IP Spoofing

Alberto Fernández López (lume@lume.infonegocio.com)
Abel Crespo Vázquez (abelcv@wanadoo.es)
Escuela Superior de Ingeniería Informática de Ourense (Universidad de Vigo)
Última revisión del documento: 1 de Junio de 2002.

Índice del documento

Breve historia del IP Spoofing

En Abril de 1989 un artículo titulado *"Security Problems in the TCP/IP Protocol Suite"* de S.M. Bellovin de los laboratorios de AT&T Bell, fue el primero en identificar el IP Spoofing como un riesgo real a las redes de computadoras. Bellovin describe cómo Robert Morris, creador del infame *Internet Worm*, dedujo la forma en que TCP crea los números de secuencia y falsificó una secuencia de paquetes TCP. Este paquete TCP incluía la dirección destino de su víctima y usando IP spoofing Morris fue capaz de obtener acceso a root del sistema atacado sin un ID de usuario o contraseña.

Un concepto erróneo común es ese “IP spoofing” que puede usarse para esconder la IP mientras navega por las Web, charla en el IRC, envía correo electrónico, etc. Esto generalmente no es verdad. Falsificando la IP de origen, las contestaciones van a ser dirigidas erradamente, con lo cual no se puede crear una conexión normal. Sin embargo, el IP spoofing es una parte íntegra de muchos ataques en la red que no necesitan ver las contestaciones (blind spoofing)

Introducción al IP Spoofing

El *Internet Protocol* (IP) (RFC791) mantiene dos y sólo dos funciones. Define un datagrama que puede ser enviado a través de Internet, y proporciona unos medios para fragmentar datagramas en paquetes y reensamblar paquetes en el datagrama original.

Cita del RFC791: *"El Internet Protocol está específicamente limitado en alcance a proporcionar las funciones necesarias para entregar un paquete de bits (un datagrama de Internet) de una fuente a un destino sobre un sistema de redes interconectadas. No hay ningún mecanismo para aumentar la fiabilidad de datos end-to-end, control de flujo, secuenciamiento, u otros servicios comúnmente encontrados en los protocolos host-a-host. El protocolo de Internet puede sacar provecho de los servicios de sus redes para proporcionar varios tipos y calidades de servicio."*

Descripción de un Datagrama IP

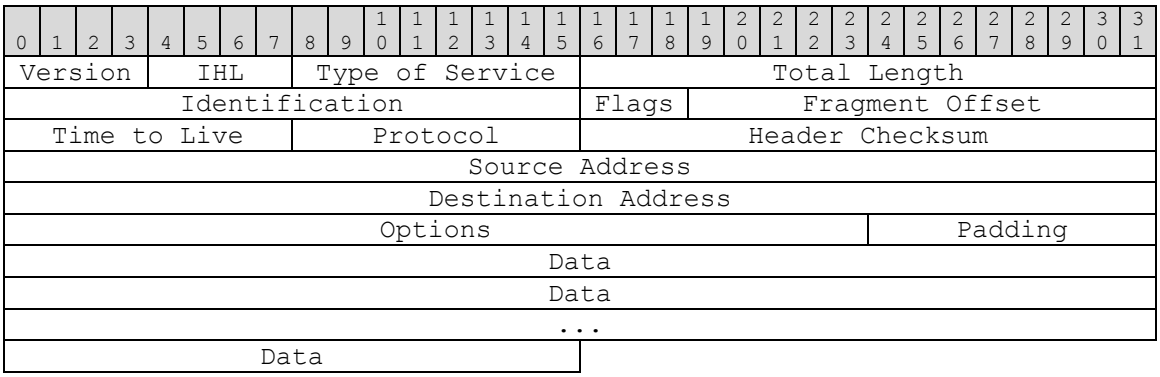


Figura 1: Descripción de un datagrama IP (RFC791)

Nótese que en la 4 línea de la descripción se requiere la *dirección de origen* del datagrama. En la forma más simple de una falsificación de dirección IP, el atacante sólo necesita crear un paquete que contiene una dirección de origen falsa y enviarlo a Internet escribiéndolo en el dispositivo utilizado para la comunicación con Internet. Para el atacante no experto, hay una herramienta llamada *iptest*, parte del paquete de seguridad libre y públicamente disponible *ipfilter* que automáticamente crea paquetes con el propósito de probar configuraciones de routers y otros elementos de seguridad.

La infraestructura de Internet consiste principalmente en un conjunto de computadoras de entrada (*gateways*) y enrutadores de paquetes (*routers*). Estos sistemas tienen múltiples interfaces hardware. Mantienen tablas de enrutamiento para decidir por qué interfaz de salida enviar un paquete basándose en la interfaz de entrada por el que llegó y la dirección IP destino que en él se especifica. Cuando un paquete falsificado llega a un elemento de la infraestructura, éste dirigirá el paquete fielmente hacia la dirección de destino, exactamente como lo habría hecho con un paquete legítimo.

Cómo utilizar el IP Spoofing

El IP spoofing es un método de engaño, y se puede usar de la misma manera que muchos otros métodos de engaño. Veamos algunas denominaciones de técnicas que utilizan el IP spoofing:

- *"Man in the middle".* Se olfatean paquetes en un eslabón entre dos puntos.

- *"Routing redirection"*. Se redirige la información enrutada desde el host original al host del atacante.

- *"Source routing"*. Se remiten paquetes individuales desde el host del atacante.

- *"Blind spoofing"*. Se predicen contestaciones de un servidor y se envían órdenes, aunque no se consiguen respuestas.

- *"Flooding"*. Se envía gran cantidad de paquetes intentando causar una denegación de servicios.

Obviamente las posibilidades que ofrece el IP spoofing son muchas. Si a estas técnicas añadimos la ingeniería social el IP spoofing se convierte en una poderosa herramienta para llevar a cabo multitud de ataques y propósitos no demasiado lícitos. He aquí algunas ideas rápidas sobre como el IP spoofing podría usarse:

- El IP spoofing se dirige usualmente a ocultar la identidad de un atacante, sobre todo cuando la denegación de servicios (DoS) es la meta del ataque.

- El IP spoofing se usa para simular que una computadora es otra como forma de convencer a la víctima, por ejemplo, que un ataque llega desde una Universidad, cuando de hecho llega de un enemigo.

- El IP spoofing puede usarse para crear la impresión que un sitio particular está actuando malévolamente con la intención de crear fricciones o llevar a que un defensor acuse falsamente a una tercera parte inocente.

- El IP spoofing puede usarse para apoyar otra actividad que desarrolle el atacante, y necesite ganar la confianza del atacado. Por ejemplo, un vendedor de productos de seguridad de información podría hacer ataques con dirección IP falsa para convencer a un cliente de la necesidad de sus servicios.

- El IP spoofing puede usarse para convencer a la víctima de que muchos sitios están participando en un ataque cuando sólo un número pequeño de individuos es responsable, de forma que la víctima considera imposible el localizar a los atacantes.

- El IP spoofing puede usarse para observar cómo una víctima reacciona y para determinar qué respuestas es posible obtener ante un determinado ataque.

Otra manera de ver este problema se refiere al efecto neto sobre la información en los sistemas de información.

- Corrupción de información: Las direcciones de IP se usan a menudo en Internet como base para la toma de decisiones. Por ejemplo, para las actualizaciones de DNS se designa a menudo que sólo pueden venir de unos servidores específicos. Con direcciones IP falsificadas el sistema DNS entero podría adulterarse, causando que todos los paquetes serían redirigidos a través de los servidores enemigos.

- Denegación de servicios: Internet es básicamente una red frágil que depende de la conducta apropiada y buena fe de los participantes para su apropiado funcionamiento. Sin grandes cambios en la forma en que Internet trabaja, la denegación de servicios es casi imposible prevenir. Por ejemplo, el mismo ataque de DNS podría usarse para causar el rechazo extendido de servicios, o quizá para modificar los mecanismos de entrega de paquetes de forma que la información viaje en círculos a través de la columna vertebral de Internet.

Éstos son sólo algunos de los ejemplos de lo que se puede hacer con IP spoofing. Sin demasiado esfuerzo, podrían crearse muchos otros ejemplos.

Prevención del IP Spoofing

Con la actual tecnología IP es imposible eliminar los paquetes falsificados, pero sí podremos evitar que circulen. Aunque para esta solución hay que tener en cuenta que individualmente poco podemos hacer para conseguirlo, sin embargo, como comunidad sería perfectamente factible.

En lugar de que todos los elementos de la infraestructura redirijan todos y cada uno de los paquetes que le llegan, cada elemento de la infraestructura debería tener una simple regla: Dirigir sólo paquetes que podrían venir legítimamente de la interfaz por la que el paquete ha llegado.

Esto puede parecer complicado, pero realmente no es. De hecho, la tecnología para hacer esto ya existe, y siempre ha existido: Es lo que conocemos como filtrado de paquetes.

Virtualmente cada router y gateway que hoy existe permite el filtrado de paquetes basado en su interfaz de entrada, dirección de origen y dirección destino. Éste es un componente necesario para su funcionamiento y es la base para la forma en que se dirigen todos los paquetes.

El único cambio que debería ser hecho es en estos routers y gateways, y sería forzar a las estructuras de red a que estén legítimamente en su lugar. En otras palabras, los routers y gateways deben negarse a dirigir paquetes ridículos. Aquí están algunos de los ejemplos más simples de paquetes malintencionados conocidos:

- La dirección IP 127.0.0.1 únicamente se usa para enrutamiento interno de paquetes desde un host a si mismo. No hay ningún datagrama legítimo que deba atravesar un router o gateway con esta dirección de origen. De hecho, enrutar estos paquetes es peligroso puesto que se podrían utilizar para falsificar paquetes del *localhost* que a menudo tienen privilegios especiales. Un reciente ataque que causa denegación de servicios se realiza enviando un paquete al puerto de eco de un host poniendo como dirección origen 127.0.0.1 y el puerto de origen como el propio puerto de eco. La función del puerto de eco es devolver cualquier paquete que llegue a su origen. Si el paquete proviene del propio host, y de su propio puerto de eco, se crea un bucle infinito que, en muchos casos, termina desactivando la computadora.

- La dirección IP 0.0.0.0 no es legítima. De hecho, no existe ninguna dirección IP legítima que deba cruzar gateways conteniendo un 0 en alguno de los elementos de la dirección (son las conocidas como direcciones IP de broadcast) Desgraciadamente, muchos routers utilizan los .0. en sus tablas de enrutamiento como convención para indicar alguna dirección de 0 a 255 (el rango entero).

- La especificación de IP incluye direcciones reservadas para redes privadas, diseñadas únicamente para uso interno. No hay ninguna razón legítima para enrutar paquetes con estas direcciones de origen (RFC1597) Estos rangos de dirección incluyen 10. *. *. *, 172.16-32. *. *, y 192.168. *. * (donde * indica algún valor de 0 a 255). Ningún paquete debe ser enrutado a través de Internet con estas direcciones como su fuente o destino.

El próximo paso para eliminar la falsificación IP se dirige a imponer estándares en los routers, gateways y cada elemento de la infraestructura.

Generalmente, Internet se divide en servidores de su columna vertebral, que proporcionan servicios de transporte de paquetes de área ancha, redes privadas propiedad de compañías, instituciones, agencias gubernamentales, y proveedores de servicios de Internet (ISP) que proporcionan conexiones entre elementos de la columna vertebral y las redes privadas.

- Redes Privadas: Cada red privada debe:
  - Impedir la entrada o salida de la organización de todos los paquetes malintencionados conocidos.
  - Prevenir la entrada de paquetes con direcciones de origen internas a la red.
  - Prevenir la salida de paquetes con direcciones de origen externas.
  - Prevenir la entrada de paquetes con direcciones de destino externas.
  - Prevenir la salida de paquetes con direcciones de destino internas.

- ISPs: Cada ISP debe:
  - Impedir que todos los paquetes malintencionados conocidos entren o salgan de su infraestructura.
  - Prevenir que cualquier paquete entrante de cualquiera de sus clientes con una dirección origen que no pertenece al rango de direcciones asignadas a ese cliente salga de la red del cliente.
  - Prevenir que cualquier paquete con una dirección destino que no esté en el rango de direcciones de su cliente entre en la red de su cliente.
  - Prevenir que cualquier paquete con una dirección IP no legítima de su ISP salga de su red.
  - Prevenir que cualquier paquete originado fuera de su red y no destinado para sus direcciones IP legítimas entre en su red.
  - Prevenir el tráfico entrante del cliente con la dirección origen del cliente.
  - Prevenir el tráfico saliente del cliente con la dirección destino del cliente.

- Redes de la columna vertebral: Cada proveedor debe:
  - Impedir que todos los paquetes malintencionados conocidos entren o salgan de su infraestructura.
  - Prevenir la entrada de paquetes originados en un ISP con dirección de origen que no pertenezca legítimamente a ese ISP.
  - Prevenir la entrada en la red de un ISP cualquier paquete no destinado a una dirección legítima de ese ISP.
  - Prevenir la entrada en su red de paquetes mal enrutados desde cualquier otro proveedor de la columna vertebral.
  - Impedir el enrutamiento de cualquier paquete a cualquier otro servidor de la columna vertebral a no ser que ellos puedan enrutar legítimamente ese paquete para que alcance su destino.

Para los servidores de la columna vertebral, esto requiere algún esfuerzo, sin embargo el alto volumen de información que mueven justifica un poco el esfuerzo en la protección.

¿Qué ocurre si...?

- ¿Qué ocurre si una red privada ignora las reglas? Sería de esperar que muchas redes privadas ignorarán tal regla, sea por ignorancia, a propio intento o por desatención. Pero aún cuando todas las redes privadas ignoren todas las reglas, las reglas para ISPs impedirán que una IP falsificada circule por la infraestructura global.

- ¿Qué ocurre si un ISP ignora las reglas? Si un ISP ignora las reglas y permite la falsificación de IP, los servidores de la columna vertebral pueden proteger al resto de Internet. Es decir, los clientes del ISP en cuestión, estarían expuestos a ataques por parte de otros clientes de ese ISP, pero no se podrían extender más allá de las redes de ese ISP.

- ¿Qué ocurre si un servidor de la columna vertebral ignora las reglas? Si todos estos servidores ignoran las reglas, a menos que los ISPs y las redes privadas sigan las reglas, continuarán habiendo falsificaciones de IP.

- ¿Qué ocurre si hay varias combinaciones que ignoran las reglas? Dependiendo de las combinaciones específicas, tendremos más o menos falsificaciones de IP. Cuantas más redes privadas, ISPs y servidores de la columna vertebral sigan las reglas, más difícil lo tendrá el atacante para falsificar su IP.

Otras objeciones

- Muchos ISPs y servidores de la columna vertebral no quieren tomar la responsabilidad sobre el contenido de la información que circula en Internet. Simplemente como una compañía de teléfono, no quieren asumir un rol de inspección del contenido de la información, sólo quieren ser un servicio de entrega.

- El coste de aplicar las reglas es despreciable. Si la aplicación de las reglas se tomase como una parte normal de la instalación y del proceso de mantenimiento, llevaría solamente unos minutos de esfuerzo.

- Hay un gran número de personas que quieren una falta total de restricciones en la información que fluye a través de Internet. Particularmente, estamos de acuerdo con el principio de flujo de información libre, excepto en casos dónde la libertad de una persona influye en la libertad de otros.

Ataques que se apoyan en IP Spoofing

En la sección anterior hemos visto qué tipos de ataques se pueden llevar a cabo con IP spoofing. Aquí nos adentraremos un poco más profundamente ya no en técnicas en general sino en ataques particulares que han sido y siguen siendo los más conocidos y utilizados, y posiblemente también los más peligrosos.

Ataques por denegación de servicios (DoS)

Los ataques por denegación de servicios se dirigen principalmente a causar la "caída" de una máquina de Internet, bien sea, colapsando su red de información, agotando sus recursos de cara a la red o explotando fallos que causen el "cuelgue" de la máquina o su reinicio.

Veremos entonces varios de los ataques DoS más conocidos y utilizados en los últimos años que se apoyan en mayor o menor medida en el IP spoofing.

TCP Syn Flooding

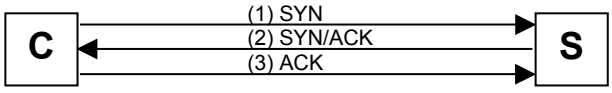
Existe un código fuente circulando por la red que permite realizar ataques por denegación de servicios creando conexiones TCP "parcialmente abiertas". Este código está siendo utilizado para atacar sitios conectados a Internet. En la actualidad no hay todavía una completa solución para este problema.

Cualquier sistema conectado a Internet y proporcionando servicios de red basados en TCP (como un servidor Web, servidor FTP o un servidor de correo) está potencialmente sujeto a este ataque. Note que además los ataques lanzados a hosts específicos pueden ser lanzados también a sus routers y otros sistemas de servicios de red si éstos tienen activos otros servicios TCP (por ejemplo, el eco) Las consecuencias del ataque pueden variar en función del sistema atacado.

Descripción

Cuando un sistema (cliente) intenta establecer una conexión TCP a un sistema que proporciona un servicio (servidor), el cliente y el servidor intercambian una sucesión fija de mensajes. Esta técnica de conexión se aplica en todas las conexiones TCP, por ejemplo telnet, Web, correo electrónico, etc.

El sistema cliente comienza enviando un mensaje SYN al servidor. El servidor reconoce el mensaje SYN enviando un mensaje SYN/ACK al cliente. El cliente finaliza el establecimiento de conexión respondiendo con un mensaje ACK. La conexión entre el cliente y el servidor está entonces abierta, y los datos específicos del servicio pueden ser intercambiados entre el cliente y el servidor.

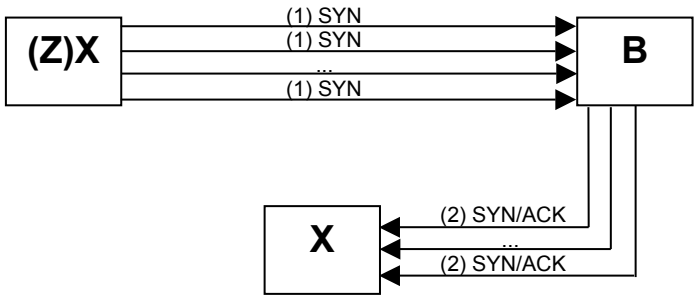


(4) El cliente C y el servidor S pueden ahora intercambiar datos.

Figura 2: Saludo de tres direcciones.

La posibilidad del ataque se encuentra en el punto dónde el sistema servidor ha enviado el mensaje SYN/ACK al cliente pero aún no ha recibido el mensaje ACK. Esto es lo que se conoce como conexión “parcialmente abierta”. El servidor ha construido en su memoria una estructura de datos en las que se describen todas las conexiones pendientes. Esta estructura de datos es de tamaño finito, y puede ser desbordada creando demasiadas conexiones parcialmente abiertas intencionalmente.

Crear conexiones parcialmente abiertas es fácil apoyándose en el IP spoofing. El sistema atacante envía mensajes SYN al sistema servidor de la víctima, éstos parecen ser legítimos pero de hecho el atacante ha creado los paquetes con una dirección IP de origen de un sistema inalcanzable, con lo cual el supuesto sistema cliente es incapaz de responder a los mensajes SYN/ACK que devuelve el servidor. Esto significa que el último mensaje de ACK nunca se enviará al sistema servidor de la victima.



En (1) el host atacante envía un gran número de peticiones SYN al objetivo. (Z)X es el host Z “disfrazado” como X.
En (2) el host objetivo responde con SYN/ACKs a lo que él cree es el origen de los SYNs que le llegan.

Figura 3: Esquema de un ataque TCP Syn Flooding

La estructura de datos de conexiones parcialmente abiertas se llenará en el futuro, entonces el sistema servidor será incapaz de aceptar cualquier nueva conexión entrante hasta que la tabla se vacíe. Normalmente existe un timeout asociado con cada conexión pendiente, para que las conexiones parcialmente abiertas expiren en el futuro y el servidor víctima se recupere. Sin embargo, el sistema atacante puede continuar enviando paquetes falsificados que piden simplemente nuevas conexiones mucho más rápido de lo que el sistema de la víctima tarda en expirar las conexiones pendientes.

En la mayoría de los casos, la víctima de tal ataque tendrá dificultad para aceptar alguna nueva conexión entrante. En estos casos, el ataque no afecta a las conexiones entrantes existentes ni a la posibilidad de originar conexiones salientes.
Sin embargo, en algunos casos, el sistema puede agotar la memoria, colgarse o quedar inoperativo.

La situación del sistema atacante se disimula porque la fuente de los paquetes SYN es a menudo inverosímil. Cuando el paquete llega al servidor de la víctima, no hay manera de determinar su verdadera fuente. Esto ocurre porque la red transmite paquetes basándose en la dirección de destino. La única manera de validar la fuente de un paquete sería utilizando el filtrado entrante de paquetes. (Vea la sección *Cómo evitar el IP Spoofing*)

Impacto

Sistemas que proporcionan servicios TCP a la comunidad Internet pueden ser incapaces de proporcionarlos mientras están siendo atacados y durante algún tiempo después del cese del ataque. El propio servicio no se daña por el ataque, sólo la posibilidad de ofrecerlo. En algunos casos, el sistema puede agotar su memoria, colgarse o quedar inoperativo.

Solución

Con la actual tecnología IP no hay ninguna solución aceptable para este problema. Sólo el correcto filtrado de paquetes podría ayudar a mitigar el problema. Vea la sección *Prevención del IP Spoofing* para más detalles.

Detección del ataque

Los usuarios del sistema servidor atacado no notan nada raro puesto que las demandas de conexión falsificadas no cargan el sistema notoriamente. El sistema todavía puede establecer conexiones salientes. El problema se notará cuando se intente acceder a alguno de los servicios del sistema de la víctima.

Para verificar que el ataque está ocurriendo, verifique el estado del tráfico de la red del sistema. Esto puede hacerse por ejemplo con *netstat*. Demasiadas conexiones con estado “SYN_RECEIVED” pueden indicar que el sistema está siendo atacado.

Smurf

Hay informes de proveedores de servicios de red (NSPs), proveedores de servicios de Internet (ISPs) y otros sitios sobre ataques por denegación de servicios en los que se ven involucrados paquetes ICMP de petición de eco falsificados enviados a direcciones IP de broadcast. Estos ataques pueden producir grandes cantidades de paquetes ICMP de respuesta de eco que se envían desde un sitio “intermediario” a una víctima, con lo que la red de la víctima se congestionará.

Descripción

Los dos componentes principales del ataque por denegación de servicios *smurf* son el uso de paquetes ICMP de petición de eco falsificados y el direccionamiento de paquetes a direcciones IP de broadcast.

El Protocolo de Control de Mensajes en Internet (ICMP) se usa para manejar errores e intercambiar mensajes de control. ICMP puede usarse para determinar si una máquina responde en Internet. Para hacer esto, se envía una petición ICMP de eco a una máquina. Si la máquina recibe este paquete, devuelve un paquete ICMP de respuesta de eco. Una implementación común de este proceso es el comando “ping” que es incluido con muchos sistemas operativos y paquetes de software de red. ICMP se usa para controlar el estado e información de errores incluyendo la notificación de congestión en la red y otros problemas de transporte de red. ICMP también puede ser una valiosa herramienta para el diagnóstico de problemas en la red.

En las redes IP, un paquete puede dirigirse a una máquina individual o puede transmitirse a una red entera. Cuando un paquete se envía a la dirección IP de broadcast de una red local desde una máquina de la propia red el paquete es transmitido a todas las máquinas de esa red. Cuando un paquete se envía a una dirección IP de broadcast fuera de la red local, el paquete es transmitido a todas las máquinas de la red objetivo (siempre y cuando los routers por los que transita el paquete estén configurados para permitir el paso a este tráfico).

Las direcciones IP de broadcast normalmente son direcciones con la parte del host con todos sus bits a 1. Por ejemplo, la IP de broadcast de la red 10.0.0.0 es 10.255.255.255. Si se tiene dividida una red de clase A en 256 subredes, la dirección IP de broadcast para la subred 10.50.X.X sería 10.50.255.255. Las direcciones de red con todos los bits a cero en la parte del host, como por ejemplo, 10.50.0.0, también pueden producir una respuesta de broadcast.

En el ataque *smurf*, los atacantes utilizan paquetes ICMP de petición de eco dirigidos a IP de broadcast de máquinas remotas para generar ataques por denegación de servicios. Hay tres partes en estos ataques: el atacante, el intermediario y la víctima (Nótese que el intermediario también puede ser una víctima)

El intermediario recibe un paquete ICMP de petición de eco dirigido a la IP de broadcast de su red. Si el intermediario no filtra el tráfico ICMP dirigido a direcciones de broadcast, muchas máquinas en la red recibirán este paquete y devolverán un paquete ICMP de respuesta de eco. Cuando todas las máquinas en una red responden a esta petición de eco, el resultado puede ser la congestión severa de la red.

Cuando los atacantes crean estos paquetes, no usan la dirección de IP de su propia máquina como dirección de origen. Crean paquetes falsificados que contienen como dirección de origen la dirección IP de la víctima. El resultado es que cuando todas las máquinas del sitio intermediario responden a las demandas de eco, estas respuestas se envían a la máquina de la víctima. La red de la víctima estará sujeta a una congestión que puede potencialmente dejarla inutilizable. Aunque no se ha etiquetado al intermediario como una víctima, podría sufrir estos mismos ataques.

Se han desarrollado herramientas automatizadas que permiten enviar estos ataques al mismo tiempo a intermediarios múltiples, causando que todos los intermediarios dirijan sus respuestas a la misma víctima. También se han desarrollado herramientas para buscar routers de la red que no filtran el tráfico a direcciones IP de broadcast y redes donde múltiples máquinas responden. Estas redes pueden usarse como intermediarias en los ataques.

Impacto

Tanto la víctima como el intermediario de este ataque pueden sufrir una degradación de los servicios de red, tanto en sus redes interiores como en su conexión a Internet, hasta el punto de no poder utilizarlos.

Un tráfico alto puede causar serias bajadas de rendimiento en pequeños y medianos ISP que proporcionan servicios a intermediarios o a víctimas del ataque. Los ISP más grandes soportarán mejor el alto tráfico, pero también notarán una disminución en el rendimiento de la red.

Solución

La solución al problema pasa por evitar que paquetes con direcciones IP falsificadas circulen por la red y evitar también la circulación de paquetes que tienen como dirección destino una dirección IP de broadcast. Vea la sección *Prevención del IP Spoofing*.

Como solución añadida sería recomendable para el intermediario en el ataque configurar su sistema operativo para impedir que la máquina responda a los paquetes ICMP enviados a direcciones IP de broadcast.

Si un intruso viola la seguridad de una máquina de su red, puede intentar lanzar un ataque *smurf* contra su red desde dentro. En este caso, el intruso usaría la máquina comprometida para enviar el paquete ICMP de petición de eco a la dirección IP de broadcast de la red local. Este tráfico no viaja a través de los routers para alcanzar las máquinas en la red local, con lo cual, el filtrado que ha hecho en sus routers no es suficiente para prevenir esto.

Land

Descripción

Algunas aplicaciones TCP/IP son vulnerables (causan el cuelgue de la máquina) a paquetes que son construidos de una manera particular: un paquete SYN en el cual la dirección y el puerto de origen son iguales que los de destino. Land es una herramienta que aprovecha esta vulnerabilidad.

Impacto

Cualquier usuario remoto puede enviar paquetes falsificados que pueden colgar un servidor.

Solución

No hay una solución inmediata para evitar este tipo de ataques. La solución pasa por evitar que paquetes falsificados circulen por la red (Vea la sección *Prevención del IP Spoofing*) y mantener su software actualizado.

Para conseguir acceso a un sistema remoto, los intrusos intentan iniciar conexiones creando paquetes con dirección IP de origen falsa. Aprovechando aplicaciones que usan la autenticación basada en direcciones IP se consigue al acceso al sistema atacado (y probablemente como root). Es posible dirigir los paquetes a través de los firewalls si no se configuran para filtrar paquetes entrantes cuya dirección de origen está en el dominio local. Es importante notar que el ataque descrito es posible aún cuando ningún paquete de respuesta puede alcanzar al atacante.

Configuraciones potencialmente vulnerables:

- Routers a redes externas que soportan múltiples interfaces interiores.
- Routers con dos interfaces que soportan subredes en la red interna.
- Proxy firewalls que utilizan la IP de origen para la autenticación.

Servicios que son vulnerables al IP Spoofing:

- SunRPC y NFS
- Los comandos “r” de Unix
- X Windows
- Cualquier otra aplicación que use la dirección IP origen para la autenticación.

Cualquier servicio que use *Kerberos* para la autenticación no es vulnerable a un ataque IP spoofing.

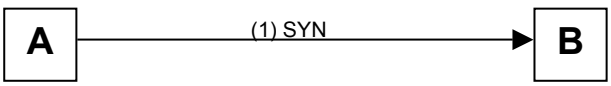
Información Previa

Notación y esquemas

A	Host objetivo
B	Host con el que A establece la relación *trusted host*
X	Host no alcanzable
Z	Host atacante
(Z)B	Host Z “disfrazado” como host B

Figura 4: Notación que utilizaremos en los próximos esquemas.

Los esquemas deben ser interpretados como el siguiente ejemplo:



A y B son hosts, sistemas participando en una conversación basada en TCP.
SYN es un ejemplo de bit de control de la cabecera TCP
(1) es un tick, un paso de tiempo. Es indiferente el tiempo que transcurre entre cada paso.

Figura 5: Ejemplo de esquema.

En la figura 5 el host A le está enviando un segmento TCP al host A con el bit de SYN activado. A menos que se indique lo contrario, no nos importa la porción de datos de dicho segmento TCP.

Trusted Hosts

En el mundo Unix, la confianza se da fácilmente. Por ejemplo, si se tiene una cuenta de usuario en la máquina A y otra en la máquina B, para facilitar la conexión de una a la otra se establece una relación entre ambas.

En el directorio *home* de la máquina A se crea un archivo *.rhosts* que contenga la siguiente información 'B nombre_usuario_B' y en el directorio de la máquina B se crea otro *.rhosts* similar 'A nombre_usuario_A'. Ahora, se podría usar cualquier comando "r" para conectar de una a otra cuenta si necesidad de verificar contraseñas, estos comandos permitirán la autenticación en base a las direcciones IP, es decir, permitirá o negará el acceso dependiendo de la dirección IP del solicitante.

Como alternativa, el root de una máquina puede establecer una configuración similar en */etc/hosts.equiv*, la diferencia está en que se haría a nivel de host, y no a nivel de usuario.

Rlogin

Rlogin es simplemente un protocolo cliente-servidor que utiliza TCP como medio de transporte. Permite a un usuario identificarse remotamente desde un host a otro, y si el host destino confía en el origen no pedirá la contraseña. En lugar de esto, habrá comprobado la identidad del cliente analizando su dirección IP. Por tanto, como en el ejemplo anterior, se podrá usar *rlogin* para acceder remotamente desde A a B o viceversa sin necesidad de introducir contraseñas.

Internet Protocol (IP)

Ya hemos visto cuál es la tarea del Internet Protocol en la introducción de este documento (*Introducción al IP spoofing*), pero vamos a recordar algunas propiedades significativas de este protocolo que tendremos que tener especialmente en cuenta aquí.

- IP es el protocolo más empleado de todos los protocolos TCP/IP ya que casi todo el tráfico TCP/IP está encapsulado en datagramas IP.
- Su trabajo es el de enrutar paquetes de la red y no ofrece ningún mecanismo de comprobación (es un protocolo "sin conexión") Es decir, IP simplemente envía datagramas y confía en que lleguen intactos a su destino. Si no lo hacen, IP puede intentar enviar un mensaje ICMP de error al origen, aunque, por supuesto, este paquete también puede extraviarse. (Recordemos que ICMP significa Internet Control Message Protocol, y se utiliza para informar sobre las condiciones en las que se encuentra una red y sobre los errores que se van produciendo)
- IP no mantiene ninguna información sobre el estado de la conexión.
- Cada datagrama IP es enviado sin ninguna relación con el último enviado o el siguiente a mandar.
- Un datagrama IP posee dos campos de encabezamiento de 32 bits para la información de las direcciones IP (origen y destino)

Todas estas propiedades unidas al hecho de que es sencillísimo modificar la pila de IP para permitir la elección de una dirección IP arbitrariamente en los campos de origen y destino convierten al protocolo IP en algo fácilmente modificable, y por tanto poco fiable.

Transmission Control Protocol (TCP)

TCP es el protocolo orientado a la conexión, el protocolo de transporte en el que se puede confiar plenamente dentro del sistema TCP/IP.

"Orientado a la conexión" significa simplemente que dos hosts que quieran intercambiar datos deben establecer previamente una conexión.

La seguridad se consigue a través de diferentes modos, pero los dos que nos conciernen en este momento son: secuenciación de datos e identificación.

TCP asigna números secuenciales a cada segmento e identifica todos los segmentos de datos recibidos desde el otro extremo (Se revisa la secuencia de números, no los segmentos en sí)

Estas características hacen que TCP sea mucho más difícil de adulterar que IP.

Números secuenciales, identificaciones y otras indicaciones

Dado que TCP posee una seguridad bastante aceptable, debe ser capaz de recuperar datos perdidos, duplicados, o fuera de servicio. Asignando una secuencia de números a cada byte transmitido, y requiriendo una identificación para cada uno recibido del extremo opuesto, TCP puede garantizar una transmisión sin errores. El extremo receptor utiliza la secuencia de números para asegurar el orden correcto de los datos y eliminar bytes duplicados.

Los números secuenciales del TCP se pueden imaginar como contadores de 32 bits. Se encuentran en un rango desde el 0 hasta el 4.294.967.295. Cada byte de datos intercambiado en una conexión TCP (junto a otros indicadores) va secuenciado. El campo del número secuencial en la cabecera TCP contendrá el número secuencial correspondiente al primer byte de datos en el segmento TCP.

El campo del número de identificación (ACK) en la cabecera TCP muestra el valor del siguiente número secuencial esperado, y también identifica todos los datos hasta este número de ACK menos uno.

Para el control del flujo, TCP envía un paquete para decirle al otro extremo cuántos datos puede "buffear". Dado que este paquete es de 16 bits, se puede notificar un máximo de 65535 bytes. El objetivo de este método es enviar una notificación desde un TCP al otro sobre la amplitud de la secuencia de números a emplear de manera que sea aceptable.

Otros indicadores en la cabecera TCP a mencionar son RST (reset), PSH (push) y FIN (finish).

Si se recibe un RST, se corta inmediatamente la comunicación. Los RST se envían normalmente cuando un extremo recibe un segmento que simplemente no tiene relación con la conexión que está establecida (veremos un ejemplo más tarde).

El indicador PSH le dice al receptor que pase tan pronto como sea posible todos los datos que se han ido almacenando a la aplicación correspondiente.

El indicador FIN es la manera en que una aplicación comienza el amable cierre de la conexión (el corte de una conexión es un proceso de 4 direcciones). Cuando un extremo recibe un FIN, lo ACKea (autentifica) y ya no espera recibir más datos (sin embargo el envío es todavía posible).

Establecimiento de una conexión TCP

Para poder intercambiar datos usando TCP, los hosts deben establecer una conexión. TCP establece una conexión siguiendo un proceso de 3 pasos llamado *saludo de tres direcciones*. Si la máquina A está utilizando un cliente de rlogin y desea conectar a un daemon de rlogin en la máquina B, el proceso es el siguiente:

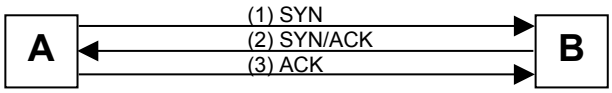


Figura 6: Esquema del "saludo de tres direcciones".

En (1) el programa cliente le está diciendo al servidor que quiere iniciar una conexión. Este es el único propósito del indicador SYN. El cliente le está diciendo al server que el campo de secuencia numérica es válido, y que debería ser comprobado. El cliente configurará el campo de secuencia numérica en la cabecera TCP a su ISN (Initial Sequence Number, *número inicial de la secuencia*).

El server, al recibir este segmento (2) responderá con su propio ISN (por lo tanto el flag SYN está activado) y una autentificación (ACK) del primer segmento enviado por el cliente (que será el ISN_del_cliente + 1).

El cliente entonces ACKea (autentifica) el ISN del servidor (3). Ahora ya puede tener lugar la transferencia de datos.

El ISN y el incremento de los números secuenciales

Es importante entender cómo son elegidos los números secuenciales inicialmente, y cómo cambian con respecto al tiempo.

El número secuencial inicial se cambia a 1 cuando el host se inicializa. TCP llama a esta variable "tcp_iss" dado que se trata del número secuencial inicial de envío (*initial send sequence number*). La otra variable de número secuencial, "tcp_irs" es el número secuencial inicial de recepción (*initial receive sequence number*) y se establece al crearse la conexión de 3 direcciones que tratamos antes. No nos preocuparemos por las distinciones entre las dos variables.

El ISN se incrementa en 128.000 cada segundo, lo que provoca que el contador de ISN de 32-bits quede agotado cada 9.32 horas si no se establece ninguna conexión. Sin embargo, cada vez que se establece un connect(), el contador es incrementado en 64.000. Esto es así para hacer mínimo el riesgo de que datos de una vieja conexión, con las mismas direcciones IP y puertos origen y destino, puedan llegar y mezclarse con los datos de la nueva conexión (Aquí se aplica el concepto del tiempo de espera de *2MSL* que no analizaremos)

Si los números secuenciales fuesen elegidos al azar cuando llega una conexión, no se podría garantizar que esos números secuenciales fuesen distintos de los empleados en una conexión anterior. Si una porción de datos quedase retenida en mitad de su recorrido y después consiguiese llegar a su destino interfiriendo con los envíos de la nueva conexión, nada bueno podría ocurrir.

Puertos

Para garantizar el acceso simultáneo al módulo de TCP, TCP provee una interfaz de usuario llamado puerto. Los puertos son utilizados por el *kernel* para identificar procesos de red y son estrictamente entidades de transporte (es decir, al IP no le importa su presencia).

Junto a una dirección IP, un puerto TCP forma lo que hemos llamado extremo de una comunicación de red. De hecho, en un momento dado cualquier conexión de Internet puede ser descrita por 4 números: la dirección IP de inicio y su puerto , y la dirección IP de destino y el correspondiente puerto de destino.

Los servers suelen ceñirse a puertos corrientes para que puedan ser localizados a través de puertos estándar en sistemas diferentes. Por ejemplo, el daemon de *rlogin* se encuentra en el puerto TCP 513.

Ataque

Introducción

El IP Spoofing se compone de varios pasos, que resumiremos ahora brevemente y analizaremos luego con más profundidad. Primero, se elige el host objetivo. A continuación descubrimos un indicio de “confianza” con otro host, es decir, nos lleva a un *trusted host*. Entonces se desactiva el trusted host, y se hace un muestreo de los números secuenciales de TCP del objetivo. Se usurpa la personalidad del trusted host, se averiguan los números secuenciales correspondientes, y se intenta la conexión a un servicio que sólo requiera identificación basada en direcciones. Si sale bien, el atacante ejecuta un simple comando para dejar una puerta trasera en el sistema.

Qué es necesario

- host objetivo
- trusted host
- host atacante (con acceso de root)
- software de IP spoofing

Generalmente el ataque se hace desde la cuenta root del host atacante contra la cuenta de root del objetivo.

IP Spoofing es un “ataque ciego”

Un factor que muchas veces no se analiza pero que es crítico en el IP spoofing es el hecho de que el ataque es ciego.

El atacante va a suplantar la identidad de un trusted host para poder saltarse la seguridad del host objetivo. Lo que el host objetivo cree es que está manteniendo una conversación con un host amigo mientras que, en realidad, el atacante está sentado en alguna oscura esquina de Internet, falsificando paquetes de este trusted host y a la vez enfrascado en una batalla DoS con el propio trusted host.

Los datagramas IP enviados con la dirección IP falsificada alcanzan su objetivo sin problemas (recordemos que IP es un protocolo “sin conexión”, cada datagrama es enviado sin tener en cuenta lo que pase con el otro extremo), pero los datagramas que

el host objetivo envía de vuelta (destinados al trusted host) se pierden. El atacante nunca los ve. Los routers que intervienen conocen dónde se supone que tienen que ir los datagramas. Se supone que van hacia el trusted host. En lo que respecta al nivel de red, allí es dónde fueron originados y allí es a donde van. Por supuesto una vez que los datagramas son enrutados hacia allí y la información es desmultiplexada y llega al TCP, se desecha (el TCP del trusted host no puede responder)

Por lo tanto el atacante tiene que ser inteligente y saber qué fue enviado, y saber qué respuesta está buscando el server. El atacante no puede ver lo que el host objetivo le envía, pero puede predecir lo que le enviará; eso unido al conocimiento con certeza de lo que enviará, le permite al atacante librarse de esta "ceguera".

Encontrando trusted hosts

Después de elegir un objetivo el atacante debe averiguar los posibles trusted hosts disponibles (daremos por hecho que el host objetivo confía en alguien)

Averiguar en quién confía un host puede no ser fácil. Un *showmount -e* puede mostrarnos a dónde se exportan los archivos del sistema, y *rcpinfo* también puede ofrecernos información interesante. Si se tiene abundante información sobre el host, no debería ser difícil. Si todo esto falla, probar direcciones IP vecinas en un esfuerzo de fuerza bruta puede ser una opción viable.

Desactivación del trusted host

Una vez que el atacante ha encontrado el trusted host, debe desactivarlo. Dado que el atacante va a hacerse pasar por él, debe asegurarse de que este host no reciba ningún tráfico de la red e interfiera en su conexión.

Existen muchas maneras para hacer esto, por ejemplo el TCP Syn Flooding (Vea la sección *Ataques por denegación de servicios – TCP Syn Flooding*)

Muestreo de los números secuenciales y predicción

Ahora el atacante necesita hacerse una idea de dónde se encuentra el TCP del host objetivo de entre el espacio de la secuencia numérica de 32 bits. El atacante conecta a un puerto TCP del host objetivo (SMTP es una buena elección) justo antes de lanzar el ataque y completa el saludo de tres direcciones con dicho host. El proceso es exactamente como en la figura 6, excepto que el atacante guardará el valor del ISN enviado por el host objetivo.

A menudo, este proceso se repite varias veces y el último ISN enviado se almacena. El atacante necesita saber cuál es el RTT (*round trip time*, tiempo de ida y vuelta) desde el objetivo a su host (El proceso puede repetirse varias veces, y se calcula una media de todos los RTTs hallados) El RTT es necesario para poder predecir con seguridad el siguiente ISN. El atacante tiene un punto de referencia (el último ISN enviado) y conoce también cómo funciona el incremento de los números secuenciales (128.000/segundo y 64.000 por cada connect) y ahora tiene una idea bastante aproximada de cuánto tardará un datagrama IP en viajar por Internet hasta alcanzar al objetivo (aproximadamente la mitad del RTT, dado que la mayoría de las veces las rutas son simétricas).

Después de que el atacante haya conseguido esta información, inmediatamente se procede a la siguiente fase del ataque (si otra conexión TCP llegase a algún puerto del

objetivo antes de que el atacante haya podido continuar con el ataque, el ISN real tendría una diferencia de 64.000 con el ISN previsto).

Cuando el segmento "spoofeado" recorre su camino hasta el objetivo, pueden pasar varias cosas dependiendo de la exactitud de la predicción del atacante:

- Si el número secuencial está exactamente donde el TCP receptor espera que esté, los datos que llegan serán colocados en la siguiente posición disponible del buffer receptor.
- Si el número secuencial es menor que el valor esperado el byte de datos se considera como una repetición de la transmisión, y es desechado.
- Si el número secuencial es mayor que el valor esperado pero todavía dentro de los límites de la capacidad del buffer, el byte de datos se considera que es un byte futuro, y es controlado por el TCP, pendiente de la llegada de los bytes que faltan antes que él. Si llega un segmento con un número secuencial mayor que el valor esperado y que no está dentro de los límites de la capacidad del buffer el segmento es excluido, y TCP enviará un segmento de respuesta con el número secuencial esperado.

Alteración

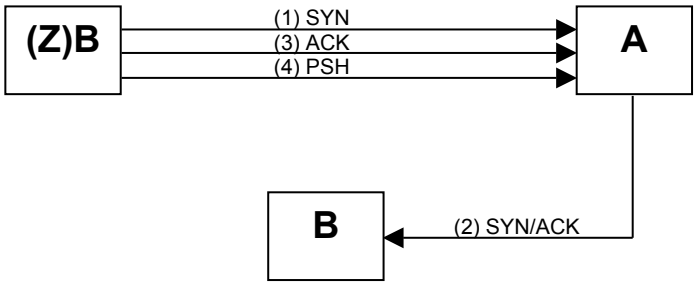


Figura 7: Inicio de la conexión “a ciegas” utilizando IP spoofing.

El host atacante falsifica su dirección IP para que sea la del trusted host (el cual todavía debería estar sufriendo los efectos del ataque DoS) y envía su petición de conexión al puerto 513 del host objetivo (1).

En (2), el host objetivo responde a la petición de conexión "spoofeada" con un SYN/ACK, que recorrerá su camino hasta el trusted host (el cual, si pudiera procesar este segmento entrante, lo consideraría un error, e inmediatamente envía un RST al host objetivo) Si todo va de acuerdo con lo previsto, el SYN/ACK será ignorado por el trusted host. Después de (1), el atacante puede descansar un poco para darle tiempo al host objetivo para enviar el SYN/ACK (el atacante no puede ver este segmento).

Entonces, en (3) el atacante envía un ACK al host objetivo conteniendo el número secuencial previsto (más uno, porque estamos ACKeándolo). Si el atacante acierta en su predicción, el host objetivo aceptará el ACK.

(4) El host objetivo establece la conexión y la transferencia de datos puede comenzar.

Generalmente, después de establecer la conexión, el atacante insertará una *backdoor* en el sistema que le permitirá llevar a cabo nuevas intrusiones de una manera más fácil. Con un "*cat + + >> ~/.rhosts*" suele bastar. Esta es una buena idea por varias razones: es rápido, permite accesos más simples, y no es interactivo. Recuerda, el atacante no puede ver el tráfico que proviene del host objetivo, por lo tanto todas las respuestas caerán en el olvido.

Por qué funciona

El IP spoofing funciona porque los servicios de *trust* entre ordenadores (como los trusted hosts) solamente basan su seguridad en autentificaciones de las direcciones de red. Dado que el IP es fácilmente engañable, la falsificación de direcciones no es difícil. La parte más complicada del ataque es la predicción de los números secuenciales, porque es ahí donde las suposiciones y conjeturas entran en escena.

Reducir las dudas y las adivinanzas al mínimo es básico para tener más posibilidades de éxito. Incluso un sistema que utilice los *TCP wrappers* de Wietse Venema es vulnerable al ataque. Los TCP wrappers se basan en los hostnames o en direcciones IP para las autentificaciones de los usuarios.

Medidas Preventivas

Descubrir los ataques

Si supervisa paquetes usando software como *netlog* y se encuentra paquetes en su interfaz externa que tienen IP de origen y destino dentro del dominio local, usted está siendo atacado.

Otra manera de descubrir la falsificación de IP es comparar los logs de acceso al sistema en su red interna. Si el ataque ha tenido éxito, se podrá conseguir una entrada de acceso remoto en el log de la máquina víctima, mientras que en la máquina de origen (también en su dominio) no habrá ninguna entrada correspondiente al inicio de ese acceso remoto.

No confiar en nadie

Una sencilla solución para prevenir este ataque es no utilizar la autentificación basada en direcciones. Desactivar los comandos "r", borrar todos los archivos *.rhosts* y vaciar el archivo */etc/hosts.equiv*. Esto forzará a todos los usuarios a emplear otras medidas de acceso remoto (telnet, ssh, skey, etc.)

Filtrado de Paquetes

Si su host tiene una conexión directa a Internet, puede utilizar su router como ayuda.

Primero asegúrese de que únicamente hosts de su propia LAN interna pueden participar en relaciones de trust (ningún host interno debe ser trusted host de otro sistema externo a la LAN, o viceversa)

Luego utilice el filtrado de paquetes para evitar que paquetes con una dirección de origen que pertenezca a su LAN llegado desde Internet pueda llegar hasta el interior. Vea la sección *Prevención del IP Spoofing* para más detalles.

Métodos Criptográficos

Un método obvio para evitar el IP spoofing es obligar a que todo el tráfico de la red sea encriptado y/o autentificado. Mientras se debaten otras posibles soluciones, puede establecerse ésta como medida estándar de seguridad.

Número secuencial inicial aleatorio

Dado que los números secuenciales no son escogidos aleatoriamente (o incrementados aleatoriamente) el ataque funciona. Bellovin aporta un parche para TCP que implica una partición del espacio dedicado al número secuencial. Cada conexión tendría su propio espacio separado de número secuencial. Los números secuenciales serían incrementados como antes, sin embargo, no habría ninguna relación obvia o apreciable entre la numeración en estos espacios. Se sugiere la fórmula siguiente:

ISN=M+F(localhost, localport, remotehost, remoteport)

Donde M es el cronómetro de 4 microsegundos y F es un hash criptográfico. F no debe ser calculable desde el exterior o el atacante podría todavía averiguar la secuencia numérica. Bellovin sugiere que F sea un hash del id de la conexión y un vector secreto (un número aleatorio, o un número de host secreto relacionado combinado con el tiempo que lleva encendida la máquina)

Anexo: Secuestro de conexiones (Hijacking)

No profundizaremos demasiado en el ataque que describimos a continuación puesto que no se trata realmente de una técnica de IP spoofing. A pesar de todo merece una mención especial.

Una vez que los intrusos tienen el acceso a root en un sistema, pueden utilizar una herramienta que circula por la red que permite modificar el kernel de Unix dinámicamente. Esta modificación les permite secuestrar terminales existentes y conexiones de login de cualquier usuario en el sistema.

Secuestrando las conexiones existentes, los intrusos pueden saltarse las contraseñas *one-time* y cualquier otro modelo de autenticación fuerte atrapando la conexión cuando la autenticación está completa. Por ejemplo, un usuario legítimo se conecta a un sitio remoto a través de un login o sesión terminal, el intruso secuestra la conexión después de que el usuario haya completado la autenticación al sitio remoto, con lo cual, ahora el sitio remoto se ve comprometido.

Sin mucho esfuerzo nos percatamos de que la combinación de estas dos herramientas (IP spoofing y Hijacking) podrían proporcionar acceso a un intruso a una amplia gama de sistemas, lo cual las hace extremadamente peligrosas.

Web Spoofing

Analizaremos aquí este tipo de ataques que aunque no son explícitamente técnicas IP spoofing guardan una estrecha relación en lo que se refiere al contexto engañoso que el atacante crea para poder burlar a la víctima y llevar a cabo sus propósitos.

Este agujero de seguridad de Internet puede poner en peligro la privacidad de los usuarios del World Wide Web y la integridad de sus datos. El ataque puede llevarse a cabo en los sistemas de hoy, poniendo en peligro a los usuarios de los navegadores web más comunes, incluyendo Netscape Navigator y Microsoft Internet Explorer.

El Web spoofing permite a un atacante crear una “copia oculta” del World Wide Web entero. Los accesos a la Web oculta están canalizados a través de la máquina del atacante, permitiéndole monitorizar todas las actividades de las víctimas incluyendo contraseñas y números de cuenta. El atacante puede también enviar datos falsos o engañosos a los servidores Web en nombre de la víctima o a la víctima en nombre de cualquier servidor Web. En pocas palabras, el atacante observa y controla todo lo que la víctima hace en el Web.

Ataques Spoofing

En un ataque spoofing el atacante crea un contexto engañoso para obligar a la víctima a tomar una decisión de seguridad impropia. El atacante prepara un mundo falso pero convincente alrededor de la víctima. La víctima hace algo que sería apropiado si el mundo falso fuese real. Desgraciadamente, actividades que parecen razonables en un mundo falso pueden tener efectos desastrosos en el mundo real.

Estos ataques son posibles en el mundo físico así como en el electrónico. Por ejemplo, ha habido varios incidentes en que los delincuentes prepararan un cajero automático ficticio y lo colocan en un área comercial. Las máquinas aceptaban tarjetas de crédito y pedían el número PIN de la tarjeta. Una vez la máquina tenía el PIN, podía hacer dos cosas: bien comer la tarjeta o bien devolverla alegando un “funcionamiento defectuoso” En cualquier caso, los delincuentes tenían bastante información para copiar la tarjeta de la víctima y usar el duplicado. En estos ataques, las víctimas fueron engañadas por el contexto que vieron: la situación de las máquinas, su tamaño, la manera en que fueron decorados y la apariencia de sus pantallas electrónicas.

Las personas que usan computadoras toman decisiones de seguridad a menudo basadas en las señales contextuales que ellos ven. Por ejemplo, usted podría decidir teclear su número de cuenta bancaria, porque usted cree que está visitando la página Web de su banco. Esta creencia podría adquirirse porque la página tiene un estilo particular, por que la URL del banco aparece en la línea de direcciones del navegador, o por alguna otra razón.

Para apreciar el rango y severidad de posibles ataques spoofing debemos centrarnos más profundamente en las dos partes de la definición del spoofing: decisiones de seguridad y contexto.

Decisiones de Seguridad

Por “decisión de seguridad” queremos decir cualquier decisión que una persona toma y que puede llevar a resultados indeseables si se elige una mala opción. Decidir divulgar información sensible, por ejemplo tecleando una contraseña o un número de cuenta es un ejemplo de decisión de seguridad. Escoger descargar un documento de la red es una decisión de seguridad, puesto que en muchos casos un documento es capaz de contener elementos malévolos que dañan a la persona que lo recibe.

Incluso la decisión de aceptar la exactitud de la información mostrada por su computadora puede ser una decisión de seguridad. Por ejemplo, si usted decide comprar acciones basándose en la información que recibe por un teletipo, usted está confiando en que la información que proporcionó el teletipo es correcta. Si alguien pudiera mostrarle una información incorrecta a través del teletipo, lo comprometerán en una transacción que usted no habría hecho si fuera de otra forma, y que le costará dinero.

El contexto

Un navegador presenta muchos tipos de contexto en los que los usuarios podrían confiar para tomar decisiones. El texto y las imágenes en una página Web pueden darle la impresión de que la página viene de un determinado sitio, por ejemplo, la presencia de un logotipo corporativo implica que la página la originó una cierta corporación.

La apariencia de un objeto le podría causar una cierta impresión, por ejemplo, usted podría pensar que está ante una ventana popup del navegador cuando lo único que ve es sólo un rectángulo que muestra una imagen de un popup. A los objetos gráficos particulares como una caja de dialogo de “abrir archivo” se les reconoce inmediatamente y se les supone un cierto propósito, que viniendo desde un navegador posiblemente sea engañoso. Los usuarios Web experimentados reaccionan a tales señales del mismo modo que los chóferes experimentados reaccionan a las señales sin leerlas.

Los nombres de objetos pueden llevar el contexto. Las personas deducen a menudo lo que está en un archivo por su nombre. ¿Manual.doc es un manual de usuario? Podría ser un amable documento, o podría no ser en absoluto un documento. Las URLs son otro ejemplo. ¿MICR0S0FT.COM es la dirección de una compañía de software? Durante algún tiempo esa dirección llevaba a otro sitio totalmente diferente.

A menudo recibimos el contexto según el orden cronológico de los eventos. Si dos cosas pasan al mismo tiempo, naturalmente pensamos que están relacionadas. Por ejemplo, si hace clic en la página de su banco y aparece un cuadro de diálogo username/password, usted asume que debe teclear su nombre y contraseña para el banco. Si usted hace clic en un enlace y un documento comienza a transmitirse inmediatamente, usted asume que el documento viene del sitio en donde ha hecho clic. Cualquiera de estas asunciones podría estar equivocada.

Si usted sólo ve una ventana del navegador cuando un evento ocurre, usted no podría comprender que el evento se causó por otra ventana que se esconde detrás de la ventana activa.

Los modernos diseñadores de interfaces de usuario dedican su tiempo intentando inventar señales contextuales que guiarán a las personas para comportarse apropiadamente, aún cuando no notan explícitamente las señales. Mientras esto es normalmente beneficioso, puede volverse un peligro cuando las personas están acostumbradas a confiar en el contexto que no siempre es correcto.

Web Spoofing

El Web spoofing es un tipo de timo electrónico en el que el atacante crea una convincente pero falsa copia de un Web entero. Las falsas Web son idénticas a las reales: tiene las mismas páginas y enlaces. Sin embargo, el atacante controla el Web falso, para que todo el tráfico entre el navegador de la víctima y el Web real pase por sus manos.

Consecuencias

Una vez que el atacante puede observar y modificar cualquier dato que va de la víctima a los servidores Web, así como controlar todo el tráfico de retorno de los servidores Web a la víctima, el atacante tiene muchas posibilidades, incluyendo la vigilancia y la adulteración.

- Vigilancia. El atacante puede mirar el tráfico pasivamente, grabando qué páginas visita la víctima y los contenidos de esas páginas. Cuando la víctima rellena un formulario, los datos introducidos se transmiten a un servidor Web que el atacante puede grabar también, junto con la contestación devuelta por el servidor. El comercio electrónico se realiza a través de formularios, esto significa que el atacante puede observar cualquier número de cuenta o contraseña que la víctima introduzca.

  El atacante pude llevar a cabo la vigilancia aún cuando la víctima tiene una conexión “segura” (normalmente vía *Secure Sockets Layer*) al servidor, es decir, aún cuando el navegador de la víctima muestra el icono de conexión segura (normalmente una imagen con una cerradura o una llave)

- Adulteración. El atacante también es libre de modificar cualquiera de los datos que viajan en cualquier dirección entre la víctima y el Web. El atacante puede modificar datos del formulario enviado por la víctima. Por ejemplo, si la víctima está comprando un producto on-line, el atacante puede cambiar el número de producto, la cantidad o la dirección de envío.

  El atacante puede también modificar los datos devueltos por un servidor Web, por ejemplo insertando material ofensivo causando malos entendidos entre la víctima y el servidor.

Duplicando el Web entero

Podríamos pensar que es difícil para el atacante “duplicar” el Web entero, pero no es así. El Web entero está disponible on-line, el servidor del atacante puede simplemente sacar una página del Web real cuando necesita proporcionar una copia de la página en el Web falso.

Cómo funciona el ataque

La clave de este ataque es que el atacante se sitúe entre la víctima y el resto del Web. Este tipo de técnica se denomina “man in the middle” en términos de seguridad.

Reescritura de URLs

El primer truco del atacante es volver a escribir las URLs en alguna página Web de un servidor real para que apunten al servidor del atacante. Asumiendo que el servidor del atacante está en la máquina www.hacker.org, el atacante rescribe un URL original agregando http://www.hacker.org al frente del URL. Por ejemplo http://home.netscape.com se vuelve http://www.hacker.org/http://home.netscape.com (La técnica de reescritura de URLs se ha usado para otras razones en algunos sitios Web como por ejemplo Anonymizer)

En este ejemplo el navegador de la víctima pide la página al servidor www.hacker.org puesto que la URL comienza con http://www.hacker.org. El resto del URL le sirve al atacante para saber dónde puede conseguir el documento real.

Lo que ocurre al hacer clic en el enlace es lo siguiente: (1) el navegador de la víctima solicita la página al servidor del atacante; (2) el servidor del atacante solicita la página al servidor real; (2) el servidor real proporciona la página al servidor del atacante; (4) el servidor del atacante rescribe las URLs de la página añadiendo http://www.hacker.org al inicio; (5) el servidor del atacante proporciona la versión reescrita a la víctima.

Como todas las URLs de la página han sido reescritas, si la víctima sigue un nuevo enlace de la página, la página se solicitará de nuevo al servidor del atacante. La víctima permanecerá atrapada en el Web falso del atacante, y puede seguir enlaces durante todo el tiempo sin abandonar el servidor del atacante.

Formularios

Si la víctima rellena un formulario en una página de un Web falso, el resultado parece haber sido manipulado propiamente. Los formularios falsificados trabajan normalmente puesto que están estrechamente integrados en los protocolos Web básicos: el envío de formularios y el envío de las respuestas después de haberlos rellenado forma parte del HTML ordinario. Desde que cualquier URL puede falsificarse, pueden falsificarse los formularios.

Cuando la víctima rellena y envía un formulario, los datos adjuntos van al servidor del atacante. El atacante en este momento puede observar e incluso modificar los datos adjuntos a su antojo antes de reenviarlos al servidor real. El servidor del atacante puede también modificar los datos devueltos en la respuesta al envío del formulario.

Las conexiones “seguras” no ayudan

Una propiedad desafortunada de este ataque es que incluso funciona cuando la víctima pide una página vía conexión “segura”. Si la víctima hace un acceso de Web “seguro” (un acceso de Web usando Secure Sockets Layer) en un Web falso, todo parecerá normal: la página se entregará, y el indicador de conexión seguro (normalmente una imagen de una cerradura o una llave) se encenderá.

El navegador de la víctima dice que tiene una conexión segura por que realmente la tiene. Desgraciadamente la conexión segura es a www.hacker.org y no al lugar donde la víctima piensa que es. El navegador de la víctima piensa que todo está bien: se debía acceder a www.hacker.org a través de una conexión segura. El indicador de conexión segura sólo le da un falso sentido de seguridad a la víctima.

Empezando el Ataque

Para comenzar un ataque, el atacante debe atraer a la víctima de algún modo a su Web falso. Hay varias maneras de hacer esto. Un atacante podría poner un enlace a un Web falso en una página Web popular. Podría mandar un correo electrónico a la víctima mostrándole ciertos contenidos para atraerlo. O bien, podría insertar un link a su Web falso en un motor de búsqueda.

Completando la ilusión

El ataque descrito es bastante eficaz, pero no es perfecto. Hay algún contexto que puede dar pistas a la víctima de que el ataque se está produciendo. Sin embargo, también es posible para el atacante eliminar todas estas pistas.

Tales evidencias no son demasiado difíciles de eliminar porque los navegadores son muy personalizables. La posibilidad de que una página Web controle la conducta del navegador es a menudo deseable, pero cuando la página es hostil puede ser peligroso.

Línea de estado

La línea de estado es una línea de texto en el fondo de la ventana del navegador que despliega varios mensajes, típicamente el estado de las transferencias pendientes.

El ataque descrito como hasta ahora deja dos tipos de evidencia en la línea de estado. Primero, cuando el ratón se sitúa encima de un enlace Web, la línea de estado muestra la URL a la que apunta el enlace. Así, la víctima podría notar que la URL se ha reescrito. Segundo, cuando una página se está transfiriendo, la línea de estado muestra brevemente el nombre del servidor que ha sido contactado. Así, la víctima podría ver que es www.hacker.org la URL que se muestra cuando se esperaba alguna otra dirección.

El atacante puede cubrir estas dos señales agregando un programa de JavaScript a cada página reescrita. JavaScript permite escribir en la línea de estado, y permite ligar eventos con acciones de JavaScript, con lo cual el asaltador puede arreglar las cosas para que la línea de estado participe en el timo mostrando siempre lo que se debería ver en la Web real. Así el contexto engañoso se vuelve más convincente aún.

Línea de Dirección

La línea de dirección del navegador muestra la URL de la página que se está viendo actualmente. La víctima puede teclear una URL en la línea de dirección enviando al navegador a esa URL. En el ataque descrito como hasta ahora se mostraría en la línea de dirección una URL reescrita, dándole una posible indicación a la víctima de que el ataque está en marcha.

Esta pista también se puede eliminar mediante JavaScript. Un programa de JavaScript puede esconder la línea de situación real, o también puede reemplazarla por una línea de dirección falsa que parece correcta y está en el lugar esperado. La línea de dirección falsa puede mostrar el URL que la víctima espera ver. JavaScript también puede permitir a la víctima teclear URLs normalmente en la línea de dirección y rescribirlas antes de que se acceda a ellas.

Viendo la fuente del documento

Hay una pista que el atacante no puede eliminar, pero es muy improbable que se descubra.

Usando la opción del navegador que permite visualizar el “código fuente” del documento, la víctima puede estudiar el código HTML de la página mostrada actualmente. Buscando URLs reescritas en el código HTML la víctima puede descubrir el ataque. Desgraciadamente, el código HTML es difícil de leer para los usuarios principiantes y muy pocos usuarios se molestan en mirar el código HTML de los documentos que están visitando, por lo que esto proporciona una protección muy pequeña.

Una pista relacionada está disponible si la víctima escoge la opción “ver información del documento”. Esta opción muestra información como por ejemplo la URL real del documento, permitiendo a la víctima detectar el ataque. Como anteriormente, esta opción casi nunca se usa por lo que no proporcionará mucha protección.

Bookmarks (Marcadores de enlaces favoritos)

Hay varias maneras en que la víctima podría abandonar accidentalmente el Web del atacante. Accediendo a un marcador o saltando a una URL utilizando la opción del navegador “Abrir dirección” la víctima podría volver al mundo real. La víctima podría volver a entrar en el Web falso pulsando el botón “Atrás”.

Los marcadores también pueden jugar en contra de la víctima, puesto que es posible añadir al marcador de enlaces favoritos una página en un Web falso. Saltando a tal marcador la víctima volvería a un Web falso.

Rastreando al atacante

Algunas personas han sugerido que este ataque puede detenerse encontrando y castigando al atacante. Es cierto que el servidor del atacante debe revelar su situación para llevar a cabo el ataque, y esa evidencia de esa situación estará ciertamente disponible después de que el ataque se descubra.

Desgraciadamente, esto no ayudará mucho en la práctica porque los atacantes irrumpirán en la máquina de alguna persona inocente y lanzarán el ataque desde allí. Se usarán máquinas ajenas en estos ataques por la misma razón que la mayoría de los ladrones de bancos hacen sus atracos en automóviles robados.

Soluciones

El Web spoofing es un peligroso y casi indetectable ataque de seguridad que se puede llevar a cabo en el Internet de hoy. Afortunadamente hay algunas medidas de protección que puede llevar a cabo.

Solución a corto plazo

La mejor defensa es seguir estos tres pasos:

- Desactive JavaScript en su navegador para que el atacante sea incapaz de esconder las evidencias del ataque.
- Asegúrese de que la línea de dirección de su navegador siempre es visible.
- Preste atención a las URLs desplegadas en la línea de dirección de su navegador, asegurándose de que siempre apunta al servidor que usted piensa que se conecta.

Esta estrategia disminuirá el riesgo de ataque significativamente, aunque todavía podría caer en algún ataque si no es consciente de mirar la línea de dirección habitualmente.

En la actualidad JavaScript, ActiveX y Java tienden a facilitar las falsificaciones y otros ataques a la seguridad, por lo que sería recomendable desactivarlos. Haciéndolo perderá alguna funcionalidad útil, pero puede resarcirse de esta pérdida activando estas opciones cuando visita un sitio confiable que los requiere.

En la actualidad, JavaScript, ActiveX, y Java todos tienden a facilitar engañando y otra seguridad ataca, para que nosotros recomendamos que usted los desactiva. Haciendo lo causarán perder alguna funcionalidad útil así, pero usted puede resarcirse de mucha de esta pérdida encendiendo estos rasgos selectivamente cuando usted visita un sitio confiado que los requiere.

Solución a largo plazo.

Nosotros no conocemos una solución a largo plazo totalmente satisfactoria a este problema. Aunque apostaríamos por el uso de certificados y las firmas digitales.

IP Spoofing desde el punto de vista jurídico

El IP spoofing supone fingir que se es un ordenador diferente al que se está utilizando.

En el Código penal español anterior al vigente, una de las circunstancias agravantes en la comisión de los delitos consistía en 'emplear astucia, fraude o disfraz'. Dicha redacción no pasó íntegramente al Código penal de 1995, sino que se vio modificada, quedando como agravante la de 'ejecutar el hecho mediante disfraz'.

La razón de ser de esta agravante consiste en que se considera de mayor reprochabilidad moral una acción cometida ocultando la identidad que una acción cometida a 'cara descubierta'.

Bajo ningún concepto debemos confundir el término “disfraz” aplicable a los ilícitos en Internet con el disfraz o uso de máscara de los códigos penales. Nos hallamos ante conceptos diferentes ya que el “disfraz” digital es un ilícito autónomo y no una disposición moral de un delincuente.

Cada uno de los datagramas que un ordenador envía a otro contiene, básicamente, las siguientes secuencias de datos:

- Identificación del destino.
- Identificación del remitente.
- Datos strictu sensu.

Partiendo de la anterior configuración, uno de los sistemas que existen para facilitar las comunicaciones en Internet, consiste en la declaración de 'ordenadores de confianza' (*trusted hosts*). A un ordenador puede instruírsele que los datagramas que llegan de otro determinado ordenador son datagramas en los que se debe confiar y, por tanto, acceder a la conexión, seleccionando así quién tiene acceso y quién no. La mecánica para decidir si se confía o no consiste en la verificación de la parte del datagrama que contiene la identificación del remitente.

Por tanto, el IP spoofing consiste en enviar al ordenador destino una serie de datagramas con una cabecera trucada en la que se ha alterado la identificación del remite ya que se ha suprimido la cabecera propia y se ha incluido una cabecera falsa.

Esta técnica no es nada fácil, pues primero deben averiguarse los datos de un ordenador B en quien el ordenador A tiene confianza, para después fingir ser B.

Diferencias con la figura delictiva de la negativa o degradación de servicio.

Podemos afirmar que no existen excesivos parecidos entre ambas conductas:

- El IP spoofing tiene como finalidad la de lograr una conexión como root en un ordenador ajeno. La negativa o degradación de servicio consiste en impedir o ralentizar las conexiones de otro ordenador.

- El IP spoofing permite al cracker acceder a información contenida en otro ordenador. La negativa o degradación de servicio nunca permite la obtención de información alguna.

Se necesita una gran habilidad técnica para hacer IP spoofing, no necesitándose ninguna para ejecutar el bombardeo que provoca la degradación o denegación de servicio.

Ante tales diferencias y aplicándoles a ambas conductas los principios de desvalor de acción y desvalor de resultado, así como la necesaria proporcionalidad de las penas (a mayores desvalores, mayor pena), el tratamiento penológico de ambas conductas debe ser más dura para el ilícito del IP spoofing que del ilícito de la degradación o negativa de servicio.

El desvalor de acción es más reprochable en el IP spoofing. El cracker no sólo debe poseer mejor preparación técnica (lo cual no es objeto del Derecho Penal), sino que, además, en el acto ejecutado interviene su pericia.

Una conducta en la que deben planificarse los actos y éstos deben encadenarse sucesivamente para lograr el resultado, es moralmente más condenable que otra conducta en la que sólo se necesita ejecutar un programa de software.

En lo que respecta al desvalor de resultado, los daños producidos difieren según el ilícito. En la negativa o degradación de servicio, los daños son temporales, corresponden al daño que se produce por la no realización del servicio, y su duración se limita al lapso del ataque. En los supuestos de IP spoofing los daños no pueden medirse bajo una óptica temporal sino cualitativa, referida a la destrucción o copiado de datos del ordenador atacado.

No cabe una valoración en abstracto de en cuál de los supuestos se produce un mayor desvalor de resultado. A priori no puede valorarse el perjuicio ocasionado, sino que se deberá analizar caso por caso el resultado producido. Si los ordenadores cuyo servicio se impide o dificulta pertenecen a una compañía de e-commerce, el perjuicio económico causado será mayor que el producido a una página personal. Teóricamente también puede darse el caso de que los datos obtenidos mediante la técnica de IP spoofing tengan un contenido económico mayor que el del perjuicio ocasionado mediante una negativa o degradación.

Las anteriores reflexiones nos llevan a concluir que el elemento del desvalor de resultado no podrá tenerse en cuenta al tratar penológicamente a ambas figuras, debiendo únicamente utilizarse como referencia el elemento del desvalor de acción.

La técnica del IP spoofing fue la utilizada por el mítico Kevin Mitnick para introducirse en los ordenadores de la compañía Pacific Bell, produciéndose un duelo personal entre el mismo y Tsutomu Shimomura, duelo ampliamente novelado en diversas publicaciones, y que culminó con la detención, juicio y condena de Mitnick.

Tras su salida de prisión (libertad condicional por buen comportamiento), Mitnick compareció ante el Senado de los EEUU en marzo del año 2000, donde intervino para señalar los cuatro talones de Aquiles en la seguridad de la red, y que él clasificaba en los siguientes:

- Acceso físico a los ordenadores.
- Diseño de las redes.
- Sistemas operativos y software.
- Personal que opera los ordenadores.

Resulta cuando menos sorprendente que un delincuente condenado comparezca ante una institución para, según sus palabras, 'presentar mis recomendaciones para que la Comisión pueda crear legislación efectiva'.

No obstante, ello es lógico, puesto que la legislación sobre los delitos cometidos en la Red se halla absolutamente en pañales, encontrándonos con noticias tales como las subvenciones por parte de algunas cámaras legislativas de las Comunidades Autónomas para que sus miembros compren un ordenador portátil y aprendan a manejarlos.

La técnica del IP spoofing no halla acogida alguna en el Código Penal español. Es más, no puede condenarse la entrada en un ordenador ajeno si no se utiliza la información contenida en el mismo.

En este aspecto, la legislación española carece del más absoluto conocimiento de la realidad.

Referencias

Security Problems in the TCP/IP Protocol Suite
Steven M. Bellovin.
http://www.ja.net/CERT/Bellovin/TCP-IP_Security_Problems.html

Web Spoofing: An Internet Con Game
Edward W. Felten, Dirk Balfanz, Drew Dean, and Dan S. Wallach.
Department of Computer Science, Princeton University.
http://www.cs.princeton.edu/sip/pub/spoofing.php3

TCP SYN Flooding and IP Spoofing Attacks
CERT Advisory CA-1996-21
http://www.cert.org/advisories/CA-1996-21.html

IP Spoofing Attacks and Hijacked Terminal Connections
CERT Advisory CA-1995-01
http://www.cert.org/advisories/CA-1995-01.html

IP Denial-of-Service Attacks
CERT Advisory CA-1997-28
http://www.cert.org/advisories/CA-1997-28.html

Smurf IP Denial-of-Service Attacks
CERT Advisory CA-1998-01
http://www.cert.org/advisories/CA-1998-01.html

Enmascaramiento
Redacción de Derecho de Internet
http://www.derecho-internet.org/teoria.php?teoria_id=31