida.

En este esquema se utilizan dos Routers: uno exterior y otro interior. El Router exterior tiene la misión de bloquear el tráfico no deseado en ambos sentidos: hacia la red interna y hacia la red externa. El Router interior hace lo mismo con la red interna y la DMZ (zona entre el Router externo y el interno).

Es posible definir varios niveles de DMZ agregando más Routers, pero destacando que las reglas aplicadas a cada uno deben ser distintas ya que en caso contrario los niveles se simplificarían a uno solo.

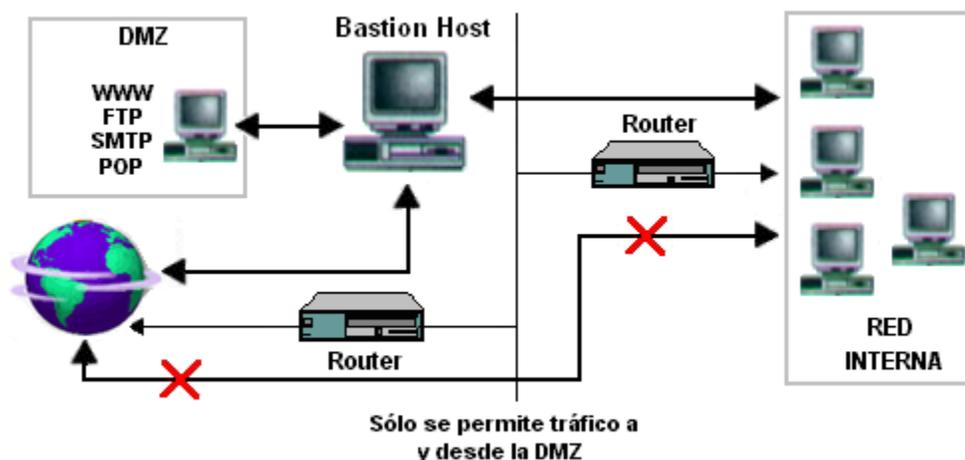


Gráfico 8.5 – Screened Hosted

Como puede apreciarse la Zona Desmilitarizada aísla físicamente los servicios internos, separados de los servicios públicos. Además, no existe una conexión directa entre la red interna y la externa.

Los sistemas Dual–Homed Host y Screened pueden ser complicados de configurar y comprobar, lo que puede dar lugar, paradójicamente, a importantes agujeros de seguridad en toda la red. En cambio, si se encuentran bien configurados y administrados pueden brindar un alto grado de protección y ciertas ventajas:

1. Ocultamiento de la información: los sistemas externos no deben conocer el nombre de los sistemas internos. El Gateway de aplicaciones es el único autorizado a conectarse con el exterior y el encargado de bloquear la información no solicitada o sospechosa.
2. Registro de actividades y autenticación robusta: El Gateway requiere de autenticación cuando se realiza un pedido de datos externos. El registro de actividades se realiza en base a estas solicitudes.
3. Reglas de filtrado menos complejas: Las reglas del filtrado de los paquetes por parte del Router serán menos compleja dado a que él sólo debe atender las solicitudes del Gateway.

Así mismo tiene la desventaja de ser intrusivos y no transparentes para el usuario ya que generalmente este debe instalar algún tipo de aplicación especializada para lograr la comunicación. Se suma a esto que generalmente son más lentos porque deben revisar todo el tráfico de la red.

8.2.2.6 INSPECCIÓN DE PAQUETES

Este tipo de Firewalls se basa en el principio de que cada paquete que circula por la red es inspeccionado, así como también su procedencia y destino. Se aplican desde la capa de Red hasta la de Aplicaciones. Generalmente son instalados cuando se requiere seguridad sensible al contexto y en aplicaciones muy complejas.

8.2.2.7 FIREWALLS PERSONALES

Estos Firewalls son aplicaciones disponibles para usuarios finales que desean conectarse a una red externa insegura y mantener su computadora a salvo de ataques que puedan ocasionarle desde un simple “cuelgue” o infección de virus hasta la pérdida de toda su información almacenada.

8.2.3 POLÍTICAS DE DISEÑO DE FIREWALLS

Tesis "Seguridad Informática: Sus <input type="checkbox"/>
Implicancias e Implementación". <input type="checkbox"/>
Copyright Cristian F. Borghello 2001 <input type="checkbox"/>
webmaster@cfbsoft.com.ar <input type="checkbox"/>
www.cfbsoft.com.ar <input type="checkbox"/>

Las políticas de accesos en un Firewalls se deben diseñar poniendo principal atención en sus limitaciones y capacidades pero también pensando en las amenazas y vulnerabilidades presentes en una red externa insegura.

Conocer los puntos a proteger es el primer paso a la hora de establecer normas de seguridad. También es importante definir los usuarios contra los que se debe proteger cada recurso, ya que las medidas diferirán notablemente en función de esos usuarios.

Generalmente se plantean algunas preguntas fundamentales que debe responder cualquier política de seguridad:

- ¿Qué se debe proteger?. Se deberían proteger todos los elementos de la red interna (hardware, software, datos, etc.).
- ¿De quién protegerse?. De cualquier intento de acceso no autorizado desde el exterior y contra ciertos ataques desde el interior que puedan preverse y prevenir.

Sin embargo, podemos definir niveles de confianza, permitiendo selectivamente el acceso de determinados usuarios externos a determinados servicios o denegando cualquier tipo de acceso a otros.

- ¿Cómo protegerse?. Esta es la pregunta más difícil y está orientada a establecer el nivel de monitorización, control y respuesta deseado en la organización. Puede optarse por alguno de los siguientes paradigmas o estrategias:

a. Paradigmas de seguridad

- Se permite cualquier servicio excepto aquellos expresamente prohibidos.
- Se prohíbe cualquier servicio excepto aquellos expresamente permitidos. La más recomendada y utilizada aunque algunas veces suele acarrear problemas por usuarios descontentos que no pueden acceder a tal cual servicio.

b. Estrategias de seguridad

- Paranoica: se controla todo, no se permite nada.
- Prudente: se controla y se conoce todo lo que sucede.
- Permisiva: se controla pero se permite demasiado.
- Promiscua: no se controla (o se hace poco) y se permite todo.

- ¿Cuánto costará?. Estimando en función de lo que se desea proteger se debe decidir cuanto es conveniente invertir.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

8.2.4 RESTRICCIONES EN EL FIREWALL

La parte más importante de las tareas que realizan los Firewalls, la de permitir o denegar determinados servicios, se hacen en función de los distintos usuarios y su ubicación:

1. **Usuarios internos con permiso de salida para servicios restringidos:** permite especificar una serie de redes y direcciones a los que denomina **Trusted (validados)**. Estos usuarios, cuando provengan del interior, van a poder acceder a determinados servicios externos que se han definido.
2. **Usuarios externos con permiso de entrada desde el exterior:** este es el caso más sensible a la hora de vigilarse. Suele tratarse de usuarios externos que por algún motivo deben acceder para consultar servicios de la red interna.

También es habitual utilizar estos accesos por parte de terceros para prestar servicios al perímetro interior de la red. Sería conveniente que estas cuentas sean activadas y desactivadas bajo demanda y únicamente el tiempo que sean necesarias.

8.2.5 BENEFICIOS DE UN FIREWALL

Los Firewalls manejan el acceso entre dos redes, y si no existiera, todas las computadoras de la red estarían expuestas a ataques desde el exterior. Esto significa que la seguridad de toda la red, estaría dependiendo de que tan fácil fuera violar la seguridad local de cada maquina interna.

El Firewall es el punto ideal para monitorear la seguridad de la red y generar alarmas de intentos de ataque, el administrador será el responsable de la revisión de estos monitoreos.

Otra causa que ha hecho que el uso de Firewalls se halla convertido en uso casi imperativo es el hecho que en los últimos años en Internet han entrado en crisis el número disponible de direcciones IP, esto ha hecho que las intranets adopten direcciones sin clase, las cuales salen a Internet por medio de un “traductor de direcciones”, el cual puede alojarse en el Firewall.

Los Firewalls también son importantes desde el punto de vista de llevar las estadísticas del ancho de banda “consumido” por el trafico de la red, y que procesos han influido más en ese trafico, de esta manera el administrador de la red puede restringir el uso de estos procesos y economizar o aprovechar mejor el ancho de banda disponible.

Los Firewalls también tienen otros usos. Por ejemplo, se pueden usar para dividir partes de un sitio que tienen distintas necesidades de seguridad o para albergar los servicios WWW y FTP brindados.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

8.2.6 LIMITACIONES DE UN FIREWALL

La limitación más grande que tiene un Firewall sencillamente es el hueco que no se tapa y que coincidentemente o no, es descubierto por un intruso. Los Firewalls no son sistemas inteligentes, ellos actúan de acuerdo a parámetros introducidos por su diseñador, por ende si un paquete de información no se encuentra dentro de estos parámetros como una amenaza de peligro simplemente lo deja pasar. Más peligroso aún es que ese intruso deje Back Doors, abriendo un hueco diferente y borre las pruebas o indicios del ataque original.

Otra limitación es que el Firewall “NO es contra humanos”, es decir que si un intruso logra entrar a la organización y descubrir passwords o los huecos del Firewall y difunde esta información, el Firewall no se dará cuenta.

El Firewall tampoco provee de herramientas contra la filtración de software o archivos infectados con virus, aunque es posible dotar a la máquina, donde se aloja el Firewall, de antivirus apropiados.

Finalmente, un Firewall es vulnerable, él NO protege de la gente que está dentro de la red interna. El Firewall trabaja mejor si se complementa con una defensa interna. Como moraleja: “cuanto mayor sea el tráfico de entrada y salida permitido por el Firewall, menor será la resistencia contra los paquetes externos. El único Firewall seguro (100%) es aquel que se mantiene apagado”⁵

⁵ HERNÁNDEZ, Roberto. Firewalls: Seguridad en las redes e Internet. Boletín de Política Informática N° 2. Página 7. España. 2000.

8.3 ACCESS CONTROL LISTS (ACL)

Las Listas de Control de Accesos proveen de un nivel de seguridad adicional a los clásicos provistos por los Sistemas Operativos. Estas listas permiten definir permisos a usuarios y grupos concretos. Por ejemplo pueden definirse sobre un Proxy una lista de todos los usuarios (o grupos de ellos) a quien se le permite el acceso a Internet, FTP, etc. También podrán definirse otras características como limitaciones de anchos de banda y horarios.

8.4 WRAPPERS

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

Un Wrapper es un programa que controla el acceso a un segundo programa. El Wrapper literalmente cubre la identidad de este segundo programa, obteniendo con esto un más alto nivel de seguridad. Los Wrappers son usados dentro de la seguridad en sistemas UNIXs. Estos programas nacieron por la necesidad de modificar el comportamiento del sistema operativo sin tener que modificar su funcionamiento.

Los Wrappers son ampliamente utilizados, y han llegado a formar parte de herramientas de seguridad por las siguientes razones:

- Debido a que la seguridad lógica esta concentrada en un solo programa, los Wrappers son fáciles y simples de validar.
- Debido a que el programa protegido se mantiene como una entidad separada, éste puede ser actualizado sin necesidad de cambiar el Wrapper.
- Debido a que los Wrappers llaman al programa protegido mediante llamadas estándar al sistema, se puede usar un solo Wrapper para controlar el acceso a diversos programas que se necesiten proteger.
- Permite un control de accesos exhaustivo de los servicios de comunicaciones, además de buena capacidad de Logs y auditorias de peticiones a dichos servicios, ya sean autorizados o no.

El paquete Wrapper más ampliamente utilizado es el TCP-Wrappers, el cual es un conjunto de utilidades de distribución libre, escrito por Wietse Venema (co-autor de SATAN, con Dan Farmer, y considerado el padre de los sistemas Firewalls) en 1990.

Consiste en un programa que es ejecutado cuando llega una petición a un puerto específico. Este, una vez comprobada la dirección de origen de la petición, la verifica contra las reglas almacenadas, y en función de ellas, decide o no dar paso al servicio. Adicionalmente, registra estas actividades del sistema, su petición y su resolución.

Algunas configuraciones avanzadas de este paquete, permiten también ejecutar comandos en el propio sistema operativo, en función de la resolución de la petición. Por ejemplo, es posible que intente detectar una posible máquina atacante, en el caso de un intento de conexión, para tener más datos a la hora de una posible investigación. Este tipo de comportamiento raya en la estrategia paranoica, ya vista cuando se definió la política de seguridad del firewall.

Con lo mencionado hasta aquí, puede pensarse que los Wrappers son Firewall ya que muchos de los servicios brindados son los mismos o causan los mismos efectos: usando

Wrappers, se puede controlar el acceso a cada máquina y los servicios accedidos. Así, estos controles son el complemento perfecto de un Firewall y la instalación de uno no está supeditada a la del otro.

8.5 DETECCIÓN DE INTRUSOS EN TIEMPO REAL

La seguridad se tiene que tratar en conjunto. Este viejo criterio es el que recuerda que los sistemas de protección hasta aquí abordados, si bien son eficaces, distan mucho de ser la protección ideal.

Así, debe estar fuera de toda discusión la conveniencia de añadir elementos que controlen lo que ocurre dentro de la red (detrás de los Firewalls).

Como se ha visto, la integridad de un sistema se puede corromper de varias formas y la forma de evitar esto es con la instalación de sistemas de Detección de Intrusos en Tiempo Real, quienes:

- Inspeccionan el tráfico de la red buscando posibles ataques.
- Controlan el registro de los servidores para detectar acciones sospechosas (tanto de intrusos como de usuarios autorizados).
- Mantienen una base de datos con el estado exacto de cada uno de los archivos (Integrity Check) del sistema para detectar la modificación de los mismos.
- Controlan el ingreso de cada nuevo archivo al sistema para detectar Caballos de Troya o semejantes.
- Controlan el núcleo del Sistema Operativo para detectar posibles infiltraciones en él, con el fin de controlar los recursos y acciones del mismo.
- Avisan al administrador de cualquiera de las acciones mencionadas.

Cada una de estas herramientas permiten mantener alejados a la gran mayoría de los intrusos normales. Algunos pocos, con suficientes conocimientos, experiencia y paciencia serán capaces de utilizar métodos sofisticados (u originales) como para voltear el perímetro de seguridad (interna + externa) y serán estos los casos que deban estudiarse para integrar a la política de seguridad existente mayor conocimiento y con él mayor seguridad.

8.5.1 INTRUSIÓN DETECTION SYSTEMS (IDS)

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

Un sistema de detección de intrusos es un componente más dentro del modelo de seguridad de una organización. Consiste en detectar actividades inapropiadas, incorrectas o anómala desde el exterior–interior de un sistema informático.

Los sistemas de detección de intrusos pueden clasificarse, según su función y comportamiento en:

- **Host–Based IDS:** operan en un host para detectar actividad maliciosa en el mismo.
- **Network–Based IDS:** operan sobre los flujos de información intercambiados en una red.
- **Knowledge–Based IDS:** sistemas basados en Conocimiento.

- **Behavior-Based IDS:** sistemas basados en Comportamiento. Se asume que una intrusión puede ser detectada observando una desviación respecto del comportamiento normal o esperado de un usuario en el sistema.

La idea central de este tipo de detección es el hecho de que la actividad intrusiva es un conjunto de actividades anómalas. Si alguien consigue entrar de forma ilegal al sistema, no actuará como un usuario comprometido; su comportamiento se alejará del de un usuario normal.

Sin embargo en la mayoría de las ocasiones una actividad intrusiva resulta del agregado de otras actividades individuales que por sí solas no constituyen un comportamiento intrusivo de ningún tipo. Así las intrusiones pueden clasificarse en:

- **Intrusivas pero no anómalas:** denominados Falsos Negativos (el sistema erróneamente indica ausencia de intrusión). En este caso la actividad es intrusiva pero como no es anómala no es detectada. No son deseables, porque dan una falsa sensación de seguridad del sistema.
- **No intrusivas pero anómalas:** denominados Falsos Positivos (el sistema erróneamente indica la existencia de intrusión). En este caso la actividad es no intrusiva, pero como es anómala el sistema “decide” que es intrusiva. Deben intentar minimizarse, ya que en caso contrario se ignorarán los avisos del sistema, incluso cuando sean acertados.
- **No intrusiva ni anómala:** son Negativos Verdaderos, la actividad es no intrusiva y se indica como tal.
- **Intrusiva y anómala:** se denominan Positivos Verdaderos, la actividad es intrusiva y es detectada.

Los detectores de intrusiones anómalas requieren mucho gasto computacional, ya que se siguen normalmente varias métricas para determinar cuánto se aleja el usuario de lo que se considera comportamiento normal.

8.5.1.1 CARACTERÍSTICAS DE IDS

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

Cualquier sistema de detección de intrusos debería, sea cual sea el mecanismo en que esté basado, **debería** contar con las siguientes características:

- Debe funcionar continuamente sin supervisión humana. El sistema debe ser lo suficientemente fiable para poder ser ejecutado en background dentro del equipo que está siendo observado. Sin embargo, no debe ser una “caja negra” (debe ser examinable desde el exterior).
- Debe ser tolerante a fallos en el sentido de que debe ser capaz de sobrevivir a una caída del sistema.
- En relación con el punto anterior, debe ser resistente a perturbaciones. El sistema puede monitorizarse a sí mismo para asegurarse de que no ha sido perturbado.
- Debe imponer mínima sobrecarga sobre el sistema. Un sistema que relentiza la máquina, simplemente no será utilizado.
- Debe observar desviaciones sobre el comportamiento estándar.

- Debe ser fácilmente adaptable al sistema ya instalado. Cada sistema tiene un patrón de funcionamiento diferente y el mecanismo de defensa debe adaptarse de manera sencilla a esos patrones.
- Debe hacer frente a los cambios de comportamiento del sistema según se añaden nuevas aplicaciones al mismo.
- Debe ser difícil de “engañar”.

8.5.1.2 FORTALEZAS DE IDS

- Suministra información muy interesante sobre el tráfico malicioso de la red.
- Poder de reacción para prevenir el daño.
- Es una herramienta útil como arma de seguridad de la red.
- Ayuda a identificar de dónde provienen los ataques que se sufren.
- Recoge evidencias que pueden ser usadas para identificar intrusos.
- Es una “cámara” de seguridad y una “alarma” contra ladrones.
- Funciona como “disuasor de intrusos”.
- Alerta al personal de seguridad de que alguien está tratando de entrar.
- Protege contra la invasión de la red.
- Suministra cierta tranquilidad.
- Es una parte de la infraestructura para la estrategia global de defensa.
- La posibilidad de detectar intrusiones desconocidas e imprevistas. Pueden incluso contribuir (parcialmente) al descubrimiento automático de esos nuevos ataques.
- Son menos dependientes de los mecanismos específicos de cada sistema operativo.
- Pueden ayudar a detectar ataques del tipo “abuso de privilegios” que no implica realmente ninguna vulnerabilidad de seguridad. En pocas palabras, se trata de una aproximación a la paranoia: “todo aquello que no se ha visto previamente es peligroso”.
- Menor costo de implementación y mantenimiento al ubicarse en puntos estratégicos de la red.
- Dificulta el trabajo del intruso de eliminar sus huellas.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

8.5.1.3 DEBILIDADES DE IDS

- No existe un parche para la mayoría de bugs de seguridad.
- Se producen falsas alarmas.
- Se producen fallos en las alarmas.
- No es sustituto para un buen Firewall, una auditoría de seguridad regular y una fuerte y estricta política de seguridad.

8.5.1.4 INCONVENIENTES DE IDS

- La alta tasa de falsas alarmas dado que no es posible cubrir todo el ámbito del comportamiento de un sistema de información durante la fase de aprendizaje.
- El comportamiento puede cambiar con el tiempo, haciendo necesario un re-entrenamiento periódico del perfil, lo que da lugar a la no disponibilidad del sistema o la generación de falsas alarmas adicionales.
- El sistema puede sufrir ataques durante la fase de aprendizaje, con lo que el perfil de comportamiento contendrá un comportamiento intrusivo el cual no será considerado anómalo.

8.6 CALL BACK

Este procedimiento es utilizado para verificar la autenticidad de una llamada vía modem. El usuario llama, se autentifica contra el sistema, se desconecta y luego el servidor se conecta al número que en teoría pertenece al usuario.

La ventaja reside en que si un intruso desea hacerse pasar por el usuario, la llamada se devolverá al usuario legal y no al del intruso, siendo este desconectado. Como precaución adicional, el usuario deberá verificar que la llamada-retorno proceda del número a donde llamó previamente.

8.7 SISTEMAS ANTI-SNIFFERS

Esta técnica consiste en detectar Sniffers en el sistema. Generalmente estos programas se basan en verificar el estado de la placa de red, para detectar el modo en el cual está actuando (recordar que un Sniffer la coloca en Modo Promiscuo), y el tráfico de datos en ella.

8.8 GESTION DE CLAVES “SEGURAS”

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

Como ya se vio en el capítulo anterior (ver Tabla 7.4), si se utiliza una clave de 8 caracteres de longitud, con los 96 caracteres posibles, puede tardarse 2.288 años en descifrarla (analizando 100.000 palabras por segundo). Esto se obtiene a partir de las 96^8 (7.213.895.789.838.340) claves posibles de generar con esos caracteres.

Partiendo de la premisa en que no se disponen de esa cantidad de años para analizarlas por fuerza bruta, se deberá comenzar a probar con las claves más posibles, comúnmente llamadas Claves Débiles.

Según demuestra el análisis de +NetBuL⁶ realizado sobre 2.134 cuentas y probando 227.000 palabras por segundo:

⁶ +NetBul. Tabla de Tiempos del John the Ripper 1.4. SET N°15-0x07. Junio de 1998.
<http://www.vanhackez.co/set> - <http://www.thepentagon.com/paseante>

- Con un diccionario 2.030 palabras (el original de John de Ripper 1.04), se obtuvieron 36 cuentas en solo 19 segundos (1,77%).
- Con un diccionario de 250.000 palabras, se obtuvieron 64 cuentas en 36:18 minutos (3,15%).

Otro estudio⁷ muestra el resultado obtenido al aplicar un ataque, mediante un diccionario de 62.727 palabras, a 13.794 cuentas:

- En un año se obtuvieron 3.340 contraseñas (24,22%).
- En la primera semana se descubrieron 3.000 claves (21,74%).
- En los primeros 15 minutos se descubrieron 368 palabras claves (2,66%).

Según los grandes números vistos, sería válido afirmar que: es imposible encontrar ¡36 cuentas en 19 segundos!. También debe observarse, en el segundo estudio, que el porcentaje de hallazgos casi no varía entre un año y una semana.

Tal vez, ¿esto sucedió porque existían claves nulas; que corresponde al nombre del usuario; a secuencias alfabéticas tipo 'abcd'; a secuencias numéricas tipo '1234'; a secuencias observadas en el teclado tipo 'qwer'; a palabras que existen en un diccionario del lenguaje del usuario?. Sí, estas claves (las más débiles) son las primeras en ser analizadas y los tiempos obtenidos confirman la hipótesis.

Este simple estudio confirma nuestra mala elección de contraseñas, y el riesgo se incrementa si el atacante conoce algo sobre la víctima (Ingeniería Social) ya que podrá probar palabras relacionadas a su persona o diccionarios orientados.

8.8.1 NORMAS DE ELECCIÓN DE CLAVES

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

Se debe tener en cuenta los siguientes consejos:

1. No utilizar contraseñas que sean palabras (aunque sean extranjeras), o nombres (el del usuario, personajes de ficción, miembros de la familia, mascotas, marcas, ciudades, lugares, u otro relacionado).
2. No usar contraseñas completamente numéricas con algún significado (teléfono, D.N.I., fecha de nacimiento, patente del automóvil, etc.).
3. Elegir una contraseña que mezcle caracteres alfabéticos (mayúsculas y minúsculas) y numéricos.
4. Deben ser largas, de 8 caracteres o más.
5. Tener contraseñas diferentes en máquinas diferentes. Es posible usar una contraseña base y ciertas variaciones lógicas de la misma para distintas máquinas.
6. Deben ser fáciles de recordar para no verse obligado a escribirlas. Algunos ejemplos son:
 - Combinar palabras cortas con algún número o carácter de puntuación:
soy2_yo3

⁷ KLEIN, Daniel V. Foiling the Cracker: A Survey of, and Improvement to, Password Security.

- Usar un acrónimo de alguna frase fácil de recordar: A rio Revuelto Ganancia de Pescadores → ArRGdP
- Añadir un número al acrónimo para mayor seguridad: A9r7R5G3d1P
- Mejor incluso si la frase no es conocida: Hasta Ahora no he Olvidado mi Contraseña → aHoelIo
- Elegir una palabra sin sentido, aunque pronunciable: taChunda72, AtajulH, Wen2Mar
- Realizar reemplazos de letras por signos o números: En Seguridad Más Vale Prevenir que Curar → 35M\ /Pq<

8.8.2 NORMAS PARA PROTEGER UNA CLAVE

La protección de la contraseña recae tanto sobre el administrador del sistema como sobre el usuario. Al comprometer una cuenta se puede estar comprometiendo todo el sistema.

La siguiente frase difundida en UseNet resume algunas de las reglas básicas de uso de la contraseña: “Un password debe ser como un cepillo de dientes. Úsalo cada día; cámbialo regularmente; y NO lo compartas con tus amigos”.

Algunos consejos a seguir:

1. No permitir ninguna cuenta sin contraseña. Si se es administrador del sistema, repasar este hecho periódicamente.
2. No mantener las contraseñas por defecto del sistema. Por ejemplo, cambiar las cuentas de Root, System, Test, Demo, Guest, etc.
3. Nunca compartir con nadie la contraseña. Si se hace, cambiarla inmediatamente.
4. No escribir la contraseña en ningún sitio. Si se escribe, no debe identificarse como tal y no debe identificarse al propietario en el mismo lugar.
5. No teclear la contraseña si hay alguien mirando. Es una norma tácita de buen usuario no mirar el teclado mientras alguien teclea su contraseña.
6. No enviar la contraseña por correo electrónico ni mencionarla en una conversación. Si se debe mencionar no hacerlo explícitamente diciendo: “mi clave es...”.
7. No mantener una contraseña indefinidamente. Cambiarla regularmente. Disponer de una lista de contraseñas que puedan usarse cíclicamente (por lo menos 5).

Muchos sistemas incorporan ya algunas medidas de gestión y protección de las contraseñas. Entre ellas podemos citar las siguientes:

1. Número de intentos limitado. Tras un número de intentos fallidos, pueden tomarse distintas medidas:
 - Obligar a reescribir el nombre de usuario (lo más común).
 - Bloquear el acceso durante un tiempo.
 - Enviar un mensaje al administrador y/o mantener un registro especial.
2. Longitud mínima. Las contraseñas deben tener un número mínimo de caracteres (se recomienda 7 u 8 como mínimo).

3. Restricciones de formato. Las contraseñas deben combinar un mínimo de letras y números, no pueden contener el nombre del usuario ni ser un blanco.
4. Envejecimiento y expiración de contraseñas. Cada cierto tiempo se fuerza a cambiar la contraseña. Se obliga a no repetir ciertas cantidad de las anterior. Se mantiene un periodo forzoso entre cambios, para evitar que se vuelva a cambiar inmediatamente y se repita la anterior.
5. Ataque preventivo. Muchos administradores utilizan crackeadores para intentar atacar las contraseñas de su propio sistema en busca de debilidades.

8.8.3 CONTRASEÑAS DE UN SÓLO USO

Las contraseñas de un solo uso (One–Time Passwords) son uno de los mecanismos de autenticación más seguros, debido a que su descubrimiento tan solo permite acceder al sistema una vez. Además, en muchas ocasiones se suelen utilizar dispositivos hardware para su generación, lo que las hace mucho más difíciles de descubrir.

Ejemplos de este tipo de contraseñas serian las basadas en funciones unidireccionales (sencillas de evaluar en un sentido pero imposible o muy costoso de evaluar en sentido contrario) y en listas de contraseñas.

Se distinguen tres tipos de contraseñas de un solo uso:

1. Las que requieren algún dispositivo hardware para su generación, tales como calculadoras especiales o tarjetas inteligentes (Token Cards).
2. Las que requieren algún tipo de software de cifrado especial.
3. Las que se basan en una lista de contraseñas sobre papel.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

La tarjeta genera periódicamente valores mediante a una función secreta y unidireccional, basada en el tiempo y en el número de identificación de la misma.

El usuario combina el número generado por la tarjeta con su palabra de paso para obtener el password de entrada, lo que le protege en caso de robo o perdida.

8.9 SEGURIDAD EN PROTOCOLOS Y SERVICIOS

Se ha visto en capítulos anteriores la variedad de protocolos de comunicaciones existentes, sus objetivos y su funcionamiento. Como puede preverse todos estos protocolos tienen su debilidad ya sea en su implementación o en su uso. A continuación se describen los problemas de seguridad más comunes y sus formas de prevención.

Nuevamente no se verán los detalles sobre el funcionamiento de cada uno de ellos, simplemente se ofrecerán las potenciales puertas de entrada como fuentes de ataques que ni siquiera tienen por qué proporcionar acceso a la máquina (como las DoS por ejemplo).

De esta forma, si cada servicio ofrecido es un posible problema para la seguridad, parece claro que lo ideal sería no ofrecer ninguno, poseer una máquina completamente aislada del resto; evidentemente, hoy en día no es posible en la mayor parte de los sistemas.

Por lo tanto, ya que es necesaria la conectividad entre equipos, se ha de ofrecer los mínimos servicios necesarios para que todo funcione correctamente; esto choca frontalmente con las políticas de la mayoría de fabricantes y empresas, que por defecto mantienen la mayoría de servicios abiertos al instalar un equipo nuevo: es responsabilidad del administrador preocuparse de cerrar los que no sean estrictamente necesarios.

8.9.1 NETBIOS

Estos puertos (137–139 en TCP y UDP) son empleado en las redes Microsoft® para la autenticación de usuarios y la compartición de recursos. Como primera medida debe minimizarse la cantidad de recursos compartidos y luego debe evitarse permitir el acceso global a esos dispositivos, ya que es posible el acceso de intrusos desde cualquier lugar externo a la red.

8.9.2 ICMP

A fin de prevenir los ataques basados en bombas ICMP, se deben filtrar todos los paquetes de redirección y los paquetes inalcanzables.

8.9.3 FINGER

Típicamente el servicio Finger (puerto 79 en TCP) ha sido una de las principales fuentes de problemas. Este protocolo proporciona información detallada de los usuarios de una estación de trabajo, estén o no conectados en el momento de acceder al servicio.

La información suministrada suele ser de mucha utilidad para un atacante: datos del usuario, hábitos de conexión, cuentas inactivas. Está claro que esto es fácilmente aprovechable por un intruso para practicar ingeniería social contra esos usuarios.

Es básico deshabilitar este servicio, restringir su acceso a unos cuantos equipos de la red local o utilizar versiones de Finger que permiten especificar la información que se muestra al acceder al servicio.

8.9.4 POP

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

El servicio POP (puertos 109 y 110 en TCP) utilizado para que los usuarios puedan acceder a su correo sin necesidad de montar un sistemas de archivos compartidos. Se trata de un servicio que se podría considerar peligroso, por lo que (como el resto, pero este especialmente) debemos deshabilitarlo a no ser que sea estrictamente necesario ofrecerlo; en ese caso debemos restringir al máximo los lugares y usuario desde los que se puede acceder.

Mediante POP se genera un tránsito peligroso de contraseñas a través de la red. Se ofrece tres modelos distintos de autenticación: uno basado en Kerberos, apenas utilizado, otro basado en un protocolo desafío–respuesta, y el otro basado en un simple nombre de usuario con su password correspondiente.

Este último, el más usado en todo tipo de entornos, es un excelente objetivo para un intruso con un Sniffer. Los usuarios suelen configurar sus clientes para que chequeen el buzón de correo cada pocos minutos, con lo que a intervalos muy cortos envían su clave a un puerto conocido de una máquina conocida; al realizar toda esta comunicación en texto claro, un atacante no tiene más que interceptar la sesión POP para averiguar nombres de usuario y claves (a parte de poder leer el correo).

8.9.5 NNTP

El servicio NNTP (puerto 119 en TCP) se utilizado para intercambiar mensajes de grupos de noticias entre servidores de News. Los diferentes demonios encargados de esta tarea suelen discriminar conexiones en función de la dirección o el nombre de la máquina cliente para decidir si ofrece el servicio a un determinado host, y si es así, concretar de que forma puede acceder a él (sólo lectura, sólo ciertos grupos, etc.).

De esta forma, los servidores NNTP son muy vulnerables a cualquier ataque que permita falsear la identidad de la máquina origen, como el IP Spoofing.

Los problemas relacionados con las News no suelen ser excesivamente graves desde un punto de vista estrictamente técnico, pero en ocasiones sí que lo son aplicando una visión global. Por ejemplo, habría que evaluar el daño que le supone a la imagen de la organización el que un atacante envíe mensajes insultantes o pornográficos utilizando el nombre o los recursos de la misma.

Realmente, es muy poco probable que se necesite ofrecer este servicio, por lo que lo más razonable es deshabilitarlo. Generalmente sólo existen servidores de noticias en grandes organizaciones, y si se debe administrar equipo con este servicio la mejor forma de protegerlo es utilizando un buen firewall.

8.9.6 NTP

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

NTP (puerto 123 en UDP y TCP) es un protocolo utilizado para sincronizar relojes de máquinas de una forma muy precisa; a pesar de su sofisticación no fue diseñado con una idea de robustez ante ataques, por lo que puede convertirse en una gran fuente de problemas si no está correctamente configurado.

Son muchos los problemas de seguridad relacionados con un tiempo correcto; el más simple y obvio es la poca fiabilidad que ofrecerá el sistema de Log a la hora de determinar cuándo sucedió determinado evento.

Otro problema inherente a NTP se refiere a la planificación de tareas: si el reloj tiene problemas, es posible que ciertas tareas no se lleguen a ejecutar, que se ejecuten varias veces, o que se ejecuten cuando no han de hacerlo; esto es especialmente peligroso para tareas de las que depende la seguridad (como los backups).

No obstante, muy pocos sistemas necesitan la precisión de NTP, por lo que es habitual tener este servicio deshabilitado. En la mayoría de ocasiones el propio reloj de la máquina, o un protocolo mucho más simple (como Time), es más que suficiente para sincronizar equipos.

8.9.7 TFTP

TFTP es un protocolo de transferencia de archivos (puerto 69 basado en UDP) que no proporciona ninguna seguridad. Por tanto en la mayoría de sistemas es deseable (obligatorio) que este servicio esté desactivado. Al utilizar este servicio en ningún momento se solicita un nombre de usuario o una clave, lo que da una idea de los graves problemas de seguridad que ofrece este servicio.

“Gracias” a este protocolo se han implementado algunas de las últimas vulnerabilidades del Internet Information Server®.

8.9.8 FTP

Un problema básico y grave de FTP (puerto 21 en TCP) es que ha sido diseñado para ofrecer la máxima velocidad en la conexión, pero no para ofrecer la seguridad; todo el intercambio de información, desde el Login y password del usuario en el servidor hasta la transferencia de cualquier archivo, se realiza en texto claro, con lo que un atacante no tiene más que capturar todo ese tráfico y conseguir así un acceso válido al servidor. Incluso puede ser una amenaza a la privacidad de los datos el hecho de que ese atacante también pueda capturar y reproducir (y modificar) los archivos transferidos.

Para solucionar este problema es conveniente dar acceso FTP a pocos usuarios bien identificados y que necesiten utilizarlo, concientizándolos de la utilidad de aplicaciones que cifren todo el tráfico de información (como SSH por ejemplo).

8.9.8.1 FTP ANÓNIMO

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

El servicio FTP se vuelve especialmente preocupantes cuando se trata de configurar un servidor de FTP anónimo; muchos de estas máquinas situadas en universidades y empresas se convierten en servidores de imágenes pornográficas, de Warez (copias ilegales de programas comerciales), etc. Conseguir un servidor de FTP anónimo seguro puede llegar a ser una tarea complicada.

El usuario Anónimo debe conectar a un entorno restringido del sistema y sólo a ese.

8.9.8.2 FTP INVITADO

El otro tipo de acceso FTP es el denominado invitado (guest). La idea de este mecanismo es muy sencilla: se trata de permitir que cada usuario conecte a la máquina mediante su login y su contraseña, pero evitando que tenga acceso a partes del sistema de archivos que no necesita para realizar su trabajo; se conectará a un entorno restringido de forma similar a lo que sucede en los accesos anónimos.

Para poder crear fácilmente entornos FTP restringidos a cada usuario, es conveniente instalar programas para este fin en la máquina servidor. Estos servidores permiten crear usuarios invitados configurando el entorno al que van a conectarse los usuarios, su estructura de directorios–archivos y sus permisos a los recursos.

8.9.9 TELNET

El protocolo TELNET (TCP, puerto 23) permite utilizar una máquina como terminal virtual de otra a través de la red, de forma que se crea un canal virtual de comunicaciones similar (pero mucho más inseguro) a utilizar una terminal físicamente conectada a un servidor.

TELNET es el clásico servicio que hasta hace unos años no se solía deshabilitar nunca: lo más normal es que este servicio esté disponible para que los usuarios puedan trabajar remotamente, al menos desde un conjunto de máquinas determinado.

Evidentemente, reducir al mínimo imprescindible el conjunto de sistemas desde donde es posible la conexión es una primera medida de seguridad; no obstante, no suele ser suficiente.

TELNET no utiliza ningún tipo de cifrado, por lo que todo el tráfico entre equipos se realiza en texto claro. Cualquier intruso con un Sniffer puede capturar el Login y el password utilizados en una conexión otorgando a cualquiera que lea esos datos un acceso total a la máquina destino. Es muy recomendable no utilizar TELNET para conexiones remotas, sino sustituirlo por aplicaciones equivalentes pero que utilizan cifrado para la transmisión de datos (SSH o SSL–Telnet por ejemplo).

8.9.10 SMTP

La mala configuración del servicio SMTP (puerto 25 en TCP) utilizado para transferir correo electrónico entre equipos remotos; suele ser causante del Mail Bombing y el Spam redirigido.

Por lo general se recibirá correo de un número indeterminado de máquinas, y no se podrá bloquear el acceso a SMTP. No obstante, en este caso podemos aplicar unas medidas de seguridad simples, como realizar una consulta inversa a DNS para asegurarnos de que sólo máquinas registradas envían correo o no permitir que el sistema reenvíe correo que no provenga de direcciones registradas bajo su dominio.

8.9.11 SERVIDORES WWW

Tesis "Seguridad Informática: Sus <input type="checkbox"/>
Implicancias e Implementación". <input type="checkbox"/>
Copyright Cristian F. Borghello 2001 <input type="checkbox"/>
webmaster@cfbsoft.com.ar <input type="checkbox"/>
www.cfbsoft.com.ar <input type="checkbox"/>

Hoy en día las conexiones a servidores web son sin duda las más extendidas entre usuarios de Internet. En la actualidad mueve a diario millones de dólares y es uno de los pilares fundamentales de muchas empresas: es por tanto un objetivo muy atractivo para cualquier intruso.

Los problemas de seguridad relacionados con el protocolo HTTP se dividen en tres grandes grupos en función de los datos a los que pueden afectar:⁸

- **Seguridad en el servidor:** es necesario garantizar que la información almacenada en la máquina servidora no pueda ser modificada sin autorización, que permanezca

⁸ HUERTA, Antonio Villalón. Seguridad en Unix y redes. (Versión 1.2)–Digital Open Publication License v.10 o Later. 2 de Octubre de 2000. Capítulo 11–Página 190. <http://www.kriptopolis.com>.

disponible y que sólo pueda ser accedida por los usuarios a los que les esté legítimamente permitido.

- **Seguridad en la red:** cuando un usuario conecta a un servidor web se produce un intercambio de información entre ambos; es vital garantizar que los datos que recibe el cliente desde el servidor sean los mismos que se están enviando (esto es, que no sufran modificaciones de terceros), y también garantizar que la información que el usuario envía hacia el servidor no sea capturada, destruida o modificada por un atacante.
- **Seguridad en el cliente:** es necesario garantizar al usuario que descarga páginas de un servidor no va a perjudicar a la seguridad de su equipo. Se deben evitar Applets maliciosos, programas con virus o simples cuelgues al acceder a las páginas de la organización. Ante hechos de esta especie seguramente la persona dejará de visitarlas, con la consecuente pérdida de imagen (y posiblemente un cliente) de esa entidad.

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

Asegurar el servidor implica (aparte de las medidas habituales) medidas excepcionales dedicadas al servidor de Web y su entorno de trabajo.

Sea cual sea el servidor utilizado (IIS, Apache, NCSA, Netscape, etc.), es necesario seguir un consejo básico: minimizar el número de usuarios en la máquina y minimizar el número de servicios ofrecidos en ella; aunque lo normal es que una máquina dedicada a cualquier tarea, sea también el servidor Web, es recomendable que dicho servidor sea un equipo dedicado sólo a esa tarea.

Los problemas relacionados con servidores Web suelen proceder de errores de programación en los CGIs ubicados en el servidor. La capacidad del CGI para comunicarse con el resto del sistema que alberga las páginas es lo que le otorga su potencia, pero también lo que causa mayores problemas de seguridad: un fallo en estos programas suele permitir a cualquier "visitante" ejecutar órdenes en el sistema.

Una medida de seguridad básica es ejecutar el demonio servidor bajo la identidad de un usuario con privilegios mínimos para que todo funcione correctamente, pero nunca como Administrador, Root o cuenta del sistema.

Para garantizar la seguridad de los datos que circulan entre un cliente y el servidor es casi obligatorio cifrar dichos datos (mediante SSL o utilizando Certificados Digitales por ejemplo).

8.10 CRIPTOLOGÍA

8.10.1 HISTORIA

En el año 500 a.C. los griegos utilizaron un cilindro llamado “scytale” alrededor del cual enrollaban una tira de cuero. Al escribir un mensaje sobre el cuero y desenrollarlo se veía una lista de letras sin sentido. El mensaje correcto sólo podía leerse al enrollar el cuero nuevamente en un cilindro de igual diámetro.

Durante el Imperio Romano Julio Cesar empleo un sistema de cifrado consistente en sustituir la letra a encriptar por otra letra distanciada a tres posiciones más adelante. Durante su reinado, los mensajes de Julio Cesar nunca fueron descifrados.

En el S. XII Roger Bacon y en el S. XV León Batista Alberti inventaron y publicaron sendos algoritmos de encriptación basados en modificaciones del método de Julio César.

Durante la segunda guerra mundial en un lugar llamado Bletchley Park (70 Km al norte de Londres) un grupo de científicos trabajaba en Enigma, la máquina encargada de cifrar los mensajes secretos alemanes.

En este grupo se encontraban tres matemáticos polacos llamados Marian Rejewski, Jerzy Rozycki, Henryk Zygalski y “un joven que se mordía siempre las pieles alrededor de las uñas, iba con ropa sin planchar y era más bien bajito. Este joven retraído se llamaba Alan Turing y había sido reclutado porque unos años antes había creado un ordenador binario. Probablemente poca gente en los servicios secretos ingleses sabía lo que era un ordenador (y mucho menos binario)... pero no había duda que sólo alguien realmente inteligente podía inventar algo así, cualquier cosa que eso fuese... Era mucho más abstracto que todos sus antecesores y sólo utilizaba 0 y 1 como valores posibles de las variables de su álgebra.”⁹.

Sería Turing el encargado de descifrar el primer mensaje de Enigma y, cambiar el curso de la guerra, la historia y de... la Seguridad Informática actual.

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

8.10.2 CRIPTOGRAFÍA

La palabra **Criptografía** proviene etimológicamente del griego Κρυπτοζ (Kriptos–Oculto) y Γραφειν (Grafo–Escritura) y significa “arte de escribir con clave secreta o de un modo enigmático”¹⁰.

Aportando luz a la definición cabe aclarar que la Criptografía hace años que dejó de ser un arte para convertirse en una técnica (o conjunto de ellas) que tratan sobre la protección (ocultamiento ante personas no autorizadas) de la información. Entre las disciplinas que engloba cabe destacar la Teoría de la Información, la Matemática Discreta, la Teoría de los Grandes Números y la Complejidad Algorítmica.

Es decir que la **Criptografía** es la ciencia que consiste en transformar un mensaje inteligible en otro que no lo es (mediante claves que sólo el emisor y el destinatario conocen),

⁹ Extraído de <http://www.kriptopolis.com>

¹⁰ LUCENA LÓPEZ, Manuel José. Criptografía y Seguridad en Computadores. Dpto. de Informática Universidad de Jaén. Edición virtual. España. 1999. <http://www.kriptopolis.com>. Capítulo 2–Página 23.

para después devolverlo a su forma original, sin que nadie que vea el mensaje cifrado sea capaz de entenderlo.

El mensaje cifrado recibe el nombre **Criptograma**



Gráfico 8.6 – Criptograma

La importancia de la Criptografía radica en que es el único método actual capaz de hacer cumplir el objetivo de la Seguridad Informática: “mantener la Privacidad, Integridad, Autenticidad...” y hacer cumplir con el **No Rechazo**, relacionado a no poder negar la autoría y recepción de un mensaje enviado.

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

8.10.3 CRIPTOANÁLISIS

Es el arte de estudiar los mensajes ilegibles, encriptados, para transformarlos en legibles sin conocer la clave, aunque el método de cifrado empleado siempre es conocido.

8.10.4 CRIPTOSISTEMA

“Un Criptosistema se define como la quintupla **(m,C,K,E,D)**, donde:

- **m** representa el conjunto de todos los mensajes sin cifrar (texto plano) que pueden ser enviados.
- **C** Representa el conjunto de todos los posibles mensajes cifrados, o criptogramas.
- **K** representa el conjunto de claves que se pueden emplear en el Criptosistema.
- **E** es el conjunto de transformaciones de cifrado o familia de funciones que se aplica a cada elemento de **m** para obtener un elemento de **C**. Existe una transformación diferente **E_K** para cada valor posible de la clave **K**.
- **D** es el conjunto de transformaciones de descifrado, análogo a **E**.

Todo Criptosistema cumple la condición **D_K(E_K(m)) = m** es decir, que si se tiene un mensaje **m**, se cifra empleando la clave **K** y luego se descifra empleando la misma clave, se obtiene el mensaje original **m**.¹¹

Existen dos tipos fundamentales de Criptosistemas utilizados para cifrar datos e información digital y ser enviados posteriormente después por medios de transmisión libre.

- a. **Simétricos o de clave privada:** se emplea la misma clave **K** para cifrar y descifrar, por lo tanto el emisor y el receptor deben poseer la clave. El mayor inconveniente

¹¹ LUCENA LÓPEZ, Manuel José. Criptografía y Seguridad en Computadores. Dpto. de Informática Universidad de Jaén. Edición virtual. España. 1999. <http://www.kriptopolis.com>. Capítulo 2–Página 24.

que presentan es que se debe contar con un canal seguro para la transmisión de dicha clave.

- b. **Asimétricos o de llave pública:** se emplea una doble clave conocidas como K_p (clave privada) y K_P (clave Pública). Una de ellas es utilizada para la transformación E de cifrado y la otra para el descifrado D. En muchos de los sistemas existentes estas clave son intercambiables, es decir que si empleamos una para cifrar se utiliza la otra para descifrar y viceversa.

Los sistemas asimétricos deben cumplir con la condición que la clave Publica (al ser conocida y sólo utilizada para cifrar) no debe permitir calcular la privada. Como puede observarse este sistema permite intercambiar claves en un canal inseguro de transmisión ya que lo único que se envía es la clave pública.

Los algoritmos asimétricos emplean claves de longitud mayor a los simétricos. Así, por ejemplo, suele considerarse segura una clave de 128 bits para estos últimos pero se recomienda claves de 1024 bits (como mínimo) para los algoritmos asimétricos. Esto permite que los algoritmos simétricos sean considerablemente más rápidos que los asimétricos.

En la práctica actualmente se emplea una combinación de ambos sistemas ya que los asimétricos son computacionalmente más costosos (mayor tiempo de cifrado). Para realizar dicha combinación se cifra el mensaje m con un sistema simétrico y luego se encripta la clave K utilizada en el algoritmo simétrico (generalmente más corta que el mensaje) con un sistema asimétrico.

Después de estos Criptosistemas modernos podemos encontrar otros no menos importantes utilizados desde siempre para cifrar mensajes de menos importancia o domésticos, y que han ido perdiendo su eficacia por ser fácilmente criptoanalizables y por tanto “reventables”. Cada uno de los algoritmos clásicos descritos a continuación utilizan la misma clave K para cifrar y descifrar el mensaje.

8.10.4.1 TRANSPOSICIÓN

Tesis "Seguridad Informática: Sus Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

Son aquellos que alteran el orden de los caracteres dentro del mensaje a cifrar. El algoritmo de transposición más común consiste en colocar el texto en una tabla de n columnas. El texto cifrado serán los caracteres dados por columna (de arriba hacia abajo) con una clave K consistente en el orden en que se leen las columnas.

Ejemplo: Si $n = 3$ columnas, la clave K es (3,1,2) y el mensaje a cifrar “SEGURIDAD INFORMATICA”.

1	2	3
S	E	G
U	R	I
D	A	D
	I	N
F	O	R
M	A	T
I	C	A

El mensaje cifrado será: “GIDNRTASUD FMIERAIOAC”

8.10.4.2 CIFRADOS MONOALFABÉTICOS

Sin desordenar los símbolos del lenguaje, se establece una correspondencia única para todos ellos en todo el mensaje. Es decir que si al carácter A le corresponde carácter D, esta correspondencia se mantiene durante todo el mensaje.

8.10.4.2.1 Algoritmo de César

Es uno de los algoritmos criptográficos más simples. Consiste en sumar 3 al número de orden de cada letra. De esta forma a la A le corresponde la D, a la B la E, y así sucesivamente. Puede observarse que este algoritmo ni siquiera posee clave, puesto que la transformación siempre es la misma. Obviamente, para descifrar basta con restar 3 al número de orden de las letras del criptograma.

Ejemplo: Si el algoritmo de cifrado es:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
```

Entonces el mensaje cifrado será:

```
S E G U R I D A D   I N F O R M A T I C A
V H J X U L G D G   L Q I R U P D W L F D
```

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

8.10.4.2.2 Sustitución General

Es el caso general del algoritmo de César. El sistema consiste en sustituir cada letra por otra aleatoria. Esto supone un grado más de complejidad aunque como es de suponer las propiedades estadísticas del texto original se conservan en el criptograma y por lo tanto el sistema sigue siendo criptoanalizable.

8.10.5 ALGORITMOS SIMÉTRICOS MODERNOS (LLAVE PRIVADA)

La mayoría de los algoritmos simétricos actuales se apoyan en los conceptos de **Confusión** y **Difusión** vertidos por Claude Shannon sobre la Teoría de la Información a finales de los años cuarenta.

Estos métodos consisten en ocultar la relación entre el texto plano, el texto cifrado y la clave (Confusión); y repartir la influencia de cada bit del mensaje original lo más posible entre el mensaje cifrado (Difusión).

El objetivo del presente no es entrar en detalles de cada uno de los muchos algoritmos existentes, por lo que sólo se dará una idea de su funcionamiento y complejidad.

8.10.5.1 REDES DE FEISTEL

Este algoritmo no es un algoritmo de cifrado per se, pero muchos de los vistos a continuación lo utilizan como parte vital en su funcionamiento. Se basa en dividir un bloque

de longitud n (generalmente el texto a cifrar) en dos mitades, L y R. Luego se define un cifrado de producto interactivo en el que la salida de cada ronda es la entrada de la siguiente.

8.10.5.2 DES

Data Encryption Standard es el algoritmo simétrico más extendido mundialmente. A mediados de los setenta fue adoptado como estándar para las comunicaciones seguras (Estándar AES) del gobierno de EE.UU. En su principio fue diseñado por la NSA (National Security Agency)¹² para ser implementado en hardware, pero al extenderse su algoritmo se comenzó a implementar en software.

DES utiliza bloques de 64 bits, los cuales codifica empleando claves de 56 bits y aplicando permutaciones a nivel de bit en diferentes momentos (mediante tablas de permutaciones y operaciones XOR). Es una red de Feistel de 16 rondas, más dos permutaciones, una que se aplica al principio y otra al final.

La flexibilidad de DES reside en que el mismo algoritmo puede ser utilizado tanto para cifrar como para descifrar, simplemente invirtiendo el orden de las 16 subclaves obtenidas a partir de la clave de cifrado.

En la actualidad no se ha podido romper el sistema DES criptoanalíticamente (deducir la clave simétrica a partir de la información interceptada). Sin embargo una empresa española sin fines de lucro llamado Electronic Frontier Foundation (EFF)¹³ construyo en Enero de 1999 una máquina capaz de probar las 2^{56} claves posibles en DES y romperlo sólo en tres días con fuerza bruta.

A pesar de su caída DES sigue siendo utilizado por su amplia extensión de las implementaciones vía hardware existentes (en cajeros automáticos y señales de video por ejemplo) y se evita tener que confiar en nuevas tecnologías no probadas. En vez de abandonar su utilización se prefiere suplantar a DES con lo que se conoce como cifrado múltiple, es decir aplicar varias veces el mismo algoritmo para fortalecer la longitud de la clave

8.10.5.2.1 DES Múltiple

Consiste en aplicar varias veces el algoritmo DES (con diferentes claves) al mensaje original. El más conocidos de todos ellos el Triple-DES (T-DES), el cual consiste en aplicar 3 veces DES de la siguiente manera:

1. Se codifica con la clave K_1 .
2. Se decodifica el resultado con la clave K_2 .
3. Lo obtenido se vuelve a codificar con K_1 .

La clave resultante es la concatenación de K_1 y K_2 con una longitud de 112 bits.

En 1998 el NIST (National Institute of Standards Technology) convocó a un concurso para poder determinar un algoritmo simétrico seguro y próximo sustituto de DES. Se aceptaron

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

¹² National Security Agency (NSA): <http://www.nsa.gov:8080>

¹³ Electronic Frontier Foundation (EFF) actualmente ya no se encuentra trabajando pero puede visitarse su sitio en <http://www.eff.com>

15 candidatos y a principios del año 2000 los 5 finalistas fueron MARS, RC-6, Serpent y TwoFish y Rijndael (que en octubre sería el ganador).

8.10.5.3 IDEA

El International Data Encryption Algorithm fue desarrollado en Alemania a principios de los noventa por James L. Massey y Xuejia Lai.

Trabaja con bloques de 64 bits de longitud empleando una clave de 128 bits y, como en el caso de DES, se utiliza el mismo algoritmo tanto para cifrar como para descifrar.

El proceso de encriptación consiste ocho rondas de cifrado idéntico, excepto por las subclaves utilizadas (segmentos de 16 bits de los 128 de la clave), en donde se combinan diferentes operaciones matemáticas (XORs y Sumas Módulo 16) y una transformación final.

“En mi opinión, él es el mejor y más seguro algoritmo de bloques disponible actualmente al público.”¹⁴

8.10.5.4 BLOWFISH

Este algoritmo fue desarrollado por Bruce Schneier en 1993. Para la encriptación emplea bloques de 64 bits y permite claves de encriptación de diversas longitudes (hasta 448 bits).

Generalmente, utiliza valores decimales de π (aunque puede cambiarse a voluntad) para obtener las funciones de encriptación y desencriptación. Estas funciones emplean operaciones lógicas simples y presentes en cualquier procesador. Esto se traduce en un algoritmo “liviano”, que permite su implementación, vía hardware, en cualquier controlador (como teléfonos celulares por ejemplo).

8.10.5.5 RC5

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

Este algoritmo, diseñado por RSA¹⁵, permite definir el tamaño del bloque a encriptar, el tamaño de la clave utilizada y el número de fases de encriptación. El algoritmo genera una tabla de encriptación y luego procede a encriptar o desencriptar los datos.

8.10.5.6 CAST

Es un buen sistema de cifrado en bloques con una clave de 128 bits, es muy rápido y es gratuito. Su nombre deriva de las iniciales de sus autores, Carlisle, Adams, Stafford Tavares, de la empresa Northern Telecom (NorTel).

CAST no tiene claves débiles o semidébiles y hay fuertes argumentos acerca que CAST es completamente inmune a los métodos de criptoanálisis más potentes conocidos.

¹⁴ SCHNEIER, Bruce. Applied Cryptography. Segunda Edición. EE.UU. 1996

¹⁵ RSA Labs: <http://www.rsa.com>. No confundir con el algoritmo de clave pública del mismo nombre RSA (Rivest-Shamir-Adleman)

8.10.5.7 RIJNDAEL (EL NUEVO ESTÁNDAR AES)

Este nuevo algoritmo belga mezcla de Vincent Rijmen y Joan Daemen (sus autores) sorprende tanto por su innovador diseño como por su simplicidad práctica; aunque tras él se esconda un complejo trasfondo matemático.

Su algoritmo no se basa en redes de Feistel, y en su lugar se ha definido una estructura de “capas” formadas por funciones polinómicas reversibles (tienen inversa) y no lineales. Es fácil imaginar que el proceso de descifrado consiste en aplicar las funciones inversas a las aplicadas para cifrar, en el orden contrario.

Las implementaciones actuales pueden utilizar bloques de 128, 192 y 256 bits de longitud combinadas con claves de 128, 192 y 256 bits para su cifrado; aunque tanto los bloques como las claves pueden extenderse en múltiplo de 32 bits.

Si bien su joven edad no permite asegurar nada, según sus autores, es altamente improbable que existan claves débiles en el nuevo AES. También se ha probado la resistencia al criptoanálisis tanto lineal como diferencial, asegurando así la desaparición de DES.

8.10.5.8 CRIPTOANÁLISIS DE ALGORITMOS SIMÉTRICOS

El Criptoanálisis comenzó a extenderse a partir de la aparición de DES por sospechas (nunca confirmadas) de que el algoritmo propuesto por la NSA contenía puertas traseras. Entre los ataques más potentes a la criptografía simétrica se encuentran:

- **Criptoanálisis Diferencial:** Ideado por Biham y Shamir en 1990, se basa en el estudio de dos textos codificados para estudiar las diferencias entre ambos mientras se los está codificando. Luego puede asignarse probabilidades a ciertas claves de cifrado.
- **Criptoanálisis Lineal:** Ideado por Mitsuru Matsui, se basa en tomar porciones del texto cifrado y porciones de otro texto plano y efectuar operaciones sobre ellos de forma tal de obtener probabilidades de aparición de ciertas claves.

Sin embargo, estos métodos, no han podido ser muy eficientes en la práctica. En el momento después de que un sistema criptográfico es publicado y se muestra inmune a estos dos tipos de ataques (y otros pocos) la mayor preocupación es la longitud de las claves.

8.10.6 ALGORITMOS ASIMÉTRICOS (LLAVE PRIVADA–PÚBLICA)

Ideado por los matemáticos Whitfield Diffie y Martín Hellman (DH) con el informático Ralph Merkle a mediados de los 70, estos algoritmos han demostrado su seguridad en comunicaciones inseguras como Internet. Su principal característica es que no se basa en una única clave sino en un par de ellas: una conocida (Pública) y otra Privada.

Actualmente existen muchos algoritmos de este tipo pero han demostrado ser poco utilizables en la práctica ya sea por la longitud de las clave, la longitud del texto encriptado generado o su velocidad de cifrado extremadamente largos.

DH está basado en las propiedades y en el tiempo necesario para calcular el valor del logaritmo de un número extremadamente alto y primo.

8.10.6.1 RSA

Este algoritmo fue ideado en 1977 por Ron Rivest, Adi Shamir y Leonard Adleman (RSA). Es el más empleado en la actualidad, sencillo de comprender e implementar, aunque la longitud de sus claves es bastante considerable (ha pasado desde sus 200 bits originales a 2048 actualmente).

RSA es la suma de dos de los algoritmos mas importantes de la historia: el Máximo Común Divisor de Euclides (Grecia 450–377 A.C.) y el último teorema de Fermat (Francia 1601–1665).

Se emplean las ventajas proporcionadas por las propiedades de los números primos cuando se aplican sobre ellos operaciones matemáticas basadas en la función módulo. En concreto, emplea la función exponencial discreta para cifrar y descifrar, y cuya inversa, el logaritmo discreto, el muy difícil de calcular.

Los cálculos matemáticos de este algoritmo emplean un número denominado Módulo Público, N , que forma parte de la clave pública y que se obtiene a partir de la multiplicación de dos números primos, p y q , diferentes y grandes (del orden de 512 bits) y que forman parte de la clave privada. La gran propiedad de RSA es que, mientras que N es público, los valores de p y q se pueden mantener en secreto debido a la dificultad que entraña la factorización de un número grande.

La robustez del algoritmo se basa en la facilidad para encontrar dos números primos grandes frente a la enorme dificultad que presenta la factorización de su producto. Aunque el avance tecnológico hace que cada vez sea más rápido un posible ataque por fuerza bruta, el simple hecho de aumentar la longitud de las claves empleadas supone un incremento en la carga computacional lo suficientemente grande para que este tipo de ataque sea inviable.

Sin embargo, se ha de notar que, aunque el hecho de aumentar la longitud de las claves RSA no supone ninguna dificultad tecnológica, las leyes de exportación de criptografía de EE.UU. imponían, hasta el 20 de septiembre de 2000, un límite a dicha longitud por lo que el su uso comercial de RSA no estaba permitido, ya que la patente pertenecía a los laboratorios RSA. Desde esta fecha su uso es libre.

8.10.6.1.1 Ataques a RSA

Si un atacante quiere recuperar la clave privada a partir de la pública debe obtener p y q a partir de N , lo cual actualmente es un problema intratable si los números primos son lo suficientemente grandes (alrededor de 200 dígitos).

Vale decir que nadie a demostrado que no pueda existir un método que permita descifrar un mensaje sin usar la clave privada y sin factorizar N . Así, aunque el algoritmo es

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

bastante seguro conceptualmente, existen algunos ataques que pueden ser efectivos al apoyarse sobre deficiencias en la implementación y uso del mismo.

El ataque que con mayores probabilidades de éxito es el **ataque de intermediario**, que en realidad puede darse sobre cualquier algoritmo de clave pública. Supongamos:

... que A quiere establecer una comunicación con B, y que C quiere espiarla. Cuando A le solicite a B su clave pública K_B , C se interpone, obteniendo la clave de B y enviado a A una clave falsa K_C , creada por él. Cuando A codifique el mensaje, C lo intercepta de nuevo, lo decodifica con su clave propia y emplea K_B para codificarlo y enviarlo a B... ni A ni B sospecharán nunca de lo sucedido.

La única manera de evitar esto consiste en asegurar a A que la clave pública de B es auténtica. Para ello esta debería ser firmada por un amigo común que, actuando como Autoridad Certificadora, certifique su autenticidad.

Otros ataques (como el de claves débiles, el de texto plano escogido, el de módulo común, y el de exponente bajo) aprovechan vulnerabilidades específicas de algunas implementaciones.

8.10.6.2 CURVAS ELÍPTICAS (CEE)

Las curvas elípticas fueron propuestas por primera vez para ser usadas en aplicaciones criptográficas en 1985 de forma independiente por Miller y Koblitz. Las curvas elípticas en sí llevan estudiándose durante muchos siglos y están entre los objetos más ricamente estructurados y estudiados de la teoría de números.

La eficiencia de este algoritmo radica en la longitud reducida de las claves, lo cual permite su implementación en sistemas de bajos recursos como teléfonos celulares y Smart Cards. Puede hacerse la siguiente comparación con RSA, obteniendo el mismo nivel de seguridad:

- CCE de 163 bits = RSA de 1024 bits
- CCE de 224 bits = RSA de 2048 bits

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

Otros algoritmos asimétricos conocidos son **ElGamal** (basado en el Problema de los Logaritmos Discretos de Diffie–Hellman DH), **Rabin** (basado en el problema del cálculo de raíces cuadradas módulo un número compuesto), **DSS** y **LUC**.

8.10.7 AUTENTIFICACIÓN

Es de destacar que muchas de estas definiciones, pueden ser encontradas en el texto del Proyecto de “Ley de Firma Digital” (ver Anexo Leyes) actualmente con media sanción.

Se entiende por Autenticación cualquier método que permita garantizar alguna característica sobre un objeto dado. Interesa comprobar la autenticación de:

- a. Un Mensaje mediante una firma: se debe garantizar la procedencia de un mensaje conocido, de forma de poder asegurar que no es una falsificación. A este mecanismo se lo conoce como **Firma Digital** y consiste en asegurar que el mensaje **m** proviene del emisor **E** y no de otro.

- b. Un Usuario mediante una contraseña: se debe garantizar la presencia de un usuario autorizado mediante una contraseña secreta.
- c. Un Dispositivo: se debe garantizar la presencia de un dispositivo válido en el sistema, por ejemplo una llave electrónica.

8.10.7.1 FIRMA DIGITAL

Una firma digital se logra mediante una Función Hash de Resumen. Esta función se encarga de obtener una “muestra única” del mensaje original. Dicha muestra es más pequeña y es muy difícil encontrar otro mensaje que tenga la misma firma. Suponiendo que B envía un mensaje m firmado a A, el procedimiento es:

- a. B genera un resumen del mensaje $r(m)$ y lo cifra con su clave privada.
- b. B envía el criptograma.
- c. A genera su propia copia de $r(m)$ usando la clave pública de B asociada a la privada.
- d. A compara su criptograma con el recibido y si coinciden el mensaje es auténtico.

Cabe destacar que:

- 1. Cualquiera que posea la clave pública de B puede constatar que el mensaje proviene realmente de B.
- 2. La firma digital es distinta en todos los documentos: si A firma dos documentos produce dos criptogramas distintos y; si A y B firman el mismo documento m también se producen dos criptogramas diferentes.

Las funciones Hash están basadas en que un mensaje de longitud arbitraria se transforma en un mensaje de longitud constante dividiendo el mensaje en partes iguales, aplicando la función de transformación a cada parte y sumando todos los resultados obtenidos.

Actualmente se recomienda utilizar firmas de al menos 128 bits (38 dígitos) siendo 160 bits (48 dígitos) el valor más utilizado.

8.10.7.1.1 MD5

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

El Message Digest 5 (resultado mejorado sobre el MD4 original de Ron Rivest) procesa los mensajes de entrada en bloques de 512, y que produce una salida de 128 bits.

Siendo m un mensaje de b bits de longitud, se alarga m hasta que su longitud sea 64 bits inferior a un múltiplo de 512. Esto se realiza agregando un 1 y tantos ceros como sea necesario. A continuación se agregan 64 bits con el valor de b comenzando por el byte menos significativo.

A continuación se realizan 64 operaciones divididas en 4 rondas sobre estos bloques de 512 bits. Finalmente, se suman y concatenan los bloques obteniendo la firma deseada de m .

8.10.7.1.2 SHA-1

El Secure Hash Algorithm fue desarrollado por la NSA, y genera firmas de 160 bits a partir de bloques de 512 bits del mensaje original.

Su funcionamiento es similar al MD5, solo variando la longitud de los bloques y la cantidad de operaciones realizadas en las 5 rondas en las que se divide el proceso.

Otros algoritmos utilizados para obtener firmas digitales son: DSA (Digital Signature Algorithm) y el RIPE–MD160.

8.10.8 PGP (PRETTY GOOD PRIVACY)

Este proyecto de “Seguridad Bastante Buena” pertenece a Phill Zimmerman quien decidió crearlo en 1991 “por falta de herramientas criptográficas sencillas, potentes, baratas y al alcance del usuario común. Es personal. Es privado. Y no es de interés para nadie más que no sea usted... Existe una necesidad social en crecimiento para esto. Es por eso que lo creé.”¹⁶

Actualmente PGP es la herramienta más popular y fiable para mantener la seguridad y privacidad en las comunicaciones tanto para pequeños usuarios como para grandes empresas.

8.10.8.1 FUNCIONAMIENTO DE PGP

8.10.8.1.1 Anillos de Claves

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

Un anillo es una colección de claves almacenadas en un archivo. Cada usuario tiene dos anillos, uno para las claves públicas y otro para las claves privadas.

Cada una de las claves, además, posee un identificador de usuario, fecha de expiración, versión de PGP y una huella digital única hexadecimal suficientemente corta que permita verificar la autenticidad de la clave.

8.10.8.1.2 Codificación de Mensajes

Como ya se sabe, los algoritmos simétricos de cifrado son más rápidos que los asimétricos. Por esta razón PGP cifra primero el mensaje empleando un algoritmo simétrico con una clave generada aleatoriamente (clave de sesión) y posteriormente codifica la clave haciendo uso de la llave pública del destinatario. Dicha clave es extraída convenientemente del anillo de claves públicas a partir del identificador suministrado por el usuario

Nótese que para que el mensaje pueda ser leído por múltiples destinatarios basta con que se incluya en la cabecera cada una de las claves públicas correspondientes.

8.10.8.1.3 Decodificación de Mensajes

Cuando se trata de decodificar el mensaje, PGP simplemente busca en la cabecera las claves públicas con las que está codificado, pide una contraseña para abrir el anillo de claves privadas y comprueba si se tiene una clave que permita decodificar el mensaje.

¹⁶ “Porque escribí PGP”. Declaraciones de Phill Zimmerman. <http://www.pgpi.com> – <http://pgp.org>

Nótese que siempre que se quiere hacer uso de una clave privada, habrá que suministrar la contraseña correspondiente, por lo que si este anillo quedara comprometido, el atacante tendría que averiguar dicha contraseña para descifrar los mensajes.

No obstante, si el anillo de claves privadas quedara comprometido, es recomendable revocar todas las claves almacenadas y generar otras nuevas.

8.10.8.1.4 Compresión de Archivos

PGP generalmente comprime el texto plano antes de encriptar el mensaje (y lo descomprime después de desencriptarlo) para disminuir el tiempo de cifrado, de transmisión y de alguna manera fortalecer la seguridad del cifrado ante el criptoanálisis que explotan las redundancias del texto plano.

PGP utiliza rutinas de compresión de dominio público creadas por Gailly–Adler–Wales (basadas en los algoritmos de Liv–Zemple) funcionalmente semejantes a las utilizadas en los softwares comerciales de este tipo.

8.10.8.1.5 Algoritmos Utilizados por PGP

Las diferentes versiones de PGP han ido adoptando diferentes combinación de algoritmos de signatura y cifrado eligiendo entre los estudiados. Las signatura se realizan mediante MD5, SHA–1 y/o RIPE–MD6. Los algoritmos simétricos utilizados pueden ser IDEA, CAST y TDES y los asimétricos RSA y ElGamal.

8.10.9 ESTEGANOGRAFÍA

Consiste en ocultar en el interior de información aparentemente inocua, otro tipo de información (cifrada o no). El texto se envía como texto plano, pero entremezclado con mucha cantidad de “basura” que sirve de camuflaje al mensaje enviado. El método de recuperación y lectura sólo es conocido por el destinatario del mensaje y se conoce como “separar el grano de la paja”.

Los mensajes suelen ir ocultos entre archivos de sonido o imágenes y ser enormemente grandes por la cantidad extra de información enviada (a comparación del mensaje original).

8.11 COMERCIO ELECTRÓNICO

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

El comercio electrónico abarca todos los conceptos relacionados con procesos de mercado entre entidades físicas o jurídicas pero a través de redes de telecomunicaciones.

El principal requisito que debe tener una transacción electrónica es la **Seguridad** además de:

- **Confidencialidad (anonimato):** la identidad del comprador no es conocida por el vendedor; nadie, excepto el banco, debería conocer la identidad del comprador; el banco debería ignorar la naturaleza de la compra y; un tercero no debería poder acceder a la información enviada.

- **Autenticación:** permite a cada lado de la comunicación asegurarse de que el otro es quien dice ser.
- **Integridad:** evita que un tercero pueda modificar la información enviada por cualquiera de las partes.
- **No Repudio o Irrefutabilidad:** permite, a cada lado de la comunicación, probar fehacientemente que el otro lado ha participado: el origen no puede negar haberlo enviado y el destino no puede negar haberlo recibido.
- **Flexibilidad:** aceptar todas las posibles formas de pago existentes.
- **Eficiencia:** el costo del servicio no debe ser mayor que el precio del producto o servicio.

8.11.1 DINERO ELECTRÓNICO

Como ya se mencionó, si alguien desea verificar la autenticidad de un mensaje (un banco por ejemplo) debe poseer la clave pública del emisor. Es decir que una persona que se dedique a autenticar documentos deberá poseer una cantidad considerable de claves almacenadas. Este problema se soluciona aplicando un Certificado Digital (CD) emitido y firmado por una Autoridad Certificadora (AC).

El CD es un documento firmado digitalmente por la AC y establece una relación entre una persona y su llave pública.

La idea es que cualquiera que conozca la llave pública de la AC puede autenticar un CD de la misma manera que se autentifica cualquier documento físico. Si se confía en la AC, entonces se puede confiar que la clave pública que figura en el Certificado es de la persona que dice ser.

Luego, si una persona firma un documento y anexa su CD, cualquiera que conozca la clave pública de la AC (una única clave) podrá verificar la autenticidad del documento.

El Estándar internacional para CD más aceptado y extendido en la actualidad es el denominado X.509.

8.11.1.1 CERTIFICADOS X.509

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

Este certificado se basa en la Recomendación X.509 de CCITT llamada "The Directory–Autentication Framework", que data de 1988 y actualmente se encuentra en su versión 3.

Un Certificado es esencialmente una Clave Pública y un Identificador, firmados digitalmente por una autoridad de certificación, y su utilidad es demostrar que una clave pública pertenece a un usuario concreto.

El estándar X.509 sólo define la sintaxis de los certificados, por lo que no está atado a ningún algoritmo en particular, y contempla los siguientes campos:

1. **Versión:** Indica si la versión del certificado X.509 es la 1 (defecto), 2 ó 3.
2. **Número de serie:** Es un número entero asignado por la AC emisora y que identifica unívocamente al certificado dentro del conjunto de certificados emitidos.

3. **Firma:** Identifica al algoritmo utilizado por la AC para firmar el certificado.
4. **Emisor:** El nombre del emisor identifica a la entidad que ha firmado el certificado.
5. **Validez:** Indica el intervalo de tiempo en el que el certificado es válido.
6. **Usuario o Sujeto:** Es un nombre distinguible X.500 que identifica de forma unívoca al poseedor del certificado; y la nomenclatura de nombres distinguibles (DN: Distinguished Names).
7. **Clave pública del usuario:** Contiene la clave pública del usuario junto con el identificador del algoritmo con el que se ha de utilizar.
8. **Identificadores únicos de emisor y de usuario:** Es una cadena de bits opcional que identifica al emisor o al usuario en el caso de que su DN sea reutilizado con el paso del tiempo.
9. **Campos de extensión:** Permiten la adición de nuevos campos a la estructura sin que por ello se tenga que modificar la definición del certificado.

La Firma, realizada por la AC emisora, permite que aquellas entidades que deseen realizar comunicaciones con la persona poseedora del certificado, puedan comprobar que la información que éste contiene es auténtica (suponiendo que confíen en la AC emisora).

Una vez que los certificados han sido firmados, se almacenan en servidores de directorios y/o transmitidos por cualquier medio (seguros o no) para que estén disponibles públicamente.

Los certificados tienen un periodo de vida limitado, el cual está especificado en el campo Validez, y que viene determinado por la política de la AC emisora. Sin embargo, en algunas ocasiones especiales la seguridad de la clave privada asociada puede verse comprometida, por lo que la utilización de la correspondiente clave pública ha de ser evitada. En tal caso, la AC emisora puede revocar el certificado para prevenir su uso.

8.11.1.2 SSL

Secure Sockets Layers es un protocolo seguro de Internet diseñado en 1994 por Netscape Communication Corporation® y posteriormente adoptado por otros navegadores. Es utilizado para cualquier comunicación donde deba establecerse un canal seguro (al solicitarse clave o número de tarjeta de crédito por ejemplo).

En la pila TCP/IP, se ubica entre la capa TCP (Transporte) y la de Aplicación, por lo que es muy flexible ya que puede ser utilizado en cualquier aplicación que utilice TCP/IP (Mail, HTTP, FTP, News, etc.) aunque actualmente sólo se implementa sobre HTTP. Para diferenciar las páginas comunes HTTP de las protegidas se utiliza la denominación HTTPS conectado mediante el puerto 443.

SSLv3 supera algunas limitaciones de sus versiones anteriores y ofrece estas características:

- **Cifrado de datos:** los datos viajan cifrados mediante algunos de los algoritmos vistos. Para el intercambio de datos entre servidor y cliente se utilizan algoritmos simétricos (DES-TDES, RC4, IDEA) y para la clave de sesión (utilizada para los algoritmos anteriores) cifrado asimétrico (típicamente RSA).

- **Fragmentación de datos:** en el emisor se fragmentan los datos en bloques para volver a reensamblarlos en el receptor.
- **Compresión de datos:** se puede aplicar un algoritmo de compresión a los datos.
- **Autenticación de servidores:** el usuario puede verificar la identidad del servidor al que se conecta y al que puede mandar datos confidenciales.
- **Integridad de mensajes:** las modificaciones intencionales o accidentales, de la información, en el viaje por el canal inseguro son detectadas.
- **Autenticación del cliente:** permite al servidor conocer la identidad del usuario, con el fin de decidir si este puede acceder a cierta información protegida. Esta autenticación no siempre debe darse.

Al reunir estas características, la comunicación se realiza en dos fases:

- **Saludo (Handshaking):** los interlocutores se identifican mutuamente empleando, habitualmente, certificados X.509. Tras el intercambio de claves públicas, los dos escogen una clave de sesión simétrica para el intercambio de datos.
- **Comunicación:** se produce el intercambio de información propiamente dicho, que se codifica mediante las claves de sesión ya establecidas.

De aquí en adelante, durante la sesión segura abierta, SSL proporciona un canal de comunicaciones seguro entre el servidor y el clientes a través del cual se intercambiará cifrada la siguiente información:

- La URL del documento solicitado.
- El contenido del documento solicitado.
- Los contenidos de cualquier formulario enviado desde el navegador.
- Las cookies enviadas desde el navegador al servidor y viceversa.
- Los contenidos de las cabeceras HTTP.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfssoft.com.ar □ www.cfssoft.com.ar □
--

8.11.1.2.1 Limitaciones y Problemas de SSL

1. Debido a la limitación de exportación del gobierno de los EE.UU. sobre los productos criptográficos, las versiones de los navegadores distribuidas legalmente más allá de sus fronteras operan con nada más que 40 bits de longitud de clave, frente a los 128 ó 256 bits de las versiones fuertes.

Claves tan cortas facilitan los ataques de fuerza bruta, dependiendo de los recursos informáticos disponibles. Este serio problema ganó notoriedad en los medios de comunicación cuando en 1995 un estudiante francés, Damien Doligez, fue capaz de descifrar un mensaje cifrado con SSL en pocos días utilizando la red de computadoras de su Universidad.

2. SSL **sólo** garantiza la confidencialidad e integridad de los datos en tránsito, pero nunca antes ni después. Por lo tanto, si se envían datos personales al servidor, SSL solamente asegura que no serán modificados ni espiados mientras viajan desde el navegador hasta el servidor. Lo que el servidor haga con ellos, está más allá de la competencia de este protocolo.

3. SSL **no** garantiza la identidad del servidor al que se conecta el usuario. Podría suceder que el servidor seguro contase con un certificado perfectamente válido y que estuviera suplantando la identidad de algún otro servidor seguro bien conocido. Por consiguiente, es de extrema importancia que se compruebe siempre el certificado del sitio web para cerciorarse de que no se está conectando a un web falsificado.
4. El servidor identifica al navegador incluso aunque éste no se autentique mediante certificados. Cuando un usuario se conecta a un servidor, rutinariamente le comunica ciertos datos como su dirección IP, tipo y versión de navegador, sistema operativo, y otros.
5. Actualmente SSL solamente se utiliza para comunicaciones web seguras, por lo que otros servicios de Internet, como el correo electrónico, no irán cifrados a pesar de utilizar SSL para el envío de formularios o la recuperación de páginas web. Por esto, se debe usar S/MIME, PGP o algún otro software criptográfico para correo.

8.12.1.2.2 Ventajas de SSL

1. SSL v3.0 goza de gran popularidad y se encuentra ampliamente extendido en Internet, ya que viene soportado por los dos principales navegadores del mercado, Netscape Navigator[®] e Internet Explorer[®].
2. SSL proporciona un canal de comunicaciones seguro entre los servidores web y los clientes (los navegadores), pero su uso no se limita a la transmisión de páginas web. Al encontrarse entre los niveles de transporte y de aplicación, potencialmente SSL puede servir para securizar otros servicios, como FTP, correo, telnet, etc.
3. El usuario no necesita realizar ninguna acción especial para invocar el protocolo SSL, basta con seguir un enlace o abrir una página cuya dirección empieza por *https://*. El navegador se encarga del resto.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

8.11.1.3 TLS

Transport Layer Security es un protocolo estandarizado por el IETF¹⁷. Está basado en SSL v3 (y es totalmente compatible) pero incorpora algunas mejoras y se destaca por no ser de una empresa privada.

8.11.1.4 SET

Secure Electronic Transaction es un protocolo definido por las empresas VISA, MasterCard, Microsoft, IBM, Netscape, Verisign, GTE y otras; exclusivamente para realizar comercio electrónico con tarjetas de crédito.

SET es un conjunto de protocolos, normas y especificaciones de seguridad, que constituyen una forma estándar para la realización de transacciones, reproduciendo en un entorno electrónico el pago con tarjeta de crédito física.

¹⁷ IETF: Internet Engineering Task Force. <http://www.ietf.org>

Además de poseer todas las características de SSL, el sistema autentifica los titulares de las tarjetas, los comerciantes y los bancos, garantiza la confidencialidad de la información de pago y asegura que los mensajes no sean manipulados.

La diferencia fundamental entre SSL y SET es que este último establece diferentes entidades (Cliente, Vendedor, Banco) y un protocolo de comunicaciones entre el Vendedor y el Banco. Cada una de estas entidades debe certificarse previo realizar cualquier transacción y cada mensaje queda firmado para evitar modificaciones y repudio posteriores.

Esta diferencia puede apreciarse cuando se piensa que SSL sólo protege un número de tarjeta de crédito, por ejemplo, cuando se envía del cliente al comerciante, sin embargo no hace nada para la validación de ese número, no chequea su autorización, permite que el comerciante lo almacene, etc. SET cubre todas estas debilidades ofreciendo seguridad a las entidades intervinientes.

La implantación del protocolo SET aporta una serie de beneficios:

- Autentifica los titulares de las tarjetas de crédito, los comerciantes y los bancos que intervienen en la operación. La autenticación asegura que los participantes en la operación comercial sean quienes dicen ser: el consumidor sabe en qué comercio está comprando y; el comercio está seguro de que quien está comprando es realmente el titular del instrumento de pago. La autenticación se realiza a través de certificados digitales que tanto el comerciante como el comprador poseen.
- Garantiza la máxima confidencialidad de la información del pago. Toda la información que viaja por la red, durante el intercambio de identidades y datos, está protegida contra cualquier intromisión o captura con métodos criptográficos.
- Asegura que los mensajes financieros no sean manipulados dentro del circuito del proceso de pago. La integridad y la autenticidad se basan en la generación de firmas digitales.

La utilización de un documento firmado con la clave privada del cliente y encriptada con la clave pública del receptor puede apreciarse en el Gráfico 8.7.

1. El Cliente Firma el documento de compra, mediante su Clave Privada.
2. El Cliente Encripta los datos, mediante la Clave Pública del Vendedor.
3. El Vendedor descifra, mediante su Clave Privada, los datos encriptados por el Cliente.
4. El Vendedor comprueba la integridad y autenticidad de los datos (Firma del Cliente), mediante la Clave Pública del mismo.

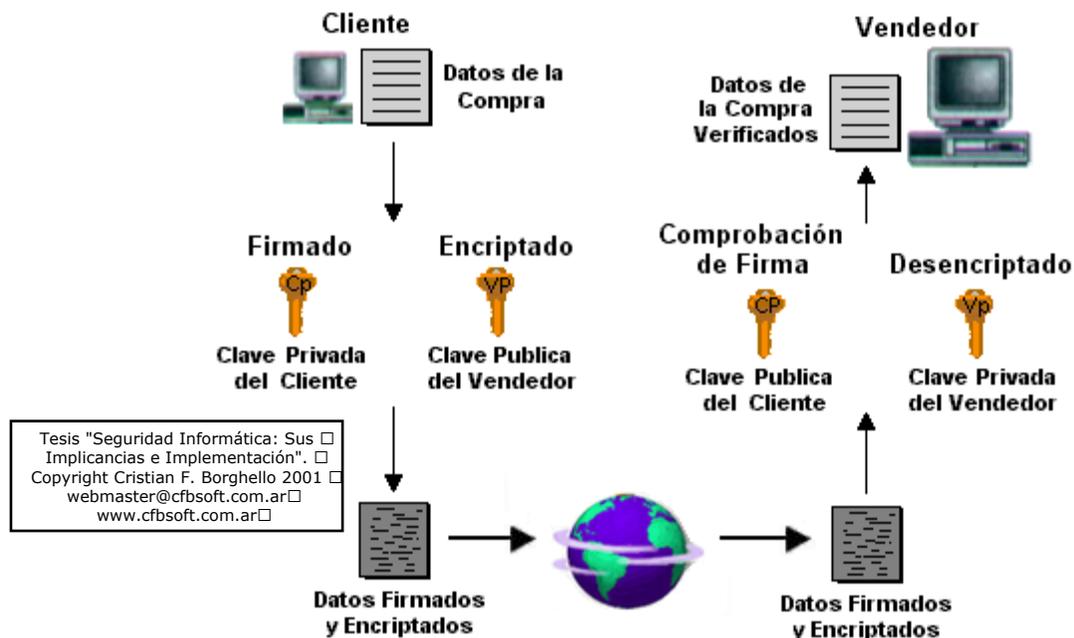


Gráfico 8.7 – Proceso Encriptado-Firmado de SET

SET utiliza algoritmos de encriptación como SHA-1, DES y RSA ya que estos son compatibles con los certificados existentes, aunque en próximas versiones se piensa dar soporte a algoritmos de curvas elípticas.

8.12 OTROS PROTOCOLOS DE SEGURIDAD

8.12.1 SSH

El protocolo Secure Shell fue desarrollado en 1995 por Tatu Ylonen para permitir un logueo seguro en terminales remotas, evitando el viaje de passwords en claro por redes inseguras; mediante el uso de comunicaciones cifradas. El protocolo SSH se establece en tres niveles:

1. **Nivel de Transporte:** En este nivel se procede a la autenticación del servidor, el establecimiento de un canal cifrado (confidencialidad), chequeo de integridad de los mensajes, y un identificador único de sesión. Típicamente esta conexión se realiza mediante TCP/IP.

En cuanto a los algoritmos empleados:

- a. Para el intercambio de claves: Diffie-Hellman.
- b. Algoritmos de clave pública para encriptación y autenticación del servidor: DSA, Certificados X.509 y Certificados PGP.
- c. Algoritmos de clave simétrica: 3DES, BlowFish e IDEA.
- d. Algoritmos de integridad: SHA1 y MD5.

Todos estos son utilizados con claves de 128 bits.

2. **Nivel de Autenticación del Usuario:** En este nivel se supone establecida la encriptación e integridad del canal y la autenticación del servidor. Para la autenticación del usuario el SSH ofrece varias posibilidades:
 - Autenticación del usuario basada en claves Pública–Privada: la autenticación del usuario se establece en base a la posesión de la clave privada. El servidor SSH conoce la clave pública del usuario. Este es el modo recomendado por los fabricantes que implementan SSH.
 - Autenticación del usuario basada en passwords. Hay que señalar que el password no viaja encriptado, sino el canal por el que va el password es el que se mantiene encriptado (el nivel de Transporte es un túnel). Es tarea del servidor la validación del password según su base de datos.
 - Autenticación del usuario basada en procedencia del Host: en esta situación hay que proteger las claves privadas del Host por parte del usuario. Es una autenticación parecida a la ofrecida por otros sistemas de logueo por lo que es completamente desaconsejable.
3. **Nivel de Conexión:** Es el protocolo encargado de multiplexar el “túnel encriptado” en varios canales lógicos, de forma de obtener múltiples sesiones para la ejecución de canales remotos.

Tesis "Seguridad Informática: Sus Implicancias e Implementación". □ Copyright Cristian F. Borghello 2001 □ webmaster@cfbsoft.com.ar □ www.cfbsoft.com.ar □
--

8.12.2 S/MIME

El protocolo MIME Seguro fue propuesto por la empresa RSA y después de su aparición fue propuesto como estándar por la IETF pero por problemas de derechos y restricciones de patentes no pudo ser posible.

S/MIME utiliza técnicas similares a PGP e incorpora certificados X.509. Aunque no cuente con el apoyo necesario para ser considerado un estándar, está implementado en muchos programas de correo electrónico. Tiene la ventaja sobre PGP, que al utilizar Autoridades de Certificación, es ideal para ser utilizado por empresas y para el comercio electrónico.

8.12.3 SOCKS

En sus orígenes este protocolo fue aprobado por el IETF como un estándar para la autenticación ante un Firewall. Actualmente, y combinado con SSL provee las bases para construir VPN altamente seguras.

Socks permite la conexión de equipos situados tras un Firewall. Inicialmente fue pensado para permitir el acceso desde una red interna a servicios disponibles en el exterior, sin embargo puede emplearse en sentido contrario, para el acceso desde el exterior de la organización (protegida con un Firewall).

La conexión es validada por el sistema de autenticación y después el servidor Socks actúa de intermediario con la aplicación situada en el servidor destino.

Socks actúa de “envoltura” sobre el protocolo UDP–TCP permitiendo que los equipos protegidos por el Firewall puedan conectarse a una red insegura, utilizando su propia dirección y devolviendo los resultados al cliente.

Debe notarse que Socks sólo autentifica las conexiones pero no produce ningún tipo de cifrado de los datos por lo que se hace necesario combinarlo con algún algoritmo que si lo haga (SSH, SSL, PPTP, IPsec, etc).

Tesis "Seguridad Informática: Sus Implicancias e Implementación". Copyright Cristian F. Borghello 2001 webmaster@cfbsoft.com.ar www.cfbsoft.com.ar
--

8.12.4 KERBEROS

En 1993 el MIT crea el proyecto Athena, y basándose en la mitología griega con su perro de tres cabezas y una cola de serpiente vigilando la entrada a Hades (el infierno), nace Kerberos.

Kerberos es un sistema de seguridad que provee autenticación a través de redes inseguras. Su objetivo es restringir los accesos sólo a usuarios autorizados y poder autenticar los requerimientos a servicios, asumiendo un entorno distribuido abierto, en el cual los usuarios en las estaciones de trabajo acceden a estos servicios a través de una red.

Los modelos de autenticación hasta ahora vistos son, principalmente, de dos tipos:

- **Recursos:** el usuario indica el recurso al que desea acceder mediante un cliente verificado.
- **Usuario:** El usuario se ve obligado a verificar su autenticidad cada cierto tiempo.

En estos sistemas se tiene una dificultad esencial: la password viaja en forma permanente por la red estando a merced de cualquier tipo de ataque que se desee realizar.

Kerberos fue creado para mitigar este problema de forma tal que el usuario necesita autorización para comunicarse con el servidor (y esta es confiable), se elimina la necesidad de demostrar el conocimiento de información privada y de que esta viaje a través de la red.

Kerberos provee un servidor de autenticación centralizado, cuya función es autenticar a los usuarios frente a servidores y a estos frente a los usuarios. La tecnología Kerberos está basado en tres objetos de seguridad (tres cabezas):

- **Autenticación Service (AS):** Autentifica los usuarios y les proporciona un ticket para realizar la comunicación con el servidor de Tickets.
- **Tickets Gratin Service (TGS):** Proporciona las credenciales necesarias para la comunicación con el servidor que proporciona los servicios.
- **Autenticador:** es un certificado testigo construido por el cliente o el servidor para probar las identidades y la actualidad de la comunicación; solo puede ser utilizado una vez.

Un Servidor KDC (Kerberos Distribution Center) alojado en el AS mantiene una base de datos de sus clientes (usuarios y servicios) y sus respectivas claves simétricas privadas utilizando DES (aunque actualmente se encuentra en desarrollo versiones de Kerberos empleando RSA):

- **La Clave Secreta del Usuario:** esta clave es conocida únicamente por el usuario y por Kerberos y tiene la finalidad de autenticar al usuario frente a Kerberos. El AS comparte una única clave secreta con cada servidor, las cuales fueron distribuídas físicamente o de otra de forma segura.

- **La Clave de Sesión:** clave secreta generada por Kerberos luego de verificar al usuario y expedida al mismo con el objetivo de autentificar el intercambio de un par de usuarios que definen una sesión. Esta clave tiene un “tiempo de vida” predeterminado y conocida únicamente por aquellos para los cuales fue generada.

Existen dos tipos de credenciales utilizadas por el modelo:

- **Ticket:** es un certificado testigo expedido a un cliente para solicitar los servicios de un servidor. Este Ticket contiene el ID del usuario y su dirección en la red y es encriptado usando la clave secreta compartida por el AS y el cliente, garantizando que este ha sido autenticado recientemente (el mismo tiene un período de validez).
- **Autenticador:** Es un testigo construido por el cliente y enviado al AS para probar su identidad. Sólo cuando el servidor descifra el Ticket, y verifica que el ID del usuario es auténtico, otorga el servicio requerido.

8.12.4.1 RESUMEN DE KERBEROS

En el Gráfico 8.7 puede apreciarse el funcionamiento de las distintas entidades intervinientes en Kerberos y su función:

- Solicitud de un Ticket de acceso.**
- Ticket + Clave de Sesión.**
- Solicitud de un Ticket de acceso al servicio.**
- Ticket + Clave de Sesión.**
- Solicitud de Servicio.**
- Autenticador del Servidor.**

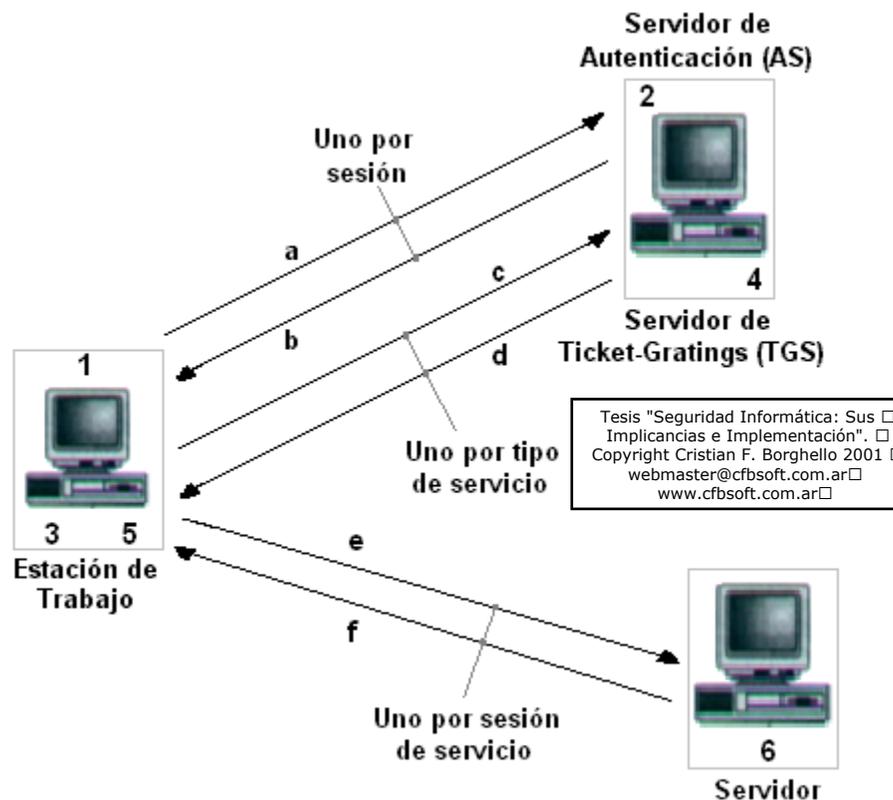


Gráfico 8.8 – Proceso de Kerberos

El proceso de Autenticación se divide en dos etapas:

- Autenticación de Usuario
 1. Un usuario desde una Estación de Trabajo requiere un servicio.
 2. AS verifica el correcto acceso del usuario a la Base de Datos, crea un Ticket y una Clave de Sesión. Los resultados son encriptados usando la clave derivada de la password del usuario.
- Autenticación de Servicio
 3. La Estación solicita la password al usuario y la utiliza para desencriptar el mensaje, luego envía al TGS el Ticket y el Autenticador que contienen el Nombre de Usuario, la Dirección de red y el Tiempo de Vida.
 4. El TGS desencripta el Ticket y el Autenticador, verifica la solicitud y crea un Ticket para ser enviado al Servidor.
 5. La Estación de Trabajo envía el Ticket y el Autenticador al Servidor.
 6. El Servidor verifica que el Ticket y el Autenticador coincidan, luego permite al Servicio.

8.12.4.2 PROBLEMAS DE KERBEROS

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

La filosofía de Kerberos está basado en una fuerte centralización del sistema, ya que para su correcto funcionamiento se debe disponer de forma permanente del servidor, de forma que si este falla toda la red se vuelve inutilizable por no disponer de forma para desencriptar los mensajes que circulan por ella. Este concepto es una contradicción a la teoría de sistemas distribuidos, sobre el que se basa el modelo que rige cualquier red (si una máquina falla el resto puede seguir su funcionamiento, sino a pleno, al menos correctamente).

Otra falencia es que casi toda la seguridad reside en el servidor que mantiene la base de datos de claves, por lo que si este se ve comprometido, toda la red estará amenazada.

Por último, la implementación de Kerberos, actualmente, acarrea algunos inconvenientes ya que se debe realizar un proceso de “Kerberización” sobre cada programa que se desee utilizar, suponiendo esto un conocimiento y tiempo considerable no siempre disponible. Si bien este inconveniente está siendo subsanado en diversas versiones aún no se cuenta con la standardización suficiente para su extensión masiva.

8.13 VPN—REDES PRIVADAS VIRTUALES

La tecnología de VPN proporciona un medio para usar el canal público de Internet como un canal apropiado para comunicar los datos privados. Con la tecnología de encriptación y encapsulamiento, una VPN básica, crea un pasillo privado a través de una red insegura. Es decir que la red pública sólo proporciona la infraestructura para enviar los datos.

El objetivo fundamental de una VPN es proteger los datos durante la transmisión a través de la red, permitiendo el uso de redes públicas como si fueran privadas (virtualmente privadas). Esta protección previene el mal uso, modificación, uso no autorizado e interrupciones de acceso a la información mientras atraviesa los distintos segmentos de la red (o redes).

Una VPN no protege la información mientras está alojada en el origen, o cuando llega y se aloja en su destino. También puede dejar expuesto los datos durante alguno de los procesos de encriptación en la red (redes internas antes de la encriptación o redes externas después de la desencriptación). Una VPN solo protege los aspectos de protección en la comunicación, no protege la información alojada en el disco, en pantalla, o impresas.

8.13.1 REQUERIMIENTOS DE UNA VPN

- **Escalabilidad:** Esto significa poder decidir cuanta información puede manejarse al mismo tiempo, y efectivamente poder hacerlo.
- **Performance:** Este uno de los puntos críticos, la VPN debe estar preparada para manejar una gran cantidad de tráfico si es que va a trabajar en un ambiente corporativo.
- **Disponibilidad:** Las soluciones VPN están siendo adoptadas estratégicamente por las organizaciones para proveer accesos externos y eliminar altos costos de conectividad, por lo que su disponibilidad debe estar asegurada en todo momento.
- **Transparencia:** La VPN necesita ser fácil de usar y entender para el usuario, que lo utilizará sin saber como exactamente trabaja, una vez que han sido definidos los “túneles” de protección de la información. Una buena política de distribución debe permitir a la VPN determinar cuando encriptar y cuando enviar texto claro, pidiéndole al usuario únicamente su autenticación para proveer el acceso.
- **Fácil de administrar:** una VPN que se instale en una mediana o gran empresa debe ser fácil de administrar, como todo producto de seguridad, donde la administración y el control centralizado de la solución es clave. El módulo de control debe tener una simple vía para diseñar la política de seguridad, y una fácil distribución de esa política en todos los puntos de la empresa.
- **Interoperatividad:** Una completa VPN debe poder comunicarse con otros productos VPN.
- **Encriptación:** La solución VPN debería ofrecer distintos tipos de encriptación, que se utilizarán de acuerdo a las necesidades de cada segmento de la red. El estándar actual para la encriptación comercial es DES o 3DES, pero existen otras alternativas como BlowFish o CAST (168 bit).
- **Seguridad:** Uno de los requerimientos más importantes antes de implementar la VPN, es contar con políticas y procedimientos de seguridad definidos. La red virtual sólo resuelve un problema específico, y su configuración debe estar basada en una política que haya contemplado el análisis del riesgo que debemos atacar con la instalación de esta herramienta. Esto hace que sea atractivo combinar la flexibilidad de los protocolos VPN con la seguridad proporcionada por IPSEC.

8.13.2 L2TP

Layer To Tunneling Protocol es un protocolo estándares del IETF que ha sido ampliamente implementado. L2TP encapsula las tramas del protocolo punto a punto (PPP Point to Point Protocol) que van a enviarse a través de redes.

Cuando está configurado para utilizar IP como su transporte, L2TP se puede utilizar como protocolo de túnel VPN en Internet. Las tramas PPP encapsuladas se pueden cifrar o comprimir. Cuando los túneles L2TP aparecen como paquetes IP, aprovechan la seguridad IPsec estándar para obtener una fuerte protección de integridad, reproducción, autenticidad y privacidad. L2TP se diseñó específicamente para conexiones Cliente–Servidor de acceso a redes y para conexiones Gateway a Gateway

8.13.3 PPTP

Point to Point Tunneling Protocol (antecesor de L2TP) fue diseñado para proporcionar comunicaciones autenticadas y cifradas entre un cliente y un Gateway o entre dos Gateways (sin necesitar una infraestructura de clave pública) utilizando un ID de usuario y una contraseña. Apareció por primera vez en 1996, dos años antes de la disponibilidad de IPsec y L2TP y su objetivo era la simplicidad en su diseño, la compatibilidad multiprotocolo y la capacidad de cruzar una amplia gama de redes IP.

8.13.4 IPSEC

Tesis "Seguridad Informática: Sus
Implicancias e Implementación". □
Copyright Cristian F. Borghello 2001 □
webmaster@cfbsoft.com.ar □
www.cfbsoft.com.ar □

El IETF ha desarrollado, en principios de 1995, un conjunto de estándares para la seguridad del protocolo IP conocida como IPsec. Este estándar es válido para IPv4 y IPv6, y provee un marco que permite a dos o más partes el uso de distintos algoritmos de encriptación y métodos de autenticación en una misma sesión de comunicación. Esta flexibilidad permite incorporar esta tecnología para integrar distintos participantes a bajo costo, sin necesidad de dispositivos adicionales.

Por primera vez el protocolo IP (capa de red y superiores) se modifica para proporcionar seguridad. IPsec proporciona autenticación de origen, comprobación de integridad y, opcionalmente, confidencialidad de contenido.

El equipo emisor protege los datos antes de la transmisión y el equipo receptor los descodifica una vez que los ha recibido. IPsec se basa en claves criptográficas (independientes de los algoritmos utilizados) y se puede utilizar para proteger equipos, sitios, dominios, comunicaciones de aplicaciones, usuarios de acceso telefónico. Como parte de un completo plan de seguridad que utiliza controles rigurosos y seguridad periférica, IPsec asegura la protección de los datos que transmite.

IPsec elimina el requisito de la implementación de seguridad en la aplicación bajando la seguridad al nivel de red. Esto permite a las aplicaciones permanecer independientes de la infraestructura de seguridad subyacente. Los datagramas IP se protegen sin tener en cuenta la aplicación que inicialmente generó el tráfico.

En otras palabras, las aplicaciones no son compatibles con IPsec. Las reglas de seguridad las define el administrador sin tener en cuenta qué aplicación se ejecuta; IPsec es transparente para las aplicaciones.

IPsec define una familia de protocolos diseñados para ser empleados con los datagramas IP:

- Authentication Header (AH)–Protocolo IP 51: utilizado para garantizar la integridad de los datos, proporciona protección antirreproducción y protege la autenticación del Host. AH provee autenticación, integridad y protección a la replica (una transacción sólo debe llevarse a cabo una vez) asegurando partes de la cabecera IP del paquete.
- Encapsulating Security Payload (ESP)– Protocolo IP 50: incluye las características de AH y agrega, opcionalmente, la confidencialidad de los datos asegurando los paquetes IP que siguen a la cabecera

La aplicación de estos dos protocolos puede verse en el siguiente gráfico:

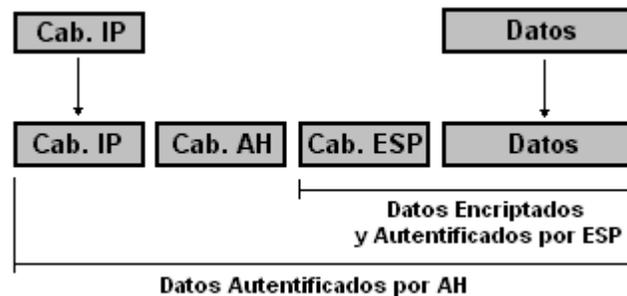


Gráfico 8.9 – Fuente: MONSERRAT COLL, Francisco Jesús. Seguridad en los protocolos TCP/IP. Página 30.
<http://www.rediris.es/ftp>

Es importante tener en cuenta que ni AH ni ESP proporcionan los algoritmos criptográficos reales para implementar las características especificadas anteriormente, solo aprovechan los algoritmos criptográficos y de autenticación existentes.

Los servicios proporcionados por IPsec pueden aplicarse en dos modos a los datagramas IP:

- **Modo Normal:** empleado para realizar comunicaciones entre equipos finales (punto a punto). En este caso toda la comunicación es encriptada y los equipos intermedios no pueden desencriptar el contenido de los datagramas. Este modo permite la total confidencialidad de la comunicación.
- **Modo Túnel:** los datagramas son enviados en claro hacia equipos intermedios (Router o Firewall), este encripta los datos y los envía al exterior. En el otro extremo del túnel se realiza el proceso de desencriptado y se envía el datagrama en claro hacia el equipo destino. Esto permite a los equipos de la red interna visualizar los datos y sólo encriptar los que salen al exterior.

8.14 INVERSIÓN

Tesis "Seguridad Informática: Sus
Implicancias e Implementación".
Copyright Cristian F. Borghello 2001
webmaster@cfbsoft.com.ar
www.cfbsoft.com.ar

Los costos de las diferentes herramientas de protección se están haciendo accesibles, en general, incluso para las organizaciones más pequeñas. Esto hace que la implementación de mecanismos de seguridad se dé prácticamente en todos los niveles: empresas grandes, medianas, chicas y usuarios finales. Todos pueden acceder a las herramientas que necesitan y los costos (la inversión que cada uno debe realizar) va de acuerdo con el tamaño y potencialidades de la herramienta .

Pero no es sólo una cuestión de costos, los constantes cambios de la tecnología hacen que para mantener un nivel parejo de seguridad, se deba actualizar permanentemente las herramientas con las que se cuenta. Como los intrusos mejoran sus armas y metodologías de penetración de forma incesante, el recambio y la revisión constantes en los mecanismos de seguridad se convierten en imprescindibles. Y éste es un verdadero punto crítico.

Según Testers, “esto es tan importante como el tipo de elementos que se usen”. Sin duda, éstos deben ser las que mejor se adapten al tipo de organización. Pero tan importante como eso es el hecho de conocer exactamente cómo funcionan y qué se puede hacer con ellos. “Es prioritario saber los riesgos que una nueva tecnología trae aparejados”.

PROTECCIÓN.....	1
8.1 VULNERAR PARA PROTEGER	2
8.1.1 ADMINISTRACIÓN DE LA SEGURIDAD.....	2
8.1.2 PENETRATION TEST, ETHICAL HACKING O PRUEBA DE VULNERABILIDAD.....	3
8.1.3 HONEYPOTS–HONEYNETS	5
8.2 FIREWALLS.....	5
8.2.1 ROUTERS Y BRIDGES	6
8.2.2 TIPOS DE FIREWALL.....	7
8.2.2.1 Filtrado de Paquetes.....	7
8.2.2.2 Proxy–Gateways de Aplicaciones.....	7
8.2.2.3 Dual–Homed Host	8
8.2.2.4 Screened Host	8
8.2.2.5 Screened Subnet.....	9
8.2.2.6 Inspección de Paquetes.....	10
8.2.2.7 Firewalls Personales	10
8.2.3 POLÍTICAS DE DISEÑO DE FIREWALLS.....	10
8.2.4 RESTRICCIONES EN EL FIREWALL	11
8.2.5 BENEFICIOS DE UN FIREWALL.....	12
8.2.6 LIMITACIONES DE UN FIREWALL.....	12
8.3 ACCESS CONTROL LISTS (ACL).....	13
8.4 WRAPPERS	13
8.5 DETECCIÓN DE INTRUSOS EN TIEMPO REAL	14
8.5.1 INTRUSIÓN DETECTION SYSTEMS (IDS)	14
8.5.1.1 Características de IDS.....	15
8.5.1.2 Fortalezas de IDS	16
8.5.1.3 Debilidades de IDS.....	16
8.5.1.4 Inconvenientes de IDS	17
8.6 CALL BACK	17

8.7 SISTEMAS ANTI-SNIFFERS.....	17
8.8 GESTION DE CLAVES “SEGURAS”.....	17
8.8.1 NORMAS DE ELECCIÓN DE CLAVES.....	18
8.8.2 NORMAS PARA PROTEGER UNA CLAVE	19
8.8.3 CONTRASEÑAS DE UN SÓLO USO	20
8.9 SEGURIDAD EN PROTOCOLOS Y SERVICIOS.....	20
8.9.1 NETBIOS.....	21
8.9.2 ICMP.....	21
8.9.3 FINGER	21
8.9.4 POP	21
8.9.5 NNTP.....	22
8.9.6 NTP	22
8.9.7 TFTP	23
8.9.8 FTP	23
8.9.8.1 FTP Anónimo.....	23
8.9.8.2 FTP Invitado.....	23
8.9.9 TELNET	24
8.9.10 SMTP.....	24
8.9.11 SERVIDORES WWW	24
8.10 CRIPTOLOGÍA.....	26
8.10.1 HISTORIA	26
8.10.2 CRIPTOGRAFÍA.....	26
8.10.3 CRIPTOANÁLISIS	27
8.10.4 CRIPTOSISTEMA	27
8.10.4.1 Transposición	28
8.10.4.2 Cifrados Monoalfabéticos	29
8.10.5 ALGORITMOS SIMÉTRICOS MODERNOS (LLAVE PRIVADA).....	29
8.10.5.1 Redes de Feistel.....	29
8.10.5.2 DES.....	30
8.10.5.3 IDEA	31
8.10.5.4 BlowFish.....	31

8.10.5.5 RC5	31
8.10.5.6 CAST.....	31
8.10.5.7 Rijndael (el nuevo estándar AES).....	32
8.10.5.8 Criptoanálisis de Algoritmos Simétricos	32
8.10.6 ALGORITMOS ASIMÉTRICOS (LLAVE PRIVADA–PÚBLICA).....	32
8.10.6.1 RSA	33
8.10.6.2 Curvas Elípticas (CEE)	34
8.10.7 AUTENTIFICACIÓN	34
8.10.7.1 Firma Digital.....	35
8.10.8 PGP (PRETTY GOOD PRIVACY)	36
8.10.8.1 Funcionamiento de PGP.....	36
8.10.9 ESTEGANOGRAFÍA	37
8.11 COMERCIO ELECTRÓNICO.....	37
8.11.1 DINERO ELECTRÓNICO	38
8.11.1.1 Certificados X.509	38
8.11.1.2 SSL	39
8.11.1.3 TLS.....	41
8.11.1.4 SET.....	41
8.12 OTROS PROTOCOLOS DE SEGURIDAD.....	43
8.12.1 SSH	43
8.12.2 S/MIME	44
8.12.3 SOCKS.....	44
8.12.4 KERBEROS	45
8.12.4.1 Resumen de Kerberos	46
8.12.4.2 Problemas de Kerberos	47
8.13 VPN–REDES PRIVADAS VIRTUALES.....	47
8.13.1 REQUERIMIENTOS DE UNA VPN.....	48
8.13.2 L2TP	48
8.13.3 PPTP	49
8.13.4 IPSEC	49
8.14 INVERSIÓN.....	50

