

Conociendo al Enemigo

EL ATACANTE INFORMÁTICO

Protocolos de Comunicación
Ambientes Operativos
Buffer Overflow
DoS
Exploits

CAPÍTULO 3

AMBIENTES OPERATIVOS

Rookits

Virus

Criptografía

Metodologías y Estándares



Jhon Cesar Arango Serna

www.itforensic-la.com

CAPITULO 3

AMBIENTES OPERATIVOS

LINUX/UNIX

Para entender la numerosa función de los ataques, usted debe tener una comprensión básica del sistema operativo LINUX, debido a su popularidad es una plataforma ideal para servidores y como un sistema operativo para lanzar ataques. Este capítulo presenta una apreciación global del sistema operativo de LINUX y describe conceptos subyacentes que se exigen entender los diversos ataques explicados a lo largo del libro.

UNIX es una bestia bonita pero extraña, introducido hace más de 30 años como un proyecto de investigación a AT&T, el sistema operativo de UNIX se usa ampliamente a lo largo del mundo en los servidores y sistemas de estación de trabajo.

Mucho del Internet se construyo con UNIX, en recientes años, el proyecto de código abierto de LINUX (como OpenBSD, GNU/Linux, y otros) ha ayudado a llevar LINUX al escritorio (desktop) e incluso a los dispositivos móviles, LINUX es muy poderoso. Son miles las personas que han trabajado en LINUX en vías de su desarrollo durante los últimos años, han perfeccionado rutinas y han creado numerosas herramientas.

Muchos sistemas LINUX tienen gran fiabilidad, alto rendimiento y rasgos de seguridad fuertes. Dado a los aportes de una comunidad global que utilizan este sistema como una herramienta de investigación. Gracias a Esto y a su relación íntima con el Internet, papel crítico del software libre y los movimientos de código abiertos, los administradores del sistema pueden encontrar una variedad de herramientas libremente disponible en Internet y pueden hacer preguntas a una comunidad grande y relativamente amistosa de LINUX a través de listas de correo grupos de noticias.

Pero LINUX es también una bestia extraña, por dos razones en particular. Primero: No hay un solo sistema operativo llamado LINUX. En cambio, LINUX es una familia de sistemas operativos puesta al día por muchos distribuidores que compiten entre ellos, cada uno con metas y visiones diferentes. Varias variantes populares de LINUX/UNIX incluyen:

- Solaris de Sun Microsystems.
- HP-UX (11iv3) by Hewlett Packard.
- IRIX de Sgi (Su nuevo nombre es Silicon Graphics).
- AIX de IBM.
- SCO, de Santa Cruz Operation.

- BSD de BSDi.
- FreeBSD una versión freeware de BSD.
- OpenBSD, otra versión gratis de BSD el cual se ha catalogado como el systema operativo más seguro.
- Linux, una fuente abierta de UNIX creada por Linux Torvalds, disponible para descarga libre y distribuida comercialmente por una variedad de vendedores en la que se incluye: Ubuntu, OpenSuse, Fedora, Centos, Mandriva, Debian, Linux Mint, Slackware, Gentoo y FreeBsd, entre otras.
- SunOS, el sistema operativo más viejo de Sun Microsystems.

ESCOGIENDO UNA DISTRIBUCIÓN DE LINUX¹



Un paso crucial a la hora de construir un servidor Linux seguro, está en la selección de la distribución que beneficiara las necesidades perfiladas en su política de seguridad.

Esta decisión determina el éxito de mantener un servidor de Linux seguro. Pero qué hace una distribución exactamente "Segura"? A continuación encontrará un juego de criterios que debe considerar antes de tomar esta decisión:

- ¿Tiene el distribuidor un mecanismo muy conocido para informar de las vulnerabilidades de seguridad encontradas en su distribución?
- ¿El distribuidor hace públicas las advertencias de seguridad que indica a los usuarios de vulnerabilidades encontrado en su distribución?
- ¿Qué tan a menudo resuelve los problemas de seguridad se resueltos encontrados?
- ¿El distribuidor posee un sitio especializado Web o FTP donde reside la información de seguridad (parches, actualizaciones de seguridad, etc.)?

¹ <http://tuela.com/wp-content/uploads/2009/10/distros.jpg>

- ¿Qué tan a menudo el distribuidor arroja las versiones revisadas de la distribución en uso? Es probable que el horario en que se permiten las descargas sea tan lento que prolongue la exposición de vulnerabilidades conocidas.
- ¿Ofrece el distribuidor un método fácil de usar para instalar y poner al día paquetes del software?
- ¿El vendedor abre el código fuente de sus productos a la comunidad global para mejorar seguridad de Linux y/o las herramientas?
- ¿Desde cuando existe el Distribuidor?
- ¿Las versiones anteriores han sido bien soportadas y seguras?

Como con cualquier otra opción de software de Linux no hay ningún ganador, pero usted puede tomar la decisión más educada teniendo en cuenta las preguntas anteriores.

A continuación miraremos algunas distribuciones de Linux examinando un poco sus rasgos de seguridad.



Para muchos el nombre de Red Hat equivale a Linux, ya que Probablemente se trata de la compañía de linux más popular del mundo. Fundada en 1995 por Bob Young y Marc Ewing, red Hat Inc solo ha mostrado beneficios recientemente gracias a otros servicios en lugar de la distribución en si. Aun y así, Red Hat es la primera elección para muchos profesionales y parece que seguirá siendo un peso pesado durante mucho tiempo. Ha podido consagrar recursos considerables al rastrear, diseminar y la resolver fallas de seguridad en los paquetes que distribuye. A la fecha se puede considerar como el vendedor de Linux más exitoso.

Los paquetes no son los más actuales, una vez se anuncia una nueva versión beta, las versiones de los paquetes se mantienen, excepto para actualizaciones de seguridad. Como resultado se obtiene una distribución bien probada y estable. El programa de betas y las facilidades para enviar fallos están abiertas al público y hay un gran espíritu en las listas de correo.

Su manejador de paquetes (RPM) permite la actualización constante sobre los programas donde se halla encontrado y resuelto problemas de seguridad. Esto se recomienda hacerlo una vez cada mes, siempre y cuando la actualización no sea crítica en cuyo caso debe hacerse inmediatamente. Todo esto con el fin de garantizar la seguridad de su sistema y el de estar corriendo siempre la versión reciente.

Actualmente Red Hat ha dividido el negocio en dos áreas distintas, por una parte promociona el proyecto Fedora para usuarios finales, el cual saca tres versiones al año, manteniendo los paquetes de Red Hat para usuarios corporativos, que se mantienen más tiempo, y garantizan su estabilidad.

Red Hat Linux se ha convertido en la distribución linux dominante en servidores en todo el mundo.. Otra de las razones del éxito de Red Hat es la gran variedad de servicios populares que ofrece la compañía. Los paquetes de software son fácilmente actualizables usando la Red Hat Network, un repositorio oficial de software e información. Una larga lista de servicios de soporte son accesibles en la compañía y, aunque no siempre baratos, tienes virtualmente asegurado un excelente soporte de personal altamente cualificado. La compañía ha desarrollado incluso un programa de certificación para popularizar su distribución, el RHCE (Certificado de Ingeniería de Red Hat), academias y centros examinadores están disponibles en el casi todas partes del mundo.



Debian GNU/Linux inició su andadura de la mano de Ian Murdock en 1993. Debian es un proyecto totalmente no-comercial; posiblemente el más puro de los ideales que iniciaron el movimiento del software libre. Cientos de desarrolladores voluntarios de alrededor del mundo contribuyen al proyecto, que es bien dirigido y estricto, asegurando la calidad de una distribución conocida como Debian. En cualquier momento del proceso de desarrollo existen tres ramas en el directorio principal: "estable", "en pruebas" e "inestable" (también conocida como "sid").

Cuando aparece una nueva versión de un paquete, se sitúa en la rama inestable para las primeras pruebas, si las pasa, el paquete se mueve a la rama de pruebas, donde se realiza un riguroso proceso de pruebas que dura muchos meses. Esta rama solo es declarada estable tras una muy intensa fase de pruebas.

Como resultado de esto, la distribución es posiblemente la más estable y confiable, aunque no la más actualizada. Mientras que la rama estable es perfecta para servidores con funciones críticas, muchos usuarios prefieren usar las ramas de pruebas o inestable, más actualizadas, en sus computadores personales. Debian es también famosa por su reputación de ser difícil de instalar, a menos que el usuario tenga un profundo conocimiento del hardware de la computadora. Compensando este fallo está "apt-get" un instalador de paquetes Debian.



SuSE es otra compañía orientada a los escritorios, aunque variedad de otros productos para empresas están disponibles. La distribución ha recibido buenas críticas por su instalador y la herramienta de configuración YaST, desarrollada por los desarrolladores de la propia SuSE. La documentación que viene con las versiones comerciales, ha sido repetidas veces evaluada como la más completa, útil y usable con diferencia a la de sus competidores. SuSE Linux 7.3 recibió el premio "Producto del año 2001" que entrega el Linux Journal. La distribución tiene un gran porcentaje de mercado en Europa y América del norte, pero no se vende en Asia y otras partes del mundo.

El desarrollo de SuSE se realiza completamente a puerta cerrada, y no se lanzan betas públicas para probar. Siguen la política de no permitir descargar el software hasta tiempo después de que salgan a la venta las versiones comerciales. A pesar de todo, SuSE no entrega imágenes ISO de fácil instalación de su distribución, usando el software empaquetado para la gran mayoría de su base de usuarios.

Novell ha comprado a esta compañía, y esta haciendo inversiones importantes en mantener y desarrollar esta distribución, a nivel corporativo, pero sin olvidarse del usuario final (Compra de Ximan, y la reciente liberación del instalador YaST bajo licencia GPL), y ha seguido la misma estrategia que Redhat, dejando SuSE para SOHO (Small Office, home, en español, pequeñas oficinas y usuarios domésticos), y creando una distribución para entornos empresariales (Novell Linux Desktop para escritorio, basada en gnome, y Novell Open Enterprise Server, para servidores).

ARQUITECTURA

SISTEMAS DE ARCHIVOS

Aunque los discos duros pueden ser muy chicos, aún así contienen millones de bits, y por lo tanto necesitan organizarse para poder ubicar la información. Éste es el propósito del sistema de archivos.

El sistema de archivos nativo de Linux es el EXT2. Ahora proliferan otros sistemas de archivos con journalising (si se arranca sin haber cerrado el sistema, no necesitan hacer un chequeo sino que recuperan automáticamente su último estado), los más conocidos son EXT3, EXT4, ReiserFS, Entre Otros

EXT2 (SECOND EXTENDED FILESYSTEM)

El sistema de ficheros EXT2 fue desarrollado originalmente por Remy Card quien es un programador y desarrollador de origen Francés el cual ha aportado mucha de su investigación al proyecto GNU/Linux. Particularmente Remy Card desarrolló el sistema de ficheros ext2 para los sistemas operativos RedHat, Fedora y Debian, Este sistema de ficheros tiene un tipo de tabla FAT de tamaño fijo, donde se almacenan los inodos.

Los inodos son una versión muy mejorada de FAT, donde un puntero inodo almacena información del archivo (ruta o path, tamaño, ubicación física). En cuanto a la ubicación, es una referencia a un sector del disco donde están todos y cada una de las referencias a los bloques del archivo fragmentado. Estos bloques son de tamaño especificable cuando se crea el sistema de archivos, desde los 512 bytes hasta los 4 kB, lo cual asegura un buen aprovechamiento del espacio libre con archivos pequeños. Los límites son un máximo de 2 TB de archivo, y de 4 TB de partición.

EXT3 (THIRD EXTENDED FILESYSTEM)

La principal diferencia de EXT2 con EXT3 es que EXT3 dispone de un registro por diario o mayormente conocido como “journaling”. Así mismo EXT3 puede ser montado y usado como un sistema de archivos EXT2. Otra diferencia importante es que EXT3 utiliza un árbol binario balanceado (árbol AVL) e incorpora el asignador de bloques de disco.

Aunque su velocidad y escalabilidad es menor que sus competidores, como JFS, ReiserFS o XFS, tiene la ventaja de permitir actualizar de EXT2 a EXT3 sin perder los datos almacenados ni formatear el disco y un menor consumo de CPU.

El sistema de archivo EXT3 agrega a EXT2 lo siguiente:

- Registro por diario.
- Índices en árbol para directorios que ocupan múltiples bloques.
- Crecimiento en línea.

EXT4 (FOURTH EXTENDED FILESYSTEM)

Ext4 es un sistema de archivos con bitácora (en inglés: Journaling) que fue concebida como una mejora compatible de ext3. Ext4 fue publicado como estable el 25 de diciembre de 2008 en la versión 2.6.28 del núcleo Linux y desde entonces se encuentra disponible para el uso en sistemas de producción.

El sistema de archivos ext4 es una notable mejora sobre ext3 mucho más de la que fue ext3 sobre ext2. La mayor mejora del sistema de archivos ext3 sobre ext2 fue añadir el soporte de journaling (bitácoras). En cambio ext4 modifica importantes estructuras de datos del sistema de archivo tales como aquellas destinadas a almacenar los archivos de datos. El resultado es un sistema de archivos con un mejorado diseño, mejores características, rendimiento y confiabilidad.

Características principales

- Soporte de volúmenes de hasta 1 exabyte (260 bytes) y archivos con tamaño hasta 16 terabytes.
- Capacidad de reservar un área contigua para un archivo denominada "extents", la cual puede reducir y hasta eliminar completamente la fragmentación de archivos.
- Menor uso del CPU.
- Mejoras en la velocidad de lectura y escritura.

Actualmente, el ext4 es compatible con su anterior versión, el ext3, esto quiere decir que se puede montar como una partición ext3. También se pueden montar las particiones ext3 como ext4, aunque, si la partición ext4 usa extent (una de las mayores mejoras), la compatibilidad con la versión anterior, y por lo tanto, montar la partición como ext3, no es posible. La opción extent no es usada por defecto.

HPFS (HIGH PERFORMANCE FILE SYSTEM)

Fue creado específicamente para el sistema operativo OS/2 para mejorar las limitaciones del sistema de archivos FAT. Fue escrito por Gordon Letwin y otros empleados de Microsoft, y agregado a OS/2 versión 1.2, en esa época OS/2 era todavía un desarrollo conjunto entre Microsoft e IBM.

Se caracteriza por permitir nombres largos, metadatos e información de seguridad, así como de autocomprobación e información estructural. Otra de sus características es que, aunque poseía tabla de archivos como FAT, ésta se encontraba posicionada físicamente en el centro de la partición, de tal manera que redundaba en menores tiempos de acceso a la hora de leerla o

escribirla.

REISERFS

ReiserFS es un sistema de archivos de propósito general, diseñado e implementado por un equipo de la empresa Namesys, liderado por Hans Reiser. Actualmente es soportado por Linux y existen planes de futuro para incluirlo en otros sistemas operativos. También es soportado bajo windows de forma no oficial, aunque por el momento de manera inestable y rudimentaria. A partir de la versión 2.4.1 del núcleo de Linux, ReiserFS se convirtió en el primer sistema de ficheros con journal en ser incluido en el núcleo estándar. También es el sistema de archivos por defecto en varias distribuciones, como SuSE (excepto en openSuSE 10.2 que su formato por defecto es ext3), Xandros, Yoper, Linspire, Kurumin Linux, FTOSX, Libranet y Knoppix.

Con la excepción de actualizaciones de seguridad y parches críticos, Namesys ha cesado el desarrollo de ReiserFS (también llamado reiser3) para centrarse en Reiser4, el sucesor de este sistema de archivos.

ReiserFS ofrece funcionalidades que pocas veces se han visto en otros sistemas de archivos:

- **Journaling.** Esta es la mejora a la que se ha dado más publicidad, ya que previene el riesgo de corrupción del sistema de archivos.
- **Reparticionamiento con el sistema de ficheros montado y desmontado.** Podemos aumentar el tamaño del sistema de ficheros mientras lo tenemos montado y desmontado (online y offline). Para disminuirlo, únicamente se permite estando offline (desmontado). Namesys nos proporciona las herramientas para estas operaciones, e incluso, podemos usarlas bajo un gestor de volúmenes lógicos como LVM o EVMS.
- **Tail packing,** un esquema para reducir la fragmentación interna comparado con EXT2 y EXT3 en el uso de archivos menores de 4k, ReiserFS es normalmente más rápido en un factor de 10–15. Esto proporciona una elevada ganancia en las news, como por ejemplo Usenet, caches para servicios HTTP, agentes de correo y otras aplicaciones en las que el tiempo de acceso a ficheros pequeños debe ser lo más rápida posible.

Algunas de las desventajas de ReiserFS son:

- Los usuarios que usen como sistema de ficheros ext2, deben formatear sus discos, aunque no así los que usen ext3.
- ReiserFS en versiones del kernel anteriores a la 2.4.10 se considera inestable y no se recomienda su uso, especialmente en conjunción con NFS
- Algunas operaciones sobre archivos no son síncronas bajo ReiserFS, lo que pueden causar comportamientos extraños en aplicaciones fuertemente basadas en locks de archivos.
- No se conoce una forma de desfragmentar un sistema de archivos ReiserFS, aparte de un volcado completo y su restauración.
- Tempranas implementaciones de ReiserFS (anteriores a la incluida en el kernel 2.6.2), eran susceptibles de problemas de escrituras fuera de orden, lo que provocaba que archivos siendo escritos durante una caída del sistema, ganaran un pico de bytes extras de basura en el siguiente montado del sistema de archivos. La implementación actual de journaling, es correcta en este aspecto, manteniendo el journaling ordenado, del estilo de ext3.

ZFS (ZETTABYTE FILE SYSTEM)

Es un sistema de ficheros desarrollado por Sun Microsystems para su sistema operativo Solaris. El significado original era “Zettabyte File System”, pero ahora es un acrónimo recursivo.

El anuncio oficial de ZFS se produjo en Septiembre del 2004. El código fuente del producto final se integró en la rama principal de desarrollo de Solaris el 31 de octubre del 2005 y fue lanzado el 16 de noviembre de 2005 como parte del build 27 de OpenSolaris.

ZFS fue diseñado e implementado por un equipo de Sun liderado por Jeff Bonwick. ZFS destaca por su gran capacidad, integración de los conceptos anteriormente separados de sistema de ficheros y administrador de volúmenes en un solo producto, nueva estructura sobre el disco, sistemas de archivos ligeros, y una administración de espacios de almacenamiento sencilla.

XFS

XFS es un sistema de archivos de 64 bits con journaling de alto rendimiento creado por SGI (antiguamente Silicon Graphics Inc.) para su implementación de UNIX llamada IRIX. En mayo del 2000, SGI liberó XFS bajo una licencia de código abierto.

XFS se incorporó a Linux a partir de la versión 2.4.25, cuando Marcelo Tosatti (responsable de la rama 2.4) lo consideró lo suficientemente estable para incorporarlo en la rama principal de desarrollo del kernel. Los programas de instalación de las distribuciones de SuSE, Gentoo, Mandriva, Slackware, Fedora Core, Ubuntu y Debian ofrecen XFS como un sistema de archivos más. En FreeBSD el soporte para solo lectura de XFS se añadió a partir de Diciembre de 2005 y en Junio de 2006 un soporte experimental de escritura fue incorporado a FreeBSD7.0CURRENT.

SWAP

La swap es un espacio reservado en tu disco duro para poder usarse como una extensión de memoria virtual de tu sistema. Es una técnica utilizada desde hace mucho tiempo, para hacer creer a los programas que existe más memoria RAM de la que en realidad existe. Es el propio sistema operativo el que se encarga de pasar datos a la swap cuando necesita más espacio libre en la RAM y viceversa.

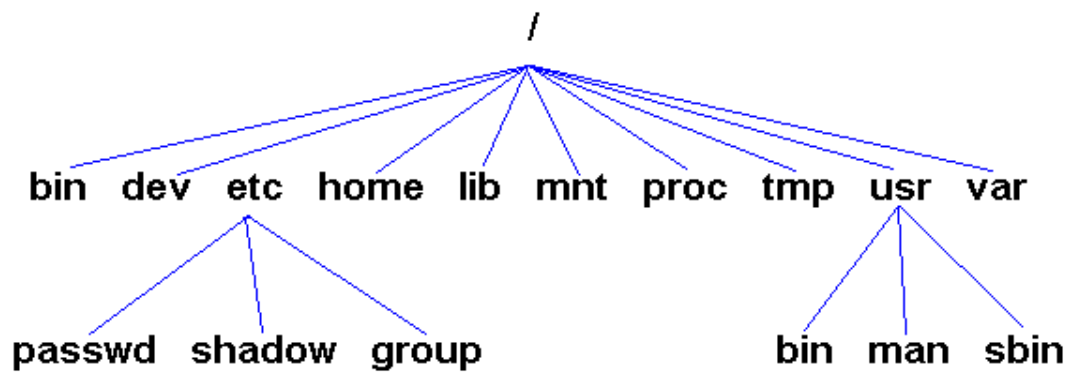
En Linux, la memoria total disponible por el sistema está formada por la cantidad de memoria RAM instalada + la swap disponible.

El acceso a la swap (disco duro) es más lento que el acceso a la memoria RAM, por lo que si nuestro ordenador está muy cargado de trabajo y hace un uso intensivo de la swap, la velocidad del sistema disminuirá. Un uso muy intensivo y continuado de la swap es un indicativo de que necesitamos más memoria en nuestro sistema para que funcione desahogado con el uso que le estamos dando.

En linux generalmente se usa como minimo una partición dedicada a swap (aunque también se puede tener un fichero swap).

ESTRUCTURA DEL SISTEMA DE ARCHIVOS

Todo alrededor de LINUX o de UNIX, se basa en la estructura del Sistema de Archivos (File System) ya que todos se trata como un archivo: Los procesos, los dispositivos y los archivos como tal. Explorar el sistema de archivos de LINUX es como viajar a través de una ciudad, con diversos directorios actuando como las calles para conducirlo a los edificios, que son archivos individuales. Aunque son muchas las versiones de LINUX/UNIX el siguiente esquema representa la estructura general de estas versiones:



El primer nivel es conocido como el directorio “Raíz” (Root), simplemente porque a partir de esta ubicación se encuentran el resto de los directorios que están bajo ella. El directorio raíz convenientemente es nombrado "/". (Se llega con el comando `cd /`). A partir de este directorio se visualizan otros directorios que contienen la información del sistema que incluye configuración del sistema, ejecutables del sistema, datos del usuario entre otros. La siguiente tabla muestra el significado de cada uno de estos directorios:

DIRECTORIO	PROPOSITO
/	Es el directorio Raíz.
/bin y/o /sbin)	Contiene ejecutables críticos necesarios para el arranque del sistema.
/dev	Contiene todos los dispositivos del sistema (terminales, unidades de disco, modem, etc.)
/etc	Contiene los archivos de configuración del sistema, incluyendo grupos y usuarios.
/home	Aquí se localiza el directorio de los usuarios.
/lib	Aquí reside las librerías compartidas de los programas.
/mnt y/o /media	Es el punto donde se almacena temporalmente otro sistema de archivos exportado, como Usb, CD Rom, discos esclavos, Etc.
/proc	Imágenes de los procesos que actualmente se estan corriendo en el sistema.
/tmp	Archivos, temporales que son eliminados cada vez que se enciende el equipo.
/usr	Una variedad de archivos críticos, incluye: utilidades estándar del sistema (/usr/bin), manuales (/usr/man), encabezados de programas de C (/usr/include) y ejecutables de administración (/usr/sbin).

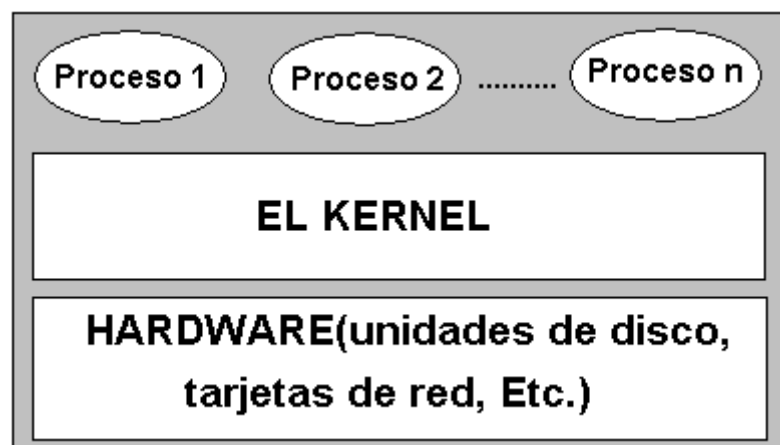
/var	Directorio donde se almacena los log's del sistema, para luego ser usados en cuestiones de administración. Incluye: registro de las acciones sobre el sistema, paginas web visitadas, correos enviados, entre otras.
------	--

EL KERNEL Y LOS PROCESOS

Los sistemas de UNIX y/o LINUX tienden a tener una arquitectura modular, con un corazón central y varios programas alrededor de dicho corazón. En una máquina de UNIX/LINUX, el programa especial en el corazón del sistema se llama, “Kernel”. El Kernel es el corazón y cerebro del sistema y controla las funciones críticas del sistema, como las interacciones con el hardware y la administración de los recursos de los diferentes usuarios. Cuando un programa corre, necesita acceder a componentes de hardware como discos, cintas o interfaces de red, el Kernel proporciona las funciones necesarias para acceder a este hardware. Cuando un programa se ejecuta en un sistema UNIX/LINUX, el Kernel lanza un proceso para ejecutar dicho programa. Un proceso contiene el código ejecutable del programa que se esta corriendo y la memoria que le fue asociada. Programas de usuario, herramientas administrativas, y incluso algunos servicios (como servidores Web o servidores de Correo) son procesos en la máquina.

Un sistema UNIX/LINUX tiene a menudo centenares o incluso miles de procesos activos en cualquier momento dado. Sin embargo, en la unidad central de proceso (CPU) sólo un proceso puede correr en cualquier momento dado. El Kernel hace malabares en la CPU entre todos los procesos activos fijando cuando debe de correr cada uno para que el procesador del sistema pueda compartirse entre los procesos.

Adicionalmente, el kernel asigna cuidadosamente y maneja la memoria usada por los procesos. Cada proceso tiene su propio set de memoria limitada, y el Kernel impide a un proceso acceder a la memoria usada por otro proceso. Con esta capacidad de protección de memoria, un proceso renegado que intente leer o borrar la memoria de otro proceso será detenido por el kernel.



Muchos procesos en un sistema UNIX/LINUX corren sin que el usuario se de cuenta de su existencia, pero son procesos que manejan la información crítica del servidor, como un spool de paginas a ser enviadas a la impresora, la búsqueda efectiva de paginas Web, capacidades de asignación dinámica de direcciones de red. Estos procesos, que prestan un servicio a los usuarios y que normalmente se inician automáticamente una vez de enciende el sistema son conocidos como demonios “daemons”. Se les da este nombre basados en la función que ellos realizan, se reconocen por tener una “d” al final del nombre el proceso. Por ejemplo, httpd es un demonio para el servidor Web que permite a los clientes visualizar paginas Web, demonios por el estilo están sshd, ftpd, xinitd, entre otros.

PONIENDO EN MARCHA PROCESOS AUTOMÁTICAMENTE

Todos los procesos que corren en un sistema de UNIX/LINUX, desde el poderoso servidor Web hasta el humilde generador de caracteres, tienen que ser activados por el kernel o algún otro proceso que active su funcionamiento. Durante el encendido del sistema, el Kernel activa un demonio llamado “init”, que es el padre de otros procesos que corren sobre la máquina.

El trabajo de “init” es terminar la carga completa del sistema, ejecutando una variedad de procesos que lo complementan. Dependiendo de las versiones de Unix o de Linux varia la ubicación de estos Scripts que se pueden encontrar en la siguiente ruta: /etc/init.d y /etc/rc.d .

el “init” también empieza una serie de procesos asociado a los servicios de red. Estos activan demonios los cuales escuchan sobre que puerto entra el trafico, y actúan recíprocamente con los usuarios. Algunos de los servicios de red más comunes iniciados por demonios a través del “init” incluyen:

Httpd:	Un servidor Web, manejando HTTP o demandas de HTTPS.
Sendmail:	Una aplicación común de UNIX/LINUX de un servidor de correo electrónico.
NFS:	Sistemas de Archivos de Red, originalmente creado por Sun Microsystems, usado para compartir archivos entre los sistemas UNIX.

Cuando el “init” empieza a trabajar conectado a una red, el proceso asociado con el servicio escucha al trafico de la red entrante. Por ejemplo, la mayoría los servidores de Web escuchan en el puerto TCP 80, mientras los servidores de correo electrónico escuchan por el puerto TCP 25.

Algunos servicios de red, como Web, correo y el compartir archivos, normalmente tienen mucho tráfico entrante, para lo cual necesitan estar constantemente listos para manipular la demanda que ello implica, este tipo de procesos son cobijados por el “init”.

Otros servicios, como Ftp, no son de acceso frecuente, solo esperan a ser solicitados por algún cliente de la red. Este tipo de procesos son llamados Demonios de Internet o initd, que tiene como fin esperar las solicitudes de servicios que son pedidos frecuentemente.

Dependiendo de las distribuciones de Linux, usted podrá encontrar un archivo de configuración llamado inetd.conf, el cual contiene todos los demonios de Internet que debe tener en cuenta en caso de una solicitud o puede encontrar un archivo llamado xinetd.conf el cual incluye un directorio llamado xinetd.d, donde se crea un archivo por cada servicio que se desee habilitar.

Inetd es activado por el demonio de init durante el encendido del sistema. Una vez activado, el inetd consulta su archivo de configuración, localizado en el directorio de /etc el cual es llamado inetd.conf o xinetd.conf. Este archivo de configuración dice al inetd que tráfico debe escuchar de acuerdo a ciertos servicios. Los puertos TCP y UDP para estos servicios son definidos en el archivo /etc/services el cual simplemente contiene un nombre de servicio, número del puerto, y indicación si un servicio es TCP (Orientado a Conexión) o UDP (No orientado a Conexión).

Cuando el tráfico llega a la máquina destino solicitando un servicio específico que esta identificado en /etc/inetd.conf o /etc/xinetd.conf, el inetd activa el programa asociado con el servicio. El proceso del servicio solicitado manipula el tráfico y se detiene una vez finaliza la solicitud. Inetd continúa esperando más tráfico para ese servicio y otros.

Se activan numerosos servicios normalmente usando inetd entre los cuales se incluyen:

Echo:	Un servicio para replicar los caracteres enviados por la red.
Chargen:	Un servicio que genera una lista repetida de caracteres.
Ftpd:	El demonio para la transferencia de archivos.

Para hacer que inetd escuche cierto servicio en particular, se debe especificar en el archivo de configuración /etc/inetd.conf o en el directorio de /etc/xinetd.d para el caso de /etc/xinetd.conf. A continuación se explica las características que debe tener estos servicios en los casos xinetd.conf ya que actualmente es el más utilizado:

Xinetd.conf

Este archivo en algunas distribuciones de Linux, reemplazo al inetd.conf y se caracteriza por su potencia en opciones y servicios. A diferencia de inetd.conf este archivo normalmente tiene un directorio asociado donde se crea un archivo por cada servicio que se desee habitar. El siguiente es un ejemplo de xinetd.conf (recuerde que las líneas que empiezan por #, no se toman en cuenta):

```
#
# Simple configuration file for xinetd
#
# Some defaults, and include /etc/xinetd.d/

defaults
{
    instances                = 60
    log_type                  = SYSLOG authpriv
    log_on_success             = HOST PID
    log_on_failure             = HOST
    only_from                  = 128.138.193.0 128.138.204.0
    cps                        = 25 30
}

includedir /etc/xinetd.d

# fin de archivo
```

Note que la instrucción includedir hace un llamado al directorio /etc/xinetd.d que es donde residen los archivos que contiene los servicios a ser habilitados. Miremos el siguiente ejemplo, suponga que al listar el contenido de dicho directorio (ls) visualizara el nombre de 2 archivos: ftp y telnet; el contenido de cada archivo respectivamente podría ser:

```
# Archivo ftp
service ftp
{
    socket_type              = stream
    wait                     = no
    nice                      = 10
    user                     = root
    server                    = /usr/etc/in.ftpd
    server_args               = -l
    instances                 = 4
    log_on_success            += DURATION HOST USERID
    access_times              = 2:00-9:00 12:00-24:00
}

# Archivo telnet
service telnet
{
    socket_type              = stream
```

wait	= no
nice	= 10
user	= root
server	= /usr/etc/in.telnetd
rlimit_as	= 8M
disabled	= no
}	

Sin importar si es inetd.conf o xinetd.conf, son varios los campos que describen la forma es que se comporta dicho servicio. A continuación se explicará los campos principales:

Id: Este atributo se usa para identificar un servicio. Es útil porque existen servicios que pueden usar protocolos diferentes y pueden necesitar parámetros diferentes en el archivo de configuración. Por defecto, el id del servicio es igual que el nombre de servicio, para xinetd el id del servicio es el mismo nombre de archivo ubicado en xinetd.d.

Socket_type: Los posibles valores para este atributo son:

stream	Servicios basados en Stream.
dgram	Servicios basados en datagramas.
Raw	Servicio que requiere acceso directo a IP.
seqpacket	Servicio que requiere transmision secuencial del datagrama en forma segura.

Protocolo: Se determina el protocolo que es empleado por el servicio. El protocolo debe existir en /etc/protocols. Si este atributo no se define, el protocolo predefinido empleado por el servicio se usará.

Wait: Normalmente los servicios “dgram” necesitan este parámetro activo, mientras que los servicios “stream” no lo requieren, sin embargo hay algunas excepciones. Con este parámetro activo (wait o yes) inetd o xinetd espera a que el programa servidor libere el socket de red antes de empezar a escuchar más solicitudes de conexión en ese socket. Con el parámetro inactivo (nowait o no), inetd o xinetd continua escuchando más solicitudes de conexión tan pronto como ha lanzado el programa servidor.

User:	determina el usuario que ejecutara el proceso en el servidor. El nombre del usuario debe existir en /etc/passwd. Este atributo es de gran cuidado ya que muchos servicios se ejecutan como "root" o administrador.
Instances:	determina el número de servicios que pueden ser simultáneamente activos (el valor por defecto es ningún límite).
Nice:	Determina la prioridad del servicio.
Server_args:	Determina los argumentos pasados al servicio.
only_from:	Determina para quienes está disponible el servicio. Su valor es una lista de direcciones IP que pueden especificarse en cualquier combinación ya sea numerico de 4 octetos (172.28.17.0), en forma de factor (172.28.), nombre de una red definida en /etc/networks, un nombre de un host o una direccion Ip y su máscara en forma reducida (172.28.16.0/32).
Disable:	Este es un valor lógico que puede ser "yes" o "no". Uno de estos parámetros producirá que el servicio este o no disponible.
No_access:	Determina para quienes no está disponible el servicio. Su valor puede especificarse de la misma manera como el valor del atributo de only_from.
Access_time:	determina los intervalos de tiempo cuando el servicio estará disponible. Un intervalo tiene el formato: hora:minutos-hora:minuto . Horas pueden ir de 0 a 23 y minutos de 0 a 59.

Más allá del init y inetd, otra manera de empezar procesos automáticamente es a través del demonio del cron. Cron es un demonio que puede ejecutar procesos en un horario específico. Los administradores frecuentemente acostumbran mediante cron a fijar procesos automáticos regulares para aliviar el trabajo de administración del sistema.

Por ejemplo si usted desea ejecutar un programa que examina el sistema para escanear los archivos en busca de virus todas las noches a medianoche o a las 3:00 Am cada 2 días, esto lo podrá realizar a través de un cron. Cron lee uno o más archivos de configuración, conocidos como crontabs, para determinar qué ejecutará y cuándo. Estos archivos del crontab se almacenan dependiendo de las versiones de UNIX/LINUX normalmente en /usr/lib/crontab y /etc/crontab.

Así como administradores del sistema acostumbran a usar este demonio para facilitar su trabajo sin requerir de su presencia, los atacantes también emplean el cron para lograr su trabajo de explotar el sistema. Un atacante con acceso a una máquina de la víctima podría editar los archivos del crontab para correr varios comandos sobre la víctima. Comandos que pueden incluir la negación de un servicio el cierre de un programa en un momento crítico, una puerta trasera (backdoor) a cierta hora para garantizar el acceso remoto a la máquina, o cualquier otro tipo de ataque cronometrado contra el sistema.

A pesar de que init, inetd o cron son procesos que corren automáticamente sobre la máquina. También se pueden ejecutar o terminar de manera manual por usuarios y administradores. Siempre que usted ejecuta un programa en una máquina de UNIX/LINUX bajo la ventana de terminal a través de la línea de comando, un proceso empieza a trabajar el programa. Cuando un usuario ejecuta un programa, el proceso resultante corre normalmente con los permisos del usuario que activó el programa.

Un ejemplo se puede indicar con el servicio de Web server, que corre con el demonio httpd, si se deseara detener este servicio se ejecutaría el siguiente comando:

```
# /etc/init.d/httpd stop
```

Otra forma de ejecutar este comando seria

```
# service httpd stop
```

Para iniciarlo nuevamente, se teclea:

```
# /etc/init.d/httpd start
```

Otra forma de ejecutar este comando seria

```
# service httpd start
```

A parte de los demonios, los procesos aplican también a los comandos como ls, dir, who, finger, Etc. Cuando un usuario ejecuta un comando al mismo tiempo se está ejecutando un proceso, dicho comando se busca en una variedad de directorios que están pre-asignados dependiendo del usuario. La búsqueda se hace en el directorio propio del usuario "." y luego en el "camino de búsqueda" para ese usuario. El camino de búsqueda del usuario es una variable que contiene todos los directorios en que buscare por defecto, Para entender mejor esto teclee el siguiente comando:

```
$ echo $PATH
```

Usted podría obtener una respuesta como:

```
/usr/bin:/usr/local/bin:/usr/bin/X11:/usr/X11R6/bin:/home/jca/bin
```

Esto significa que cuando usted ejecute un comando, primero lo buscara en /usr/bin en caso tal de que no lo encuentre seguirá con /usr/local/bin, o si no /usr/bin/X11 o /usr/X11R6/bin y finalmente en /home/jca/bin.

Es muy peligroso tener el directorio actual, ".", en el camino de búsqueda. Para entender por qué, considera lo que pasa cuando usted teclea un orden normal, como ls para listar el contenido del directorio actual, pero usted tiene "." en su camino de búsqueda. Lo que significa que usted podrá crear un script con el nombre de "ls" bajo dicho directorio que ejecute algo diferente al comando original. Los atacantes les encanta ver "." En el camino de búsqueda ya que pueden depositar troyanos con los mismos nombres de comandos que saben que un usuario normalmente ejecutara y la verdad es que el programa engaña al usuario extrayéndole la contraseña, le niega un servicio, y así sucesivamente.

INTERACTUANDO CON LOS PROCESOS

El kernel asigna a cada proceso que corre sobre la máquina, una única identificación de de proceso llamado Id o Pid que es un número entero que hace referencia al proceso.

Los usuarios pueden correr el comando "ps" para desplegar una lista de los procesos que actualmente se ejecutan. La orden "ps" también puede usarse para mostrar los pids, nombres de programas, utilización de CPU, y otros aspectos de cada programa que se esté ejecutando.

A continuación veremos un ejemplo de los procesos que corren sobre una instalación típica de Linux, mostrada con el comando ps -aux:

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	1.7	0.0	1368	476	?	S	20:24	0:03	init
root	8	0.0	0.0	0	0	?	SW	20:24	0:00	[mdrecoveryd]
root	651	0.0	0.1	1428	560	?	S	20:25	0:00	syslogd -m 0
rpc	676	0.0	0.1	1512	552	?	S	20:25	0:00	portmap
rpcuser	704	0.0	0.1	1556	712	?	S	20:25	0:00	xinetd
root	832	0.0	0.0	1360	480	?	S	20:25	0:00	/usr/sbin/apmd -p
root	886	0.0	0.2	2620	1228	?	S	20:25	0:00	/usr/sbin/sshd
root	960	0.0	0.3	4600	1812	?	S	20:25	0:00	sendmail: accepti
root	997	0.0	0.1	1536	616	?	S	20:25	0:00	cron
xfs	1051	0.3	0.6	4856	3552	?	S	20:25	0:00	xfs -droppriv -da
root	1069	0.0	0.1	1380	552	?	S	20:25	0:00	anacron
daemon	1087	0.0	0.1	1404	524	?	S	20:25	0:00	/usr/sbin/atd
root	1096	0.0	0.0	1344	400	tty1	S	20:25	0:00	/sbin/mingetty tt
root	1097	0.0	0.0	1344	400	tty2	S	20:25	0:00	/sbin/mingetty tt
root	1289	0.0	0.6	7188	3184	?	S	20:26	0:00	magicdev --sm-cli
root	1291	2.0	2.2	28896	11532	?	S	20:26	0:01	nautilus start-he
root	1323	0.0	0.2	2508	1340	pts/0	S	20:26	0:00	bash
root	1459	0.0	0.1	2736	776	pts/0	R	20:27	0:00	ps -aux

Tenga en cuenta que en este ejemplo, se eliminaron algunas líneas para hacerlo más fácil leer. En la anterior lista usted puede ver los procesos init, crond, y xinetd corriendo en el sistema. Adicionalmente, se muestra el shell del usuario que ha ingresado en el sistema (un programa llamado bash) y como puede ver también se muestra el proceso que generó esta lista: ps.

Una manera de actuar recíprocamente con procesos es enviarles un signo (signal). Un signo es un mensaje especial que interrumpe un proceso. Uno de los signos más comunes es el "TÉRMIN" (Abreviación de Terminate) que indica al kernel para detener cierto proceso que se este ejecutando. Otro de los signos frecuentemente usados es el (HUP) que causará que muchos procesos (especialmente el inetd o xinetd) releen sus archivos de configuración. Un usuario puede ejecutar el comando "kill" para enviar un signo a un proceso específico refiriéndose al ID del proceso. Similarmente, el comando "killall" se usa para enviar un signo a un proceso refiriéndose a su nombre. Los comandos "kill" y "killall" no se usan necesariamente para matar, detener o terminar procesos. Por ejemplo, suponga que el Administrador del sistema o un atacante altera la configuración de xinetd, y crea o deshabilita un servicio en el directorio /etc/xinetd.d/. Para lograr que el sistema asuma los cambios, debe obligar a xinetd a releer sus archivos de configuración. Por tanto la persona interesada podría ejecutar la siguiente orden para terminar un proceso:

```
# kill -HUP 919
```

O, alternativamente, el administrador podría usar el comando "killall" para referirse al nombre del proceso. Note que la orden "killall" no mata todos los procesos. Apenas envía un signo a un proceso, con el nombre entrado por el usuario o administrador:

```
# killall -HUP xinetd
```

Ahora que tenemos una leve comprensión de procesos, pongamos nuestra atención a otros conceptos fundamentales de UNIX: cuentas y grupos.

CUENTAS Y GRUPOS

Debido a que Unix o Linux es un sistema multiusuario, para poder ingresar al sistema se requiere de una cuenta de usuario, teniendo en cuenta que cada proceso activo corre con los permisos de una cuenta dada. Igualmente cada cuenta pertenece a un grupo.

Por tanto para usted poder ingresar a un sistema, es obligación el poseer una cuenta.

Analicemos cómo están configuradas las cuentas en un sistema de UNIX/LINUX.

Los actuales sistemas Unix o Linux guardan información acerca de los usuarios en dos archivos, /etc/passwd y /etc/shadow; las versiones viejas solo era el archivo /etc/passwd. Estos archivos son usados por el programa “login” para validar los usuarios que ingresan al sistema y preparar el entorno de trabajo inicial.

Todos los usuarios de un sistema Unix o Linux pueden leer el archivo /etc/passwd, por que tiene permiso de lecturas para todo tipo de usuario. Sin embargo, solamente el usuario “root” (Administrador) puede leer el archivo /etc/shadow, que contiene las contraseñas cifradas.

El archivo /etc/passwd

Existe una línea en /etc/passwd por cada usuario y para ciertos nombres de presentación utilizados por el sistema. Cada una de estas líneas contiene una secuencia de campos, separados por dos puntos. El siguiente ejemplo muestra un archivo /etc/passwd de una instalación por defecto de sistema Linux:

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/var/spool/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
mailnull:x:47:47:/:/var/spool/mqueue:/dev/null
rpm:x:37:37:/:/var/lib/rpm:/bin/bash
ntp:x:38:38:/:etc/ntp:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin
xfs:x:43:43:X Font Server:/etc/X11/fs:/bin/false
gdm:x:42:42:/:var/gdm:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
nscd:x:28:28:NSCD Daemon:/:/bin/false
ident:x:98:98:pident user:/:/sbin/nologin
radvd:x:75:75:radvd user:/:/bin/false
pcap:x:77:77:/:var/arpwatch:/sbin/nologin
jca:x:500:500:Jhon Cesar Arango:/home/jca:/bin/bash
```

El primer campo de una línea en el archivo /etc/passwd contiene el nombre de presentación del usuario.

El segundo campo contiene la letra x. (En las versiones anteriores este campo contenía una contraseña cifrada, de lo que se derivaba una

debilidad en la seguridad. La utilización de una x siempre proporciona un cierto grado de protección, pero sigue siendo una debilidad, ya que un intruso podría identificarla. En las Versiones recientes la contraseña cifrada está en /etc/shadow.).

Los campos tercero y cuarto son el ID de usuario e ID de grupo, respectivamente.

En el quinto campo se colocan comentarios. Generalmente este campo contiene el nombre del usuario y con frecuencia también contiene su número de despacho y el número de teléfono.

El sexto campo es el directorio propio, es decir, el valor inicial de la variable HOME.

El campo final designa al programa que el sistema ejecuta automáticamente cuando el usuario abre una sesión. Éste se denomina shell de presentación del usuario. El shell estándar, sh (bash), es el programa de inicio por defecto. Así, si el campo último está vacío, sh será el programa de inicio del usuario.

La información referente al nombre de presentación root está incluida en la primera línea del archivo /etc/passwd. El ID de usuario de root es 0, su directorio propio es el directorio raíz, representado por /, y el programa inicial que el sistema ejecuta para root es el shell estándar, sh, ya que el último campo está vacío.

Como se puede ver, en el ejemplo anterior, el archivo /etc/passwd contiene nombres de presentación usados por el sistema para su funcionamiento y para administración del sistema. Entre ellos se incluyen los siguientes ID de presentación: daemon, bin, adm, halt, Entre otros. También se incluyen nombres de presentación utilizados para la conexión en red, como uucp usado para la operación de la red de área local. El programa de inicio para cada uno de estos nombres de presentación puede encontrarse en el último campo de la línea asociada en el archivo /etc/passwd.

El archivo /etc/shadow

Existe una línea en /etc/shadow por cada línea del archivo /etc/passwd. El archivo /etc/shadow contiene información acerca de la contraseña de un usuario y datos referentes al envejecimiento de la contraseña. Por ejemplo, el archivo asociado al ejemplo anterior es el siguiente:


```

root:$1$ÓolP1ô1X$/Q171zopIt.yoqyAcF7jE/:11922:0:99999:7:::
bin:*:11922:0:99999:7:::
daemon:*:11922:0:99999:7:::
adm:*:11922:0:99999:7:::
lp:*:11922:0:99999:7:::
sync:*:11922:0:99999:7:::
shutdown:*:11922:0:99999:7:::
halt:*:11922:0:99999:7:::
mail:*:11922:0:99999:7:::
news:*:11922:0:99999:7:::
uucp:*:11922:0:99999:7:::
operator:*:11922:0:99999:7:::
games:*:11922:0:99999:7:::
gopher:*:11922:0:99999:7:::
ftp:*:11922:0:99999:7:::
nobody:*:11922:0:99999:7:::
vcsa:!!:11922:0:99999:7:::
mailnull:!!:11922:0:99999:7:::
rpm:!!:11922:0:99999:7:::
ntp:!!:11922:0:99999:7:::
rpc:!!:11922:0:99999:7:::
xfs:!!:11922:0:99999:7:::
gdm:!!:11922:0:99999:7:::
rpcuser:!!:11922:0:99999:7:::
nfsnobody:!!:11922:0:99999:7:::
nscd:!!:11922:0:99999:7:::
ident:!!:11922:0:99999:7:::
radvd:!!:11922:0:99999:7:::
pcap:!!:11922:0:99999:7:::
jca:$1$yâp6èĒŦŦ$jXMnsdZ0iaIIiTezZDfQe0:11922:0:99999:7:::

```

El primer campo de la línea contiene el nombre de presentación. Para usuarios con contraseñas.

El segundo campo contiene la contraseña cifrada para ese nombre de presentación. Este campo puede tener NP (No Password) cuando no existe contraseña para ese nombre de presentación y * para los nombres de presentación propios del sistema. Ninguna de estas cadenas (NP, y *) pueden ser nunca la versión cifrada de una contraseña válida, por lo que es imposible presentarse con uno de estos nombres al sistema, ya que cualquier respuesta dada a la petición «Password:» dejará de producir una coincidencia con los contenidos de este campo. De este modo estos nombres de presentación están efectivamente bloqueados.

El tercer campo indica el número de días entre el 1 de enero de 1970 y el día en que la contraseña fue modificada la última vez.

El cuarto campo indica el número mínimo de días requerido entre cambios de la contraseña. Un usuario no puede cambiar su contraseña de nuevo hasta que transcurra ese número de días.

El quinto campo indica el número máximo de días que una contraseña es válida. Transcurrido ese número de días, el usuario se ve forzado a cambiar la contraseña.

El sexto campo indica el número de días antes de la expiración de una contraseña que el usuario es avisado. Se enviará un mensaje de aviso a un usuario cuando éste se presente para notificarle que su contraseña está a punto de expirar dentro de esos días.

El séptimo campo indica el número de días de inactividad permitido a este usuario. Si ese número de días transcurre sin que el usuario se presente, su línea de presentación se bloquea.

El octavo campo indica la fecha absoluta (especificada mediante el número de días desde el 1 de enero de 1970; por ejemplo, 9800 es el 3 de mayo de 1996) a partir de la cual el nombre de presentación ya no puede ser utilizado.

El noveno campo es una opción que actualmente no es utilizada, pero que puede serlo en el futuro.

En las versiones anteriores de Unix y de Linux, el archivo `/etc/passwd` contenía las contraseñas cifradas para los usuarios en el segundo campo de cada línea. Puesto que los usuarios ordinarios pueden leer este archivo, un usuario autorizado, o un intruso que hubiera tenido acceso a un nombre de presentación, podría también ganar acceso a otros nombres de login. Para hacer esto, el usuario, o el intruso, ejecuta un programa (como `john the ripper`²) para cifrar palabras desde un diccionario de palabras o cadenas comunes formadas a partir de nombres, utilizando el algoritmo del sistema UNIX/LINUX para cifrar contraseñas (que no se mantiene en secreto), y compara los resultados con las contraseñas cifradas en el sistema. Si encuentra una coincidencia, el intruso tiene acceso a los archivos de un usuario. Esta vulnerabilidad ha sido reducida colocando una `x` en el segundo campo del archivo `/etc/passwd` y usando el archivo `/etc/shadow`.

CONTROL DE PRIVILEGIOS

En Linux O Unix , el acceso de los usuarios a los distintos archivos y directorios se limita mediante la concesión de permisos. Hay tres tipos básicos de permisos:

- De lectura: permite a los usuarios leer el archivo especificado.
- De escritura: permite a los usuarios modificar el archivo especificado.
- De ejecución: permite a los usuarios ejecutar el archivo especificado.

Cuando se asignan estos permisos, Linux o Unix guarda un registro de los mismos que posteriormente aparece reflejado en las listas de archivos.

² Enlace de John the ripper

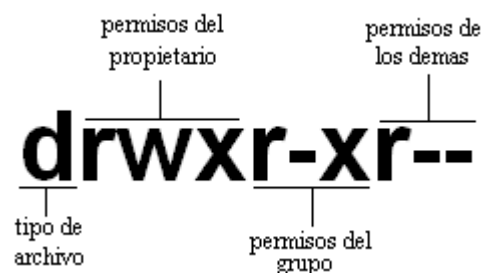
El estado de los permisos de cada uno de los archivos se expresa mediante marcas. Las marcas de permiso son:

- r : acceso de lectura.
- w : acceso de escritura.
- x : acceso de ejecución.

El comando “ls -l” o “dir -l” para linux, muestra el contenido de un directorio, esta lista puede contener directorios o archivos, observemos el ejemplo siguiente:

-rw-----	1	root	root	10200	oct	1	21:18	boot.log
-rw-----	1	root	root	960	oct	1	21:20	cron
drwxr-xr-x	2	lp	root	4096	sep	2	22:43	cups
-rw-r--r-x	1	root	root	6762	oct	1	21:16	dmesg
drwxr-xr-x	2	root	root	4096	feb	26	2002	fax
drwxr-xr-x	2	root	root	4096	ago	23	22:54	gdm
-rw-r--r--	1	root	root	59445	oct	1	21:16	ksyms.0
-rw-r--r--	1	root	root	59445	sep	15	19:55	ksyms.1
-rw-r--r--	1	root	root	59445	sep	2	22:39	ksyms.2
-rw-r--r--	1	root	root	19136220	oct	1	21:18	lastlog
-rw-----	1	root	root	554	oct	1	21:18	maillog
-rw-----	1	root	root	47573	oct	1	21:19	messages
-rw-r--r--	1	root	root	16539	sep	2	22:46	rpmpkgs
drwxr-xr-x	2	root	root	4096	oct	1	21:20	sa
-rw-----	1	root	root	450	oct	1	21:18	secure
-rw-----	1	root	root	0	sep	2	22:46	spooler
drwxr-xr-x	2	root	root	4096	abr	8	08:07	vbox
-rw-rw-r--	1	root	utmp	31104	oct	1	21:18	wtmp

Observe que la primera columna de la anterior lista contiene 10 campos, los cuales se muestran a continuación:



Existen tres clases de permisos para los archivos y directorios que se corresponden con las tres clases de usuario: el propietario (o usuario) del archivo o directorio, el grupo al que pertenece el propietario y los otros usuarios del sistema. El primer campo indica el tipo de archivo (si es un archivo, un directorio u otro), los tres siguientes campos hacen referencia a los permisos del propietario; las tres siguientes a los miembros del grupo del propietario y las tres últimas a los otros usuarios.

Note que en el ejemplo anterior, la salida correspondiente al denominado “dmesg”, indica: El primer campo indica que es un archivo, los tres siguientes rw-, muestran que el propietario del archivo puede leerlo (r) y

escribirlo (w). El segundo grupo de tres caracteres, r--, indican que los miembros del grupo pueden leer el archivo, pero no pueden escribirlo, ni ejecutarlo. Los tres últimos caracteres, r-x, muestran que todos los demás pueden leer y ejecutar el archivo, pero no escribirlo.

El sistema octal

En el sistema octal, los números representan permisos. La siguiente tabla, resume el esquema octal y lo que representa cada número.

r	w	x	Equivalente Octal
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Si se utilizan valores octales puros, hay que añadirlos juntos, lo que deriva un número final que expresa todos los permisos concedidos, Pero para facilitar las cosas, es posible reducir rápidamente los permisos del propietario, de los grupos y de otros usuarios a un número de tres dígitos utilizando estos valores:

- 0 = Sin permisos.
- 1 = Ejecución.
- 2 = Escritura.
- 3 = Escritura y ejecución (actualmente no se utiliza mucho).
- 4 = Lectura.
- 5 = Lectura y ejecución.
- 6 = Lectura y escritura.
- 7 = Todo el conjunto: lectura, escritura y ejecución.

Por ejemplo, suponga que quiere darle al archivo “dmesg” permisos de lectura y ejecución al usuario, lectura al grupo y ningún permiso a los demás. El comando “chmod” permite cambiar estos permisos y facilita su utilización si usted conoce el equivalente octal, así por ejemplo; si ejecutamos el comando:

```
# chmod 340 dmesg
```

Notará que los permisos del archivo, cambiaron al ejemplo anteriormente mencionado.

En seguridad este comando es bastante utilizado para colocar permisos a archivos que deseamos que ciertos usuarios ejecuten mas no que modifiquen, se podría asignar fácilmente a un archivo o directorio el permiso 751 que indica:

- El propietario puede leerlo, escribirlo y ejecutarlo (7).
- El grupo puede leerlo y ejecutarlo (5).
- El mundo (usuarios externos) sólo pueden ejecutarlo (1).

PROGRAMAS SUID Y SGID

Existen dos bits adicionales de permiso asociados a un archivo: los bits SUID y SGID. SUID representa el identificador del usuario y SGID representa el identificador del grupo. Cuando se ejecutan programas con permisos, éstos se comportan como si pertenecieran a identificadores de usuarios distintos. Cuando se ejecuta un programa SUID, su identificador de usuario efectivo es el mismo que el del usuario propietario del programa en el sistema de archivos, independientemente de quién esté ejecutando realmente el programa. SGID es similar, salvo que cambia el identificador de grupo.

Veamos el siguiente ejemplo, a veces usuarios o procesos tienen una razón legítima por acceder a un archivo que ellos no tienen permisos. Considere lo que pasa cuando un usuario desea cambiar su contraseña. El usuario tiene que entrar a modificar los archivos `etc/passwd` y el `etc/shadow`.

Sin embargo, estos archivos solo son de propiedad del “root” y solo pueden ser modificados por él, entonces ¿Cómo puede cambiar un usuario normal su contraseña sin tener que importunar a los administradores del sistema cada vez que necesite temporalmente las propiedades del administrador?

La respuesta la da otra capacidad de los sistemas LINUX/UNIX llamado SetUID (“Set User ID”). Con esta capacidad, un programa particular puede configurarse para siempre se ejecute con los permisos de su dueño, y no con los permisos del usuario que lanzó el programa. Recuerde, normalmente cuando un usuario empieza un proceso, el proceso corre con los permisos del usuario.

Por tanto, en nuestro ejemplo anterior, el usuario correrá un programa de SetUID especial llamado “passwd” para cambiar una contraseña. El programa del passwd está configurado por defecto para ejecutarse como “root”. Es decir, sin tener en cuenta quién ejecuta este comando, corre con permisos del “root”. Lo que nos permite decir, que este tipo de programas convierten temporalmente a usuarios normales en administradores.

Así, si deseamos hacer que el comando “dmesg” lo pueda ejecutar cualquier usuario con las propiedades del dueño original se podría teclear:

```
# chmod 4741 dmesg
```

Aunque la función SUID/SGID puede ser de gran utilidad, también compromete la seguridad del sistema. Los programadores normalmente hacen todo lo posible por garantizar cierta seguridad en sus programas

SUID. Los problemas de seguridad de los programas surgen cuando el programa ejecuta una línea de comandos, la cual puede activar un shell o ejecutar un archivo que los usuarios pueden modificar para que contenga sus propios comandos.

A pesar de que algunos programas SUID son necesarios, es mejor reducirlos al mínimo. Esto se logra explorando regularmente los sistemas de archivos a través del comando find. (Ejecutar el comando “man find” para entender los parámetros del siguiente ejemplo)

```
# find / -user root -perm 4000 -print
```

RELACION DE CONFIANZA EN MAQUINAS LINUX/UNIX

Anteriormente las maquinas en Linux podrían ser configurados de forma de que un equipo confiara en otro. Estas relaciones de confianza se realizaban a través de las opciones de algo llamado los “Los Comandos R” y mediante el archivo “/etc/hosts.equiv”.

Los comandos R incluyen normalmente un grupo de comandos que se derivan de los sistemas BSD. Estos comandos son: “rlogin” para el inicio de sesión remoto; “rsh” y “rexec” para la ejecución remota de comandos de shell; y “rcp”, para la copia de archivos remotos.

Las herramientas que proporcionan Telnet y los comandos R son adecuadas, pero tiene una serie de problemas que se pueden solucionar con un programa que puede aportar una solución: SSH.

Tanto Telnet como los comandos R están lejos de ser seguros. El mecanismo de autenticación de rhosts, basado en la confianza y que usan los comandos R, es especialmente peligroso: confía en las direcciones IP del origen para identificar a los usuarios, por lo que puede ser interceptado con facilidad. El uso de tcpd o de xinetd para detectar los intentos de falsificación y las conexiones rechazadas eliminan parte del riesgo de este método, pero no son infalibles.

Telnet no está tan mal en cuanto a que pide siempre la contraseña antes de conceder el acceso, pero sufre de otra debilidad que también comparten los comandos R: debido a que las contraseñas, como todos los demás datos, se envían como texto plano a través de la red, son muy susceptibles a los ataques mediante el rastreo de paquetes (“Sniffing”).

Telnet y Rlogin tienen también una serie de problemas desde el punto de vista del usuario. La copia remota de archivos no es ni mucho menos apropiada. Puede que se considere a FTP como el mecanismo de transferencia de archivos, pero es muy poco manejable y muy difícil de pasar a scripts. Rcp es fácil de usar, pero no funcionará en absoluto hasta que el usuario autorice la equivalencia en el ámbito de cuentas entre las máquinas.

SSH, Secure Shell Service, Servicio de shell seguro

El servicio Secure Shell, SSH, intenta solucionar estos problemas de manera excelente. Entre todas las desventajas, destaca el uso de un potente cifrado para los datos transmitidos, con lo que ni las contraseñas ni otros datos pueden ser robados ni siquiera por atacantes que escuchen los datos que se están transmitiendo. El uso del cifrado impide también los ataques en los que el intruso entra en una conexión existente y cambia los datos en las dos direcciones. Los ataques de esta índole pueden usarse para añadir comandos a las sesiones, incluso en los casos en que la sesión haya sido autenticada mediante una segura contraseña de un solo uso. SSH usa otras técnicas criptográficas para efectuar una potente autenticación de los hosts y de los clientes. Esto significa que se puede tener un alto grado de confidencialidad a la que sólo los usuarios autorizados tienen permiso para conectarse. SSH se preocupa también por la facilidad de uso y tiene soporte completo para ejecutar aplicaciones X11 a través del canal autenticado y cifrado.

Cuando un cliente se conecta a un servidor SSH, verifica que el servidor sea realmente la máquina a la que se quería conectar. El cliente y el servidor intercambian claves de cifrado (de modo que impide a los espías que se aprendan las claves). El servidor autentifica entonces al cliente, usando el mecanismo de rhosts, la autenticación tradicional basada en la contraseña, o bien (de manera más segura) la autenticación RSA. Una vez que el cliente ha sido autenticado, el servidor lanza una shell o ejecuta un comando, a petición del cliente.

El método de autenticación usado por SSH se basa en criptografía o cifrado de clave pública, también conocida como criptografía asimétrica. La criptografía de clave pública cuenta con la existencia de un par asimétrico de claves: una clave pública y una clave privada. Las dos claves están relacionadas, pero no es posible deducir la clave privada desde la pública sin tener que hacer una búsqueda tanteando por todo el espacio de claves. La clave pública se usa para el cifrado y la clave privada para el descifrado. La clave privada debe mantenerse secreta, pero la pública puede ser emitida libremente por canales de comunicación insegura. Un agente remoto puede usar la clave pública para cifrar un flujo de datos que debería ser privado; sólo un agente con la clave privada correcta será capaz de leer los datos cifrados.

SSH usa también la autenticación para verificar la identidad del cliente y de los hosts de los servidores. Esto elimina todos los ataques basados en la suplantación de identidades de host mediante el falseamiento del DNS, del encaminamiento o de la dirección IP.

SERVICIOS COMUNES DE UNIX/LINUX

Cuando se instala un sistema operativo como Linux o Unix, normalmente hay un conjunto de servicios que se instalan por defecto; esto es bueno para los proveedores del sistema operativo ya que se evitan el soporte que acarrea una instalación, pero es muy peligroso porque muchos de estos servicios no se requieren en un servidor y son usados por los atacantes como puntos de vulnerabilidad sobre una red.

Podemos ver los diferentes servicios que un sistema Unix o Linux ofrece como potenciales puertas de entrada al mismo, o al menos como fuentes de ataques que ni siquiera tienen porque proporcionar acceso a la máquina (como las negaciones de servicio). De esta forma, si cada servicio ofrecido es un posible problema para nuestra seguridad, parece claro que lo ideal sería no ofrecer ninguno, poseer una máquina completamente aislada del resto; evidentemente, esto no suele ser posible hoy en día en la mayor parte de los sistemas. Por tanto, ya que es necesaria la conectividad entre equipos, hemos de ofrecer los mínimos servicios necesarios para que todo funcione correctamente; esto choca frontalmente con las políticas de la mayoría de fabricantes de sistemas Unix/Linux, que por defecto mantienen la mayoría de servicios abiertos al instalar un equipo nuevo: es responsabilidad del administrador preocuparse de cerrar los que no sean estrictamente necesarios.

Típicos ejemplos de servicios que suele ser necesario ofrecer son sendmail o ftp; en estos casos es necesaria una correcta configuración para que solo sea posible acceder a ellos desde ciertas máquinas, a través de xinetd.d. También es una buena idea sustituir estos servicios por equivalentes cifrados, como la familia de aplicaciones ssh, y concienciar a los usuarios para que utilicen estos equivalentes: hemos de recordar siempre (y recordar a los usuarios) que cualquier conexión en texto claro entre dos sistemas puede ser fácilmente capturada por cualquier persona situada en una máquina intermedia lo cual coloca en juego la seguridad de sistema y de la red completa. Tenga en cuenta que aparte de puertas de entrada, los servicios ofrecidos también son muy susceptibles de ataques de negación de servicio (DoS).

Para determinar la seguridad de un sistema usted podría desactivar o remover todos los servicios que su red no utilizara. Para determinar qué servicios son o no de importancia, usted debe tener claro el propósito de su servidor o equipo y así determinar los servicios mínimos necesarios, los servicios más comunes en una instalación por defecto son:

- Bluetooth
- Cron
- Cups
- Haldaemon
- Iptables
- Rsyslog
- Sendmail
- Sshd

XWindow

El entorno XWindow proporciona herramientas increíblemente potentes, pero que si no son correctamente configuradas pueden convertirse en peligrosas. Este sistema está formado por una serie de piezas que trabajan conjuntamente para ofrecer al usuario final un interfaz grafica:

La más importante de ellas, sobre todo desde el punto de vista de la seguridad es el servidor X. Este programa generalmente se ejecuta en la terminal de usuario, y tiene como función principal ofrecer unas primitivas básicas de dibujo (trazado de rectas, relleno de áreas. . .) sobre la pantalla; además gestiona eventos de teclado y ratón.

Las aplicaciones X son programas de usuario que lanzan llamadas contra un servidor X, mientras que el servidor se ejecuta habitualmente en la terminal desde donde conecta el usuario las aplicaciones se pueden lanzar desde el mismo equipo o también desde una máquina más potente, de forma que aprovechamos la capacidad de procesamiento de ese equipo.

El gestor de ventanas es un caso particular de aplicación, ya que se encarga de ofrecer un entorno de trabajo más amigable al usuario que está trabajando en la terminal: dibujo de marcos, menús, cerrado de ventanas.

Laboratorio Práctico

Con el fin de ambientarnos con el sistema operativo Linux Recomendando seguir los laboratorios de la comunidad drangonjar³ en la instalación y uso de backtrack⁴

<http://labs.dragonjar.org/video-tutorial-backtrack-booteo-interfaz-grafica-y-directorios>

Gracias a nuestros amigos DragonJar y 4v4t4r por el aporte.

³ <http://www.dragonjar.org>

⁴ <http://www.backtrack-linux.org/>

AMBIENTE OPERATIVO WINDOWS NT/XP/200X/VISTA/7

Como las máquinas LINUX, las plataformas de WINDOWS NT, XP, 200X, VISTA Y WINDOWS 7 son blancos populares de atacantes. Solo consultar algunas fuentes de estadísticas de vulnerabilidades como la del FBI o la del grupo X-Force, veremos que los sistemas operativos mas atacados son aquellos basados en Windows. Adicionalmente, si usted <http://www.microsoft.com/security/about/sir.aspx#MTCSC>, encontrara que la casa Microsoft se ha preocupado por mantener al día sus sistemas operativos.

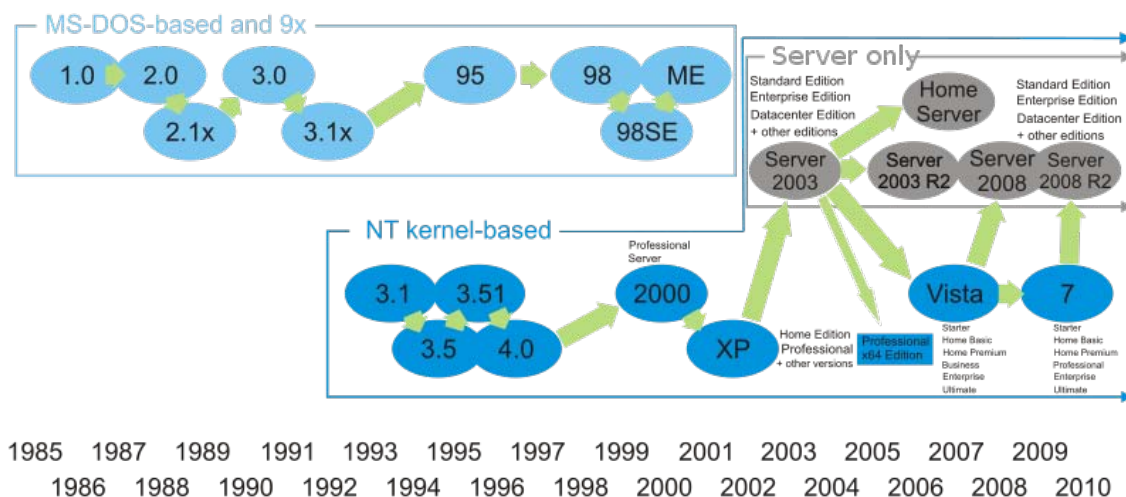
En este capítulo, echaremos una mirada a los sistemas operativos de WINDOWS NT, XP, 200X, VISTA y 7 conociendo su estructura para así analizar los mecanismos de seguridad específicos que ellos ofrecen. Veremos una breve historia de WINDOWS NT y centraremos nuestra atención a los conceptos de fundamentales, varios componentes de la arquitectura y opciones de seguridad de WINDOWS NT. Adicionalmente, examinaremos a Windows 2000 (qué realmente es WINDOWS NT 5.0) para determinar los cambios ocurridos y su impacto en la seguridad.

Este capítulo proporciona una breve apreciación global de la seguridad de WINDOWS NT y 2000 para que se pueda entender los ataques básicos descritos a lo largo del libro.

UNA BREVE HISTORIA EN EL TIEMPO⁵

Microsoft Windows

family tree



WINDOWS NT evoluciono a través de dos sistemas operativos previos: OS/2 y LAN MANAGER. Para adaptarse a la compatibilidad de estos productos, WINDOWS NT uso muchos de sus mecanismos para la conexión a red, con la

⁵ http://es.wikipedia.org/wiki/Microsoft_Windows

diferencia de una interfaz de usuario funcional y amigable. Esto aumento los esfuerzos de mercado lo cual coloco a WINDOWS NT en la cima de los mapas de ventas de sistemas operativos comerciales.

El "NT" en WINDOWS NT significa "Nueva Tecnología", también es conveniente considerar que existen diferentes compañías detrás de las versiones de UNIX, mientras que la empresa Microsoft es la única detrás de la serie WINDOWS.

Microsoft libero al mercado la versión de WINDOWS NT 3.1, luego 3.5, 3.51, 4.0, Windows Vista, Windows 2000 Server, Windows 2003 Server, Windows 2008 Server y Windows 7.

Windows 2008 Server

Windows 2008 Server es un sistema operativo multiproceso que soporta tanto redes basadas en servidores como redes punto a punto. Las características principales de este nuevo sistema operativo de red radica en la manera en que se gestiona el sistema hasta el punto de que se puede llegar a controlar el hardware de forma más efectiva, se puede controlar mucho mejor de forma remota y cambiar de forma radical la política de seguridad. Entre las mejoras que se incluyen, están:

- Nuevo proceso de reparación de sistemas NTFS: proceso en segundo plano que repara los archivos dañados.
- Creación de sesiones de usuario en paralelo: reduce tiempos de espera en los Terminal Services y en la creación de sesiones de usuario a gran escala.
- Cierre limpio de Servicios.
- Sistema de archivos SMB2: de 30 a 40 veces más rápido el acceso a los servidores multimedia.
- Address Space Load Randomization (ASLR): protección contra malware en la carga de controladores en memoria.
- Windows Hardware Error Architecture (WHEA): protocolo mejorado y estandarizado de reporte de errores.
- Virtualización de Windows Server: mejoras en el rendimiento de la virtualización.
- PowerShell: inclusión de una consola mejorada con soporte GUI para administración.
- Server Core: el núcleo del sistema se ha renovado con muchas y nuevas mejoras.

La mayoría de las ediciones de Windows Server 2008 están disponibles en x86-64 (64 bits) y x86 (32 bits). Windows Server 2008 para sistemas basados en Itanium soporta procesadores IA-64. La versión IA-64 se ha optimizado para escenarios con altas cargas de trabajo como servidores de bases de datos y aplicaciones de línea de negocios (LOB). Por ende no está optimizado para su uso como servidor de archivos o servidor de medios. Microsoft ha anunciado que Windows Server 2008 será el último sistema operativo para servidores disponible en 32 bits. Las ediciones se enumeran a continuación:

- Windows Server 2008 Standard Edition (x86 y x86-64)
- Windows Server 2008 R2 Todas las Ediciones (Solo 64Bit)
- Windows Server 2008 Enterprise Edition (x86 y x86-64)
- Windows Server 2008 Datacenter Edition (x86 y x86-64)
- Windows HPC Server 2008 (reemplaza Windows Compute Cluster Server 2003)
- Windows Web Server 2008 (x86 y x86-64)
- Windows Storage Server 2008 (x86 y x86-64)
- Windows Small Business Server 2008 (Nombre clave "Cougar") (x86-64) para pequeñas empresas
- Windows Essential Business Server 2008 (Nombre clave "Centro") (x86-64) para empresas de tamaño medio3
- Windows Server 2008 para sistemas basados en Itanium
- Windows Server 2008 Foundation Server

Windows 7

Windows 7 es la versión más reciente de Microsoft Windows, línea de sistemas operativos producida por Microsoft Corporation. Esta versión está diseñada para uso en PC, incluyendo equipos de escritorio en hogares y oficinas, equipos portátiles, "tablet PC", "netbooks" y equipos "media center".

A diferencia del gran salto arquitectónico y de características que sufrió su antecesor Windows Vista con respecto a Windows XP, Windows 7 fue concebido como una actualización incremental y focalizada de Vista y su núcleo NT 6.0, lo que permitió el mantener cierto grado de compatibilidad con aplicaciones y hardware en los que éste ya era compatible. Sin embargo, entre las metas de desarrollo para Windows 7 se dio importancia en mejorar su interfaz para volverla más accesible al usuario e incluir nuevas características que permitieran hacer tareas de una manera más fácil y rápida, al mismo tiempo en que se realizarían esfuerzos para lograr un sistema más ligero, estable y rápido.

Existen seis ediciones de Windows 7, construidas una sobre otra de manera incremental, aunque solamente se centrarán en comercializar tres de ellas para el común de los usuarios: las ediciones Home Premium, Professional y Ultimate. A estas tres, se suman las versiones Home Basic y Starter, además de la versión Enterprise, que está destinada a grupos empresariales que cuenten con licenciamiento "Open" o "Select" de Microsoft.

Starter: Es la versión de Windows 7 con menos funcionalidades de todas. Posee una versión incompleta de la interfaz Aero que no incluye los efectos de transparencia Glass, Flip 3D o las vistas previas de las ventanas en la barra de inicio y que además no permite cambiar el fondo de escritorio. Está dirigida a PC de hardware limitado —como netbooks—, siendo licenciada únicamente para integradores y fabricantes OEM. Incluye una serie de restricciones en opciones de personalización, además de ser la única edición de Windows 7 sin disponibilidad de versión para hardware de 64 bits.

Home Basic: Versión con más funciones de conectividad y personalización, aunque su interfaz seguirá siendo incompleta como en la edición Starter. Sólo estará disponible para integradores y fabricantes OEM en países en vías de desarrollo y mercados emergentes.

Home Premium: Además de lo anterior, se incluye Windows Media Center, el tema Aero completo y soporte para múltiples códecs de formatos de archivos multimedia. Disponible en canales de venta minoristas como librerías, tiendas y almacenes de cadena.

Professional: Equivalente a Vista "Business", pero ahora incluirá todas las funciones de la versión Home Premium más "Protección de datos" con "Copia de seguridad avanzada", red administrada con soporte para dominios, impresión en red localizada mediante Location Aware Printing y cifrado de archivos. También disponible en canales de venta al público.

Enterprise: Añade sobre la edición Professional de Windows 7, características de seguridad y protección de datos como BitLocker en discos duros externos e internos, Applocker, Direct Access, BranchCache, soporte a imágenes virtualizadas de discos duros (en formato VHD) y el paquete de opción multilinguaje. Únicamente se vende por volumen bajo contrato empresarial Microsoft software Assurance. También es la única que da derecho a la suscripción del paquete de optimización de escritorio MDOP.

Ultimate: Esta edición es igual a la versión Enterprise pero sin las restricciones de licenciamiento por volumen, permitiéndose su compra en canales de venta al público general, aunque Microsoft ha declarado que en lugar de publicitarse en medios comunes, será ofrecida en promociones ocasionales de fabricantes y vendedores.

Ediciones N: Las ediciones N están disponibles para actualizaciones y nuevas compras de Windows 7 Premium, Professional y Ultimate. Las características son las mismas que sus versiones equivalentes, pero no incluyen Windows Media Player. El precio también es el mismo, ya que Windows Media Player puede descargarse gratuitamente desde la página de Microsoft.

WINDOWS XP PROFESIONAL

Integra los puntos fuertes de Windows 2000 Profesional (como seguridad basada en estándares, la capacidad de administración y confiabilidad), características comerciales de Windows 98 y Windows Me (Plug and Play, interfaz de usuario sencilla y servicios de soporte). Entre sus características están:

Basado en el nuevo motor de Windows, integra la base de códigos de Windows NT y Windows 2000, que presenta una arquitectura informática de 32 bits y un modelo de memoria totalmente protegido.

Escenarios de reinicio reducidos drásticamente, elimina la mayoría de los escenarios que obligan a los usuarios finales a reiniciar los equipos en Windows NT 4.0 y Windows 95/98/Me.

Protección de códigos mejorada, la estructura de los datos importantes del núcleo son de solo lectura, por lo que los controladores y las aplicaciones no pueden corromperlas. Todos los códigos de controladores de dispositivos son de solo lectura y con protección de página.

Directivas de restricción de software mejoradas, proporciona a los administradores un mecanismo impulsado por directivas para identificar el software que se encuentra en ejecución en su entorno y controlar su capacidad de ejecución. Se utiliza en la prevención de virus y caballos de Troya y el bloqueo de software.

Sistema de cifrado de archivos con soporte para varios usuarios, cifra todos los archivos con una clave generada aleatoriamente. Los procesos de cifrado y descifrado son transparentes para el usuario. En Windows XP Profesional permite que varios usuarios tengan acceso a un documento cifrado.

Seguridad IP, ayuda a proteger los datos transmitidos a través de la red, IPSec es una parte importante de las redes virtuales privadas (VPN), que permiten a las organizaciones transmitir datos de forma segura por Internet.

Archivos y carpetas sin conexión, los usuarios pueden especificar los archivos y las carpetas de la red que necesitarán cuando se desconecten. Las carpetas sin conexión se pueden cifrar para brindar el más alto nivel de seguridad.

Consola de recuperación, proporciona una consola de línea de comandos para iniciar y detener servicios, dar formato a unidades, leer y escribir datos en una unidad local y realizar tareas administrativas.

Directiva de grupo, simplifica la administración de los usuarios, al permitir a los administradores organizarlos en unidades lógicas, como departamentos o ubicaciones, y asignar la misma configuración, incluidas las opciones de seguridad, aspecto y administración a todos los empleados del grupo.

CONCEPTOS FUNDAMENTALES DE SERVIDORES WINDOWS Y SISTEMAS WINDOWS 7

DIRECTORIO ACTIVO

El directorio activo es un servicio de directorio. El término servicio de directorio se refiere a dos cosas – un directorio donde la información sobre usuarios y recursos está almacenada, y un servicio o servicios que dejan acceder y manipular estos recursos. El directorio activo es una manera de manejar todos los elementos de una red, incluido computadoras, grupos, usuarios, dominios, políticas de seguridad, y cualquier tipo de objetos definidos para el usuario. Además de esto, provee de funciones adicionales más allá de estas herramientas y servicios.

El directorio activo está construido alrededor de la tecnología DNS y LDAP – DNS porque es el estándar en Internet y es bastante familiar, LDAP porque la mayoría de fabricantes lo soportan. Los clientes de directorio activo usan DNS y LDAP para localizar y acceder a cualquier tipo de recurso de la red. Al ser protocolos de plataforma independiente, Los computadores Unix, Linux y Macintosh pueden tener acceso a los recursos de igual modo que los clientes de Windows.

La consola MMC (*Microsoft Management Console*) se usa para implementar y gestionar el directorio activo. Las metas de directorio activo tienen dos acercamientos importantes:

- Los usuarios deben poder acceder a recursos por todo el dominio usando un único acceso o login a la red.
- Los administradores deben poder centralizar la gestión de usuarios y recursos.

La estructura de directorio activo tiene una forma jerárquica donde se localizan los objetos. Estos objetos caen en tres tipos de categorías:

- Recursos, como por ejemplo impresoras.
- Servicios, como correo, Web, FTP, etc.
- Usuarios, los cuales incluyen cuentas para conectarse, grupos de trabajo, etc.

Un objeto es únicamente identificado por su nombre y tiene una serie de atributos definidos por un esquema, que también determina la clase de objeto que se pueden almacenar en el directorio. Los atributos son las características y la información que el objeto contiene.

Cada atributo del objeto puede ser usado en diferentes clases de objetos dentro del esquema del objeto. Dicho esquema existe para que se pueda hacer modificaciones o extensiones cuando sea necesario. Hay que tener

cuidado al cambiar estos atributos una vez que estén creados, ya que podemos cambiar la estructura ya creada del directorio activo, por lo que hay que hacerlo de un modo planeado.

El dominio se observa desde un número de niveles. En la parte más alta tenemos el bosque – la colección de todos los objetos, sus atributos y reglas en el directorio activo. Los dominios se identifican por su nombre de estructura DNS. Un dominio tiene un solo nombre DNS.

Los objetos dentro de un dominio pueden estar agrupados en contenedores llamados unidades organizativas (OU). Estas unidades dan al dominio una jerarquía, facilita la administración y proporciona una imagen de la compañía en términos organizativos y geográficos.

Estas unidades organizativas pueden contener a su vez otras unidades organizativas. Normalmente, se suelen aplicar las políticas de grupo a nivel de OU, aunque también pueden ser aplicados a dominios. Se suelen dar poderes de administrador a estos OU y todo lo que contienen por debajo, aunque también se pueden delegar funciones de administrador a objetos individuales o atributos.

El directorio activo también soporta la creación de sitios, los cuales son grupos físicos más que lógicos, definidos por una o más subredes. Estos sitios son independientes del dominio y a estructura OU, y son comunes por todo el bosque. Se utilizan para controlar el tráfico de red generado por replicación, y también para referir a los clientes al controlador de dominio más cercano.

RECURSOS COMPARTIDOS

Desde una perspectiva de usuario, los recursos compartidos son una de las principales funciones de los sistemas Windows. Un recurso compartido es una conexión (normalmente remota) a un dispositivo de la red, como por ejemplo un disco duro o una impresora. Los usuarios pueden acceder a los recursos a través del Explorador de Windows o haciendo Doble Clic en el icono del Entorno de red en el escritorio, sin embargo es bueno conocer otra alternativa que se hace a través de la ventana de comandos mediante la instrucción “net use”, la cual tiene la siguiente sintaxis:

C:\> Net use \\ [Dirección IP o Nombre de Hosts] \ [Nombre del Recurso] – [Nombre de Usuario] : [Contraseña]
--

Una vez conectado a un recurso, los usuarios pueden acceder los objetos (Ejemplo, archivos, directorios o carpetas y demás), dependiendo, claro, de los permisos particulares que se aplican a estos objetos.

SERVICES PACKS Y ADVERTENCIAS CRITICAS (HOT FIXES)

Cuando se descubren vulnerabilidades, cada fabricante del sistema operativo libera actualizaciones y arreglos para cada producto del mismo; Microsoft no esta exento de esta regla. Los arreglos y actualizaciones para WINDOWS 200X/XP/VISTA/7 se clasifican en dos tipos: Service Pack y Advertencias Criticas (Hot Fixes).

Los Services Packs son, en efecto, un grupo de Advertencias Criticas integrados en un solo paquete de instalación, mientras que una Advertencia Critica solo se dirige a un problema específico como una falla en la programación que permite a un atacante romper los sistemas remotamente. Estos se pueden descargar de Internet permanentemente o se delega la función para que se haga de manera automática a través de la utilidad de Windows Update. Las Advertencias Críticas están incorporadas en los Services Packs, pero no de forma inmediata. Normalmente, los Service Packs son lanzados cuando ha pasado un tiempo razonable después del Service Pack anterior (Ejemplo, seis meses a un año).

La gran parte de administradores de sistemas, no se preocupan por estas actualizaciones por lo que un atacante podria perfectamente ensayar sus scripts para alterar o denegar el acceso a los sistemas.

CONTROL DE CUENTAS DE USUARIO

El Control de Cuentas de Usuario (UAC por sus siglas en ingles) es una tecnología e infraestructura de seguridad que Microsoft introdujo con Windows Vista. Su objetivo es mejorar la seguridad de Windows al impedir que aplicaciones maliciosas hagan cambios no autorizados en el ordenador.

Como Windows no puede diferenciar entre un usuario haciendo click sobre un botón y un programa haciendo click sobre un botón, la UAC fue implementada inicialmente para siempre advertir al usuario via una ventana de dialogo mostrada en un Escritorio Seguro (Secure Desktop), similar a la pantalla de inicio de sesión, sobre cualquier cambio en la configuración del sistema.

Windows 7, sin embargo, ahora incluye la posibilidad de configurar UAC para ocultar estos – molestos a veces – avisos cuando los usuarios cambien configuraciones de Windows. Mientras que este modo aun asegura que las aplicaciones normales no puedan sobre escribir completamente alguna llave del registro, Microsoft ha permitido que los usuarios cambien cualquier configuración de Windows sin ningún aviso advirtiendo de aquello. Sí, incluso se puede cambiar la configuración de UAC – desactivar – de tal modo que Windows no advierta nunca al usuario de estos cambios que se están llevando a cabo en el sistema.

ARQUITECTURA

La arquitectura de Windows 200X/XP/7 está dividida en dos modos, Modo del Usuario y Modo del Kernel. A continuación explicaremos el trabajo de cada uno de estos modos.

MODO USUARIO

Esta capa está compuesta de subsistemas que pasan los requerimientos de Entrada y Salida (E/S) al controlador del kernel mediante los servicios de sistema de E/S. Una aplicación siempre se ejecutará en este modo, el cual actúa como un intermediario entre las aplicaciones y los componentes del modo kernel. Este modo se divide además en el subsistema de entorno y el subsistema integral. Las aplicaciones escritas para varios sistemas operativos pueden ejecutarse sobre Windows 200X/XP/7 usando los Subsistemas de Entorno. La Interfaz de Programación de Aplicación (Application Programming Interface, API) pasa los llamados hechos por la aplicación, los cuales después de ser recibidos por el subsistema de entorno, son pasados a los componentes de ejecución del modo kernel. Los subsistemas de entorno están limitados a una dirección ya asignada y no tienen contacto directo con el hardware o los controladores de dispositivos. Comparados con el Modo Kernel, poseen una baja prioridad de ejecución y utilizan espacio del disco duro como memoria virtual cada vez que el sistema necesita memoria. El subsistema integral ejecuta varias funciones, entre las cuales se encuentran la seguridad, los servicios de la estación de trabajo y los servicios del servidor.

Este modo contiene también el Subsistema de Seguridad, también conocido como la “Autoridad de Seguridad Local (LSA)”, tiene un papel crítico en la seguridad de WINDOWS 200X/XP/7. Este subsistema de Modo de Usuario determina si los esfuerzos del inicio de sección son válidos. Cuando un usuario entra su Nombre de Usuario y la Contraseña durante el proceso de inicio de sección, el Subsistema de Seguridad envía estos datos para facilitar el llamado al Manejador de Cuentas Seguras o SAM. El SAM tiene una base de datos que se identifica con su nombre “Base de Datos SAM”. Normalmente, en esta base de datos está compuesta por dos parámetros para cada usuario, uno (llamado LM contraseña de representación) que contiene una representación de la contraseña del usuario para los propósitos de compatibilidad dirigida hacia los productos menos sofisticados de la casa Microsoft, como LanMan y Windows para Trabajo en Grupo.

El otro parámetro en la base de datos de SAM se llama los "Windows hash – (Picadillo de Windows)" y contiene la contraseña cifrada la cual es necesaria para la compatibilidad con los sistemas Windows 200X/XP/7. Este archivo podrá ser encontrado en la siguiente ruta: \WINDOWS\SYSTEM32\CONFIG, y tiene el siguiente esquema:

```
jca:1011:3466C2B0487FE39A417EAF50CFAC29C3:80030E356D15FB1942772DCFD7DD3234:::  
alfredof:1000:89D42A44E77140AAAAD3B435B51404EE:C5663434F963BE79C8FD99F535E7AAD  
8:::  
william:1012:DBC5E5CBA8028091B79AE2610DD89D4C:6B6E0FB2ED246885B98586C73B5BFB  
77:::  
silvia:1001:1C3A2B6D939A1021AAD3B435B51404EE:E24106942BF38BCF57A6A4B29016EFF6:::
```

Observe que cada línea consiste de un juego de entradas: el nombre de cuenta, un número único de identificación conocido como el ID relativo, la contraseña de representación LM, el Windows Hash, y varios campos opcionales. Cada uno de estos campos está separado por dos puntos.

Las contraseñas de representación LM y los Windows hash para cada cuenta, se genera básicamente de dos formas diferentes. En WINDOWS, la longitud de la contraseña es máxima de 14 caracteres. Efectivamente, un usuario puede teclear más de 14 caracteres para una contraseña, pero el sistema arrastrara algunos caracteres para solo tomar en cuenta una contraseña real de 14 caracteres.

La representación de LM se genera ajustando la longitud de la contraseña a exactamente 14 caracteres, o anulando los caracteres de exceso o insertando caracteres en blanco. Entonces, la cadena resultante está dividida en dos partes iguales: Un carácter de paridad (necesario para la Norma de Encriptación de Datos [DES]) que se agrega a cada parte, y cada parte se usa como una llave para el encriptación de DES de un número hexadecimal. La representación LM es increíblemente débil, un atacante puede perfectamente tomar cada parte de la clave (7 caracteres de los 14 de la contraseña) para formar la representación LM a partir de suposiciones.

Los Windows hash son mucho más fuertes, pero no imposibles de descifrar. Al igual que LM la longitud de la contraseña se ajusta a 14 caracteres exactamente. El algoritmo de encriptación MD-4 (MD-5) usa tres permutaciones sobre la contraseña original para dar como resultado una contraseña derivada o picadillo de la contraseña (Windows hash).

Hay una notable falla en el algoritmo que produce los Windows hash, y es el de no poseer un numero grande de permutaciones lo que hace que se puedan realizar ataques a base de diccionario sobre la base datos SAM

MODO KERNEL

Aunque ambos modos tienen seguridad incorporada, el Modo Kernel, reservado para la funcionalidad del sistema operativo (incluso el acceso a la memoria y hardware) es el más seguro de los dos. Este contiene unos subsistemas como son el Manejador de Entrada y Salida, El manejador de Objetos, el Monitor de Seguridad, el Manejador de Procesos, Las Llamadas de los Procedimientos Locales, el Manejador de la Memoria Virtual, y el subsistema de los controladores de la Interface Gráfica.

De todos estos subsistemas, el Monitor de Seguridad es el más importante obviamente desde nuestro punto de vista. Se encarga de verificar, aprobar o rechazar cada esfuerzo por acceder al Modo kernel, el monitor de seguridad sirve como un tipo de "guardián" del modo kernel, pero este también desempeña una función paralela para el usuario inicial y programas basados en objetos como son los archivos y directorios. Verifica que los usuarios y programas tengan los permisos apropiados antes de permitir el acceso a los objetos. Finalmente, define como audita el sistema para traducir la captura en tiempo real de los eventos ocurridos en la máquina (Even Log).

Mucha funcionalidad de sistemas operativos WINDOWS (incluyendo el Monitor de Seguridad) es basada en el Manejador de Objetos, el cual es un subsistema crítico que maneja la información sobre los objetos dentro del sistema. Los objetos incluyen archivos, directorios, dispositivos como impresoras, usb, DVD entre otros. El manejador de objetos asigna un Identificador de Objeto (OID) a cada uno cuando se crea por primera vez. Este OID perdura por el tiempo de vida del objeto y se usa para referirse a él. Siempre que un objeto se elimine (por ejemplo, cuando un usuario arrastra el icono de un archivo a la papelera de reciclaje, y luego vacía la papelera), el manejador de objetos anula el OID asignado para dicho archivo.

Finalmente, el Modo Kernel también incluye la Capa de Abstracción de Hardware (HAL). Una ventaja del HAL es que un DVD de WINDOWS puede ser instalado en diferentes plataformas de hardware.

En resumen podemos decir que el Modo Kernel está compuesto por todos los controladores y dispositivos de hardware, los cuales son los bloques de construcción de Windows 200X/XP/7.

CUENTAS Y GRUPOS

Las cuentas y los grupos son los puntos centrales en la seguridad de cada sistema operativo, incluyendo a WINDOWS 200X/XP/7. Cuentas con accesos inapropiados y grupos con privilegios inapropiados, pueden facilitar acceso ilimitado a los atacantes.

A continuación exploraremos las consideraciones de seguridad relacionados con los grupos y las cuentas de usuario.

LAS CUENTAS

En WINDOWS 200X/XP/7 hay dos tipos de cuentas: las cuentas predeterminadas y las cuentas creadas por los administradores.

Las Cuentas Predeterminadas

En los sistemas WINDOWS, se crean dos cuentas automáticamente cuando el PDC se instala y ellas son el Administrador y el Invitado. La cuenta Administrador tiene el nivel más alto de privilegios que se puedan considerar, es como la cuenta “root” de UNIX/LINUX. Es posible usar la función de Copia con WINDOWS GUI para crear cuentas adicionales con los privilegios del administrador, o alternativamente, crear completamente nuevas cuentas que son incluidas en el Administrador del Dominio para lograr el mismo efecto.

Una propiedad interesante de la cuenta Administrador es que, por defecto, no puede bloquearse no importa cuántos intentos fallidos en contraseñas supuestas o mal digitadas se utilicen. Adicionalmente, esta cuenta nunca podrá eliminarse, y sólo puede desactivar desde otra cuenta no deshabilitada con propiedades de Administrador. Crear más de una cuenta administradora es esencial; ya que la cuenta administradora por defecto, puede recibir ataques de fuerza bruta para suponer la contraseña y crear una cuenta sin privilegios y una cuenta administradora por cada Administrador es una buena práctica de seguridad que permite responsabilizar a cada individuo por las acciones hechas con propiedades de Administrador, en este caso cada administrador usaría su cuenta sin privilegios para el acceso normal al sistema; cuando requiera realizar acciones de administración solo cerraría su sección y pasaría a la cuenta administradora.

La segunda cuenta predeterminada en el sistema es la cuenta Invitado. Si se habilita esta cuenta, es un blanco fácil para los atacantes informáticos. Afortunadamente, por defecto esta cuenta está deshabilitada. Como la cuenta del Administrador, la cuenta de Invitado no puede eliminarse. Por las razones de seguridad, es conveniente mantener desactivada dicha cuenta.

Otras Cuentas

Las cuentas adicionales, como cuentas del usuario o cuentas para servicios específicos o aplicaciones, pueden ser creadas por administradores por necesidad. Muchas aplicaciones también crean cuentas durante la instalación. Mientras la cuenta Administrador y la cuenta Invitado tienen muchas restricciones, cualquier cuenta adicional puede desactivarse o puede eliminarse sin estas restricciones.

Estrategias Usadas En Sitios Seguros Sobre Las Cuentas

A simple vista las medidas expuestas a continuación pueden parecer demasiado simples, pero la realidad es que complica aún más el trabajo de un atacante a la hora de irrumpir en nuestro servidor. Una primera estrategia es el renombramiento de la cuenta del “Administrador” creado por defecto por el sistema y asignarle otro nombre como por ejemplo un nombre de usuario ficticio, esto hará que dicha cuenta sea menos visible para un atacante (claro, que un atacante experimentado puede determinar el nombre de una cuenta administradora rápidamente a través de un programa de fácil acceso llamado escáner de vulnerabilidades). También, si el nombre de la cuenta del Administrador se cambia, es una buena idea cambiar la descripción de dicha cuenta.

Una segunda estrategia es crear una cuenta sin privilegios llamada “Administrador” para actuar como una cuenta señuelo. Los atacantes pueden perseguir esta cuenta que tiene una contraseña de difícil suposición y cuyos privilegios de acceso son sumamente limitados. Con esta cuenta señuelo es posible examinar los datos de seguridad ubicados en los log del sistema para determinar si alguien ha intentado atacar la cuenta del Administrador.

GRUPOS

En la mayoría de las versiones de WINDOWS 200X/XP/7, se usan los grupos para controlar el acceso y privilegios de un conjunto de usuarios. La razón de ser de esto, es para cada usuario que ingrese al sistema se genera un esquema el cual controla su acceso y sus privilegios, estos esquemas colocan el sistema demasiado lento y pesado cuando tiene una gran cantidad de usuarios, por lo que la solución es asignar un esquema a un grupo de usuarios.

Los sistemas WINDOWS tienen dos tipos de grupos, grupos globales y locales. Los grupos globales permiten el acceso potencialmente a cualquier recurso en cualquier servidor dentro de un dominio. Los grupos locales permiten solamente el acceso en el servidor o puesto de trabajo en que fueron creados.

Los Grupos Por Defecto

Son varios los grupos que se crean por defecto cuando el PDC se instala. Algunos de éstos son grupos locales mientras otros son globales. Estos grupos (la mayoría tiene un nombre auto explicativo, salvo el grupo de Replicadores controla la función de soluciones a tolerancia a fallos) se muestra en el siguiente cuadro:

GRUPOS LOCALES	GRUPOS GLOBALES
Administradores	Administradores de Dominio
Usuarios Avanzados	Usuario de Dominio
Duplicadores	
Invitados	
Operadores de Copias	
Usuarios	

Más allá de estos grupos predefinidos, hay también grupos especiales pensados para controlar ciertos tipos de funcionalidades del sistema.

CONTROL DE PRIVILEGIOS

En Windows 200X/XP/7, la capacidad de acceder y manipular las diferentes utilidades del sistema en forma colectiva, es conocida como “privilegios”. Los privilegios está compuesta de dos áreas: los derechos y las habilidades. Los derechos son las cosas que los usuarios pueden hacer para agregar o revocar cuentas de usuarios y grupos (con algunas restricciones). Las habilidades por otro lado, no pueden agregarse o revocarse; son las capacidades incorporadas de varios grupos que no pueden alterarse. Los grupos definidos previamente vienen con un nivel particular de derechos y habilidades.

Hasta donde pueden llegar los privilegios de los usuarios logueados, los privilegios del Administrador son el nivel más alto de cualquier sección en WINDOWS 200X/XP/7, actuando un poco como la cuenta del “root” UNIX/LINUX.

BIBLIOGRAFÍA

LINUX

- SANCHEZ PRIETO, Sebastián; GARCIA POBLACION, Oscar. Linux Guía Practica.
- CARAZO GIL, Francisco Javier. Ubuntu Linux. Instalación y configuración básica en equipos y servidores.
- MCCARTY, Bill. SELinux.
- TURNBULL, James. LIEVERDINK, Peter. MATOTEK, Dennis; GONZÁLEZ CRUZ, Sergio Luis. Administración de Sistemas Linux.
- ADELSTEIN, Tom; LUBANOVIC, Bill. Administración de Sistemas Linux.

WINDOWS

- DOMÍNGUEZ ALCONCHEL, José. Microsoft Windows7: Guía de Información.
- RAYA CABRERA, José Luis; MARTÍNEZ RUIZ, Miguel Ángel; RAYA GONZÁLEZ, Laura. Aprenda Microsoft Windows Server 2003.
- RAYA CABRERA, José Luis; RAYA GONZÁLEZ, Laura; MARTÍNEZ RUIZ, Miguel Ángel. Domine Microsoft Windows Server 2008.
- PÉREZ, César. Guía de campo de Microsoft Windows XP (SP2).
- PARDO NIEBLA, Miguel. Windows XP Professional
- CRAIG, Zacker. Planning and Maintaining a MS Windows Server 2003 Network
- CHARTE, Francisco. Windows 7. Registro y Configuración.
- STANEK, William R. (1966-); CABRERIZO PASCUAL, María. Windows 7. Guía de Configuración.
- TEMPRADO MORALES, José Carlos. Windows Server 2008. Registro y Configuración.
- MARTOS RUBIO, Ana. Windows XP.

ENLACES

www.microsoft.com/es/es/default.aspx
www.fedoraproject.org/es/
www.debian.org/index.es.html
www.ubuntu.com
www.opensuse.org/es/
www.redhat.com/
www.kubuntu.org/
www.mandriva.com/
www.gentoo.org/
www.oracle.com/us/products/servers-storage/solaris/index.html
www.dragongar.org

TABLA DE CONTENIDO

AMBIENTES OPERATIVOS.....	1
<i>LINUX/UNIX</i>	<i>1</i>
<i>ESCOGIENDO UNA DISTRIBUCIÓN DE LINUX</i>	<i>2</i>
<i>ARQUITECTURA</i>	<i>5</i>
SISTEMAS DE ARCHIVOS	5
ESTRUCTURA DEL SISTEMA DE ARCHIVOS	10
EL KERNEL Y LOS PROCESOS	12
PONIENDO EN MARCHA PROCESOS AUTOMÁTICAMENTE	13
INTERACTUANDO CON LOS PROCESOS.....	19
CUENTAS Y GRUPOS	20
CONTROL DE PRIVILEGIOS	24
PROGRAMAS SUID Y SGID	27
RELACION DE CONFIANZA EN MAQUINAS LINUX/UNIX	28
SERVICIOS COMUNES DE UNIX/LINUX	30
<i>Laboratorio Práctico</i>	<i>31</i>
<i>AMBIENTE OPERATIVO WINDOWS NT/XP/200X/VISTA/7.....</i>	<i>32</i>
<i>UNA BREVE HISTORIA EN EL TIEMPO</i>	<i>32</i>
<i>Windows 2008 Server</i>	<i>33</i>
<i>Windows 7</i>	<i>34</i>
<i>WINDOWS XP PROFESIONAL.....</i>	<i>35</i>
<i>CONCEPTOS FUNDAMENTALES DE SERVIDORES WINDOWS Y SISTEMAS</i>	
<i>WINDOWS 7</i>	<i>37</i>
DIRECTORIO ACTIVO	37
RECURSOS COMPARTIDOS	38
SERVICES PACKS Y ADVERTENCIAS CRITICAS (HOT FIXES).....	39
CONTROL DE CUENTAS DE USUARIO	39
ARQUITECTURA	40
CUENTAS Y GRUPOS	42
CONTROL DE PRIVILEGIOS	45
BIBLIOGRAFÍA	46