Recomendaciones de seguridad en sistemas distribuidos de cómputo (V0.11)

Diego Bravo Estrada

En los últimos años el tema de la seguridad en las redes de computadores se ha tornado en un asunto de primera importancia dado el incremento de prestaciones de las mismas, así como la imparable ola de ataques o violaciones a las barreras de acceso a los sistemas implementados en aquellas. Los "incidentes de seguridad" reportados continúan creciendo cada vez a un ritmo más acelerado, a la par de la masificación del Internet y de la complejidad del software desarrollado [1],[2]. Este texto pretende presentar brevemente algunas recomendaciones dirigidas a los administradores e implementadores de redes informáticas, y lo he preparado a modo de síntesis breve a partir de diversos materiales difuminados en Internet. No pretendo hacer un listado exhaustivo de las responsabilidades del administrador ni explicar detalles acerca de la implementación de las recomendaciones, aunque sugiero algunos puntos de partida para esto.

Cualquier sugerencia o corrección, por favor hacérmela llegar vía mi página web [3], donde debería estar siempre la última versión de este texto.

Tabla de contenidos

1. Efectuar un análisis de riesgos

Esto se suele mencionar en la literatura como el primer paso a realizarse cuando se plantea la seguridad en un sistema [4]. La idea es muy sencilla: trazar todos los elementos que conforman nuestro sistema (hardware y software) y observar cuáles involucran más o menos riesgo. Esto desembocará en un plan de seguridad cuyo objetivo es disminuir el riesgo total del sistema, que se puede modelar como la suma de los riesgos de sus componentes:

RIESGO TOTAL = RIESGO(componente 1) + RIESGO(componente 2) ...

El riesgo de cada componente está en función directa a las pérdidas que ocasionaría el que éste deje de operar, así como en función de cuán vulnerable es dicho componente en este momento. Por ejemplo, una base de datos de clientes involucra un gran riesgo debido al gran valor que la información representa para una organización; pero una simple PC Windows de la misma organización conectada directamente al Internet (sin firewall/proxy de por medio) también lo representa, debido a que puede ser objeto de un ataque desde el exterior, con el posible riesgo de fácil propagación hacia otros computadores de nuestra red.

El riesgo no es fácil de cuantificar, siendo en general un estimador subjetivo. A modo de ejemplo podríamos plantear una fórmula como la que sigue:

RIESGO(componente) = P * V

Donde P=pérdida, es la pérdida en dinero que implicaría la inoperatividad del componente hasta su reparación, aunque se pueden agregar otros estimadores como el desprestigio ante nuestros clientes. A veces ayuda considerarlo como "el valor" que representa para la organización. V=vulnerabilidad, es tanto o más subjetiva puesto que no hay una manera segura de establecer para todos los casos si los supuestos mecanismos de protección (del componente) son o no realmente confiables. Así por ejemplo, podríamos suponer que la vulnerabilidad de ciertos documentos importantes es muy baja, puesto que están protegidos por cierto antivirus. Sin embargo, esto realmente estará en función de diversas características del antivirus, como pueden ser: recientes actualizaciones, ejecución automática, capacidad de eliminar virus locales, licencias no caducadas, configuración adecuada, etc. Pero por algo se debe comenzar. Por ejemplo, se puede asignar números como 1, 2, 3, 4, para señalar una vulnerabilidad mínima, regular, importante y peligrosa, respectivamente. Para una extensa (y compleja) explicación, ver [5].

Esto normalmente demanda el acopio de mucha información, la que muchas veces no está a cargo de una sola persona. Esto implica que todo el staff encargado de las distintas partes del sistema debe colaborar en el análisis.

2. Lo más valioso debe alejarse de lo más vulnerable

En la fórmula del "riesgo" propuesta arriba, es evidente que los componentes de nuestro sistema con algo valor y alta vulnerabilidad serán de lejos los que presenten mayor riesgo. Sin embargo, en muchos casos no es sencillo disminuir el valor de cierto componente (y por tanto la pérdida en caso de problemas), y tampoco se puede eliminar completamente la vulnerabilidad del mismo (por ejemplo, si está de cara a Internet.) En este caso lo que conviene es separar o dividir este componente en dos partes suficientemente alejadas e independientes a fin de que el riesgo total disminuya. Por ejemplo, los portales de comercio electrónico deben dar cara a Internet (siendo vulnerables en

principio) y a la vez manejar información muy costosa (como transacciones con tarjeta de crédito.) Esto los convierte en un sistema de alto riesgo. Sin embargo es casi universal la separación que se efectúa entre los componentes dedicados a dar cara a Internet (como los Web Servers) y los componentes que manipulan la información comercial (generalmente sistemas DBMS.) En términos prácticos, esto significa que el hacker no puede acceder directamente al DBMS (lo que sería catastrófico), y sólo podría atacar al Web Server, lo que en principio no acarrea mayores consecuencias.

Juguemos con los números siguiendo las ideas expuestas más arriba. Suponiendo que los datos relacionados al negocio valen para nosotros $50000, y están en un computador donde corre un Web Server IIS que no ha sido parchado adecuadamente. Entonces el riesgo sería:

R total = 50000 x 5 = 250000

Supongamos ahora que los datos están en otro computador "mínimamente vulnerable" (con V=1) adecuadamente aislado del Web Server (que todavía no ha sido parchado). La pérdida por una interrupción del servicio del Web Server, por día se ha estimado en $2000 (asumimos que ese es el tiempo que toma en restablecerse el servicio tras el ataque.) Entonces el riesgo total se convierte en:

R total = R1 + R2 = 50000 x 1 + 2000 x 5 = 60000

3. Mantener las cosas simples

Un sistema complejo es más difícil de asegurar y potencialmente proporciona una mayor cantidad de puertas abiertas a los atacantes. En general, es recomendable intentar dividir el problema mediante la simplificación de la configuración, para así identificar los puntos o rutas de control vulnerables para incrementar la seguridad.

Un ejemplo frecuentemente citado es la seguridad de una red de estaciones Windows manipulada por usuarios inexpertos. En muchos casos no resulta práctico efectuar un profundo trabajo de seguridad en las estaciones (más allá de la instalación del antivirus, por ejemplo), puesto que por un lado, son muchas, y por otro, los usuarios suelen modificar la configuración en tanto instalan y desinstalan su software sin dar aviso a nadie. En estos casos es aconsejable centrarnos en un único punto que centralice la seguridad de todas las estaciones. Una configuración de red que obligue a que todo su tráfico pase a través de un gateway permitiría crear un único punto de control para todas las estaciones, con lo cual simplificamos la administración.

Todo esto generalmente conlleva a situaciones en las que hay que hacer un compromiso. Por ejemplo, en la recomendación anterior, optamos por separar la base de datos del servidor Web con lo que la complejidad del sistema se incrementó, aunque se redujo el riesgo total.

4. Asegurar la seguridad en todos los niveles

Esto se puede expresar más sencillamente como: no confiar el sistema a un único mecanismo de seguridad.

La información fluye a través de los distintos componentes y/o capas del sistema y son muchas las instancias en las que se puede mejorar su seguridad. La recomendación estipula que utilicemos todas estas instancias a pesar de que en principio puedan parecer redundantes. Por lo general los administradores tienden a preocuparse por un único punto de acceso desde donde supuestamente hay una gran probabilidad de ser atacados (por ejemplo, la conexión a Internet.) Por tanto se invierte esfuerzo y dinero en controlar este único punto bajo la asunción de que es la única puerta de entrada a los maleantes y que por tanto, tras asegurarla, todo el sistema quedará seguro. Esto tiene dos problemas:

- Muchos ataques o "vulnerabilidades" se originan (de forma inocente o intencional) desde dentro de la organización
- El sistema que controla la "puerta" siempre puede fallar

Esto obliga a implementar la seguridad no en un único punto evidentemente vulnerable, sino en todos los lugares por donde fluye la información al interior de cada componente involucrado.

Un ejemplo típico lo constituye un filtro de paquetes TCP/IP, operando a nivel de capas 3 y 4 del modelo OSI. Estos por lo general hacen un gran servicio restringiendo accesos a servicios de red internos y se suelen colocar en la "puerta" o "gateway" que da cara a Internet o a una red insegura a modo de "firewall".

Tomando literalmente la idea de "niveles" o "capas", podríamos pensar qué hacer en este gateway para mejorar la seguridad. Son 7 las capas OSI y podríamos intentar añadir instancias de control adicionales a las mencionadas. Por ejemplo, en la capa 7 (aplicación) es frecuente y muy recomendable el empleo de proxys HTTP. Algunos firewalls proporcionan control a nivel de la capa 2 (específicamente, del MAC Address.)

Sin embargo, esto se debe complementar con otras medidas que van más allá del gateway y que se implementan a lo largo de todo el sistema (pudiéndose considerar como niveles o "capas" adicionales), tales como redes perimétricas, el uso de encriptación, etc.

5. Encriptar tanto como sea posible

La encriptación es un tema complejo pero cuya implementación resulta cada vez más sencilla conforme aparecen más productos. Los cambios del año pasado en la legislación norteamericana con respecto a la exportación de productos que encriptan, son un incentivo claro para que los desarrolladores y vendedores se interesen más en el tema.

En general, los canales de comunicación más vulnerables o de mayor cercanía al público requieren una encriptación "más fuerte", es decir, más difícil de descifrar por los curiosos o atacantes. Cierta información conlleva más riesgo que otra, y por tanto requerirá un nivel de encriptación diferenciado.

Las herramientas capaces de hacer esto son muchas, dependiendo del contexto en que nos encontremos. Por ejemplo, los sistemas DBMS más avanzados incorporan la encriptación como una opción normal para los datos almacenados, generalmente bajo esquemas propietarios.

La tecnología de encriptación de información destinada a pasar a través de la red ha evolucionado bastante, haciéndose popular el término VPN [6],[7] para hacer referencia a canales que encriptan la información de un modo más o menos transparente. Hay soluciones propietarias así como estándares relativamente implementados como IP Sec.

Ciertas aplicaciones estándar han recibido soluciones de encriptación también estándar. El caso del Web encriptado bajo SSL (HTTPS) junto con la industria de certificados digitales es el caso más conspicuo. De igual modo los estándares para correo electrónico PGP (o derivados) y S/MIME son integrados cada vez con mayor frecuencia en las aplicaciones de los usuarios finales.

Retornemos al principio. En nuestra organización deberíamos encriptar todo lo que sea posible. La razón de esto es evidente si de lo que se trata es de enviar un mensaje privado por Internet. Sin embargo, al interior de la organización la encriptación puede ayudar también. Por ejemplo, es usual que cierta información sea manejada exclusivamente por un número reducido de personas (los contratos salariales, por poner un caso.) Un caso más dramático se presenta cuando un hacker ha logrado infiltrarse en la organización: si la información está encriptada, los datos que robe le serán inútiles dado que no posee las claves correspondientes.

Naturalmente hay que sopesar los inconvenientes que trae la encriptación en términos de incomodidad de uso, costo de licencias, ciclos de CPU, etcétera; con el hecho de que cierta información es definitivamente de carácter público y por tanto no tiene sentido que esté encriptada. Por ejemplo, la información que se publica en Internet vía el Web Server de la empresa. En [8] se puede obtener regularmente mucha información interesante y actualizada sobre la encriptación.

6. No confiar en la autenticación estándar

Las redes locales, y por extensión, el Internet, NO fueron diseñados en principio para ser seguros. La mayoría del software y hardware creados hasta incluso mediados de los años 90, no tuvieron la seguridad como objetivo de primer orden. Esto implica que muchos de los sistemas que compramos y usamos en la actualidad tienen serias deficiencias, pero que por mantener la compatibilidad no se han podido corregir.

Un caso crítico es lo concerniente a la autenticación de los accesos. En la gran mayoría de casos esto se limita a proporcionar una contraseña por parte de quien pretende acceder al recurso correspondiente, cosa que funciona bien en un ambiente seguro en el que todos estamos de acuerdo en no ir más allá de este mecanismo. Sin embargo esto no siempre es cierto (y menos en Internet.) Son muchos los mecanismos que puede emplear un hacker para "capturar" las contraseñas de usuarios reales, así como para "adivinar" las mismas. Otros esquemas pretenden ser más seguros basándose en el análisis de la dirección de red de quien intenta conectarse, pero de igual modo los hackers pueden "simular" una dirección de red sin mayores inconvenientes.

En tanto los ataques son cada vez más sofisticados, es de rigor que la autenticación también sea cada vez más sofisticada, lo cual implica abandonar para siempre diversos mecanismos estándar muy arraigados. El ejemplo más paradigmático tal vez sea el comando "telnet" de los sistemas Unix, utilizado para obtener una sesión en un sistema remoto. Este comando se distribuye (y se seguirá distribuyendo sin duda) en prácticamente todas las encarnaciones

de los sistemas Unix y sus clones (como Linux.) Es tan popular y útil que incluso los sistemas Windows lo proporcionan. Sin embargo, ningún administrador serio recomendaría su uso para una conexión remota, y quizá tampoco al interior de la red. La razón estriba en que toda la información de autenticación (el nombre de usuario y su contraseña) viaja sin encriptación tal cual es, y cualquier persona con acceso a un nodo intermedio o al canal de la conexión puede extraer esta información con relativa facilidad.

Actualmente diversos productos y estándares promueven la seguridad en la autenticación. Como muestra podemos mencionar a SSH [9], S/Key [10], Kerberos [11], Tcpwrappers, etc.

7. No usar la configuración "estándar"

Esto se puede considerar una generalización de la recomendación anterior. Por lo general los sistemas operativos y las aplicaciones se instalan con una configuración determinada y de carácter genérico. En muchos casos el administrador no tiene necesidad de modificarla pues el sistema aparentemente funciona bien. Sin embargo, los atacantes al no tener conocimiento de la configuración de nuestro sistema, asumen que ésta es justamente la "estándar", y programan sus ataques basados en dicha asunción.

Leí en algún lugar acerca de un administrador cuyo servidor web IIS fue atacado con éxito por "CodeRed", pero las páginas web publicadas no fueron alteradas debido a que antes había decidido modificar en la configuración el directorio donde se guardan los documentos html.

Ciertamente este tipo de cambios tienden a complicar el sistema puesto que sólo el administrador es consciente de tales. Sin embargo esta complejidad también la padecerá el atacante. Lamentablemente muchas herramientas "automáticas" de configuración, en su afán de hacer la vida del administrador más sencilla, no permiten hacer modificaciones radicales. Esta tendencia a la homogeneidad en la configuración va en aumento.

Muchos administradores (y sus jefes) pretenden elevar la seguridad tan sólo a costa de comprar un producto costoso. Imaginémonos una ciudad en la que todas las casas comprar un mismo modelo de cerradura. Entonces lógicamente los ladrones se dedicarán sin parar a conseguir la llave correspondiente que les abrirá las puertas de todas las casas. Ahora imaginemos otra ciudad en la que todos los habitantes son cerrajeros. Seguramente los ladrones dejarán de pensar en las llaves y a no ser que encuentren otro mecanismo distinto, tendrán que mudarse.

8. La seguridad hacia el interior

Algunos reportes [12] han puesto de relieve que en una gran cantidad de casos la mayor amenaza de ataques al sistema no proviene de fuera, sino que parte desde el interior de la organización. Muchos ataques exitosos desde el exterior han necesitado de cierta ayuda inicial activada en el interior de la organización, donde por lo general nadie sospecha de este tipo de prácticas.

Por tanto, el análisis de riesgos debe incluir posibles ataques originados en el interior, incluyéndose el robo de contraseñas, la modificación de archivos de configuración, la desactivación de las barreras de protección, etc.

Un caso muy común de este tipo de ataque lo constituye el trabajador despedido o castigado que decide tomar venganza. Antes de retirarse definitivamente puede efectuar este tipo de tareas maliciosas e incluso ponerse en combinación con un atacante externo. En ciertos casos la simple introducción intencional de un virus puede acarrear efectos devastadores.

La única manera de reducir el riesgo en estos casos, consiste en planificar el acceso al sistema de modo tal que ningún elemento crítico dependa de una sola persona. Dicho de otro modo, para dañar un sistema, no debe bastar con un único individuo disconforme.

Esto muchas veces no es posible con los sistemas operativos comerciales actuales (por ejemplo, con las estaciones Windows 9x) y se hace aún más difícil si el despedido es el administrador! Algunos sistemas intentan por tanto dividir las responsabilidades en diversos administradores, pero en el sector comercial todavía falta mucho camino por recorrer. Los usuarios de Linux podrían interesarse en el sistema LIDS a fin de implementar esta recomendación.

9. Educar a los usuarios

Una de las mayores ayudas que puede recibir un hacker que intenta infiltrarse en el sistema de una organización consiste en obtener información acerca de éste. En este sentido, las prácticas empleadas por el atacante comprenden muchas veces la interacción encubierta con los usuarios de la organización a los cuales se les extrae (sin que tomen conciencia de esto) una serie de datos útiles para el hacker. El caso más evidente consiste en obtener "como jugando" una contraseña de parte de este incauto. Y ciertamente las contraseñas de los usuarios comunes suelen ser muy malas, por lo que pueden liberarlas inadvertidamente al interlocutor en medio de una conversación aparentemente inocente. Esto se suele denominar "Ingeniería Social".

La única forma de combatir esto es con educación para los usuarios. Por ejemplo, es necesario hacer que ellos sean conscientes de este tipo de ataque (o entrevista), y que una de las mejores medidas a tomarse es el uso de contraseñas suficientemente complicadas como para que no surjan de improviso en cualquier diálogo.

Otra recomendación (o norma obligatoria si se desea) consiste en que los usuarios no divulguen detalles acerca de la configuración del sistema. Esta es una práctica increíblemente extendida: Auxiliares, programadores, ingenieros, e incluso administradores, que en su deseo de hacer alarde del "super sistema" que utiliza su empresa, empiezan de pronto a nombrar todos y cada uno de sus componentes, tanto de hardware como de software, con número de versión incluido.

Esto puede ser inmediatamente aprovechado por el atacante, pues empezará a "disparar" los ataques ya conocidos para ese hardware/software específico.

10. No confiar (totalmente) en nosotros mismos

Esto puede sonar extraño, sin embargo lo único que quiero indicar es la necesidad de que otra persona verifique la seguridad de nuestro sistema. Como se sabe, existen empresas consultoras especializadas en auditar nuestra organización con este fin. Si esta última opción no es posible (por ejemplo, por el costo involucrado) entonces es de rigor solicitar a otro administrador o ingeniero que verifique las medidas que hemos considerado. En sistemas y redes complejas es muy posible que una sola persona (nosotros) hayamos dejado pasar alguna puerta insegura. Mientras más personas verifiquen nuestro trabajo, habrá más probabilidades de que éste esté adecuadamente realizado. Esta es la idea que está detrás de mucho software Open Source, siendo el Kernel de Linux el caso más conspicuo.

Naturalmente evitaremos mostrar estos detalles a personas ajenas a la organización, a fin de disminuir el conocimiento que se tiene en el exterior acerca de nuestro sistema [13].

11. Ejecutar sólo los servicios imprescindibles

Algunas personas tienen la manía de instalar los sistemas con la mayor cantidad posible de opciones que puedan entrar en el disco duro. Los administradores de sistemas seguros deben ir exactamente en sentido inverso: en un sistema de alto riesgo es de rigor que se ejecute únicamente lo imprescindible. El ejemplo más conocido corresponde a los servicios de red, los cuales muchas veces vienen configurados para estar activos tan pronto como se instala un sistema operativo, creándose automáticamente nuevas oportunidades para los atacantes.

El administrador debería asegurarse de que sólo esté instalado lo imprescindible para que el sistema siga en operación. Debe descartar incluso el software aparentemente más inofensivo. Recuerdo haber leído de un caso en el cual el sistema de manuales de Solaris tenía una vulnerabilidad explotable que permitía a cualquier usuario ejecutar programas con mayores privilegios.

Sistemas que en apariencia no tienen nada de riesgosos han presentado serias amenazas para la seguridad, como por ejemplo, las conexiones X Window, las librerías compartidas, los módulos del kernel, etc.

12. Mantenerse al día con las actualizaciones

Esta recomendación cada vez es más crítica [14]. El software, pese a los esfuerzos y la propaganda, continuará teniendo errores y puertas ocultas. Y al parecer la tendencia sigue en aumento con la complejidad del mismo. Esto implica que los vendedores deberán proporcionar parches o versiones mejoradas a sus clientes cada vez que se descubra alguna vulnerabilidad.

Lamentablemente esto no se suele tomar muy en cuenta, tanto por las empresas de software como por los administradores. Desde que se descubre una vulnerabilidad hasta que ésta puede ser aprovechada por un atacante puede pasar mucho tiempo (incluso puede ser que nunca se logre aprovechar) lo que motiva a que las casas de

software tiendan a esperar más de la cuenta para admitir la vulnerabilidad a la vez que presentan el parche correspondiente.

En sistemas operativos que involucran muchos componentes (como casi todos en la actualidad) es de esperarse que las vulnerabilidades surjan con una frecuencia extremadamente alta. Desde el punto de vista de la casa de software, esto significaría incomodar con demasiada frecuencia a los administradores (que se convertirían en eternos "parchadores") por lo que usualmente se distribuyen parches a intervalos relativamente extensos y que corrigen de un solo golpe todas vulnerabilidades halladas hasta la fecha, generalmente acompañadas de aditamentos al sistema de modo tal que no luzca como un simple parche (piénsese en los "service pack".) Obviamente, los administradores que se limitan a esperar el lanzamiento del próximo super-parche corren un gran riesgo en este lapso.

En un ambiente de alta seguridad esto no es aceptable: los parches de cada componente deben aplicarse tan pronto como sea posible. Por tanto, si no deseamos que la administración se convierta en un parchar y parchar, deberíamos asegurarnos que nuestro sistema realmente tenga pocos componentes (y por tanto, menos necesidad de parches.) Como se indicó en la recomendación anterior, en todo sistema seguro, lo más indicado es evitar que se ejecuten servicios innecesarios. He aquí un motivo más para seguir esta regla.

Otro corolario de esto es el establecimiento de una rigurosa política de actualización, para lo cual normalmente existen diversos canales en Internet proporcionados por el vendedor. De igual modo, es menester mantenerse informado acerca de las nuevas vulnerabilidades descubiertas, para lo cual existen diversas publicaciones electrónicas especializadas [15],[16].

13. Escaneos regulares

Un "scanner" es un programa que intenta indagar acerca de qué servicios proporciona un computador de la red. Una vez que se conocen estos servicios, un atacante puede centrar sus ataques hacia los mismos.

Esta herramienta es empleada regularmente por los hackers. Sin embargo, la podemos emplear en nuestro propio beneficio, puesto que así nosotros podremos descubrir si hemos dejado ciertas puertas abiertas. El scanner sólo se limita a proporcionar esta información y por tanto su uso es seguro e inofensivo.

El "escaneo" se debería hacer desde fuera de nuestra red (por ejemplo, desde un computador de Internet), así como desde su interior contra nuestras estaciones de trabajo [17].

14. Descargas de software de Internet

Como regla general, el software no debería ser descargado de Internet, sino adquirido de una fuente confiable. Sin embargo en muchos casos esto es imprescindible por lo que deberíamos seguir algunas reglas mínimas para no terminar descargando un virus o un "caballo de Troya".

En primer lugar, debemos asegurarnos que el software que descargamos es realmente lo que hemos pretendido descargar. La idea es que un atacante podría modificar los datos que estamos recibiendo e introducir modificaciones en aquello que descargamos. Luego nosotros confiadamente ejecutamos el archivo en cuestión y... la historia es conocida.

Para evitar esto, cada vez con mayor frecuencia los portales de descarga incluyen una serie de alternativas para verificar que lo que hemos descargado no ha sido alterado. Un método común consiste en el "checksum MD5" que el usuario debería aplicar a todo archivo que descargue, a fin de obtener una especie de "firma" que ha de coincidir con la que se anuncia en el portal.

Es asimismo muy recomendable que los portales de descarga utilicen certificados digitales, de modo que no seamos presas de un portal ficticio implementado por atacantes (esto último es realmente es muy difícil, pero no imposible.)

15. Establecer planes de contingencia y sistemas de respaldo

No existe ninguna garantía de que nuestro sistema sea invulnerable. Más allá de las medidas que podamos adoptar, siempre existirá la posibilidad de ser atacados. Esto nos obliga a tener presentes ciertas medidas de contingencia traducidas preferentemente en políticas de seguridad bien establecidas.

En otras palabras, debemos imaginarnos sucesivamente un conjunto de escenarios de ataques exitosos. ¿Qué hacemos si...

- Sospechamos que un hacker está atacando el firewall
- Sospechamos que ya ha tomado control del firewall
- Comprobamos que ya ha tomado control del firewall
- Sospechamos que el servidor de base de datos ha sido alterado
- Descubrimos que las PCs Windows han sido infectadas con un virus

Las posibilidades evidentemente son muchas, y las acciones a tomarse también. Sin embargo es mejor tener esto por escrito a no tener absolutamente nada el día que las quejas sobre sucesos extraños en el sistema empiecen a acaecer.

Ahora bien, cualquier administrador medianamente decente tendrá establecida una política de backups para situaciones críticas. En el caso de ataques a través de la red las consecuencias pueden obligar a requerir de estos backups, por lo que es imperativo que éstos estén actualizados y preparados para su uso.

Este tipo de análisis no debe ser extraño puesto que los sistemas pueden tener problemas por múltiples motivos que no tienen nada que ver con los ataques (por ejemplo, un fallo de hardware) que deberían también poseer planes de contingencia.

Los administradores más minuciosos podrían efectuar simulacros de desastres: No hay nada más frustrante en medio de un desastre que descubrir que los backups diarios nunca se grabaron por problemas en la unidad de cinta.

16. Mantener contacto con el proveedor de líneas de comunicación

Diversos problemas pueden aparecer en las comunicaciones desde nuestra red hacia el exterior en los cuales sólo el proveedor del enlace puede ayudarnos. Ciertos tipos de ataques necesitan de la participación de uno o más proveedores de Internet para paliarlos, como son ciertos tipos de ataque DoS (Deny of Service) ante los que nuestro firewall podría quedar indefenso. Sólo mediante el concurso de uno o más nodos de alto nivel del Internet se podría bloquear al atacante [18],[19],[20]. Por suerte los ataques más sofisticados de este tipo no son muy frecuentes.

17. No permitir conexiones directas desde la red interna a Internet

Asumimos que en Internet están los malos, y por tanto nuestra red interna no debería tomar contacto con ellos. Sin embargo el Internet es irresistible e imprescindible para muchas personas de nuestra organización, por lo que debemos buscar algún mecanismo de protección. Una de las soluciones más generalizadas a este problema la constituyen los sistemas denominados proxy. En este caso, los usuarios de nuestra red local (a los que protegemos), se conectan aparentemente a Internet, pero realmente lo hacen hacia nuestro programa proxy. La función de éste es básicamente conectarse a Internet en beneficio de los usuarios internos que lo solicitan. El resultado es que los posibles atacantes observarán al proxy conectándose, pero no podrán acceder a la red interna. Obviamente nos aseguraremos que el proxy esté bien seguro. Esto es más sencillo que cuidar a cada estación con usuarios incautos y sus sistemas operativos inseguros.

En resumen: cuando sea posible, use proxy para dar acceso a Internet para la red interna.

Los proxys pueden presentar otras ventajas, como ahorro del ancho de banda y mayor velocidad de acceso a las páginas web más populares (proxy-caché.)

En [21] se puede leer una revisión general acerca de la seguridad de los sistemas en Internet.

18. Uso de red perimétrica o zona desmilitarizada

Si Ud. tiene servidores que dan cara a Internet, entonces deberá permitir ciertas conexiones de el exterior a sus servidores. Esto es problemático y debe ser controlado.

Muchos especialistas recomiendan mantener estos servidores protegidos mediante firewalls, pero a la vez separados de la red interna donde se encuentra, por ejemplo, nuestra base de datos. La idea es que estos servidores están en una zona de peligrosidad intermedia protegidos de Internet, pero no al nivel de nuestro sistema más interno. Es por eso que se les suele asignar una red especial independiente denominada "perimétrica" o zona desmilitarizada (DMZ) con la intención de reducir la vulnerabilidad de nuestros datos más importantes mediante el alejamiento de los computadores que son blanco directo de las conexiones exteriores.

19. Prácticas de programación segura

Si su organización desarrolla software de cualquier tipo, y en especial, si este software dará cara a Internet, entonces Ud. debería asegurarse de que los desarrolladores conozcan las recomendaciones de seguridad correspondientes al lenguaje de programación o herramienta de desarrollo empleada así como del ambiente (sistema operativo o red) en que se ejecutan. De igual modo los diseñadores de la arquitectura deben considerar la seguridad de la misma desde el principio del proyecto [22],[23],[24]. Los programas que dan cara a Internet son un caso muy especial [25] donde la seguridad reviste aún más importancia a juzgar por las malas experiencias producidas.

20. Vigilancia

La vigilancia del buen funcionamiento del sistema es un asunto más complicado de lo que parece. El problema es que los ataques frecuentemente están disfrazados de conexiones más o menos válidas, y por otro lado, los sistemas de cómputo normalmente no avisan cuando son alterados, a no ser que esto se haya establecido de antemano. Los ataques generalmente tratan de aparentar que no ha ocurrido nada, a fin de conseguir hacer más y más modificaciones sin ser detectados y detenidos.

Todo esto obliga a considerar de antemano ciertos indicadores que nos ayuden a determinar si un sistema ha sido comprometido o está en proceso de serlo. Esto en ciertos casos puede requerir mucha sagacidad y un profundo conocimiento de los mecanismos internos del sistema, así como la activación de una mayor descripción de eventos. Pero generalmente este análisis no se pone en práctica hasta que los síntomas aparecen y el ataque ya está muy avanzado, lo que ha motivado el incremento de productos que intentan adelantarse y dar aviso cuando algo anda mal. A estos sistemas se les conoce como sistemas de detección de intrusiones (IDS o NIDS.) El administrador precavido hará bien en considerarlos como parte de las medidas de seguridad [26],[27],[28].

21. Establecimiento de políticas

Para terminar, una recomendación que en cierto modo engloba a todas las anteriores. El establecimiento de políticas corresponde a un mecanismo que permite asegurar que la seguridad se mantenga en todas las situaciones y se deriva

del "compromiso con la seguridad" de la organización. La idea detrás de todo esto es que nadie puede saber de antemano lo que piensa el administrador o el encargado de la seguridad sin ser informado. Muchas organizaciones no tienen esto en cuenta y se da el caso en que un gerente se limita a reunir a los empleados y decirles "no hagan nada que pueda atentar contra la seguridad de la organización, o serán castigados ..." El problema es que la gente normalmente no piensa en términos de seguridad sino en términos de cumplimiento de obligaciones y logro de resultados, y el camino más corto no siempre es el más seguro.

Las políticas justamente intentan abordar estos problemas mediante el establecimiento de normas que han de cumplir todos los miembros de la organización. Las políticas son documentos destinados a personas con responsabilidades claramente definidas y detallan cuidadosamente su alcance y aplicación. Explican tanto procedimientos informáticos como logísticos en términos de la jerarquía de la organización [29],[30].

Una ventaja colateral es la posibilidad de identificar responsables en caso de ataques exitosos (¡y así salvar su cabeza el administrador!)

El cumplimiento de las políticas pasa por hacer tomar consciencia de su importancia a los destinatarios, para lo cual es imprescindible que su aplicación sea impulsada desde el más alto nivel jerárquico. Es recomendable, por ejemplo, que sea leída y firmada a modo de compromiso por parte del destinatario, y su difusión puede estar acompañada por charlas informativas por parte del administrador o del staff responsable.

La actualización responsable de las políticas es crítica para su cumplimiento, por lo que en ciertos casos se indica la designación de un responsable de las mismas o incluso de un equipo responsable, especialmente durante su elaboración y establecimiento inicial.

22. Conclusión

Brevemente hemos pasado revista a un conjunto de recomendaciones genéricas de seguridad que pueden ser aplicadas prácticamente a cualquier red de computadores. No pretendo que sea un listado exhaustivo, y de hecho existen diversas publicaciones especializadas que sin duda trascienden lo expuesto aquí. Tan sólo he pretendido presentar un resumen de lo que a mi juicio es lo más importante, a partir de mi experiencia con sistemas Linux en Internet.

La seguridad de la red requiere ir más allá de lograr que los sistemas simplemente funcionen bien. Se requiere creatividad (ser un cerrajero original), ciertas aptitudes de legislador (para proponer políticas adecuadas), así como un considerable conocimiento de la tecnología involucrada. Una dosis de paranoya puede ser indispensable [31].

En Internet se puede encontrar numerosas listas con "los pasos" a llevarse a cabo para asegurar la red que pueden resultar útiles al lector [32],[33]. Por otro lado, cada plataforma cuenta con muchas recomendaciones de seguridad específicas que es menester conocer. Diversos portales especializados en seguridad proporcionan una fuente invaluable de información que los administradores deberían aprovechar [2],[34],[35].

23. Agradecimiento

En el Internet hay muchos héroes anónimos o casi anónimos que presentan buenas y malas ideas con la única intención de contribuir a la seguridad de este relativamente nuevo medio de comunicación. Sus diversos materiales son los que me permitieron efectuar esta pequeña síntesis, algunos de los cuales cito como referencia, aunque otros ya no son ubicables.

24. Referencias

[1] El software se torna más inseguro y complejo

http://www.counterpane.com/crypto-gram-0003.html#SoftwareComplexityandSecurity

[2] El CERT/CC

http://www.cert.org/

[3] HTTP-Diego

http://www.compulinux.com/diego

[4] 5 Steps to enterprise security

http://www.eweek.com/category/0,3660,s=25132,00.asp

[5] Extenso documento sobre administración de riesgos

http://cisac.stanford.edu/docs/soohoo.pdf

[6] VPN Source Page

http://www.internetweek.com/VPN/default.html

[7] VPN Labs

http://www.vpnlabs.org/

[8] Cryptogram newsletter

http://www.counterpane.com/crypto-gram.html

[9] OpenSSH Home

http://www.openssh.org/

[10] S/Key Introduction

http://www.surfnet.nl/innovatie/surf-ace/security/doc/skey.html

[11] Kerberos

http://web.mit.edu/kerberos/www/

[12] El enemigo más peligroso está en el interior

http://oraclepressoffice.bulletonline.com/showrelease.php?id=128

[13] Outsourcing para auditorías de seguridad

http://www.networknews.co.uk/Analysis/1129412

[14] Consecuencias de no aplicar parches

http://zdnet.com.com/2100-11-527502.html?legacy=zdnn

[15] Securityfocus Mailing lists:

http://online.securityfocus.com/archive

[16] Cert Mailing lists:

http://www.cert.org/contact_cert/certmaillist.html

[17] The Art of port scanning

http://www.dlhoffman.com/publiclibrary/software/nmap_doc.html

[18] Deny Of Service resources

http://www.denialinfo.com/

[19] Distributed Deny Of Service resources

http://staff.washington.edu/dittrich/misc/ddos/

[20] Overview of Scans and DDos Attacks

http://www.nipc.gov/ddos.pdf

[21] La seguridad en Internet

http://www.cert.org/encyc_article/tocencyc.html

[22] Shmoo: Cómo escribir código seguro

http://www.shmoo.com/securecode/

[23] (Libro) John Viega and Gary McGraw: Building Secure Software
http://cseng.aw.com/book/0,3828,020172152X,00.html

[24] Buffer overflows:
http://www.enseirb.fr/~glaume/bof/report.html

[25] IBM: Programación segura de CGI
http://www-106.ibm.com/developerworks/library/secure-cgi/

[26] Previniendo y detectando ataques con IDS's
http://online.securityfocus.com/infocus/1558

[27] Snort: Network instrusion detection system
http://www.snort.org/

[28] Tripwire: Host based IDS
http://www.tripwire.org/

[29] Recursos sobre políticas de seguridad
http://secinf.net/ipolicye.html

[30] Sobre las políticas de seguridad
http://www.sans.org/newlook/resources/policies/policies.htm

[31] "Lecciones" de seguridad tras el 9/11
http://www.counterpane.com/crypto-gram-0203.html#7

[32] Sysadmin's Security Basics
http://linux.oreillynet.com/lpt/a/linux/2001/10/18/basics.html

[33] SANS: Ten days to network security
http://rr.sans.org/securitybasics/10days.php

[34] Recursos de seguridad
http://netsecurity.about.com/

[35] Recursos de seguridad (centrado en Unix)
http://www.alw.nih.gov/Security/Docs/network-security.html