Seguridad de Servidores

Primer Taller CEDIA

3 de Marzo, 2004

Presentado por Hervey Allen
Network Startup Resource Center

Compendio

- Seguridad fisico
- Contrseñas seguras
- Los “firewalls” y protegiendo servicios
- Mejorar y parchar software
- Prevencion del ataque de “buffer overflow”
- Archivos de “logging” y avisos
- Chequeos de integridad del sistema
- Respaldos
- Seguridad del clientes
- Detectar un intruso en tu sistema
- Seguridad por obscuridad

Seguridad desde el principio

- Haz “seguro” un servidor *antes* de conectarlo a una Red.
- Incluye cosas asi:
  - Apagar servicios no necesario.
  - Usa un “tcpwrapper” con los servicios de red.
  - Planificar y instalar reglas de netfilter.
  - Instalar paquetes de controlar usuarios (cuotas, chequeos de contraseñas).
  - Aplicar parches a servicios que va a correr y al sistema operativo.
  - cont.-->

Seguridad desde el principio cont.

- Instalar sistema de detectar intrusion.
- Instalar biblioteca encontra ataques de buffer overflow.
- Configurar como va a correr y usar los logs.
- Determinar su sistema de respaldo.
- Seguridad de clientes.
- Seguridad fisico! No lo olvido.
- Inscribe en las listas de correo acerca seguridad y tu sistema operative.

Apagar servicios no necesarios

Esto es clave por la seguridad de tu sistema.

Usa los comandos asi:

- ps -aux | more
- lsof -i
- netstat -natup

Despues investigaciones si necesario usando /etc/services, man nombre, /etc/rc.d/init.d/, /etc/xinetd.d

Apagar servicios no seguros

Se puede discutir cual servicios son, pero tipicamente hablamos de:

- telnet
- ftp
- rpc
- nfs
- sendmail

Correr servicios con tcpwrapper

- Un tcpwrapper es un programa como xinetd.
- Antes usamos inetd.
- /etc/rc.d/init.d/xinetd
- /etc/xinetd.d
- Se configura servicios dentro este directorio.
- Se lo apaga con “disable = yes”
- Que provee xinetd? -->

Para que correr xinetd?

- El servicio (daemon) de xinetd “escucha” por los paquetes por todo los servicios mencionados en los archivos de configuracion en /etc/xinetd.d
- Se ahora memoria y recursos. Pero, por un servicio cargado (como httpd) mejor no usar xinetd.
- Se puede controlar varios aspetos de connecion usando el “super servidor” xinetd.

Algunos parametros de xinetd

- **Disable:** si el servicio corre o no cuando xinetd esta en uso.
- **Wait:** si o no corre multiple daemons si hay mas de una connecion.
- **User:** bajor que usuario corre el servicio.
- **Instances:** Numero maximo de connecions permitido al servicio. Si wait = nowait el defecto es “no limite” por instantes.
- **Server:** El nombre del programa de correr cuando esta conectado el servicio.
- **only_from:** Especifica de donde se accepta conecciones.
- **no_access:** Especifica de dondes no se accepta conecciones.

Parametros de xinetd cont.

- **Interface:** puede decir en que dispositivo de ethernet va a responder el servicio.
- **cps:** “cps = 10 30” significa accepta hasta 10 conecciones y si hay mas apaga el servicio por 30 segundos.
- Tambien hay los parametros Id, Type, socket_type, Protocol, Group, server_args, log_type, log_on_success, Port.
- Xinetd permite bastante control al nivel de aplicacion para controlar algunos servicios de red.
- Solamente tiene que especificar los parametros “socket_type”, “user”, “server”, y “wait” en un archivo de configuracion por xinetd.

Correr reglas de Netfilter?

Netfilter es iptables es “firewall” en la mente de la mayoria de las personas.

Si tienes un servidor que no esta pasando paquetes de IP a un red privado, no es necesario correr una tabla tan grande de Netfilter.

Se puede correr reglas para proteger encontra paquetes no muy bien formados, etc.

Rechazar icmp? (ping)

Controlar los usuarios

Bueno, esto es imposible :-) Pero, tal vez se puede restringir las contraseñas que usan y el espacio que toman.

Implentar reglas de contraseñas. Un ejemplo:

- Contraseñas nuevas deberian tener entre 6 a 14 caracteres.
- No se puede eligir una palabra, nombres, lugares, o datos personales.
- Una contraseña segura deberia contener, por lo minimo, dos numeros, dos letras, y unz mezcla de mayuscalas y minuscalas.

Controlar los usuarios cont.

Implementar cuotas a los usuarios. En Red Hat ya esta instalado el aporte para implementar las cuotas.

- /etc/fstab usrquota y/o grpquota
- /part/quota.user y /part/quota.group
- edquota -u usuario para congigurar las cuotas.
  - edquota -p artc 'awk -F: '$3 >> 499 print $1' /etc/passwd'
- quotacheck -avug
- quotaon -avug
- quotacheck, repquota, quota

Controlar los usuarios cont.

Para que los usuarios tienen que usar una contraseña bastante buena se puede usar PAM (Pluggable Authenticion Module) y cracklib.

- PAM con cracklib (PAM ya esta en Red Hat 9)

Quebrar las contraseña despues que existen:

- John the Ripper: http://www.openwall.com/john/
- Crack: http://www.crypticide.org/users/alecm
- Slurpie: http://www.ussrback.com/docs/distributed/

Aparchar y actualizar software

Aparchar o aplicar los mejoramientos al software que va a correr y a tu kernel.

En el mundo Red Hat busca el directorio “udpates”

Desde que Red Hat 9 ha salido hay 500MB de updates.

Revisa los updates y decide cuales tiene que aplicar.

Por servicios importante vaya a su sitio de web y inscribe en listas de correo para saber siempre cuando sale un update – especialmente por seguridad.

Sistemas de detectar intrusos

Tambien, se dice “Sistemas de probar la integridad de sistemas.”

En este caso estamos hablando de sistemas por tu servidor y sus archivos, no por la red.

Hay tres sistemas bien populares hoy en dia. Son:

- tripwire: http://www.tripwire.org/
- AIDE: http://www.cs.tut.fi/~rammer/aide.html
- Snort: http://www.snort.org/

Ataques de Buffer Overflow

Aprovechando del estado de memoria de un programa. Usando mas memoria temporia que hay, y el programa no lo atrapa bien este condicion. Se falla en una forma que permite el intruso tomar el proceso con los permisos del usuario de que pertenece al programa.

El proyecto libsafe (http://www.research.avayalabs.com/project/libsafe/) tiene una solucion sensible. Se lo instala su software y esto es todo.

Ataques de Buffer Overflow cont

Otro software encontra este tipo de ataque incluyen:

- **Openwall:** http://www.openwall.com/
- **Stackguard:** http://www.cse.ogi.edu/DISC/projects/immunix/StackGuard/linux.html
- **WireX:** http://immunix.org/

Son mas complicado para instalar, pero pueden ser mas completo, o util por los servidores que tienen que ser paranoiacos para mantenar seguro datos.

Logging y Reportajes

- Red Hat Server 9 ya tienen /etc/syslog.conf bien hecho. Lea “man syslog.conf” por mas detallas.
- Red Hat Server 9 corre cada dia el servicio “logwatch” que manda un mensaje al root con un reportaje hecho de los logs en /var/log.
- Vea /etc/log.d/logwatch.conf.
- Donde van los logs es importante.
- Respaldos de los logs es importante.

Sistema de respaldo

Tecnicamente no es seguridad, pero sin respaldo de los datos que haces si hay un intruso?

- Que tienes que respaldar?
- Cuantas veces tienes que correr un respaldo?
- Donde va la media de respaldo en caso de desastre?
- Que pasa en caso de desastre (terremoto?).
- Que herramientas vas a usar (tar? arkeia? cpio?)

Seguridad de los clientes

Esto es supremamente importante!

Mucha gente olvida esto. Pero un cliente (usuario que tiene acceso a tu sistema) que no es seguro es un riesgo y oyo de seguridad grande a tu servidor.

Insiste en pop/imap por ssl, ssh, scp, sftp, https, contraseñas buenas.

No use pop/imap sin ssl, webmail por http, ftp (menos anymous), ni telnet.

Seguridad fisico

Todo la seguridad en el mundo no te va a ayudar si alguien tiene acceso fisico a tu servidor y si este persona quiere causar daño.

Considera quien tiene acceso. Llaves. El espacio fisico. Donde estan los respaldos? Los logs?

Tienes una contraseña en el bootloader y un sistema de archivos encifriado? No importa, la persona puede llevar tu computador!

Detectando un intruso

Que pasa si, ya, alguien entro tu sistema y hico cambios? Como va a saber?

- Tal vez nunca vas a saber. Sin un sistema de detection de intruso, como Tripwire, es dificil saber.
- Si un programa como “ls” no corre.
- Si “netstat -natup” muestra algo como BASH escuchando a un puerto de tcp.
- Si, de repente, vea mucho mas actividad en el disco duro, por el camando “top” o en la red.
- Si esta corriendo un proceso de red escuchando por un puerto > 1024

“Seguridad por osbscuridad”

“Security from obscurity”

¡No funciona!

Monitorea tu red y vea cuanta gente esta probando los puertos de tcp/udp en tu red publico – es impresionante!

Recursos

Mas software:

- COPS: ftp://ftp.cert.org/pub/tools para revisar tu sistema y buscar cosas problematics.
- Listas de Correo/Sitios de Web:
  - CERT: http://www.cert.org/
  - BugTraq: http://www.securityfocus.com/
  - Rootshell: http://www.rootshell.com/
  - Red Hat: http://www.redhat.com/

Recursos cont.

- http://nsrc.org/security/
- Los sitios de web por cada servicio que corres:
  - http://www.apache.org/
  - http://www.ssh.com/
  - http://www.openssh.org/
  - http://www.isc.org/ (BIND)

Resumen

En resumen, para hacer “seguro” tu servidor haz lo siguiente:

- Apagar servicios no necesarios
- Apagar servicios peligrosos
- Usa contraseñas seguras
- Considera sistemas de detectar intrusos.
- Considera sistemas de chequear integridad de sistema.
- Usa logs. Protega tus logs.
- Aplica los parches encontra los ataques de buffer overflow.

Resumen cont.

- Aplicar los parches a los servicios que corres y al kernel.
- Usa un servicio de tcpwrapper, como xinetd.
- Corre respaldos!
- Considera usando reglas de netfilter.
- Inscribe al listas de correo como bugtraq y cert.
- Instala software para controlar tus usuarios como cuotas y chequeo de contraseñas.
- Insiste en clientes seguros que usan metodos de criptografia (ssh, scp, sftp, https, pop/imap/ssl) .
- Seguridad fisico de tus servidores.

Entiendes que haces

- Una solucion de seguridad mala es peor que no solucion.

- Entiendes que haces:
  - Lea todo la documentacion
  - Lea las configuraciones de muestras
  - Construye maquinas de pruebas
  - Haz preguntas
  - Inscribe en las listas de correo de anuncios por tu sistema operativo y por las aplicaciones que corres.
  - Trata de conectar desde afuera tu propia red.
  - Trata de evitar tus proprias reglas.

Conclusion

Hay una regla en el mundo de seguridad:

- Mas seguro = mas molestia

Cada tapa de seguridad que impones, en general, lo hace mas dificil para hacer tu trabajo.

Hay otra regla en el mundo de seguridad:

- Menos seguro = desastre esperando

Solo requiere una quebrada de seguridad para que tus datos pueden ser destruidos, tu negocio puede fallar, o tu trabajo puede estar en peligro.

Conclusion cont.

Con un poco trabajo se puede mejorar el nivel de seguridad de un servidor mucho.

Es una molestia, pero desafortunadamente es una molestia necesaria.

Pero, Linux esta hecho en una forma bien modular que lo hice mucho mas facil de hacer seguro. Y, mientras que haces seguro tu servidor, tambien, haces tu servidor ordenado y mas facil de administrar.