UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES

ARAGÓN"

"ESQUEMA DE SEGURIDAD PROPUESTO EN LA ELABORACIÓN DE UNA NOMINA."

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A :

EFRAÍN BARRAGÁN GONZÁLEZ

MÉXICO, 2000.

Agradecimientos

Primeramente quisiera darle gracias a Dios por darme la oportunidad de vivir y poder realizar esta carrera...

A mis padres, Gloria y Antelmo, por darme su confianza y su apoyo en todo momento. Doy gracias a Dios por darme unos padres tan caritativos, bondadosos, generosos, humildes; con una fortaleza inmensa y con un gran espíritu de lucha (del que aprendí mucho)...todo se los debo a ustedes...

A mis hermanos, Elizabeth, Juan Pablo, Josue, Raquel y Gloria, porque me siento orgulloso de ser su hermano, porque los admiro y siempre me impulsaron y recordaron que tenía que terminar este trabajo...

A Erika, por ser comprensiva, paciente y brindarme su amor (por ser mi corbata guinda cuando mi traje era oscuro)...

A mi tío Roberto por sus consejos, su confianza y por la esperanza que me hizo sentir cuando tuve apuros económicos...

A mis abuelitos Leoncio y Josafat, a quienes admiro, respeto y quiero mucho. Les agradezco su ayuda económica también...

A todos mis tíos y tías que me ayudaron de alguna forma (moral o económicamente)...

A mis compañeros de todos los niveles de escolaridad por brindarme su amistad, especialmente a Ana, Edith, Luis, Armando, Arturo, Oscar y José Luis...

A la Universidad y a todos los profesores que contribuyeron en mi formación profesional...

A los directivos de la Nómina por permitirme realizar este trabajo...

A todos mil gracias...

PRÓLOGO

INTRODUCCIÓN 1

CAPÍTULO 1: MARCO TEÓRICO 4

1.1 LA NÓMINA 5

OBJETIVOS DE LA NÓMINA 6
MÓDULOS 7
Captura y Validación de Movimientos de Personal 7
Captura, Carga y Validación de Percepciones y Deducciones 7
Proceso de Movimientos del Personal 8
Cálculo 8
Reportes 8
FUNCIONALIDAD DEL SISTEMA 9

1.2 PRINCIPALES CARACTERÍSTICAS DE LA NÓMINA 13

Tipos de Procesamiento 13
Plataforma 14
Ubicación de Clientes y Servidores 15

1.3 ANÁLISIS DE CONCEPTOS Y REQUERIMIENTOS DE SEGURIDAD 17

CONCEPTOS IMPORTANTES 17
REQUERIMIENTOS DE SEGURIDAD 17
PRIORIDADES 18
¿POR QUÉ SE NECESITA SEGURIDAD? 18
Principales Amenazas 19
Tipos de Atacantes 21
¿Cómo se Puede Proteger el Sitio? 22
Ningún Modelo de Seguridad Puede Hacerlo Todo 24
ESTRATEGIAS DE SEGURIDAD 24
Menor Privilegio 25
Defensa a Fondo 25
Punto de Choque 25
Eslabón más Débil 26
Postura de Falla Segura 26
Participación Universal 27
Diversificación de Defensa 28
Simplicidad 28
NIVELES DE SEGURIDAD 29
Requerimientos de Seguridad Fundamentales 29

1.4 ALCANCES DE LA PROPUESTA 34

CAPÍTULO 2: SEGURIDAD FÍSICA 35

2.1 NECESIDADES DE SEGURIDAD 36

ANÁLISIS DE RIESGO 36
ROBO 37
AMBIENTE 37
Fuego 38
Humo 38
Polvo 38

Temblor 38
Temperaturas Extremas 39
Insectos 39
Ruido Eléctrico y Fallas Eléctricas 39
Rayos 40
Vibración 40
Humedad 40
Agua 40
DESCUIDOS 41
Comidas y Bebidas 41
2.2 UBICACIÓN DEL EQUIPO DE CÓMPUTO 42
Identificación de los Puntos de Acceso 43
2.3 PROTECCIÓN DEL EQUIPO DE CÓMPUTO 46
ACCESO A LAS COMPUTADORAS 46
EL MEDIO AMBIENTE 47
Temperatura 47
Fuego y Humo 48
Polvo 48
Temblor 48
Insectos 49
Ruido Eléctrico y Fallas Eléctricas 49
Rayos 50
Vibración 50
Humedad 50
Agua 51
MANTENIMIENTO PREVENTIVO 52
PREVENCIÓN DE ACCIDENTES 52
2.4 CONTROL DE CINTAS 53
2.5 BITÁCORAS E INVENTARIOS 54
CAPÍTULO 3: SEGURIDAD EN EL SISTEMA OPERATIVO 55
3.1 CARACTERÍSTICAS DE SEGURIDAD DEL SISTEMA OPERATIVO 56
ANTECEDENTES 57
EL ACCESO AL SISTEMA 59
Seguridad de EEPROM 59
Control de Cuentas de Acceso 59
Entrada Directa de root 62
Seguridad del Shell 65
Accesos con ftp 67
Acceso a los Datos y Dispositivos del Sistema 68
Módulo "Conectable" de Autenticación (PAM) 72
Acceso a la Red 74
MONITOREO DE LA ACTIVIDAD EN EL SISTEMA 78
Auditoría 78
Módulo Básico de Seguridad (BSM) 80
3.2 NECESIDADES DE SEGURIDAD 83
ANÁLISIS DE RIESGOS 83
REQUERIMIENTOS DE SEGURIDAD 88

3.3 CUENTAS DE USUARIO Y SISTEMA DE ARCHIVOS 89
Grupos y Usuarios 89
Sistema de Archivos de Nómina 90
Estructura del Directorio /nomina *91*
Espejos de Sistemas de Archivos 93
3.4 CONFIGURACIÓN DE SERVICIOS 96
Configuración Requerida 96
Seguridad Física *97*
Servicios *97*
Ambiente de root *107*
Ambiente de Operadores de Nómina *107*
Ambiente de Usuarios Externos *108*
3.5 MANEJO DE LA INTEGRIDAD DE LA INFORMACIÓN 109
Prevención 109
Detección 110
Tripwire *110*
3.6 RESPALDOS 114
¿Por qué Hacer Respaldos? 114
Lo que se Debe Respaldar 115
Tipos de Respaldos 116
Estrategia de Respaldos 117
Respaldos Normales *117*
Respaldos de Archivos de Base de Datos *118*
Respaldos Quincenales *119*
Precauciones *119*
3.7 AUDITORÍA Y BITÁCORAS DEL SISTEMA 121
Monitoreo en Tiempo Real del Comportamiento del Servidor 122
3.8 OTRAS TAREAS ADMINISTRATIVAS 123
Soporte Técnico 123
Listas Discusión 123
Manuales 123
CAPÍTULO 4: SEGURIDAD EN LA BASE DE DATOS 124
4.1 CARACTERÍSTICAS DE SEGURIDAD DEL DBMS 125
División de Roles 126
Roles del Sistema *126*
Roles Definidos por el Usuario *127*
Identificación y Autenticación del Usuario 127
Usuarios de Bases de Datos *127*
Creación de Grupos *128*
El Usuario guest *128*
Alias de Usuarios *128*
Usuarios Remotos *129*
Bloqueo de Cuentas *129*
Control de Acceso Discrecional 129
Uso de vistas y Procedimientos Almacenados como Mecanismo de Seguridad *130*
Auditoría 133

ESTADO DE LA SEGURIDAD AL MOMENTO DE INSTALAR EL SERVIDOR 133
4.2 NECESIDADES DE SEGURIDAD 134
ANÁLISIS DE RIESGO 134
Restricciones de Acceso *134*
Disponibilidad *135*
Consistencia *135*
Auditoría *135*
4.3 USUARIOS GRUPOS Y PERMISOS 136
CIFRADO DE PASSWORDS 141
4.4 DISPONIBILIDAD Y CONSISTENCIA DE LA BASE DE DATOS 142
DISTRIBUCIÓN DE LA BASE DE DATOS 142
ESPEJOS 144
VERIFICACIÓN DE CONSISTENCIA DE DATOS Y RESPALDOS. 145
Comandos dbcc *145*
Respaldos *147*
4.5 AUDITORÍA 152
EL SISTEMA DE AUDITORÍA 152
OPCIONES DE AUDITORÍA 153
Observaciones de Tablas de Auditoria *156*
RECOMENDACIONES DE ADMINISTRACIÓN 157
CAPÍTULO 5: SEGURIDAD DE RED 158
5.1 FIREWALLS 159
¿QUÉ PUEDE HACER UN FIREWALL? 161
¿QUÉ NO PUEDE HACER UN FIREWALL? 161
EVOLUCIÓN DE LOS FIREWALLS 162
Filtrado de Paquetes (Primera Generación) *162*
Filtrado de Paquetes (Segunda Generación) *163*
Gateways de Aplicación (Primera Generación) *165*
Gateways de Aplicación Transparentes (Segunda Generación) *167*
Terminología para Servidores Proxy *167*
REDES PRIVADAS VIRTUALES (VPN) 168
5.2 CARACTERÍSTICAS DEL FIREWALL 171
EL FIREWALL GAUNTLET 171
Redes Confiables y Redes no Confiables *171*
Filosofía de Diseño *172*
¿Cómo se Conecta un Firewall Gauntlet? *172*
Transparencia *173*
Sistema Operativo *174*
Filtrado de Paquetes *174*
Servicios de Seguridad a Nivel Aplicación (Proxies) *174*
Registro en Bitácoras *176*
Soporte de Cambio de Dirección de Red (NAT) *176*
Autenticación Fuerte de Usuario *178*
Soporte de Redes Privadas Virtuales (VPN) *178*
Verificación de Integridad *180*
FUNCIONALIDAD DEL FIREWALL 181
Recepción del Paquete *181*

Verificación de Fuente y Destino *181*
Tipo de Petición *181*
Procesamiento de la Petición *182*

5.3 NECESIDADES DE SEGURIDAD 183

5.4 SERVICIOS DE RED 185

SERVICIOS Y VULNERABILIDADES 185
Correo Electrónico *185*
Transferencia de Archivos (FTP) *186*
Terminal Remota (telnet) y Ejecución de Comandos *186*
Protocolo de Transferencia de Hipertexto (HTTP) *187*
Sybase *188*
IDENTIFICACIÓN DE USUARIOS Y SERVICIOS UTILIZADOS 188
Grupos de Hosts y Grupos de Servicios *189*

5.5 UBICACIÓN FÍSICA DEL FIREWALL Y RED SEGURA 195

ESQUEMA DE LA RED GLOBAL (VPN) 197

5.6 ADMINISTRACIÓN DEL FIREWALL 198

ADMINISTRACIÓN 198
Respaldos del Firewall y Verificación de Integridad *198*
Administración de Cuentas *198*
Administración de Espacio en Disco *199*
Mantenimiento Preventivo y Correctivo *199*
MONITOREO DEL SISTEMA 199
¿Qué se Debe Observar? *200*
ACTUALIZACIONES 200

CONCLUSIONES 202

GLOSARIO 203

BIBLIOGRAFÍA 207

Prólogo

Cuando se administra un equipo de cómputo que ofrece servicios de red (y máxime cuando esta red es la internet), es de pensar que se esta expuesto a múltiples amenazas que van desde un ataque pasivo (observar la información que sale del equipo) hasta un ataque activo (tratar de modificar la información o bloquear los servicios).

Tales temores se deben a que, en esta misma red, se encuentran sitios donde se publican los huecos de seguridad en *"x"* servicio y muchas veces se muestra la forma de atacarlo. Esto conlleva a un incremento en la capacidad de cualquier usuario que encuentre estos sitios para entrar de forma ilegal a algún equipo y aprovechar sus recursos a su conveniencia.

A principios de 1998 alguien trato de conectarse desde un modem, en algún lugar de Estados Unidos, al equipo servidor de la Nómina. Posteriormente se registraron nuevos intentos desde algún lugar en la ciudad de México, lo cual sirvió para que se tomara más conciencia acerca de la necesidad de implementar un esquema de seguridad que estableciera un perímetro más robusto y controlado en los servicios que ofrece la Nómina. Así es como nació esta tesis.

Todo fue gradual, primeramente se instaló software de dominio público para proteger de una forma rápida y económica a los equipos de cómputo, en lo que se terminaba de diseñar el esquema de seguridad. Sin embargo, era muy difícil mantener una administración en base a seguridad por host, ya que son muchas las bitácoras que había que revisar y aun se tenía la inconveniencia de que los clientes se conectaban directamente a los servidores, lo que permite una mejor inspección del mismo. Se optó por una seguridad de red, robustesiéndola con una seguridad a nivel host menos estricta y la seguridad en la base de datos. Además, la aplicación misma ya tiene implementada una seguridad en el cliente, lo cual refuerza la seguridad en los servicios.

Posteriormente, después de que una persona de intendencia apagó accidentalmente uno de los servidores, se pensó en reforzar la seguridad física también y se incluyó en la propuesta de seguridad.

Finalmente, cabe mencionar que todo esto no se hubiese logrado sin el apoyo de los directivos de la Nómina, que facilitaron los medios y recursos para una concientización de los usuarios y la implementación de las prácticas recomendadas en este "Esquema de Seguridad Propuesto en la Elaboración de una Nómina".

- La educación al usuario: Para concientizarlos de la importancia de su participación para mantener una aplicación segura, así como detectar y reportar intrusiones.

El objetivo principal de esta tesis es: diseñar un esquema que permita cumplir las expectativas de confidencialidad, integridad y disponibilidad para la Nómina, pero se enfoca esencialmente a la seguridad física y a los servicios (a nivel sistema operativo, base de datos y red), sin tocar la seguridad en los clientes o la educación a los usuarios. Su enfoque es totalmente práctico empleando herramientas que proveen cada una de las capas que se van analizando y también es de prevención, es decir, se analiza que se quiere proteger, de quien se quiere proteger y como protegerse.

La tesis está dividida en cinco capítulos. La metodología utilizada en todos ellos, (excepto el 1) es: conocer las características que ofrecen el sitio, el sistema operativo, la base de datos o el firewall para los servicios de red; conocer las amenazas a las que se encuentran expuestos los servicios o el equipo (en caso de la seguridad física) y una estrategia para minimizar tales amenazas.

A continuación se exponen los temas que se tratan en cada capítulo:

- Capítulo 1: La Nómina. En este capítulo se analizan los módulos que componen al sistema de Nómina, sus características, enfatizando aquellas que dan la base para iniciar el análisis de servicios como la plataforma empleada y la distribución de clientes y servicios; se analizan los requerimientos de seguridad establecidos para la Nómina y se tocan conceptos de seguridad generales.

- Capítulo 2: Seguridad Física. Se enfoca a analizar las amenazas físicas a las que se encuentran expuestos los equipos de cómputo y cómo minimizar el riesgo de disponibilidad, confidencialidad e integridad de la información fuera de línea y los equipos en el centro de cómputo.

- Capítulo 3: Seguridad en el Sistema Operativo. Tiene como objetivo el hacer un análisis de las características de seguridad que tiene el sistema operativo, observar a que amenazas se enfrenta el sistema de Nómina y como lograr que tales características de seguridad las minimicen. Se propone, también, software de dominio público para subsanar aquellas herramientas del propio sistema operativo que son poco flexibles.

- Capítulo 4: Seguridad en la Base de Datos. Contempla, al igual que el capítulo anterior, un análisis de las características de seguridad del DBMS y cómo utilizarlas para minimizar las amenazas.

- Capítulo 5: Seguridad de Red. Trata de las amenazas a las que se encuentra expuesta una aplicación que utiliza la red y cómo minimizarlas con el uso de un firewall.

Cabe señalar que ningún esquema de seguridad puede garantizar que un sistema sea totalmente seguro. Además, está demostrado que el 80% de las violaciones de seguridad[1] se deben a los propios usuarios internos, por lo que todas estas medidas de seguridad no tienen ningún efecto si no se educa y concientiza adecuadamente al usuario. Aunado a esto, se encuentra el hecho que día a día se encuentran nuevos huecos de seguridad, por lo que el administrador debe estar al tanto para saber como reaccionar. No obstante, este esquema permite tener un mejor control en los servicios que se ofrecen, permitiendo a los usuarios realizar sólo aquellas tareas necesarias para realizar su trabajo y poniendo a su disposición sólo los servicios necesarios.

[1] http://www.sun.com

Capítulo 1: Marco Teórico

1.1 La Nómina

Antes de iniciar la descripción de la Nómina es necesario aclarar que, por razones de confidencialidad, no se dará el nombre verdadero de la compañía a la cual se propone este Esquema de Seguridad. En su lugar se le definirá como "La Empresa" y a sus sucursales se les llamará "Sucursal1", "Sucursal2", etc.

La Empresa, cuenta con más de 60,000 empleados de los cuales el 65% trabajan en la casa matriz y el resto se reparte en tres Sucursales. Las Sucursales se encuentran distribuidas en diferentes puntos de la ciudad de México.

Actualmente el procesamiento de la nómina se lleva a cabo en forma centralizada en la casa matriz, utilizando un esquema de procesamiento totalmente en batch. Las entradas de datos son por oficios y discos magnéticos, lo que, en caso de algún error, retarda el trámite de movimientos del personal.

Debido a las nuevas características, tendencias de la tecnología y crecimiento de la misma empresa, se ha rediseñado el Sistema de Nómina para permitir la captura de la información desde cada Sucursal, utilizando una arquitectura cliente/servidor, situando diversos clientes en las sucursales y el servidor en la casa matriz. Con este rediseño se trata de reducir los tiempos de respuesta en los movimientos de empleados.

Objetivos de la Nómina

El desarrollo de la Nómina tiene como principal objetivo: "Crear la infraestructura tecnológica de información que posibilite la descentralización y modernización de sistemas y métodos administrativos, en busca de mayor productividad en la prestación de los servicios de nómina. El sistema a desarrollar deberá permitir la distribución de la captura de la nómina en una o más sucursales. Cada sucursal deberá tener autonomía en la actualización de movimientos de sus empleados y podrá elaborar y emitir sus propios reportes. Deberá ser posible detectar incompatibilidades para personal que labore en dos o más sucursales; se deberá emitir un solo cheque por trabajador para estos casos; y deberá ser posible explotar la información de todas las sucursales. El sistema deberá tener la suficiente flexibilidad para, en un futuro, trabajar tanto en forma centralizada como parcial o totalmente descentralizada"[2].

Este planteamiento implica el intercambio de información entre las sucursales y la casa matriz por las siguientes razones:

- Se necesita consultar la información necesaria (tabuladores, nombramientos, estímulos, deducciones, percepciones, etc.) en la casa matriz desde cada sucursal para que se puedan elaborar sus reportes.

- Los reportes de gastos por conceptos de nómina deberán seguirse controlando centralmente.

- El presupuesto destinado al pago de empleados también se seguirá llevando en la casa matriz.

- Los movimientos de empleados se realizarán en las diferentes sucursales y se mantendrá una base de datos consolidada en la casa matriz.

- Para los empleados que laboran en más de una sucursal, se deberá reunir su pago en un solo cheque y lo pagará la sucursal que él designe.

Antes de entrar en los detalles de seguridad, analicemos brevemente el diseño del sistema de Nómina.

[2] "Definición de Requerimientos de La Nómina". 1995. La Empresa. p. 19.

Módulos

El sistema de Nómina está compuesto por seis módulos:

- Captura y Validación de Movimientos de Personal
- Captura, Carga y Validación de Percepciones y Deducciones
- Proceso de Movimientos del Personal
- Cálculo
- Reportes

Tales módulos se explican brevemente a continuación.

Captura y Validación de Movimientos de Personal

En este módulo se captura la información del contrato del empleado cuando se realizan nuevos ingresos, reingresos, prórrogas, remuneraciones adicionales, aumento de horas, promociones, reanudación de labores, transferencias, disminución de horas, enfermedad, suspensión, renuncia, defunción, jubilación, etc. Esta información se debe validar para verificar que el tipo de nombramiento que viene registrado en el contrato corresponde con el sueldo, el domicilio del empleado, si su total de horas no rebasa las permitidas, etc. La aplicación cliente que realiza estos procesos se encuentra en cada sucursal y alimenta la base de datos central.

Captura, Carga y Validación de Percepciones y Deducciones

El módulo de captura de percepciones y deducciones permite la entrada de información que se refiere a percepciones extraordinarias y deducciones como pagos de tiempo extra, días festivos, estímulos, prima vacacional, aguinaldo, bonos de calidad y eficiencia, descuentos por pago de impuestos, pago de pensión, servicio médico, préstamo a corto plazo, préstamo hipotecario, donaciones, inasistencias, pensión alimenticia, crédito para automóvil, SAR, pago a aseguradoras, faltante en cajas, daños a vehículos, retardos etc., siguiendo las normas establecidas para la aplicación de éstos conceptos así como la consulta de información general del empleado, sus movimientos aplicados y por aplicar, para en caso necesario, realizar la aclaración pertinente. Al igual que el módulo anterior, éste se encuentra distribuido para su operación en cada sucursal.

Proceso de Movimientos del Personal

En este módulo se revisan y se ajustan si es necesario: nombramientos, plazas, estado del empleado (vigente o no vigente), se detectan movimientos con insuficiencia presupuestal, se detectan y clasifican los movimientos que generan situaciones incompatibles en los nombramientos del empleado.

Sobre la base de movimientos retroactivos y únicos pagos se calculan sueldos así como pagos y descuentos retroactivos. También se actualiza la antigüedad, los nombramientos (lo que cobra el empleado actualmente), datos del empleado y kardex (historia de los nombramientos que ha tenido el empleado).

Se calculan los aumentos de sueldo con retroactividad. Basándose en nuevos sueldos, se actualizan los nombramientos y se calculan los pagos y descuentos que originan el aumento. Este proceso se realiza solo en la casa matriz.

Cálculo

Este módulo procesa los movimientos de nómina (percepciones, deducciones, retenciones) aplicables para el periodo quincenal en proceso. Obtiene como resultado la información necesaria para elaborar relaciones y reportes de la nómina, así como los depósitos al personal que deba recibir su pago en cuentas bancarias o cheques.

La terminación exitosa de este proceso se toma como base para acumularlo a la historia de las nóminas que han sido ejercidas. Este módulo se ejecuta en la casa matriz.

Reportes

El módulo de reportes genera todos los reportes especiales y generales de la nómina que sirven como control y apoyo de auditorías. Los siguientes son ejemplos de reportes:

- Reporte de Altas y Bajas de empleados
- Reportes de Depósitos a Bancos
- Reportes de Vales
- Reportes de Nómina por Partida
- Reporte de Descuentos Aplicados a los Empleados
- Licencias por Sucursal
- Reporte de Promociones

Funcionalidad del Sistema

Los módulos de Captura y Validación de Percepciones y Deducciones así como el de Captura de Movimientos de Personal, proporcionan los datos de entrada necesarios para los módulos de Proceso de Movimientos del Personal y el Cálculo de la Nómina. El resultado del Cálculo permite la generación de Reportes y de información necesaria para las demás entidades (aseguradoras, bancos, etc.) con las que interactua la nómina. Durante todo el proceso, los distintos módulos consultan información de los catálogos para validación y proceso de los movimientos de un empleado. Cada módulo da mantenimiento a los catálogos que accesa. Esta interacción entre los módulos se muestra en la siguiente figura:

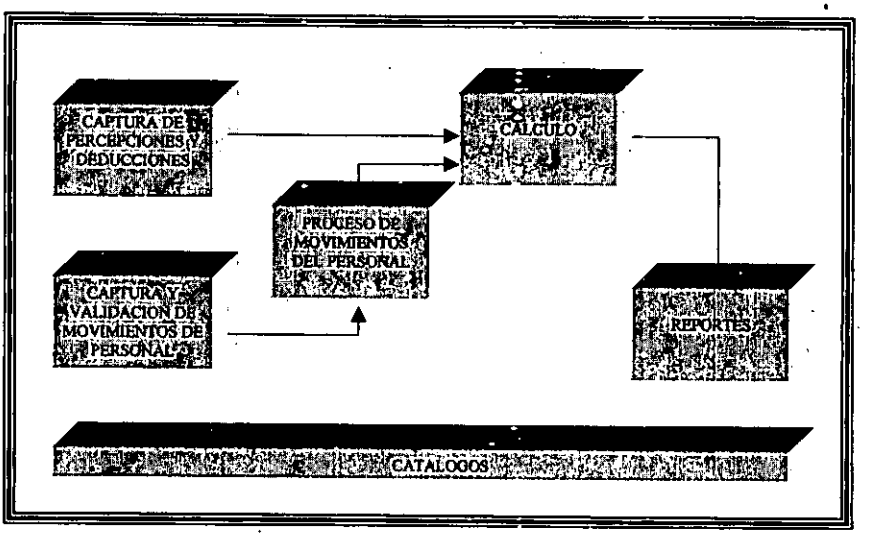

Fig. 1.1 Flujo de datos entre los módulos

La interacción de los módulos en el tiempo (Fig. 1.2) es la siguiente:

Los módulos de captura (Captura de Percepciones y Deducciones y Captura y Validación de Movimientos de Personal), se utilizan siempre y se tienen fechas de cierre de captura para una determinada quincena. Antes de que finalice la quincena, se ejecuta el módulo de Proceso de Movimientos del Personal que deja la información necesaria para el proceso de Cálculo. Los resultados del Cálculo se reflejan al ejecutar los Reportes para las diversas instituciones y dependencias internas con las que interactua, al final de la quincena. La actualización a catálogos pocas veces se realiza, ya que sólo se requiere cuando hay modificaciones a tabuladores, se agregan nuevos conceptos de pago o se modifica la normatividad.

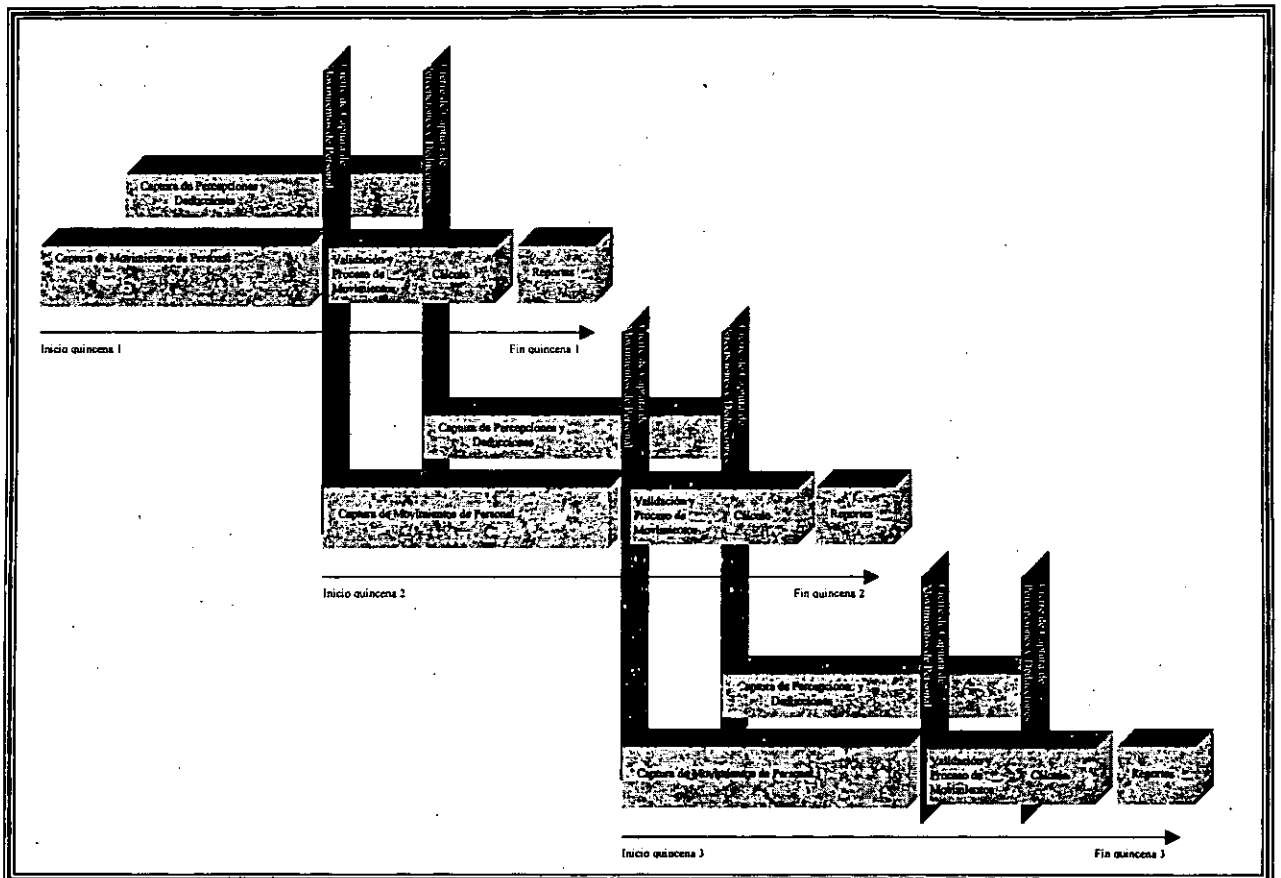

Figura 1.2 Secuencia de ejecución de los módulos en el tiempo

El sistema esta diseñado para soportar la distribución de la información necesaria a las diversas sucursales para que ellas mismas elaboren su nómina. Esta distribución confía en el funcionamiento adecuado del sistema de comunicaciones y la infraestructura de red para el envío de información y además prevé el fallo o la incapacidad de alguna de las sucursales para procesar su nómina mediante una redundancia de información en un nodo central en la casa matriz que funcionará como un centro de concentración de información.

En el esquema de comunicaciones (Fig. 1.3 y 1.4) cada Sucursal tiene todos los módulos para la captura de información de la Nómina de cada quincena. La información generada por dicha captura se envía a la base de datos central de la casa matriz.

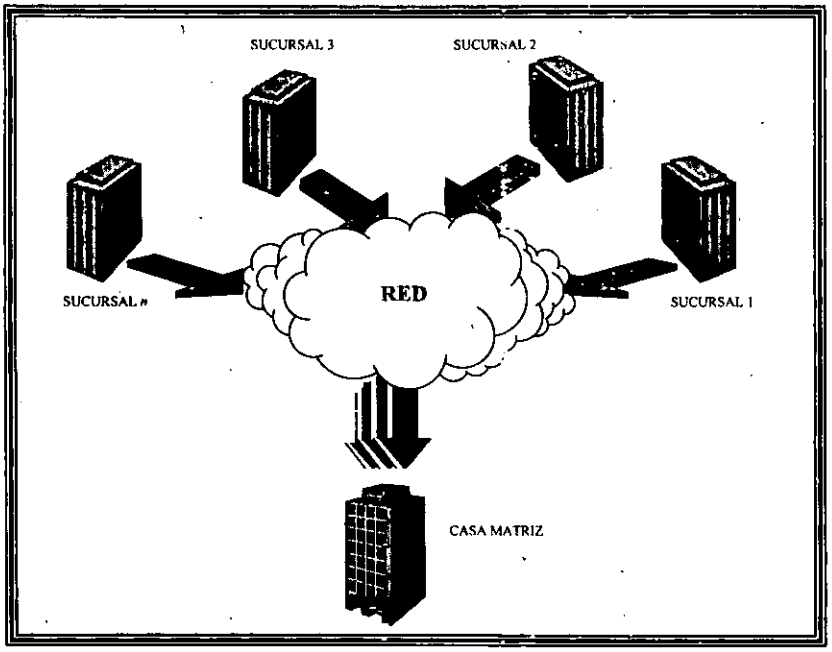

Fig. 1.3 Distribución de Información de Movimientos de Empleados

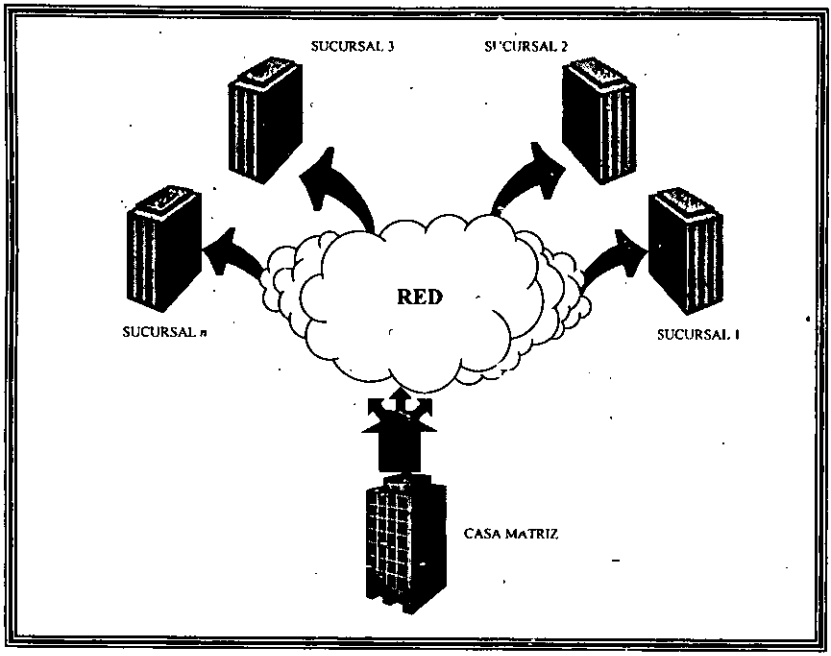

Fig. 1.3 Distribución de Información de Catálogos

Cuando las Sucursales están capturando o consultando la información de sus respectivos empleados, se apoyan en la información de los catálogos que se encuentran en la base de datos central.

1.2 Principales Características de la Nómina

El sistema esta basado en una Arquitectura Cliente/Servidor, para permitir la descentralización de la información y la utilización de equipos pequeños con distribución de procesamiento. El funcionamiento de este modelo se basa en que una aplicación "Cliente" hace peticiones de servicio. Las aplicaciones "Servidoras" reciben las peticiones de los clientes y las procesan. El servidor entonces realiza alguna acción y/o retorna un resultado al cliente. El cliente recibe el resultado (si es que hay alguno) y lo usa para continuar la tarea que esta llevando a cabo.

Las principales características de la Nómina se describen a continuación.

Tipos de Procesamiento

Con respecto al tipo de procesamiento que se utiliza para procesar la información, la Nómina es un sistema con 3 tipos de procesos (tabla 1.1):

- Batch: Para procesos que validan movimientos de empleados y el cálculo de los diferentes conceptos (tanto deducciones como percepciones) para el trabajador. Este tipo de procesamiento se caracteriza porque no hay intervención humana mientras se desarrolla. Se utiliza en estos módulos porque se necesita procesar la información en forma global, cuando se ha establecido ya una fecha de cierre o cuando se han capturado todos los movimientos del día o de la quincena.

- Procesamiento de Transacciones en Línea (también llamado OLTP por sus siglas en inglés, On-Line Transaction Processing). Se utiliza para todos los clientes de captura que modifican la información en línea y se caracteriza porque un gran número de usuarios necesitan acceso rápido a un conjunto pequeño de datos.

- Consultas de Información para el Soporte de Decisiones (DSS por sus siglas en inglés, Decision Support System). Se utiliza en la generación de reportes principalmente. Este tipo de procesamiento requiere acceso completo o a gran parte de la tabla, involucra joins entre muchas tablas y retorna resúmenes de grandes conjuntos de información.

Captura y Validación de Movimientos de Personal			
Proceso de Movimientos del Personal			
Captura y Validación de Percepciones y Deducciones			
Cálculo de la Nómina			
Reportes			

Tabla 1.1 Tipo de Procesamiento por módulo

Plataforma

La plataforma es la combinación de hardware y/o software en el cual corre la aplicación Cliente/Servidor. Los principales elementos que la conforman se listan a continuación:

Por el Lado de los Servicios

- El manejador de base de datos es el principal componente de la plataforma, ya que es quien sirve la información necesaria para la operación del sistema. El sistema, está utilizando el manejador de bases de datos Adaptive Server de Sybase, que maneja una arquitectura multihilos (multithread) con bases de datos relacionales, permite reforzar la integridad de la información, tiene un tiempo de respuesta aceptable y una alta capacidad de procesamiento de transacciones para múltiples usuarios.

- El sistema operativo es el software primario que sirve como interfaz entre el hardware y las aplicaciones. En este caso el sistema operativo es Solaris (UNIX) de SunSoft que corre sobre equipo Sun. El ambiente operativo Solaris esta basado en el estándar Unix System V Release 4, construido para alto desempeño en aplicaciones Cliente/Servidor en un ambiente de red distribuido. Permite acceso transparente e ilimitado a servidores, impresoras, bases de datos remotas y otros recursos con la escalabilidad para soportar virtualmente cualquier aplicación y configuración. Es un sistema multiprocesamiento, multihilos y multiusuario; multiprocesamiento significa la ejecución de un programa o múltiples programas simultáneamente en múltiples procesadores, multihilos es una técnica de software que divide el código de un programa en segmentos que pueden ser ejecutados en paralelo en múltiples procesadores y multiusuario porque permite que más de un usuario este ejecutando tareas a la vez.

Por el Lado de los Clientes

- Para las aplicaciones de captura, se utiliza la herramienta Power Builder de la empresa Sybase y los programas ejecutables corren sobre PC's con Sistema Operativo Windows 95. Power Builder es una herramienta que utiliza Lenguajes de Cuarta Generación (4GL) para el Desarrollo Rápido de Aplicaciones (RAD) y desde sus orígenes se desarrolló para interactuar con las Bases de Datos de Sybase. Tiene soporte completo para programación orientada a objetos, incorpora clases de objetos pre-construidos, incluyendo el objeto DataWindow para acceso y presentación de datos SQL y una interfaz gráfica de desarrollo. También soporta fácil conectividad a una amplia variedad de fuentes de datos. Es una herramienta óptima para ambientes Cliente/Servidor. La demanda de recursos para estas aplicaciones la cubre perfectamente bien una PC con el ambiente gráfico y amigable de Windows 95 o 98.

- Los clientes de batch se hicieron con librerías propias de Sybase llamadas db-libraries, incorporadas en rutinas de lenguaje C. El lenguaje C es un lenguaje de propósito general que ha sido estrechamente ligado con el sistema operativo UNIX. Proporciona una variedad de tipos de datos incluyendo apuntadores, estructuras y uniones; provee al programador de las instrucciones fundamentales de control de flujo; permite recursividad y manejo de funciones

además de la capacidad de explotar las características de multiprocesamiento del mismo sistema operativo. Las db-libraries son librerías que permiten la comunicación de los programas en C con la base de datos.

Por Parte de las Comunicaciones

- Cada Sucursal tiene una LAN en la que se encuentran todos los clientes.
- Todas las sucursales se interconectan mediante la internet hacia el servidor en la casa matriz.

Ubicación de Clientes y Servidores

Las aplicaciones clientes y servidoras pueden residir en la misma computadora o nodo; sin embargo, en un ambiente de computación distribuido por lo general residen en diferentes nodos. Los nodos se distinguen como nodos clientes y nodos servidores dependiendo de su papel.

En general, los clientes batch y DSS residen en el mismo nodo que los servidores y las aplicaciones OLTP residen en nodos separados (Fig. 1.4).

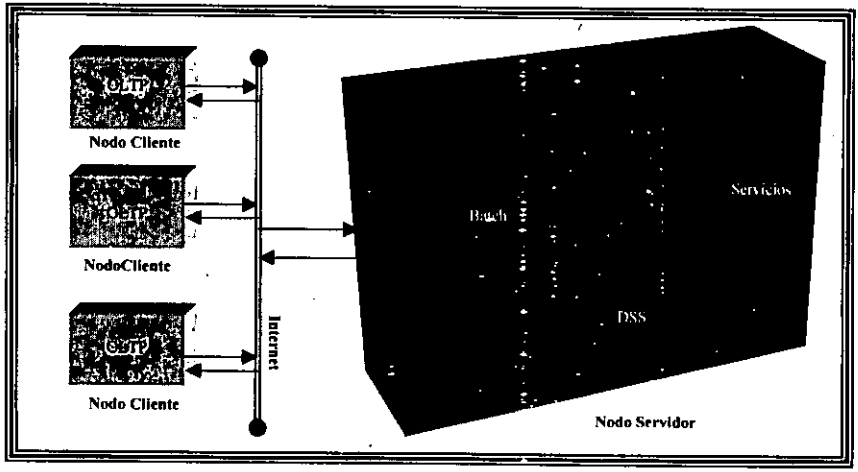

Fig. 1.4 Aplicaciones en una Sucursal y en la Casa Matriz

Además del objetivo general de funcionalidad de la Nómina expuesto en la sección 1.1, se tienen otros objetivos de seguridad para la confiabilidad y disponibilidad de la información. En la siguiente sección se inicia el tratamiento y entendimiento de los conceptos necesarios para el cabal cumplimiento de tales objetivos.

1.3 Análisis de Conceptos y Requerimientos de Seguridad

¿Qué se entiende por Seguridad?. Términos como *seguridad, protección* y *privacidad* a menudo tienen más de un significado.

Conceptos Importantes

Se dice que **"un sistema es seguro si sabemos que se comporta de la forma esperada"**. Si se espera que los datos que se metieron al sistema hoy se encuentren allí en varias semanas y que no sean leídos por alguien que no deba leerlos, entonces el sistema es seguro. A este concepto a menudo se le llama **confianza:** se confía en que el sistema preserve y proteja los datos. Pero existen otros términos que también es importante definir:

- *Confidencialidad*: Proteger la información de ser leída o copiada por alguien que no este explícitamente autorizado por el dueño de la información.
- *Integridad de los datos*: Proteger la información (incluyendo programas) de ser borrada o alterada sin los permisos del dueño.
- *Disponibilidad*: Proteger los servicios para que no sean degradados o se vuelvan indisponibles sin autorización.
- *Consistencia*: Asegurarse de que el sistema se comporta como se espera.
- *Control*: Regular el acceso al sistema.
- *Auditoría*: Los usuarios autorizados algunas veces cometen errores o aun cometen actos maliciosos. En tales casos, se necesita determinar que fue hecho, por quien y en que fecha. La única manera de lograr esto es teniendo un registro de la actividad en el sistema que identifique los actores y las acciones involucradas. A esto se le llama auditoría.

Requerimientos de Seguridad

Dentro de los requisitos de seguridad que debe cumplir este sistema se citan los siguientes:

- El sistema deberá ser diseñado de una manera que asegure la integridad de la información. Se espera el máximo grado de confiabilidad que la tecnología comercial de información pueda brindar.

- Se deberán explotar los beneficios de seguridad que ofrecen tanto el sistema operativo como el sistema manejador de base de datos. El sistema manejará diferentes niveles de seguridad, dependiendo del papel de cada usuario.

Prioridades

Todos los aspectos antes definidos nos interesan en esta propuesta de seguridad; sin embargo, los que tiene un mayor peso son los de **integridad** de la información, **consistencia** y **auditoría** ya que la modificación no autorizada de la información puede generar un comportamiento anormal en el sistema y el cálculo incorrecto del pago de los empleados o inclusive la modificación de la información por parte del personal autorizado puede traer las mismas consecuencias, por lo que también es importante saber quién lo hizo. En segundo término está la **confidencialidad** ya que existen algunos datos que no deben ser de dominio público (salario, descuentos, prestaciones, etc.) y en un tercer término la **disponibilidad**. A la disponibilidad se le considera de una prioridad más baja porque existen algunos aspectos (como las comunicaciones en la internet) que no están bajo nuestro control por lo que en algunos momentos podemos estar sin comunicaciones, pero en los aspectos que sí podemos controlar se buscará tener la disponibilidad adecuada. Cabe mencionar que existe un margen de tolerancia a fallas de hasta **tres días** para los módulos de **captura en días normales de operación** y de **un día en cierres de quincena**. Los demás procesos no tienen interacción con la internet ya que se hacen en forma local (Cálculo, Proceso de Movimientos y Reportes).

¿Por qué se Necesita Seguridad?[3]

De acuerdo al RFC[4] 1244, para la mayoría de los sitios, el interés en la seguridad de las computadoras es proporcional a la percepción de riesgos y amenazas.

El mundo de las computadoras ha cambiado radicalmente en los últimos 25 años. Hace 25 años, la mayoría de las computadoras eran centralizadas y administradas por centros de datos. Tales computadoras se mantenían en cuartos cerrados y la gente de apoyo se aseguraba que fueran cuidadosamente administradas y que estuvieran físicamente aseguradas. Los enlaces fuera del sitio eran inusuales. Eran raras las amenazas de seguridad y cuando hubo se atañían a los internos: mal uso de los recursos por usuarios autorizados, robos, vandalismo y cosas por el estilo. Estas amenazas fueron bien comprendidas y se trataron usando técnicas estándar: computadoras detrás de cuartos cerrados y contabilidad para todos los recursos.

La computación a partir de los 90's es radicalmente diferente. Muchos sistemas están en oficinas privadas y laboratorios, a menudo manejadas por individuos o personas fuera del centro de cómputo. Muchos sistemas están conectados a la internet y por lo tanto a todo el mundo.

---

[3] Chapman Brent. 1997. Construya Firewalls Para Internet. O'Reilly & Associates, Inc. México. Primera Edición en Español. p. 1-13.

[4] Los RFC (Request for comments) son documentos que contiene proposiciones para reglamentos utilizados en la Internet o información.

Los riesgos de seguridad también son diferentes ahora. El tiempo nos ha dicho "no pongas tu contraseña debajo o sobre tu escritorio" alguien la puede encontrar. Con las conexiones mundiales de la Internet, alguien puede entrar al sistema desde el otro lado del mundo y robar una contraseña a media noche cuando el edificio está cerrado. Los virus y gusanos pueden pasar de máquina en máquina. La internet permite el equivalente electrónico del ladrón que busca ventanas y puertas abiertas; ahora una persona puede verificar las vulnerabilidades de cientos de máquinas en unas cuantas horas.

Sin embargo, al conectarse en red, se obtienen muchas ventajas:

- Proporcionar mejor servicio al cliente
- Colaborar con los compañeros de trabajo
- Accesar información necesaria rápidamente

Los directivos y administradores de sistemas, tienen que comprender los riesgos de seguridad que existen, cual es el costo y riesgo de un problema de seguridad y que tipos de acciones (si hay alguna) se tienen que tomar para evitar y responder a amenazas de seguridad.

Principales Amenazas

Hay muchos tipos de ataques contra los sistemas y muchas formas de clasificarlos. En esta sección se dividen los ataques en tres categorías básicas: intrusión, negación del servicio y robo de información.

- *Intrusión*

  Los ataques más comunes a los sistemas son las intrusiones; con ellas las personas pueden utilizar otras computadoras. La mayoría de los atacantes quieren utilizarlas como si fueran usuarios legítimos.
  Los intrusos tienen docenas de formas de obtener acceso. Van desde ataques mediante manipulación social (encuentran la forma de saber el nombre de alguien con un puesto importante en la compañía), adivinación (probando combinaciones del nombre y contraseña de la cuenta hasta que funcione), explotación de servicios mal configurados, pasando por formas complicadas de entrar sin necesidad de saber el nombre o la contraseña de la cuenta y por último, utilizando la fuerza bruta (utilizan un programa que genera todas las posibles combinaciones de caracteres para adivinar una contraseña).

- *Negación del servicio*

  Un ataque de negación de servicio es el que está dirigido en su totalidad a evitar que se utilicen las computadoras. Un intruso inunda a tal grado un sistema o red (con mensajes, procesos o solicitudes) que no sea posible realizar algún trabajo.

Aunque inundar es la forma más simple y común de llevar a cabo un ataque de negación del servicio, un intruso más hábil también puede inhabilitar los servicio, volverlos a enrutar o reemplazarlos.

- *Robo de información*

  Algunos tipos de ataques permiten que el intruso obtenga información sin tener que utilizar directamente la computadora. Por lo general, se aprovechan de los servicios de internet que tienen como fin proporcionar información, haciendo que den más de lo que era su intención o dándola a las personas equivocadas.
  El robo de información no tiene que ser activo o especialmente técnico. Una persona puede llamarle y pedirle datos personales (fingiendo que tiene derecho a ello): esto es un robo *activo*. O pueden intervenir el teléfono: que en este caso se le conoce como un robo *pasivo*. De manera similar, las personas que quieran reunir información electrónica pueden indagarla activamente, fingiendo ser una máquina o un usuario con acceso válido o pueden intervenir de manera pasiva la red y esperar que la información fluya. Las intervenciones a la red generalmente se logran con la ayuda de *sniffers*, que permiten encontrar contraseñas, pero rara vez se utilizan para recabar otro tipo de información.
  En las tecnologías más comunes de redes, cualquier computadora que está en una red de área local (LAN) es capaz de ver todo el tráfico que pasa por ella. El tráfico que cruza internet puede pasar por un sinnúmero de redes LAN, cualquiera de las cuales puede ser un punto de riesgo. Los proveedores de servicio de red y los sistemas de acceso público son blancos muy comunes para intrusiones; analizadores colocados ahí pueden tener mucho éxito porque hay mucho tráfico que pasa a través de estas redes.

Las estadísticas han demostrado que, con el paso del tiempo, es menor el conocimiento que una persona debe tener y es mayor la sofisticación del ataque (fig. 1.5). Esto se debe, principalmente, a los muchos sitios en internet que ponen a disposición de quien accese allí, los programas que explotan alguna vulnerabilidad. Sin embargo, es responsabilidad de todo administrador, el conocer tales sitios y aplicar los parches recomendados por el fabricante, si es el caso.

Un ataque trae consigo múltiples consecuencia: pérdida de dinero, tiempo, reputación y robo de información confidencial.

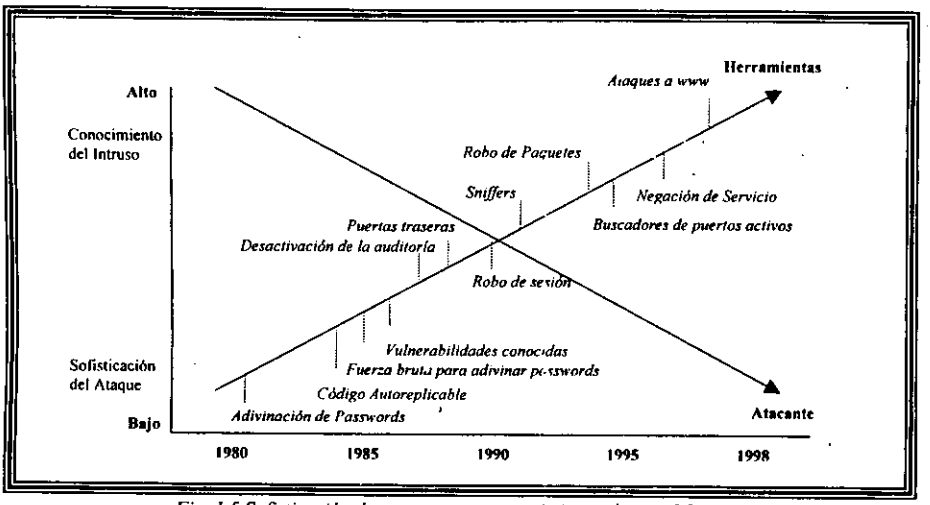

Fig. 1.5 Sofisticación de un ataque vs. conocimiento técnico del intruso

Tipos de Atacantes

Todos los atacantes comparten ciertas características. No quieren ser atrapados, así que intentan ocultarse. Si obtienen acceso a un sistema, es seguro que intentarán conservarlo, si es posible, construyendo formas adicionales de obtener acceso. La mayoría tiene algún tipo de contacto con otras personas que tienen los mismos tipos de intereses y la mayoría comparte la información que obtenga de atacar el sistema.

- *Joyriders*

  Son personas aburridas que buscan alguna diversión. Entran porque piensan que en el sistema puede haber datos interesantes; porque sería divertido utilizar las computadoras de la compañía o porque no tienen nada mejor que hacer. Tal vez quieran saber que tipo de computadoras hay o los datos que tienen. Son curiosos, pero no activamente maliciosos; sin embargo, con frecuencia dañan el sistema por ignorancia o por intentar cubrir su rastro. En especial, les atraen los sitios bien conocidos y computadoras poco comunes.

- *Vándalos*

  Los vándalos quieren causar daño, ya sea porque gozan con destruir cosas o porque el administrador del lugar no les agrada.
  Los vándalos obligan a la gente a tomarse muchas molestias para encontrarlos y detenerlos. A diferencia de muchos intrusos mundanos, los vándalos tienen carreras cortas pero llamativas. La mayoría va directo a la destrucción que es desagradable pero relativamente fácil de detectar y reparar. En la mayoría de las circunstancias, eliminar los datos o arruinar el equipo no es lo peor, en realidad, introducir cambios sutiles pero significativos en los programas o datos financieros es más difícil de detectar y reparar.

- *Score keepers*

  Muchos intrusos están entregándose a una versión actualizada de una tradición antigua: cobran fama basados en el número y tipos de sistemas a los que han entrado.
  Al igual que los joyriders y vándalos, los score keepers pueden preferir sitios de interés particular. Irrumpir en algo bien conocido, bien definido o bien ordenado significan puntos más valiosos para ellos. Sin embargo, también atacarán a cualquiera que este a su alcance; persiguen cantidad así como calidad. No necesariamente desean algo que se tenga, ni les importa en lo más mínimo las características del sitio. Pueden o no hacer daño a su paso.
  Con toda seguridad reunirán información que guardarán para utilizarla después (tal vez para intercambiarla con otros atacantes). Es probable que intenten dejar formas de volver en fecha posterior y, si es posible, utilizarán las máquinas ya atacadas como plataforma para atacar a otros.
  Estas personas se descubren mucho después de haber entrado al sistema.

- *Espías*

  Es común que la mayoría de las personas que entran sin permiso a las computadoras roben cosas que se puedan convertir en dinero o en mayor acceso (tarjetas de crédito, teléfonos o información para tener acceso a redes). Si encuentran secretos que creen que pueden vender, tal vez intenten hacerlo, pero no es su negocio principal.
  El espionaje es mucho más difícil de detectar que las entradas sin permiso comunes. No es necesario que el robo de información deje rastro e incluso las intrusiones rara vez se detectan de inmediato. Alguien que entra, copia información y sale sin alterar nada es probable que lo logre en la mayoría de los sitios.

¿Cómo se Puede Proteger el Sitio?

¿Qué medios se pueden tomar para proteger un sitio de los tipos de ataques antes mencionados?. Las personas escogen una gran variedad de modelos de seguridad o medidas que van desde las que no ofrecen seguridad alguna, pasando por lo que se llama "seguridad de ser desconocido", seguridad para host, hasta la seguridad para redes.

- *Ninguna seguridad*

  La medida más simple posible es no poner ningún esfuerzo en la seguridad y tener la mínima, cualquiera que sea, que proporcione el vendedor de forma preestablecida. Este modelo no es recomendable.

- *Seguridad a través de ser desconocido*

  Con él, un sistema es seguro sólo porque (supuestamente) nadie sabe de su existencia, contenido, medidas de seguridad o alguna otra cosa.

  Muchas personas suponen que aunque los atacantes pueden encontrarlos, no se tomarán la molestia de hacerlo. Piensan que una compañía pequeña o una máquina en casa no será de interés para los intrusos. De hecho, muchos de estos últimos no tienen interés en blancos específicos, sólo quieren entrar en tantas máquinas como sea posible. Es probable que no entren en ellas mucho tiempo, pero sí intentarán entrar y pueden hacer un daño considerable al intentar ocultar su rastro.

  Los intrusos disponen de mucho tiempo y, con frecuencia, encuentran a quien desea permanecer desconocido con solo intentar todas las posibilidades. A la larga, depender de la seguridad de ser desconocido no es una buena opción.

- *Seguridad para host*

  Es posible que el modelo más común de seguridad para computadoras sea el modelo de seguridad para host. Con él se puede reforzar la seguridad de cada máquina por separado y hacer todo el esfuerzo para evitar o reducir todos los problemas de seguridad que puedan afectar a ese host en específico.

  El principal impedimento de una seguridad efectiva para host en los ambientes modernos de cómputo es la complejidad y diversidad de esos ambientes. La mayoría incluyen máquinas provenientes de varios fabricantes, cada una con su propio sistema operativo y con su propio conjunto de problemas de seguridad. Aun cuando el sitio tenga máquinas de un solo fabricante, a menudo diferentes versiones del mismo sistema operativo presentan distintos problemas de seguridad importantes. Aunque todas estas máquinas sean de un mismo fabricante y ejecuten una sola versión del sistema operativo, diferentes configuraciones (distintos servicios activados, etc.) pueden llamar a diferentes subsistemas (y conflictos) que llevan a distintos problemas de seguridad. Esto implica una cantidad importante de trabajo directo y continuo para implementar así como mantener de manera eficaz la seguridad para el host.

  Esta seguridad también depende de las buenas intenciones y de la habilidad de quienes tienen acceso privilegiado a cualquier máquina. Conforme se incrementa el número de máquinas, también aumenta el número de usuarios privilegiados.

  Un modelo de seguridad para host puede ser altamente apropiado para sitios pequeños o con necesidades de extrema seguridad. De hecho, todos los sitios deben incluir algún nivel de seguridad para host en sus planes generales de seguridad.

- *Seguridad para redes*

  Conforme los ambientes se tornan más grandes y diversos y conforme se vuelve más difícil asegurarlos sobre la base de host por host, muchos sitios optan por un modelo de seguridad para redes. Las medidas de seguridad para redes incluyen la construcción de firewalls para proteger los sistemas y redes internas, utilizando estrictas medidas de autenticación (como las contraseñas de una sola vez), el uso de datos cifrados para proteger datos que son muy sensibles cuando transitan por la red y estrategias para esconder la arquitectura de red hacia el mundo exterior.

  La ventaja de este modelo es que se pueden proteger cientos de máquinas contra un ataque de las redes que están más allá del firewall, sin importar el nivel de seguridad de cada host. Aunque no por esto deja de ser importante la seguridad para host.

Ningún Modelo de Seguridad Puede Hacerlo Todo

Ningún modelo de seguridad puede resolver todos los problemas; ninguno puede evitar que una persona hostil con acceso legítimo dañe a propósito algún sitio u obtenga información confidencial de él. Para evitar las estrictas medidas de seguridad, un usuario legítimo puede utilizar métodos físicos. Estos pueden ir desde derramar refresco en las computadoras hasta llevarse a casa memorándums importantes. Se puede lograr la protección de accidentes y la ignorancia de manera interna y de actos maliciosos externos, pero no hay forma de protegerse de los usuarios legítimos sin dañar de forma severa la habilidad de éstos de utilizar las computadoras que se quieren proteger.

Ningún modelo de seguridad proporciona protección perfecta. Se puede esperar que las entradas sin permiso sean poco comunes, breves y baratas, pero no se espera evitarlas por completo. Aún los sitios más seguros y dedicados esperan tener un incidente que involucre la seguridad de vez en cuando.

¿Por qué preocuparse entonces?. La seguridad puede no evitar cada incidente, pero puede evitar que un incidente dañe en forma seria.

Estrategias de Seguridad[5]

Con la creciente conscientización en la seguridad, han venido desarrollándose diversas estrategias para la misma, algunas muy simples y algunas otras muy sofisticadas. Cada una, dependiendo de su grado de seguridad, conlleva a una inversión que también depende de la complejidad de ésta.

A continuación se describen las estrategias de seguridad más comunes y conocidas actualmente.

[5] Chapman Brent. 1997. Construya Firewalls Para Internet. O'Reilly & Associates, Inc. México. Primera Edición en Español. p. 45-54.

Menor Privilegio

Quizás el principio de seguridad más fundamental (cualquier tipo de seguridad, no sólo la de computadoras y redes) es el de menor privilegio. Básicamente, el principio de menor privilegio significa que cualquier objeto (usuario, administrador, programa, sistema o lo que sea) debe tener sólo los privilegios que necesita para cumplir con sus tareas asignadas (no más). Menor privilegio es un principio importante para limitar la exposición a ataques y para limitar el daño causado por ataques físicos.

Aplicar el principio de menor privilegio sugiere que se deben explotar formas de reducir los privilegios necesarios para hacer varias operaciones. Por ejemplo:

- No dar a un usuario la contraseña de root para un sistema si todo lo que necesita es reinstalar el software del reporteador. En lugar de eso, conviene escribir un programa con los privilegios necesarios para que el usuario pueda reinstalarlo.

- No permitir que un programa ejecute setuid como root si lo único que necesita es escribir a un archivo protegido. En lugar de eso, se puede permitir que el archivo pueda ser escrito por un grupo y que el programa haga setgid en vez de setuid a root.

Hay que tener cuidado con este principio, ya que es necesario hacer un buen análisis para determinar efectivamente cuales son los privilegios que se necesitan para llevar a cabo una tarea. Se puede correr el riesgo de dar menos del privilegio mínimo que necesita. Intentar cumplir con el menor privilegio con las personas, en lugar de programas, puede ser bastante peligroso. Se puede predecir con facilidad los permisos que necesitará el programa Sendmail para hacer su trabajo; los seres humanos son menos predecibles y es más fácil que se molesten y se vuelvan peligrosos si no pueden hacer lo que quieren.

Defensa a Fondo

Este principio sugiere que no se dependa sólo de un mecanismo de seguridad, sin importar cuán fuerte parezca; se deben instalar varios mecanismos que se respalden entre sí. No es deseable que la falla de un solo mecanismo de seguridad comprometa por completo toda la seguridad.

Este principio se puede llevar a cabo adoptando varios mecanismos que se den respaldo y redundancia entre si: seguridad de red, seguridad de sistema operativo, seguridad en aplicaciones y seguridad humana (educación del usuario, administración cuidadosa del sistema, etc.).

Punto de Choque

Un punto de choque obliga a los atacantes a utilizar un canal angosto que se puede monitorear y controlar.

En las redes, el punto a través del cual se conecta a la internet es un punto de choque; cualquiera que vaya a atacar el sitio desde una red externa tendrá que pasar a través de este canal.

Un punto de choque es inservible si hay una manera efectiva de que un atacante lo evite. ¿Por qué molestarse en atacar la puerta principal, que está fortificada, si la puerta trasera esta abierta?.

Un punto de choque parece como poner todos los huevos en una canasta y, por lo tanto, mala idea, pero la clave es que se trata de una canasta que se puede proteger con sumo cuidado. La alternativa contraria sería dividir la atención entre varios puntos de acceso. Si se hace así, es probable que no se pueda hacer un trabajo adecuado y no se defiendan bien tales puntos o alguien pasa por uno mientras se protege otro.

Eslabón más Débil

El punto fundamental de seguridad es que una cadena es tan fuerte como su eslabón más débil. Los atacantes inteligentes buscan el punto débil y concentran su atención en él.

Siempre hay un eslabón débil, el truco consiste en hacer que sea lo suficientemente fuerte y mantenerlo así de acuerdo con el riesgo. Por ejemplo, es muy razonable preocuparse más por personas que atacan por la red que por las que van al sitio a atacar físicamente; por lo tanto, la seguridad física puede ser el eslabón más débil.

Los modelos de seguridad para host sufren de una interacción bastante desagradable entre puntos de choque y eslabones débiles; no hay punto de choque, lo cual significa que existe un gran número de eslabones y muchos pueden ser en realidad muy débiles.

Postura de Falla Segura

Otro principio fundamental de la seguridad es que, en la medida de lo posible, los sistemas deben tener una falla segura, es decir, si van a fallar deben hacerlo de tal forma que nieguen el acceso a un atacante en lugar de dejarlo entrar. La falla también puede causar la negación del acceso a usuarios legítimos hasta que se hagan las reparaciones, pero por lo general es algo aceptable.

La aplicación más importante de este principio reside en seleccionar la postura del sitio respecto a la seguridad. ¿Se inclina por ser permisivo o restrictivo?, ¿Tiene más inclinación a fallar en dirección de la seguridad (algunos llaman a esto paranoia) o de la libertad?.

Hay dos posturas fundamentales que se pueden adoptar con respecto a decisiones de políticas de seguridad:

Postura de Negación Preestablecida

"Lo que no esta permitido expresamente, esta prohibido". Esta postura tiene sentido desde el punto de vista de la seguridad porque es una postura de falla segura. Acepta que lo que no se conoce puede causar daño.

Con esta postura, se prohibe todo por omisión; después, para determinar lo que se va a permitir, se debe:

- Examinar los servicios que necesitan los usuarios.
- Considerar cómo afectarían la seguridad tales servicios y cómo pueden proporcionarse de manera segura.
- Permitir sólo los servicios que se pueden proporcionar con seguridad y para los cuales existe una necesidad legítima.

Postura de Permiso Preestablecido

"Lo que no esta prohibido expresamente, esta permitido". La mayoría de los usuarios y administradores prefieren la postura de permiso preestabiecido. Tienden a suponer que todo estará, por omisión, permitido y que se irán prohibiendo ciertas acciones y servicios problemáticos específicos conforme sea necesario. Esto no es una postura de falla segura.

Supone que se conocen de antemano y de manera precisa cuáles son los peligros específicos, cómo explicarlos para que los usuarios los comprendan y cómo protegerse contra ellos. Adivinar qué peligros podrían estar en el sistema o en internet es, en esencia, una tarea imposible. Sencillamente hay demasiados problemas posibles y demasiada información (por decir algo, nuevos agujeros en la seguridad, nuevas formas de explotar agujeros antiguos, etc.), para mantenerse actualizado. Si no se sabe que algo es un problema no estará en la lista de "prohibido", en ese caso seguirá siendo un problema hasta darse cuenta de ello y es posible que se descubra porque alguien se aprovecha de él.

Participación Universal

Para que sean totalmente efectivos, la mayoría de los sistemas de seguridad requieren de participación universal (o por lo menos la ausencia de oposición activa) por parte del personal de un sitio.

Se necesita que todos notifiquen ocurrencias extrañas que pueden estar relacionadas con la seguridad; el administrador no puede ver todo. Se necesita que las personas seleccionen buenas contraseñas y que las cambien con regularidad.

¿Cómo hacer para que todos participen? La participación puede ser voluntaria (convenciendo a todos de que esto es una buena idea), involuntaria (alguien con suficiente autoridad y poder les dice

que tienen que cooperar o de lo contrario...) o una combinación de ambas. Vale la pena gastar mucha energía para convencer a las personas de que cooperen voluntariamente, ya que, de lo contrario, se puede caer en una guerra armamentista en la que los usuarios tratan de evadir los esquemas de seguridad y el administrador trata de bloquearlos.

Diversificación de Defensa

Así como se puede obtener seguridad adicional utilizando varios sistemas para dar profundidad a la defensa, también se puede obtener empleando varios tipos de ellos. Si todos son iguales, alguien que sepa entrar a alguno de ellos quizá penetre en todos.

La idea que sirve de apoyo a la diversificación de defensa es que utilizar sistemas de seguridad de diferentes proveedores puede reducir las posibilidades de un problema o error de configuración común que pueda comprometerlos a todos. Sin embargo, hay un balance en términos de complejidad y costo. Procurar e instalar varios diferentes es más difícil, toma más tiempo y es más costoso que procurar e instalar un solo sistema (o incluso varios idénticos).

Aunque muchos sitios reconocen que utilizar múltiples tipos de sistemas podrían incrementar de manera potencial su seguridad, con frecuencia concluyen que la diversificación de defensa requiere más trabajo de lo que vale y que las ganancias y mejorías en la seguridad potencial no valen el costo.

Simplicidad

La simplicidad es una estrategia de seguridad por dos razones. Primero, mantener las cosas sencillas las hace más fáciles de comprender; si no se entiende algo, no se puede saber si es seguro o no. Segundo, lo complejo proporciona muchos escondites que facilitan que se oculten toda clase de cosas. Los programas complejos tienen más problemas y cualquiera puede ser de seguridad.

Niveles de Seguridad[6]

El criterio estándar para la evaluación de "Computadoras Confiables" se basa en el Libro Naranja que fue publicado por el Departamento de Defensa de los Estados Unidos. Los criterios fueron desarrollados con tres objetivos en mente:

- Proporcionar a los usuarios una norma con la cual valorar el grado de confiabilidad que pueden poner en sus computadoras para el procesamiento seguro de información clasificada u otro tipo de información
- Proporcionar una guía a los fabricantes sobre como construir sus productos comerciales involucrados con la seguridad para satisfacer los requerimientos de confiabilidad para aplicaciones sensibles
- Proporcionar una base para especificar los requerimientos de seguridad en adquisiciones.

Estos criterios se pueden aplicar al conjunto de componentes que comprende un sistema confiable y no es necesario que se aplique a cada componente individual del sistema.

Requerimientos de Seguridad Fundamentales

Cualquier discusión sobre seguridad en computadoras comienza por un enunciado de requerimientos. En general, "los sistemas seguros deben controlar, a través del uso específico de características de seguridad, acceso a la información sólo a individuos autorizados". De este enunciado se derivan seis requerimientos fundamentales:

- Política de seguridad: Debe existir una política de seguridad explícita y bien definida. Debe haber un conjunto de reglas que son usadas por el sistema para determinar si alguien puede tener acceso a un objeto específico (archivo, directorio, programa, dispositivos, etc.).
- Etiquetas: Se deben asociar etiquetas de control de acceso con los objetos. Para controlar el acceso a la información almacenada en una computadora, de acuerdo a las reglas mandatorias de la política de seguridad, debe ser posible marcar cada objeto con una etiqueta que identifique el nivel de sensibilidad del objeto y/o los modos de acceso para el personal autorizado.
- Identificación: Se debe identificar a cada usuario. Cada acceso a la información debe ser restringido de acuerdo a quién la esta consultando y a la clase de información que puede manejar. Esta información de identificación y autorización debe mantenerse en forma segura y debe asociarse con cada usuario que realice alguna operación de seguridad relevante.

[6] http://www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html

- ❖ Contabilidad: La información de auditoría debe ser selectivamente mantenida y protegida para que se puedan rastrear las acciones que afecten la seguridad.

- ❖ Garantía: El sistema de cómputo debe contener mecanismos de hardware y software que puedan ser evaluados independientemente para proporcionar garantía suficiente de que el sistema refuerza los requerimientos anteriores. Estos mecanismos, por lo general, se encuentran incluidos en el sistema operativo.

- ❖ Protección continua: Los mecanismos de confiabilidad que refuerzan estos requerimientos básicos deben estar protegidos contra alteraciones y/o cambios no autorizados.

Estos requerimientos fundamentales forman las bases para el criterio de evaluación aplicable en cada división.

Los criterios están divididos en cuatro divisiones: D, C, B y A ordenados de una manera jerárquica siendo la división más alta la A. Cada división representa un mayor mejoramiento en la confidencialidad que se puede tener en el sistema. Dentro de las divisiones C y B existen un número de subdivisiones llamadas clases. Las clases también se encuentran ordenadas de una manera jerárquica. La garantía de que el diseño de estos sistemas es completo y correcto se logra probando las partes de seguridad relevantes del sistema a las que también se les llama Base de Cómputo Confiable (BCC).

División D: Protección Mínima

Esta división contiene sólo una clase. Esta reservada para aquellos sistemas que han sido evaluados pero no cubren ningún requerimiento de las clases más altas.

División C: Discrecional

Discrecional significa que se requiere conocer que protección se necesita implementar y, por lo tanto, es opcional y se logra mediante la inclusión de capacidades de auditoría de sujetos y las acciones que éstos generan.

Clase C1: Protección con Seguridad Discrecional

La Base de Cómputo Confiable de un sistema de clase C1 satisface los requerimientos de seguridad discrecional al separar los usuarios de los datos. Tiene las siguientes características:

- ➢ Control de acceso discrecional: La base de cómputo confiable define y controla los accesos entre los usuarios y objetos. Tales mecanismos (ejemplo: controles de dueño/grupo/otros y listas de control de acceso) permiten a los usuarios especificar y controlar el comportamiento de objetos a individuos o grupos definidos o ambos.

- ➢ Identificación y autenticación: La Base de Cómputo Confiable requiere que los usuarios se identifiquen. Además, usa un mecanismo protegido (por ejemplo, contraseñas) para autenticar la identidad de los usuarios. También debe proteger los datos de autenticación para que no sean accedidos por un usuario no autorizado.

- Garantía operacional: La BCC deberá mantener un dominio para su propia ejecución que la proteja de interferencias externas o alteraciones (mediante la modificación de su código o estructuras de datos). También se le llama protección de los modos de operación del sistema.

- Integridad: Las características de hardware y/o software deberán permitir validar periódicamente la operación correcta de los elementos de hardware y firmware de la Base de Cómputo Confiable.

- Pruebas de seguridad: El sistema debe probarse y trabajar como se especifica en la documentación. Se debe hacer para asegurar que no hay maneras obvias de evitar los mecanismos de seguridad de la Base de Cómputo Confiable por parte de usuarios no autorizados.

- Documentación: Debe existir una Guía de Usuario sobre Características de Seguridad, un Manual de Facilidades Confiables, Documentación de Prueba y Documentación de Diseño.

Clase C2: Protección con Acceso Controlado

Los sistemas en esta clase tienen un control de acceso discrecional más fino que los sistemas C1, permitiendo la auditoría de acciones individuales mediante procedimientos de registro, auditoría de eventos de seguridad relevantes y aislamiento de recursos. Los siguientes son los requerimientos mínimos:

- Control de acceso discrecional: Lo mismo que para la clase C1 pero además deberá proporcionar control para limitar la propagación de derechos de acceso. Deberá permitir, mediante la acción explícita del usuario o por default, proteger los objetos de accesos no autorizados. Estos controles de acceso deberán ser capaces de incluir o excluir accesos hasta la granularidad de un único usuario. Las autorizaciones a la información que son asignadas a un objeto deberán ser revocadas antes de una asignación o re-asignación a un sujeto. Ninguna información sobre las acciones anteriores de un sujeto deberá estar disponible a otro sujeto que obtenga acceso a un objeto que haya sido devuelto al sistema.

- Identificación y autenticación: Lo mismo que en C1 pero además la Base de Cómputo Confiable debe auditar las acciones individuales y quienes revisan las bitácoras deben tener acceso de sólo lectura. La auditoría debe identificar: fecha y hora del evento, usuario, tipo del evento y éxito o fallo del evento.

- Garantía operacional: Lo mismo que en la clase C1 pero agregando la capacidad de auditarlos.

- Integridad: Lo mismo que para la clase C1.

- Pruebas de seguridad: Lo mismo que para C1 pero debe incluir la capacidad de buscar debilidades que pudiesen permitir a un usuario no autorizado el acceso a datos de autenticación o auditoría.

- Documentación: Lo incluido en la clase anterior pero además deberá documentar los procedimientos para examinar y mantener la auditoría.

División B: Protección Obligatoria

En esta división las reglas de control de acceso son obligatorias, no opcionales.

Clase B1: Protección Mediante Etiquetas de Seguridad

Para alcanzar este nivel, se deben cumplir los requerimientos de la clase C2 y además:

- Mantener un control de acceso obligatorio en todos los sujetos y objetos de almacenamiento bajo el control de la Base de Cómputo Confiable, un sujeto puede leer un objeto sólo si la clasificación jerárquica en el nivel de seguridad del sujeto es mayor que o igual a la clasificación jerárquica en el nivel de seguridad del objeto y las categorías no jerárquicas en el nivel de seguridad del sujeto incluyen todas las categorías no jerárquicas en el nivel de seguridad del objeto; manejo de integridad de tales etiquetas; auditoría de los objetos etiquetados y la habilidad para mostrar la información de las etiquetas en un formato entendible.

Clase B2: Protección Estructurada

Debe cumplir con los requerimientos de la clase B1 e incluir:

- Funciones de operador y administrador separadas; la ruta de comunicaciones entre el sistema y el usuario debe ser confiable para el acceso inicial y la autenticación. Las comunicaciones por esta ruta deberán ser iniciadas exclusivamente por el usuario; debe de proporcionar una herramienta y la documentación necesaria para generar una Base de Cómputo Confiable a partir del código fuente y que compare la versión nueva con la versión anterior para asegurarse que sólo se hayan hecho los cambios planeados.

Clase B3: Dominios de Seguridad

Como B2 pero incluye:

- Los controles de acceso discrecional deben proporcionar, mediante la acción explícita del usuario o por default, protección de accesos no autorizado a los objetos y deberán ser capaces de especificar, para cada objeto, una lista de individuos y una lista de grupos con sus respectivos modos de acceso a ese objeto así como la lista de los individuos y/o grupos que no tienen acceso al mismo. La Base de Cómputo Confiable debe excluir el código que no sea esencial para reforzar la política de seguridad, tratando de minimizar la complejidad; se debe soportar un administrador de seguridad; se requieren procedimientos para la recuperación del sistema; el administrador del sistema debe poder auditar selectivamente las acciones de uno o más usuarios basado en la identidad individual y/o en el nivel de seguridad del objeto; la Base de Cómputo Confiable debe incluir un mecanismo que vigile la acumulación u ocurrencias de

eventos que puedan indicar una inminente violación de la política de seguridad. También debe incluir la documentación que muestre como probar y garantizar que el sistema funciona de esta manera.

División A: Protección Verificada

Esta división se caracteriza por el uso formal de métodos de verificación de seguridad para asegurar que los controles de acceso obligatorios y discrecionales pueden proteger de manera efectiva la información almacenada y procesada en el sistema.

Clase A1: Diseño Verificado

Los sistemas de esta clase son funcionalmente iguales a los de la clase B3 pero requieren:

- Un proceso estricto de diseño, control y verificación. Para alcanzar este nivel de seguridad el diseño debe verificarse matemáticamente y debe hacerse un análisis de los canales cubiertos y de distribución confiable. La distribución confiable significa que el hardware y el software hayan estado protegidos durante su traslado para evitar violaciones de los sistemas de seguridad.

1.4 Alcances De La Propuesta

El esquema de seguridad total de la Nómina esta dividido en tres partes. La seguridad en los clientes (tanto de captura como los de batch), la seguridad en los servicios y la seguridad física.

Esta propuesta esta enfocada a diseñar un esquema de seguridad para la parte de Seguridad Física y de los Servicios, es decir, la Base de Datos, el Sistema Operativo y los Servicios de Red.

El modelo que se utilizará para el diseño del esquema es mediante una seguridad física como una primera barrera de protección a la que se le añadirá otra capa más con la protección de los servicios de red y se utilizarán las características de seguridad que ofrecen los diferentes servicios a nivel host para agregar otra capa más (Fig. 1.5). La estrategia de este modelo es utilizar el principio de menor privilegio con la defensa a fondo que nos proporciona este modelo en capas y también adoptar la postura de negación preestablecida (que todo lo que no está expresamente permitido, esta prohibido).

Fig. 1.6 Modelo de Seguridad

En los capitulos que siguen se tocan los aspectos a considerar para la seguridad física y cada uno de estas capas. En cada uno de ellos se analizan las características de seguridad con las que cuenta el software aplicable a tal capa y analizamos que queremos proteger y de quien nos queremos proteger. Además, se analizarán otras herramientas (por lo general de dominio público) que auxilien en un mejor control de la seguridad.

Capítulo 2: Seguridad Física

2.1 Necesidades de Seguridad

La Seguridad Física se entiende como la parte de la seguridad que se encarga de las medidas para prevenir accesos no autorizados a las instalaciones, materiales o documentos y salvaguardarlos de daños o robos.

Es muy frecuente que la seguridad física sea olvidada ya que las consideraciones que se deben tener son diferentes entre las organizaciones y deben aplicarse en el sitio, es decir, no es algo que se pueda obtener de la internet o que venga pre-instalado en el sistema operativo.

Este tipo de seguridad es tan importante como las demas, ya que de nada sirve que tengamos el mejor esquema de seguridad en el sistema operativo, en la base de datos o en los servicios de red si cualquier persona puede tener contacto con el equipo y puede desconectarlo o robar algún componente del mismo.

La seguridad física en el Sistema de Nómina, se enfoca directamente a las consideraciones necesarias para las instalaciones donde se encuentran los equipos con los servicios. En éste capítulo, analizaremos lo que se quiere proteger, de quién o de qué se quiere proteger y cómo se va a proteger, que son las preguntas básicas para una estrategia de seguridad.

Análisis de Riesgo

Los recursos a proteger son todos los componentes físicos con los que funciona la Nómina o que de alguna forma tienen relación con la misma y que está bajo responsabilidad directa del Departamento de Nómina.

Primeramente, enumeremos todos los componentes que se involucran con el sistema y a los que nos interesa proteger:

- Las estaciones de trabajo (en donde se encuentra funcionando la aplicación y los servicios) y todos sus componentes (unidad de CD-ROM, unidad de cinta, teclado, ratón, monitor y discos externos). Por el papel que desempeñan en el buen funcionamiento de la Nómina, ya que en él residen los servicios necesarios para la operación, y por los datos mismos que contiene, este equipo se considera sensible.

- Las bitácoras (cuadernos de anotaciones) del administrador del sistema. Las bitácoras del administrador contienen datos importantes para una persona que desee conocer los servicios que tienen los equipos, usuarios, modificaciones hechas al sistema, parches que se han aplicado, etc., y que pueden ser de gran ayuda si los utiliza para atacarnos.

- Los inventarios. Son nuestra referencia para conocer algún faltante, su modificación o pérdida puede repercutir en un robo de equipo, software o medio de almacenamiento sin que nos demos cuenta de ello.

- Las cintas, ya que contienen los respaldos de nuestra información y ya se dijo en el primer punto que ésta se considera información sensible. Además de que nos sirve para recuperar total o parcialmente datos que se hayan perdido accidentalmente o como resultado de algún ataque en el que hayan borrado algunos.

Las amenazas a las que se encuentran expuestos se describen a continuación:

Robo

Debido a que muchas computadoras, sus componentes y manuales son de gran valor, son fácilmente robados y fácilmente vendidos. Aun las computadoras como las DEC VaxStation han sido robadas pensando que eran PC's.

El simple sentido común nos dice, "mantengamos las computadoras en un cuarto con llave". ¿Pero que tan seguro es ese cuarto?. A veces parece que esta seguro pero en realidad esta totalmente abierto.

Los ventanales son de gran riesgo ya que se pueden romper fácilmente. También es posible ver a través de ellos la forma en que están distribuidas (las computadoras) y quienes o como las operan.

Muchas veces, al cuarto de computadoras llegan conductos de aire por los que fácilmente puede deslizarse una persona. Algunas veces estos cuartos tienen pisos elevados que comparten con algunas otras áreas u oficinas y que es también una entrada potencial.

Ambiente[7]

Las computadoras son dispositivos extremadamente complicados que a menudo requieren exactamente el balance correcto de condiciones físicas y ambientales para operar propiamente. Alterar este balance puede causar un fallo de una manera inesperada e indeseable, o lo que es aun peor, puede continuar operando pero producir resultados incorrectos y corrompiendo así datos de mucho valor (la Nómina de La Empresa).

En este aspecto las computadoras se parecen mucho a las personas: no trabajan bien si están muy calientes, muy frías o sumergidas en agua sin una protección especial.

[7] Garfinkel Simson. 1996. Practical Unix & Internet Security. O'Reilly & Associates, Inc. USA. Segunda Edición. p. 357-388.

Fuego

Las computadoras no sobreviven a la exposición al fuego. Si las flamas no causan que la máquina y los circuitos se quemen, el calor puede derretir la soldadura que sostiene los componentes electrónicos.

Humo

El humo es dañino para el equipo de cómputo ya que es un potente abrasivo y se acumula en las cabezas de los discos magnéticos, discos ópticos y unidades de cinta. Una sola partícula de humo puede causar un daño severo al disco cuando éstos no están bien sellados.

Algunas veces el humo lo generan las computadoras mismas. Los fuegos eléctricos, particularmente aquellos causados por los transformadores en los monitores, pueden producir humo capaz de dañar a otros equipos y también puede ser un potente carcinógeno.

El humo que proviene de los cigarros o de las pipas es aun más dañino debido a que puede causar fallo prematuro en teclados y requerirán ser limpiados más a menudo.

Polvo

El polvo destruye los datos. Como el humo, el polvo se colecta en las cabezas de los discos magnéticos, las cintas y los discos ópticos. El polvo es abrasivo y destruirá lentamente tanto las cabezas de grabado como el medio de grabación.

La mayoría del polvo es eléctricamente conductivo. El diseño de muchas computadoras succiona grandes cantidades de aire y polvo por el sistema de ventilación. Invariablemente, una capa de polvo se irá acumulando en las tarjetas de circuitos lo que podría propiciar eventualmente un corto y ,en consecuencia, un fallo.

Temblor

Durante un temblor, las computadoras que se encuentran en lugares altos (sobre gabinetes, archiveros, etc.), son las más propensas a sufrir daño en caso de que lleguen a caerse. También el tener cosas pesadas en la vecindad, y que pueden caer sobre ellas, aumenta aun más el riesgo. Las computadoras cerca de las ventanas, especialmente en los pisos superiores de una edificio, podrían caer, lo que provocaría un daño a las personas que estuviesen abajo, además de que se dañaría la computadora.

Temperaturas Extremas

Como la gente, las computadoras operan mejor dentro de ciertos rangos de temperatura. La mayoría deberán mantenerse entre los 10 y 32 grados centígrados. Si la temperatura del ambiente que rodea a la computadora es demasiado alta, ésta no podrá enfriarse adecuadamente y los componentes internos se pueden dañar. Si la temperatura es muy fría, el sistema puede sufrir un choque térmico cuando se encienda, causando que las tarjetas de circuitos o los circuitos integrados se dañen.

Insectos

Algunas veces los insectos y otros animales pequeños encuentran su camino dentro de las computadoras. Por cierto, el término bug, se usó para describir algo que andaba mal en un programa de computadora años atrás (por 1950), cuando Grace Murray Hopper encontró una mariposilla atrapada en los contactos de los relevadores en la computadora Mark I de la Universidad de Harvard.

Los insectos tienen una extraña predilección por quedar atrapados entre los contactos de alto voltaje de las fuentes de poder. Otros parecen tener antojo insaciable por el aislante que cubre los alambres que llevan las líneas de corriente. Además, las telarañas dentro de las computadoras colectan polvo como si fueran un imán.

Ruido Eléctrico y Fallas Eléctricas

Los motores, ventiladores, equipo pesado y aun otras computadoras, algunas veces generan ruido eléctrico que puede causar problemas intermitentes con alguno de los equipos. Este ruido se puede transmitir a través del ambiente o por las líneas de potencia cercanas.

Los picos eléctricos son otro tipo de ruido eléctrico, que consiste de uno o varios picos de alto voltaje. Una aspiradora común conectada en la misma salida eléctrica de la computadora, puede generar un pico capaz de destruir la fuente de poder de la misma.

Los walkies-talkies, teléfonos celulares y otro tipo de radio transmisores, pueden causar un mal funcionamiento de la computadora cuando están transmitiendo. Los transmisores más potentes pueden causar un daño permanente a los sistemas.

Todos los radio transmisores deberán mantenerse a, al menos, metro y medio de los equipos.

Otro problema que se presenta muy a menudo, es la ausencia súbita de energía eléctrica, lo que origina la pérdida de los datos que en ese momento se estaban procesando o inclusive el daño irreparable de los mismos en el disco. Esto, especialmente en UNIX, lleva a la corrupción de sistemas de archivos del sistema operativo y requiere de una buena inversión de tiempo para repararlos o recuperarlos de respaldos, inclusive.

Rayos

Los rayos generan altas elevaciones de potencia que pueden dañar a las computadoras. Si el rayo choca con el metal del edificio, o golpea el pararrayos del edificio, la corriente resultante en su camino a la tierra puede generar un campo magnético intenso. Si las cintas donde se guardan los respaldos están cerca de la estructura metálica del edificio, pueden dañarse.

Vibración

La vibración causa diversos daños al equipo; aun la vibración que produce la gente, con el tiempo, puede ir aflojando a los conectores de sus zócalos. La vibración puede causar que los discos duros vayan saliendo de alineación e incrementa la probabilidad de un daño a los datos con su consecuente pérdida.

Humedad

La humedad es el amigo de la computadora, siempre y cuando este dentro de cierto rango. Evita la formación de cargas estáticas. Si el cuarto de las computadoras es muy seco, las descargas estáticas entre los operadores y la computadora (o entre las partes móviles de la computadora) puede destruir la información o dañar la computadora misma. Si el cuarto es demasiado húmedo, puede haber condensación en la circuitería, lo cual puede generar un corto.

Agua

El agua puede causar daños severos a una computadora. El primer daño es un corto, el cual pasa cuando el agua hace puente entre dos pistas del circuito impreso, una llevando voltaje y la otra tierra. Un corto puede causar que circule mucha corriente a través de una pista lo cual la calentará y posiblemente la fundiría; también pueden destruir componente electrónicos al circular mucha corriente por ellos.

El agua por lo general viene de las lluvias o por inundación y algunas veces de sanitarios ubicados en pisos superiores o por vandalismo.

Descuidos

Además de los problemas ambientales, las computadoras son vulnerables a un sinnúmero de accidentes.

Comidas y Bebidas

La gente necesita comer y beber para estar viva. Las computadoras, por otro lado, necesitan estar lejos de la comida y la bebida. Una de las maneras más rápidas para dejar un teclado en desuso es vaciándole refresco o una taza de café. Si este teclado es la consola del sistema, no será posible reiniciar la máquina hasta que sea reemplazado.

La comida, especialmente la grasosa, se colecta en los dedos de las personas y de allí en cualquier parte que esa gente toque. A menudo esto incluye superficies sensibles a la suciedad tales como cintas magnéticas y discos óptico. También este aceite muchas veces se pega en la superficie de las pantallas, lo que ocasiona menor visibilidad. Generalmente, la regla más simple es la más segura: Mantener toda la comida y líquidos alejados de las computadoras.

2.2 Ubicación del Equipo de Cómputo

La Empresa tiene un espacio reservado para el resguardo de diversos equipos (delicados en su mayoría) que se le denomina la Sala de Cómputo. Arriba de la sala, existen equipos de enfriamiento del aire acondicionado y del agua de los humidificadores que se encuentran en el interior de la misma.

Como se mencionó anteriormente, en este lugar se guardan diversos equipos a cargo de diferentes Departamentos, por lo que internamente la Sala esta dividida en las siguientes áreas (Fig. 2.2):

- Area de impresión. A cargo del Departamento de Procesamiento de Datos de La Empresa,
- Area de equipo de comunicaciones, del Departamento de Monitoreo,
- Area de refrigeradores y humidificadores del aire acondicionado, bajo la responsabilidad del Departamento de Mantenimiento
- Area entrega de reportes, del Departamento del Procesamiento de Datos y
- Tableros de control del Departamento de Mantenimiento.

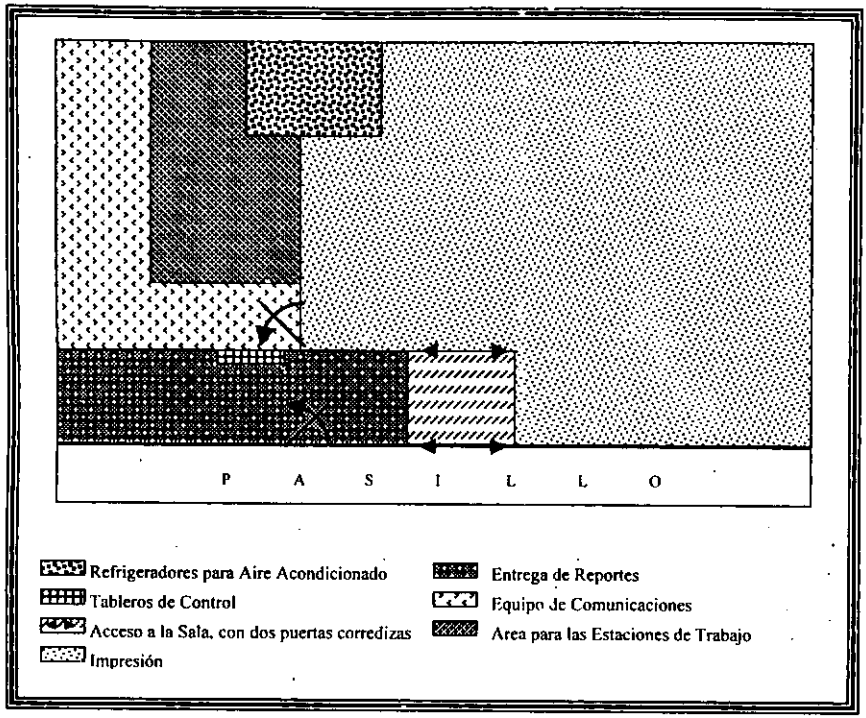

Figura 2.1 Vista Superior de la Sala de Cómputo

Identificación de los Puntos de Acceso

Sólo existe una puerta de entrada a la Sala. El área de Entrega de Reportes, tiene una puerta que sólo permite pasar a esta área pero no a la Sala. Además, esta área tiene dos ventanas; una que colinda con el pasillo y otra con la Sala y al lado derecho de la puerta de entrada a la sala se encuentra un ventanal. Esto se ilustra en la siguiente figura.

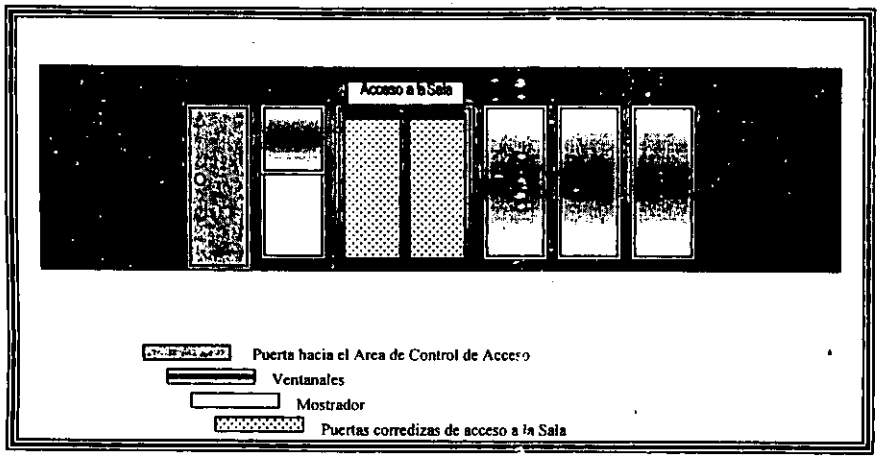

Fig. 2.2 Vista frontal (colinda con el pasillo) del acceso a la Sala

Esta sala sólo tiene una entrada "normal" y dos "potenciales"(Fig. 2.3). Se da el adjetivo de normal porque la puerta de entrada es la forma "correcta" de pasar a la Sala y la ventanilla del área de Entrega de Reportes que colinda con el interior de la Sala y el ventanal a la derecha de la entrada a la Sala son dos accesos potenciales, porque mediante un acto vandálico (rompiendo las ventanas) se puede lograr entrar (Fig. 2.3).

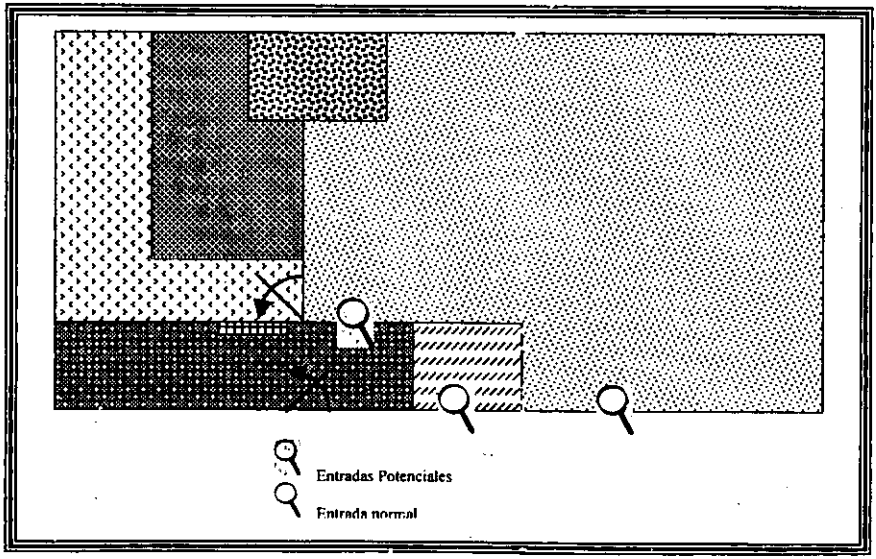

Fig. 2.3 Accesos normales y potenciales a la Sala (Vista Superior)

En el interior de la Sala, sólo al equipo de comunicaciones lo circunda un cancel, todo el demás equipo no tiene una barrera física que impida llegar a él. Esto significa que una vez que una persona logre entrar, puede desplazarse hacia cualquier área. Esto tiene algunas implicaciones serias y pone en riesgo la seguridad física del equipo de cómputo. La siguiente, es una lista de algunos riesgos a los que se encuentran expuestos los equipos:

- Desconexión accidental o a propósito del equipo. Cómo se ilustra en la Fig.2.2, el área de equipos de cómputo colinda con los refrigeradores para el aire acondicionado. Cuando éstos reciben mantenimiento, circulan por allí personas que pueden tropezar accidentalmente con los cables de alguno de los equipos y desconectarlos. Lo mismo aplica para el personal de limpieza.

- Los equipos Sun, se pueden dar de baja por hardware, ya sea apagando el switch de alimentación o con una secuencia de teclas. Especialmente, existe un equipo que tiene una consola que con una sola tecla da de baja el equipo. El personal de limpieza, a pesar de que se les prohibe tocar el equipo, cuando limpian el teclado, pueden presionar esta tecla o el switch de alimentación (de hecho ya sucedió una vez).

- Al circular gente cerca de los equipos, levantan el polvo que exista en la sala y, como se mencionó en la sección 2.1, es dañino para los equipos.

2.3 Protección del Equipo de Cómputo

Hasta aquí se han analizado las amenazas a las que se encuentran expuestos los recursos. En las siguientes secciones se darán las consideraciones para protegerlos de tales amenazas.

El plan para lograr una buena seguridad física, se basa en reducir los riesgos mencionados anteriormente mediante la adecuación de las instalaciones para la prevención de robos y accidentes, el monitoreo de las condiciones ambientales y el control del acceso a la Sala de Cómputo.

Acceso a las Computadoras

Después de analizar los accesos al equipo de cómputo, es obvio que necesitamos establecer nuestros "perímetros de seguridad". Un perímetro de seguridad es "el límite entre el resto del mundo y nuestra área segura". Ya existe un primer perímetro de seguridad, ya que se necesita una secuencia de números para poder abrir la puerta de entrada a la Sala de Cómputo y, para poder teclear la secuencia correcta, se necesita una llave para abrir la caja donde se encuentra el teclado. El contorno de la Sala no tiene otra entrada, debido a que no cuenta con ventanas ni canceles adicionales a los ilustrados en la Fig. 2.2.

Pero además de esta primera línea de defensa, se necesita un segundo perímetro que restrinja el acceso al área de computadoras. Poniendo un cancel en ésta área, podemos establecer un segundo perímetro de seguridad, porque, aunque alguien logre entrar a la sala, necesita la llave de la puerta del cancel para poder alcanzar los equipos (Fig. 2.4).

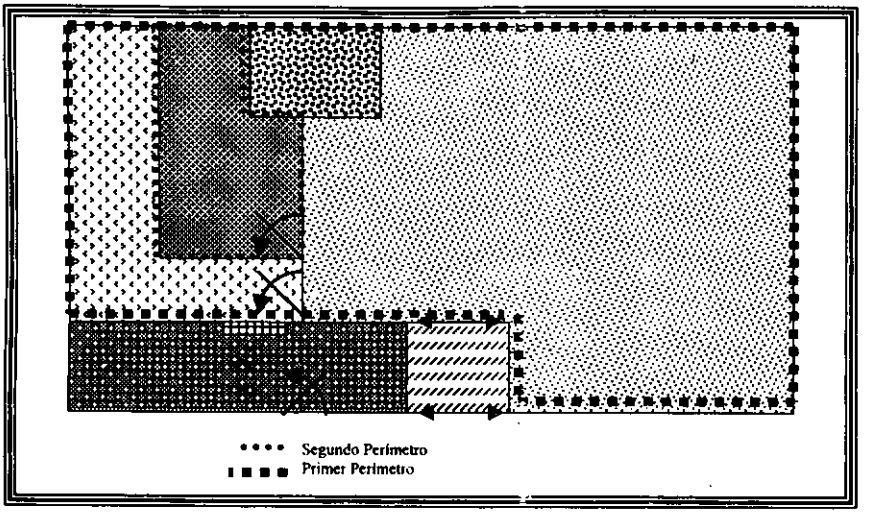

Fig. 2.4 Perímetros de Seguridad

Para controlar el acceso al primer perímetro de seguridad, sólo se deberán dar llaves y la secuencia de números a las personas autorizadas. Para controlar el acceso al segundo perímetro, sólo tendrán llaves de la puerta del cancel los administradores de los equipos.

El Medio Ambiente

Temperatura

Sun recomienda que los equipos se mantengan en un rango de temperatura que va de los 5°C a los 40°C y con una humedad relativa entre 20% y 80% sin condensación.

Para mantener la sala en condiciones ambientales operables, actualmente se tiene un piso elevado por el cual se inyecta el aire que sale de los refrigeradores a toda la sala, y mediante rejillas colocadas en los mosaicos del piso, se permite la salida de aire directamente en las áreas donde se encuentran los equipos que lo necesitan. Estos mismos refrigeradores tienen sensores que permiten controlar la temperatura a la que debe salir el aire. Además existe un termohidrógrafo en la sala que registra los niveles de temperatura y humedad.

La sala mantiene una temperatura de 20°C ± 1° y cuando esta fuera de este rango los mismos refrigeradores emiten sonidos de advertencia que se pueden escuchar en toda la Sala.

El piso falso de la Sala no se puede considerar una entrada potencial entre las diferentes áreas que existen en la Sala, ya que tienen una altura de 35 cm, por la que no es posible desplazarse. Además, para detener el mosaico, hay soportes que sostienen este piso a intervalos de 60 cm.

También, los equipos deberán tener un claro de, al menos, 10 cm para permitir la circulación de aire fresco entre ellos.

Fuego y Humo

Para la prevención de fuego se requiere tener un extintor a la mano, cerca de la entrada al cuarto donde están los equipos. Es necesario capacitar a los operadores de computadoras en la forma adecuada de usarlos. También es importante verificar, mensualmente de preferencia, el estado de cada extintor y los indicadores de carga de los mismos. Debe existir al menos un extintor cerca de donde haya equipo de cómputo y deberá haber un teléfono cerca para reportar el incidente al Departamento de Mantenimiento.

Para detectar la presencia de humo, se deben instalar sensores en el techo y en el piso falso y conectarse a un sistema de alarmas auditivas ubicada fuera y dentro de la Sala. También es necesario probarlas al menos trimestralmente. Debe haber, cuando menos, dos sensores de humo en el techo y dos en el piso en el área de equipo de cómputo.

Debido a la presencia de sensores de humo y por el daño que éste causa a los equipos, no se debe permitir fumar en el interior de la sala.

Polvo

Puesto que la Sala de Cómputo no tiene ventanas hacia el exterior y sólo pasa personal autorizado, casi no hay polvo en su interior. Sin embargo, es necesario limpiar las computadoras, teclados y monitores con aspiradora en intervalos semanales para recolectar el poco polvo que se llegara a acumular en ellas. Además, con la ayuda de los mantenimientos preventivos, se reduce aun más la posibilidad de una falla por polvo.

Temblor

Para prevenir un daño por temblor, las computadoras deben estar en mesas con base ancha para un buen soporte y a una altura no mayor a los 80 cm. Además, las computadoras deberán estar en el centro de las mismas para evitar su caída en caso de movimiento. No deberá haber objetos colgantes o adheridos al techo que puedan caer sobre ellas o algún otro mueble que por su altura pudiese tener probabilidad de dañarlas.

Insectos

Los mantenimientos preventivos también ayudarán a prevenir el anidamiento de insectos en las computadoras. Además, se deben realizar fumigaciones en periodos semestrales.

Ruido Eléctrico y Fallas Eléctricas

Para prevenir la generación de ruido eléctrico, se debe evitar conectar aparatos distintos de los componentes de las computadoras en los contactos destinados para ellas.

La carga que consume un equipo de cómputo con todos sus componentes en promedio es de 1500 Watts y una computadora con arreglo de discos consume 2500 Watts. Cada equipo debe tener protección para todos sus componentes por lo que se deben destinar los contactos para cada equipo junto con sus componentes con la protección adecuada (mediante fusibles) de las cargas permitidas (Fig. 2.5).

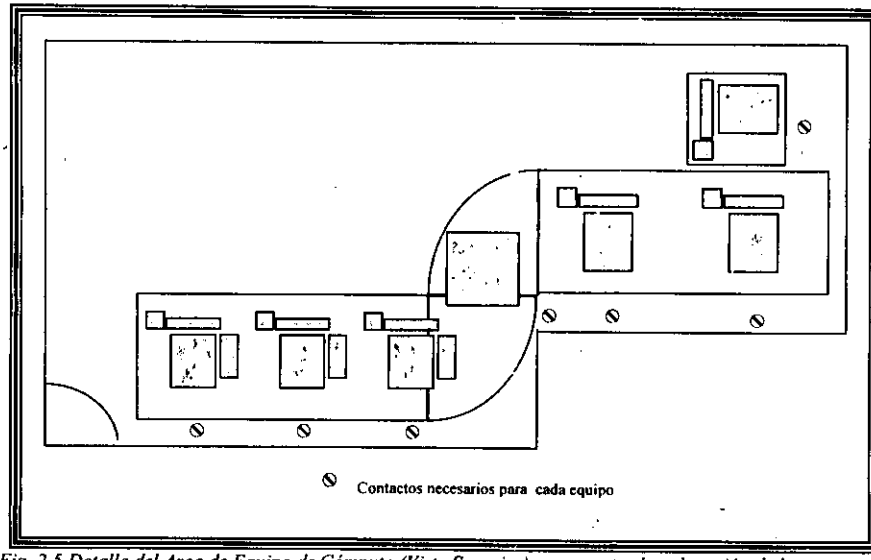

Fig. 2.5 Detalle del Area de Equipo de Cómputo (Vista Superior) que muestra la colocación de los contactos eléctricos

Las fallas de energía eléctrica son muy frecuentes, y sus daños son costosos, por lo que se debe contar con un UPS y una planta de luz. Su funcionamiento básico es el siguiente (Fig. 2.6):

La entrada de energía de la Comisión Federal de Electricidad (CFE) es la entrada del sistema. Esta energía pasa por un rectificador para convertirla en corriente directa para que cuando haya una

falla de energía, el banco de baterías (mediante la apertura y cierre de relevadores), proporcione la energía necesaria mientras se activa la planta de luz; el inversor que sigue, permite convertir en corriente alterna nuevamente la corriente directa que recibe; y por último, mediante relevadores, entra en operación la energía proporcionada por la planta de luz.

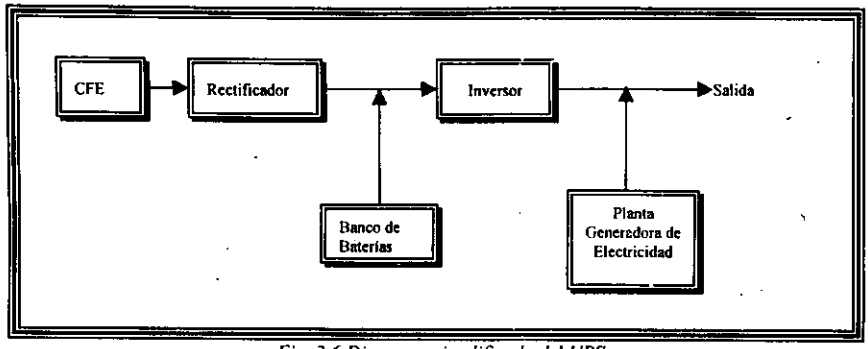

Fig. 2.6 Diagrama simplificado del UPS

Además, deben existir lámparas de emergencia en el interior de la Sala y en los lugares de paso de los empleados, debido a que, cuando entra en operación la planta de luz, no da soporte a la iluminación del edificio. Sólo protege las líneas donde se alimentan los equipos de cómputo.

Rayos

El edificio cuenta con, al menos, un pararrayos aterrizado en cada esquina del mismo para captar los rayos y dirigir sus descargas a tierra.

Para evitar problemas con los campos emitidos por los rayos, se debe evitar ubicar equipo, cintas magnéticas y discos magnéticos en las cercanías de las estructuras metálicas del edificio y de los conductores que funcionan como pararrayos.

Vibración

En el ambiente de la Sala de Cómputo de la empresa, no existe equipo que produzca vibración.

Humedad

Las Salas cuentan con dos refrigeradores que proporciona las condiciones de humedad y temperatura necesarias. Los mismos refrigeradores tienen sensores de humedad para mantenerla

dentro de las especificaciones. La humedad relativa de la Sala es de 50% y se monitorea por personal de mantenimiento mediante las lecturas al termohidrógrafo.

Agua

La Sala de Cómputo sólo tiene una fuente de agua que es la que alimenta a los refrigeradores. Aunque la probabilidad es baja de que haya una inundación, no está de más instalar sensores de agua en el piso que también enciendan la alarma auditiva en caso de inundación.

En la siguiente figura, se ilustran todos los componentes de Seguridad Física con los que debe contar la Sala de Cómputo de La Empresa.

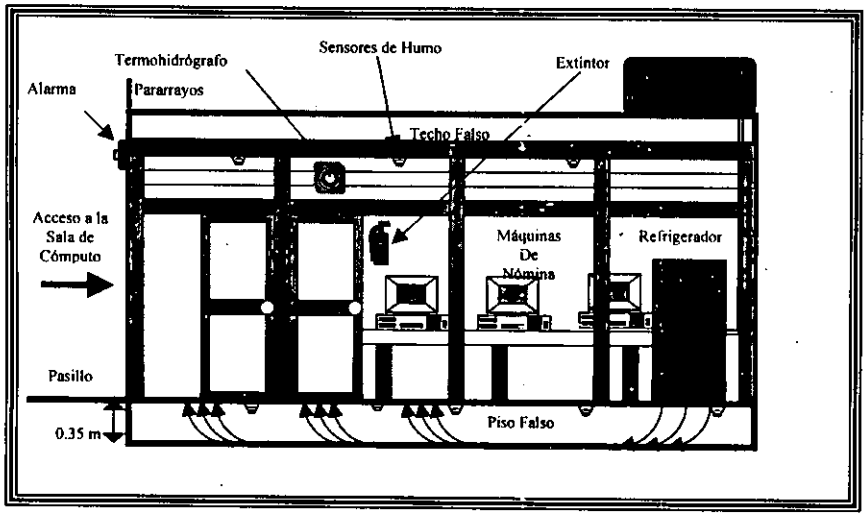

Fig. 2.7 Componentes de Seguridad Física

Mantenimiento Preventivo

El mantenimiento preventivo ayuda a prevenir los problemas que se pudiesen ocasionar por falsos contactos entre tarjetas y dispositivos de la computadora, acumulación de polvo y anidamiento de insectos, además de que sirve para revisar el estado físico de los componentes. Los mantenimientos preventivos deberán realizarse en los periodos vacacionales que son dos en el año.

Es importante también revisar el estado de los conectores de red, de los contactos de corriente y de las mesas que soportan el equipo.

Prevención de Accidentes

La buena aplicación de una Seguridad Física depende mucho de la concientización y educación de los usuarios y operadores. No hay otra manera de convencerlos que deben evitar ingerir alimentos líquidos y sólidos cerca de computadoras, que deben mantener cerrados los accesos restringidos, que no deben permitir el acceso a personas ajenas a las áreas de equipo de cómputo, etc., así como instruirlos también acerca de las condiciones bajo las cuales opera de manera apropiada un equipo de cómputo.

2.4 Control de Cintas

Los respaldos en La Empresa, se realizan en cintas de 4mm por 120 metros con capacidad de almacenamiento de 8 GB en formato de alta densidad en cartuchos con dimensiones de 7.3 cm por 5.3 cm.

El fabricante recomienda una temperatura de operación de entre 5°C y 45°C y una humedad relativa entre 20% y 80%. Recomienda también que se eviten los cambios bruscos de temperatura, que pueden darse cuando transportamos las cintas de un ambiente con una temperatura mas caliente o más fría que la de la Sala de Cómputo; si éste es el caso, antes de usar la cinta, se deberá exponer al ambiente de operación por el tiempo que estuvo fuera de él, hasta un máximo de 24 horas.

Por sus dimensiones, alcanzan fácilmente en los bolsillos por lo que deberán resguardarse en un lugar bajo llave. También es necesario que estén en otra ubicación distinta de la Sala de Cómputo y, si es posible, en otro edificio, ya que en caso de siniestro (inundación, fuego o temblor) podrían dañarse, lo que provocaría que se perdiera tanto la información de los equipos como la respaldada en las cintas.

Las cintas deben mantenerse lejos de campos magnéticos fuertes que pudieran dañarlas como monitores, conductores de los pararrayos o la estructura metálica del edificio.

El tiempo de vida de una cinta varía. Hay quienes aseguran haber hecho más de 100 operaciones con la cinta (entre lecturas, escrituras y rebobinados) o hay quienes las han grabado hasta 30 veces. Los respaldos que se hacen en las máquinas en producción se mantienen como históricos, por lo que sólo se utilizan una vez. Estas cintas deben rebobinarse o leerse al menos anualmente, una vez que se han almacenado como históricos, porque de lo contrario pueden quedar inservibles. Para el caso de los equipos de desarrollo, las cintas son reutilizadas durante medio año, tiempo en el que se escriben aproximadamente en 10 ocasiones y las operaciones de lectura son variables desde cero hasta una lectura semanal. Para detalles sobre reemplazos de cintas, ver el Capítulo 3 en la sección de respaldos o el Capítulo 4 en la misma sección.

Las unidades de cinta, además del mantenimiento preventivo que van a recibir, se deben limpiar con cintas especiales para mantener las cabezas óptimas. La limpieza debe hacerse al inicio de la semana.

2.5 Bitácoras e inventarios

Las bitácoras son cuadernos donde se anotan los cambios que los administradores hacen a cualquier componente del sistema, instalaciones de software, mantenimientos preventivos, errores reportados por el sistema, problemas y soluciones, etc, que sirven de diagnóstico para problemas similares futuros o para poder rastrear el origen de una falla. No es recomendable escribir contraseñas en ellas. Debido a que allí se describe la configuración de la máquina y software instalado e información referente a usuarios y grupos, se considera como información fuera de línea importante y, por lo tanto, también se debe mantener bajo llave.

Los inventarios son la única forma de comparar lo que tenemos con lo que deberíamos tener, por lo que una modificación o eliminación de algún componente en esa lista, evita que podamos detectar un faltante. Es recomendable que más de uno tenga una copia de los mismo, por ejemplo, el Coordinador del Area y el Administrador de los Sistemas de Cómputo.

Capítulo 3: Seguridad en el Sistema Operativo

3.1 Características de Seguridad del Sistema Operativo

Para la mayoría de la gente, una computadora es una herramienta para resolver problemas que, cuando se conecta a una red electrónica, se convierte en parte de un poderoso sistema de comunicaciones.

En el corazón de cada computadora, existe un conjunto de programas llamados "Sistema Operativo". Este es el software que controla el sistema de entrada/salida de la computadora, así como los controladores de disco, del teclado y la ejecución de programas. El sistema operativo es también un conjunto de mecanismos y políticas que ayudan a definir la distribución controlada de los recursos del sistema.

Todos los Unix pueden ser divididos en tres partes:

- El kernel o el corazón del sistema Unix. Es el sistema operativo. El kernel es un programa especial que se carga cuando la computadora inicia; controla los sistemas de entrada y salida; permite que múltiples programas se ejecuten al mismo tiempo distribuyendo el tiempo y la memoria entre ellos. El kernel incluye el sistema de archivos, el cual controla cómo se almacenan los archivos y los directorios en el disco duro de la computadora. El sistema de archivos es el principal mecanismo por el cual se refuerza la seguridad de la computadora.

- Programas de utilidad estándar. Los corren tanto los usuarios como el sistema. Algunos programas son pequeños y sirven para una única función, por ejemplo, *ls* lista archivos; *cp* los copia. Otros programas son más grandes y realizan múltiples funciones, por ejemplo, *sh* y *ksh*, que son los shells de Unix que procesan comandos de usuarios y ellos mismos son lenguajes de programación.

- Archivos de bases de datos del sistema. La mayoría de ellos son relativamente pequeños y son usados por una variedad de programas. Un archivo, */etc/passwd*, contiene la lista maestra de todos los usuarios en el sistema. Otro archivo, */etc/group*, describe los grupos de usuarios con derechos de acceso similares.

Desde el punto de vista de la seguridad de Unix, estas tres partes interactuan con una cuarta:

- Política de Seguridad, que determina como se va a proteger la computadora desde el punto de vista de los usuarios y la administración de la misma.

Antecedentes

Dennis Ritchie, uno de los fundadores del sistema operativo Unix, escribió: "No fue diseñado desde un principio para ser seguro. Fue diseñado con las características necesarias para ser útil a la seguridad"[8].

Unix es un sistema operativo multiusuario y multitarea. Multiusuario significa que el sistema operativo permite a muchas personas usar la misma computadora al mismo tiempo. Multitarea significa que cada usuario puede correr diversos programas simultáneamente.

Una de las funciones naturales de tales sistemas operativos es evitar que los usuarios o programas interfieran unos con otros, ya que sin tal protección un programa podría accidentalmente borrar archivos o aún dar de baja la computadora. Para cuidar que tales desastres no pasen, se puso en la filosofía de diseño de Unix cierta seguridad que va más allá de la mera protección de la memoria.

Debido a las muchas "historias de terror" existentes sobre problemas de seguridad de Unix, gran parte de la comunidad computacional tiene la convicción de que la frase "Seguridad en Unix" es una contradicción. Sin embargo, casi todos los huecos de seguridad que han sido encontrados en Unix han resultado de fallas en programas individuales, o en las interacciones entre ellos, y no de fallas en el diseño del propio sistema operativo. Su reputación como sistema inseguro, no viene de su diseño, sino de la práctica. Durante sus primeros quince años, fue usado principalmente en ambientes académicos, en los que la seguridad no era una preocupación.

Unix controla la manera en que los usuarios manipulan los archivos, modifican las bases de datos del sistema y usan los recursos. Desgraciadamente, tales mecanismos no ayudan mucho cuando los sistemas están mal configurados, se usan sin cuidado o contienen software con problemas. Casi todos los hoyos de seguridad que se han encontrado en Unix han resultado de este tipo de problemas y no por defectos en el diseño intrínseco del sistema.

El tipo de sistema operativo Unix que se utiliza en la Nómina es Sun OS que, al integrarle otras utilerías y ambiente de usuario, se llama ambiente operativo Solaris que es como se comercializa. Es uno de los muchos sistema Unix que existen en la actualidad y es una mezcla del Unix System V y el BSD (las dos definiciones originales de las cuales se derivan todos los demás Unix). Ofrece un amplio conjunto de características de seguridad que se acomodan a los requerimientos cambiantes de los ambientes de computación. Proporciona herramientas automatizadas que simplifican la configuración de la seguridad del sistema y reporta vulnerabilidades potenciales; ofrece servicios de archivos y directorios compartidos en forma segura así como plataformas para el desarrollo de aplicaciones seguras y el soporte de importantes estándares de seguridad internacionales.

Solaris se enfoca a cuatro áreas de la seguridad de la computadora:

- El acceso al sistema y a los datos del sistema,

[8] Citado en Garfinkel Simson. 1996. Practical Unix & Internet Security. O'Reilly & Associates Inc. USA. Segunda Edición.

- Monitoreo de la actividad en el sistema,
- Evitar acciones que amenacen la seguridad y
- Reportar problemas de seguridad.

- Monitoreo de la actividad en el sistema,
- Evitar acciones que amenacen la seguridad y
- Reportar problemas de seguridad.

El Acceso al Sistema

Seguridad de EEPROM

Todas las tarjetas de CPU de los equipos Sun tienen una EEPROM que contiene el programa "monitor" el cual controla al sistema durante el arranque. Cuando se enciende una máquina, el firmware monitor automáticamente correrá un diagnóstico. Si esta secuencia de arranque se interrumpe con la combinación de teclas stop-A, entonces la máquina quedará con el indicador interactivo del monitor (`ok`). Desde este indicador, se puede instruir al monitor que arranque el sistema desde cualquier dispositivo: CD-ROM, disco externo o aun otra máquina en la red. Para limitar esto, el monitor tiene tres modos de seguridad: no seguro (el default), comando seguro y completamente seguro.

En el modo completamente seguro se tiene que proporcionar una contraseña, llamado el "password EEPROM", antes que el monitor haga algo. Esto es un poco inconveniente en máquinas de escritorio que están expuestas a ser desconectadas accidentalmente, ya que la característica automática de arranque se interrumpiría para pedir un password EEPROM para continuar. Si el password es olvidado con el modo de seguridad completo y si la máquina se da de baja, no se podrá reiniciar. Lo único que se puede hacer para borrar el password sería cambiar el chip de la EEPROM, lo cual en algunas máquinas significa cambiar la tarjeta del CPU.

El modo de comando seguro permite que se arranque el sistema sin pedir un password EEPROM, pero limita el cambio de configuración en alguna de las variables existentes a nivel del indicador "`ok`". Es posible cambiar el password EEPROM en modo comando seguro sin saber el password anterior. Cuando la máquina está en línea, el usuario root puede cambiarlo con el comando "*eeprom*".

Finalmente, en el modo no seguro, no hay restricciones en el cambio de variables de ambiente a nivel monitor ni al momento de iniciar la máquina.

Control de Cuentas de Acceso

La primera, y más importante, forma de restringir accesos no autorizados al sistema es mediante el control de cuentas (logins). Todas las cuentas en el sistema deberán tener un password. Una cuenta sin password hace que el sistema completo sea accesible a cualquiera que adivine un nombre de usuario.

El sistema operativo también proporciona el "envejecimiento del password". Esta capacidad permite al administrador indicar tres periodos en cuanto al cambio de password: periodo máximo, periodo mínimo y días de aviso. El periodo máximo se refiere al tiempo en el que un usuario utilizará ese mismo password sin necesidad de cambiarlo. El periodo mínimo se refiere a cuanto tiempo, a partir del momento en que se modificó, el usuario permanecerá con ese mismo password sin posibilidad de cambiarlo. Y los días de aviso se refieren al número de días antes de alcanzar el

periodo máximo en los que el sistema le estará avisando al usuario que es necesario cambiar su password. Por ejemplo, si el administrador indica que el periodo máximo es de 20 días, el mínimo de 5 y los días de aviso 5; entonces el usuario, cuando cambie su password, no podrá cambiarlo en los próximos 5 días (periodo mínimo); cuando se cumplan 15 días sin que haya cambiado su password, el sistema le enviará avisos indicándole el número de días que faltan para que se bloquee la cuenta (días de aviso) y si a los 20 días el usuario no cambió su password su cuenta se bloqueará (periodo máximo).

La rutina del sistema operativo para manejo de passwords, tiene las siguientes restricciones:

- El password debe tener mínimo seis caracteres de longitud (por default)
- Los primeros seis caracteres del password deben contener al menos dos caracteres alfabéticos y al menos un caracter numérico o especial.
- Al cambiar un password, éste debe diferir en al menos tres caracteres del anterior

Los datos de las cuentas de usuario (grupo, identificador de usuarios, nombre, directorio y shell) se almacenan en un archivo en texto claro (archivo */etc/passwd*) y los passwords cifrados en otro (archivo */etc/shadow*)[9]. Además, el archivo */etc/passwd* puede leerlo cualquier usuario, pero el */etc/shadow* sólo lo puede leer root.

Predicción de Passwords

El programa que controla el acceso al sistema (*login*) sólo permite cinco intentos fallidos antes de suspender el puerto del login. Esto dificulta que un password sea adivinado a prueba y error ya que un hacker puede tener varios probables passwords, pero sólo puede intentar cinco en un tiempo. Lo que es más importante es que esto detiene un enfoque programático de tratar cada palabra de un diccionario, se puede intentar, pero lleva más tiempo.

Si se hacen cinco intentos fallidos, el proceso encargado de llevar registros de actividades (*syslogd*), enviará un mensaje de alarma a la consola. Sin embargo, registra poca información acerca del intento fallido. Se puede registrar información complementaria mediante la creación del archivo */var/adm/loginlog*:

```
$ touch /var/adm/loginlog
$ chmod 600 /var/adm/loginlog
$ chgrp sys /var/adm/loginlog
```

Sólo hasta el quinto intento fallido es cuando se registra información en esta bitácora. Esto significa que se pueden hacer cuatro intentos sin ser detectados. Ha habido varias peticiones para que SunSoft permita configurar el número de intentos fallidos.

[9] En un principio los passwords encriptados y la información de la cuenta se almacenaban juntos, pero esto permitía obtener el password y tratar de adivinarlo por fuerza bruta.

Cuentas Especiales

Hay dos maneras comunes de ingresar a un sistema: mediante el uso de una cuenta convencional de usuario (o mediante el uso de la cuenta de root) y utilizando una de las cuentas especiales. Las cuentas especiales permiten a los usuarios realizar comandos administrativos sin utilizar una cuenta de root. El administrador asigna un password a estas cuentas o las deshabilita.

La tabla 3.1 lista las cuentas del sistema y sus usos. Estas realizan funciones especiales y cada una tiene su propio identificador de grupo (GID).

Cuenta	GID	Uso
root	0	No tiene restricciones y anula las otras cuentas, protecciones y permisos. La cuenta de root tiene acceso al sistema completo. El password de esta cuenta deberá ser protegido cuidadosamente.
daemon	1	Controla el procesamiento en segundo plano (background).
bin	2	Posee casi todos los comandos.
sys	3	Es dueña de muchos de los sistemas de archivos.
adm	4	Es dueño de ciertos archivos y comandos administrativos.
sysadmin	14	Realiza tareas administrativas
lp	71	Es dueña de los objetos y archivos de datos almacenados para la impresora.
uucp	5	Posee los objetos y archivos de datos almacenados para UUCP, el programa de copia de Unix a Unix.
nuucp	9	Lo usan sistemas remotos para entrar al sistema y transferir archivos.

Tabla 3.1 Cuentas del Sistema

Deshabilitación Temporal de Acceso al Sistema

Se puede evitar el acceso de usuarios al sistema de dos formas:

- Creando el archivo /etc/nologin. Si un usuario intenta entrar mientras existe esta archivo, se despliega el contenido del mismo y se termina la conexión. No aplica a los accesos del super usuario.

- Llevando al sistema al nivel 0 o single user

Entrada Directa de root

Cualquier usuario con UID 0 (a menudo llamado superusuario o root) tiene acceso a toda la información de los usuarios en el sistema sin importar los permisos indicados por el dueño. El nombre del superusuario puede no ser root necesariamente, con el simple hecho de tener el UID 0 lo convierte en superusuario.

Algunas de las cosas que el superusuario puede hacer son:

CONTROL DE PROCESOS

- Cambiar la prioridad de ejecución de cualquier proceso,
- Enviar cualquier señal a cualquier proceso,
- Alterar los "límites duros" (hard limits) para el máximo tiempo de CPU, máximo tamaño de archivo, segmento de datos máximo, segmento de pila máximo y tamaño de los archivos core,
- Encender o apagar la auditoría,
- Pasar por alto las restricciones de entrada al momento de estarse dando de baja el sistema,
- Cambiar su Identificador de Usuario (UID) por el de otro usuario,
- Desconectar a todos los usuarios, tirar y reiniciar el sistema.

CONTROL DE DISPOSITIVOS

- Accesar cualquier dispositivo de red,
- Apagar la computadora,
- Cambiar fecha y hora,
- Leer o modificar cualquier localidad de memoria,
- Crear un nuevo dispositivo (en cualquier lugar del sistema de archivos) con el comando *mknod*.

CONTROL DE RED

- Correr servicios de red en puertos reservados,
- Reconfigurar la red,
- Poner la interfaz de red en modo promiscuo y examinar todos los paquetes en la red.

CONTROL DE LOS SISTEMAS DE ARCHIVOS

- Leer, modificar o borrar cualquier archivo o programa en el sistema,
- Ejecutar cualquier programa,
- Cambiar la etiqueta electrónica (mapa de particiones) de un disco,
- Montar y desmontar sistemas de archivos,
- Agregar, remover o cambiar cuentas de usuario,
- Habilitar y deshabilitar cuotas y contabilidad,
- Usar la llamada al sistema chroot(), la cual cambia el directorio raíz de un proceso.

Dado su imponente poder, es buena idea restringir las actividades hechas por él. La mayoría de los administradores entran directamente como root creyendo que se necesitan privilegios de root para realizar sus tareas. En Solaris, el grupo 14, a menudo llamado "*sysadmin*" da a sus miembros el privilegio para realizar tareas de administración.

Lo mejor es no dejar a nadie entrar directamente como root. Es preferible entrar con un usuario normal y cambiar posteriormente al usuario root utilizando el comando *su*. Esto también tiene la ventaja de mantener una mejor auditoria ya que el uso del comando *su* se registra en una bitácora.

Solaris por default no permite acceso directo de root desde cualquier lugar, excepto la consola. Esto es controlado por la siguiente línea en el archivo */etc/default/login*:

CONSOLE=/dev/console

Si está restringido el acceso físico de la consola en un cuarto de computadora, entonces la opción que por default maneja Solaris es más confiable aun.

Distribución de Funciones

No sólo debe vigilarse el acceso directo ilegal de root, a menudo el pertenecer a ciertos grupos especiales puede ser suficiente para comprometer severamente la seguridad. Un usuario puede ser miembro de uno o más grupos; el grupo primario se especifica en el archivo */etc/passwd* y la membresía a grupos secundarios está definida en el archivo */etc/group*. Para determinar el acceso a un archivo se usan ambos tipos de grupos: primario y secundario. Un usuario puede hacer primario cualquiera de sus grupos secundarios utilizando el comando "*newgrp*".

Algunos sistemas Unix antiguos usan la facilidad de passwords de grupo, donde los usuarios pueden cambiarse a un grupo determinado siempre y cuando conozcan el password del grupo. En Solaris, si un usuario se cambia a uno de sus grupos secundarios, no necesita password.

EL GRUPO SYS (GRUPO 3)

Hay varios comandos importantes que pueden ser ejecutados por cualquiera con identificador de grupo (GID) 3. El más importante es *ufsdump*. La idea es que los operadores realicen respaldos sin conocer el password de root.

EL GRUPO SYSADMIN (GRUPO 14)

Cualquiera en el grupo sysadmin puede correr la herramienta para administración de usuarios, grupos, hosts, impresoras, software y puertos seriales (que puede ser mediante el programa *admintool*). Con estos privilegios es posible eliminar y dar de alta nuevamente la cuenta de root con un password nuevo.

El Comando Swap User (su)

El comando "*su*" permite a cualquier usuario cambiar su UID a otro usuario a sabiendas de que conoce su password. Este es potencialmente un comando muy peligroso, de hecho, como root se puede usar *su* para cambiar a otro usuario "sin conocer su password", por lo que, al comprometerse la cuenta de root, el sistema está comprometido. Pero esto no queda ahí, porque puede tener mayores implicaciones cuando se involucra NFS (ver NFS en Acceso a la Red).

Es prudente monitorear el uso de tan poderoso comando. El archivo */etc/default/su* contiene los siguientes parámetros:

- SULOG: Define el archivo donde se registra su uso
- CONSOLE: A que terminal se envía el mensaje, si es que se especifica
- PATH: El PATH por default del nuevo shell

- SUPATH: El PATH por default si el *su* es a root
- SYSLOG: Si envía señales o no a *syslogd*

Los reportes de la actividad del comando *su* se envían al registrador del sistema (*syslogd*) y como resultado los mensajes se muestran en la consola por default. Adicionalmente, toda la actividad se registra en la bitácora */var/adm/sulog* (la configuración por default de la opción SULOG). Si *syslogd* no se está ejecutando, se pueden enviar los mensajes directamente a la consola definiendo CONSOLE en */etc/default/su*. Es poco útil el definir las variables SUPATH y PATH ya que los usuarios pueden indicarlas al momento de utilizar el comando.

Programas setuid

Si un archivo ejecutable tiene el bit setuid prendido, entonces correrá con el Identificador de Usuario Efectivo (EUID) del dueño del archivo.

El Identificador de Usuarios Real (UID) puede determinarse aun. Los programas conscientes de la seguridad deberían verificar el UID para ver si es igual que el EUID. Sin embargo el sistema de archivos de UNIX, acepta EUID en la misma manera que UID.

Esta facilidad a menudo es usada por aplicaciones multiusuario para accesar archivos que pertenecen a otros. El ejemplo más simple es el comando "passwd". Se quiere que los usuarios puedan cambiar sus passwords, esto requiere de permiso de escritura a */etc/shadow*. Sin embargo, no se quiere que escriban todo lo que quieran, por lo que sólo root tiene permisos de acceso a */etc/shadow*. Si los usuarios quieren cambiar su password, deben correr el programa "*passwd*", el cual tiene el EUID 0 (root):

```
$ ls -l /etc/shadow /bin/passwd
-r-------- 1 root sys   849 Sep 27 10:21 /etc/shadow
-r-sr-sr-x 1 root sys 96796 Jul 15  1997 /bin/passwd
```

La misma facilidad está disponible para el acceso de grupo conocido como Identificador del Grupo Efectivo (EGID).

La facilidad de setuid/setgid a menudo se usa en abuso. Si un hacker no puede llegar a usurpar la cuenta de root al menos podrá correr programas como tal usuario.

Seguridad del Shell

Hay muchos intérpretes de comandos a elegir. El programa login pone a los usuarios en su shell por default, el cual se especifica en el archivo */etc/passwd*. En Solaris ya no pueden cambiar su shell asignado por default, esto era posible en versiones anteriores (1.x). Los tipos de shells se pueden restringir creando una lista de shells permitidos en el archivo */etc/shells*.

El evitar que los usuarios cambien su shell significa que el administrador puede determinar qué archivo global de inicialización usarán para reforzar la política de seguridad. Para el bourne shell (sh) y korn shell (ksh) el archivo que lee es */etc/profile* y los usuarios de C shell (csh) usan */etc/.login*.

Estos archivos son un buen lugar para indicar algunos defaults saludables para las variables del shell tales como PATH, UMASK o LD_LIBRARY_PATH.

El Comando PATH

La versión de comandos que un usuario ejecuta esta determinada por la variable de ambiente "PATH". Es un riesgo de seguridad tener un "." en el comando PATH de un usuario, ya que esto le indica al shell que busque en el directorio actual el comando que quiere ejecutar. El usuario puede pensar que esta ejecutando el programa *ls* localizado en */usr/bin* cuando en realidad está ejecutando *./ls* que dejó un hacker. Si "." debe ir en el PATH hay que ponerlo al final. Lo que es más importante es no tener un "." en el PATH de root, de esta manera, el usuario root tiene que teclear la ruta completa del comando (*/usr/bin/ls*). Un hecho menos frecuente es que ":" al principio o al final del PATH (en el caso de ksh o sh) tiene el mismo significado que "." lo mismo que "::", así que hay que verificar esto también.

Shells Restringidos

Es un riesgo de seguridad el tener cuentas de invitados ("guest") con la que cualquiera puede entrar. Estas, por lo general, tienen nombres de cuentas como guest, visitante, demo o temp. Estos son los primeros nombres que un hacker intentaría. Si no es posible darles su propia cuenta a usuarios válidos, entonces es posible restringir la actividad de las cuentas invitadas usando shells restringidos.

Solaris proporciona los shells restringidos */usr/lib/rsh* y */usr/bin/rksh*. El primero no debe ser confundido con el comando BSD shell remoto */usr/bin/rsh*. Si los shells restringidos se especifican como los shells por default, entonces los usuarios no podrán:

- Cambiar de directorio
- Cambiar su comando PATH
- Usar la redirección del shell (ejemplo >, >>)
- Ejecutar comandos indicando la ruta absoluta

El uso de comandos se puede limitar poniendo los permitidos en el directorio especial *./bin* o el administrador puede crear otro directorio */usr/rbin* e indicarlo apropiadamente en el PATH. Es importante el examinar cada comando que se ponga en el directorio *bin* restringido. Un error clásico es permitir al usuario correr el programa *vi*. Una vez en *vi* es fácil ejecutar un shell normal "!sh" por ejemplo.

Shells Setuid

Los scripts de shell pueden correrse con EUID y EGID igual que los archivos binarios, sin embargo, son muy inseguros ya que es posible interrumpirlos potencialmente dejando al usuario en un shell con privilegios de root. Dependiendo del tipo de shell, es como se maneja el EUID:

- El Bourne Shell (sh) tiene protección contra esta amenaza. Un script de bourne shell ignorará el EUID y EGID regresándolos a UID y GID respectivamente. Sin embargo este comportamiento requiere que el shell sea ejecutado con la opción "-p".

- Korn Shell (ksh) no tiene esta característica. Sin embargo, lee */etc/suid_profile* si el EUID no es igual al UID o el EGID no es igual al GID. Pero nuevamente, se necesita la opción "-p".

- C Shell (csh) no permitirá ejecutar scripts setuid o setgid, dando un mensaje de "permiso denegado". Invocando csh con la opción "-b" deshabilitará esta característica de seguridad.

Accesos con ftp

El demonio de ftp autentica a los usuarios de acuerdo a cuatro reglas:

- El nombre del usuario debe estar en la base de datos de passwords (*/etc/passwd*) y tener un password que no sea nulo. El password lo debe proporcionar el cliente antes de realizar cualquier operación

- Si el nombre del usuario aparece en el archivo */etc/ftpusers*, ftp niega la entrada

- Si el nombre del usuario es "anonymous" o "ftp", debe existir una entrada para la cuenta "ftp" en los archivos */etc/passwd* y */etc/shadow*. Si es así, el usuario puede entrar al sistema dando como password una cuenta de correo

- Se niega el ingreso si el shell del usuario no esta listado en el archivo */etc/shells*. Si éste archivo no existe, entonces el shell debe ser uno de los que se listan a continuación

```
/usr/bin/sh   /usr/bin/csh  /usr/bin/ksh
/usr/bin/jsh  /bin/sh       /bin/csh
/bin/ksh      /bin/jsh      /sbin/sh
/sbin/jsh
```

Además, el archivo */etc/default/ftpd* controla los mensajes de bienvenida y la umask que por default utilizará ftp para la transferencia de archivos.

Acceso a los Datos y Dispositivos del Sistema

Una vez que se han puesto restricciones en el login, el siguiente paso es controlar el acceso a los datos en el sistema. Para esto, Solaris permite al administrador controlar la facilidad general para utilizar los recursos, características para el control de permisos de archivos y capacidades de auditoría.

Configuración y Verificación del Estado de la Seguridad

Hay ocasiones en las que es esencial tratar de evaluar el estado de la seguridad general del sistema y/o establecerla propiamente. Para enfrentar esto, Solaris incluye una Herramienta Automatizada para Seguridad de Acceso (ASET). ASET puede reconocer automáticamente el estado del sistema así como ponerlo en uno de los tres estados de seguridad predefinidos: bajo, medio o alto.

Cuando se corre periódicamente, ASET avisará al administrador de cualquier deficiencia de seguridad. ASET checa la existencia de un password de EEPROM el cual protege de que un individuo no autorizado levante el sistema en modo mono-usuario; verifica el uso de la variable UMASK la cual dicta los permisos por default cuando se crea un archivo; comprueba el uso de la variable PATH que indica el orden en que se harán búsquedas en los directorios para la ejecución de comandos; revisa los permisos de los archivos del sistema; examina los permisos de los directorios home; busca la existencia de nuevos programas setuid: los contenidos de los archivos *.rhosts*, */etc/passwd* y */etc/group*; y compara los tamaños de los archivos en */usr/bin* y */bin*. El administrador tiene la opción de ser notificado de problemas potenciales por correo.

Cuando se usa para poner el sistema en modo de seguridad bajo, ASET se asegura de que los permisos de archivos estén en el valor estándar de instalación. En el modo de seguridad medio, ASET trata de proporcionar la seguridad adecuada para la mayoría de los ambientes; modifica los permisos de algunos archivos del sistema (ejemplo: *ttytab*, *host.equiv*) y parámetros para restringir el acceso al sistema. En el modo de seguridad alto, ASET produce un sistema sumamente seguro; muchos archivos del sistema y parámetros se habilitan para permitir sólo el mínimo acceso, este nivel puede deshabilitar el IP Forwarding.

Con ASET, un administrador no necesita perder tiempo "cazando" hoyos de seguridad en el sistema.

Permisos en Archivos

Los datos se protegen mediante el uso de controles de acceso discrecionales en los archivos. Se pueden dar permisos de lectura, escritura y/o ejecución al dueño del archivo, al grupo o a todos los demás. La capacidad de grupos permite a los usuarios establecer conjuntos independientes de usuarios. Se puede elegir entonces los tipos de accesos que tendrán los grupos en el sistema. Además, mediante el comando *umask* se pueden indicar los permisos que por omisión tendrá cualquier archivo que genere el usuario. La forma en que trabajan los permisos se entienden bien para archivos regulares pero muchas veces no se conoce la forma en que aplican en directorios. En

la siguiente tabla se muestra el significado de los permisos para archivos y directorios, así como permisos especiales:

Tipo de Archivo	Permisos	Significado
Directorio	d-w-------	Los archivos pueden ser creados/borrados, sólo si el directorio es el directorio actual.
	dr--------	El contenido puede listarse pero los atributos del archivo no pueden ser leídos.
	d--x------	Se puede ingresar al directorio y se puede usar en rutas absolutas.
	dr-x------	Los atributos de los archivos que contiene se pueden leer.
	d-wx------	Pueden crearse/borrarse archivos aun cuando el directorio no sea el actual.
	d-------wt	Evita que los archivos sean borrados por otros con permiso de escritura. Se usa en /tmp.
	d--s--s---	Sin efecto (el setgid bit tuvo algún significado en Solaris 1.x).
Regular	-r--------	Permite leer el archivo.
	--w-------	Permite al usuario modificar o borrar el archivo.
	---x------	Sólo el dueño puede ejecutar el archivo (Los scripts de shell necesitan tanto permisos de lectura como de ejecución para poder ejecutarse).
	---s------	Se ejecuta con UID efectivo del dueño.
	------s---	Se ejecuta con el GID efectivo del grupo.
	-rw------T	No se actualiza la última fecha de modificación. Algunas veces se indica en archivos swap.
	---t------	Sin efecto (llamado el sticky bit, tuvo un significado en SunOS 3.5)
	----rwl---	Sin efecto (Bloqueo de archivo obligatorio).

Tabla 3.2 Interpretación de Permisos en Archivos y Directorios

La versión 2.6 de Solaris incorpora las listas de control de acceso (ACL) que permiten controlar los permisos a un archivo o directorio para más de un usuario o grupo e inclusive se pueden indicar los permisos con los que se crearan nuevos archivos en un directorio. Esto evita la necesidad de dar permisos a todos para realizar cierta operación. Los ACL se han implementado tanto para Sistemas de Archivos de Usuarios (UFS) así como Sistemas de Archivos en Red (NFS).

Cifrado de Archivos

Para asegurarse de que otro usuario no pueda leer la información de un archivo, se puede cifrar. Normalmente los archivos se almacenan en el disco en texto plano, pero mediante el uso de los comandos *crypt, des* o *vi –x* se hacen incomprensibles; sólo los usuarios que conocen el password con el que se cifró la información podrán verla en claro nuevamente.

UMASK

Los permisos de los archivos en UNIX es la primera línea de defensa, pero a menudo es pasada por alto. Si los archivos son creados sin especificar explícitamente sus permisos, entonces heredarán los permisos por default definidos por la variable *umask* del usuario. La umask estándar, 022, permite que los archivos sean leídos por los demás, una umask más seguras sería 077. En la tabla 3.3 se dan algunos ejemplos de restricciones con umask.

Nivel de Seguridad	umask	Lo que Deshabilita
Permisible (744)	022	Escritura para grupos y otros
Moderado (740)	027	Escritura para grupo, todos los permisos para los demás.
Moderado (741)	026	Escritura para grupo, lectura y escritura para los demás.
Severo (700)	077	Todos los permisos para grupo y otros.

Tabla 3.3 Configuración de umask para diferentes niveles de seguridad

Ejecución Periódica de Comandos

Unix tiene una utilería para la ejecución periódica de comandos llamada *cron*. Es un demonio que ejecuta la tarea especificada en el archivo *crontab* a la hora y fecha especificada. Los usuarios pueden enviar peticiones de ejecución editando su archivo con el comando *crontab*.

Para la ejecución del comando crontab, se verifica que el nombre del usuario aparezca en el archivo */etc/cron.d/cron.allow* o, si este archivo no existe, el nombre del usuario no debe aparecer en el archivo */etc/cron.d/cron.deny*. Si ninguno de los archivos existe, el usuario no puede ejecutar el comando.

Para mantener un registro de la actividad del cron, se debe especificar CRONLOG=YES en el archivo */etc/default/cron*.

Bibliotecas Compartidas

La gran ventaja de las bibliotecas compartidas es que disminuyen el uso de memoria. Cuando los programas usan la misma función de biblioteca no mantienen cada uno su propia copia en memoria. La mayoría de los comandos en */usr/bin* usan bibliotecas compartidas. Estos ejecutables son descritos como "ligados dinámicamente". Un servicio llamado "El Enlazador de Tiempo de Ejecución" los conectará a las bibliotecas adecuadas al momento de ejecución. El comando "*ldd*" puede usarse para saber cuales bibliotecas compartidas usara un programa cuando sea ejecutado:

```
$ ldd /usr/bin/ls
        libc.so.1 =>      /usr/lib/libc.so.1
        libdl.so.1 =>     /usr/lib/libdl.so.1
```

Un usuario podría crear su propia versión subversiva de bibliotecas compartidas y alterar su ruta de bibliotecas para que apunte a su versión:

```
$ touch /tmp/libc.so.1
$ setenv LD_LIBRARY_PATH /tmp

$ ldd /usr/bin/ls
      libc.so.1 =>      /tmp/libc.so.1
      libdl.so.1 =>     /tmp/libdl.so.1
```

Esto no es tan crucial para el comando *ls* pero si fuera un programa setuid entonces el creador de la nueva librería podría ejecutar cualquier pieza de código como otro usuario más privilegiado.

Es posible ligar dinámicamente un programa para que ignore LD_LIBRARY_PATH usando la variable de ambiente LD_RUN_PATH:

```
$ setenv LD_RUN_PATH /usr/lib
$ unsetenv LD_LIBRARY_PATH
$ cc -o prog prog.c
```

De esta forma, si "prog" tiene su bit setuid encendido, la variable LD_LIBRARY_PATH será ignorada. Todos los programas setuid estándar en */usr/bin* se han escrito pensando en la seguridad en bibliotecas compartidas y han sido compilados de esta manera. Sería una buena idea verificar cuales de las aplicaciones de terceros que corren con setuid también ignoran LD_LIBRARY_PATH utilizando el comando *ldd*.

Asignación de Dispositivos

Un problema tradicional al compartir dispositivos de cinta es el ganar uso exclusivo. No sólo cuando está siendo usado el dispositivo, sino al momento en que se están recuperando archivos de la cinta hasta que ésta se retira de la unidad.

Si la unidad de cinta está físicamente distante del lugar del usuario, puede haber un tiempo considerable entre el momento en que se pone la cinta en la unidad y el regreso a la estación de trabajo. Durante este tiempo un segundo usuario puede leer el dispositivo o, lo que es peor, podría sobre escribirlo.

Solaris tiene un mecanismo para prevenir este daño. Maneja dos comandos de usuario, *allocate* y *deallocate*. Estos son parte del Módulo de Seguridad Básico (BSM). Con esto los usuarios pueden reservar una unidad de cinta para su uso exclusivo antes de que baje un respaldo y hasta que saque la cinta.

La asignación de dispositivos quita los permisos de acceso de lectura/escritura a la cinta. Cuando un usuario hace la petición de uso de cinta, *allocate* checa que nadie más la este utilizando. Si está libre, *allocate* hace al usuario dueño del archivo del dispositivo.

La asignación de dispositivos es necesaria para que un sistema califique para nivel de seguridad C2.

Ejecución de Stack

Hay varias anomalías relacionadas con el stack cuando sus permisos son de lectura, escritura y ejecución. Mientras los stacks con permisos de ejecución son obligatorios para SPARC ABI[10] e Intel ABI, la mayoría de los programas pueden funcionar correctamente sin usar stacks ejecutables.

Está disponible la variable `noexec_user_stack`, en el archivo */etc/system*, para indicar si los mapeos de stack son ejecutables o no. Si la variable no esta en cero, el sistema marcará el stack de cada proceso en el sistema con permisos de lectura y escritura, pero no de ejecución.

Cuando la variable antes mencionada se iguala a uno, los intentos de ejecutar código en el stack se registrarán en el archivo */var/adm/messages*.

Módulo "Conectable" de Autenticación (PAM)

La estructura PAM permite conectar nuevas tecnologías de autenticación sin cambiar los servicios que permiten el acceso al sistema, tales como *login, ftp, telnet, rsh*, etc, pero con modificaciones de estos comandos y demonios para que tomen ventaja de estas características. PAM consiste de librerías, varios módulos y un archivo de configuración (*/etc/pam.conf*).

PAM emplea módulos conectables en tiempo de ejecución. Estos módulos se dividen en cuatro tipos, basados en su función:

- Módulos de Autenticación: Permite la autenticación de los usuarios y acepta el establecimiento, destrucción o actualización de credenciales. Proporcionan una herramienta útil para el manejo de la identificación del usuario.

- Módulos de Contabilidad: Verifica el envejecimiento del password, expiración de cuentas y restricciones de horas de acceso. Después de que el usuario es identificado con los Módulos de Autenticación, los Módulos de Contabilidad determinan si se le deberá dar entrada al usuario.

- Módulos de Manejo de Sesión: Manejan la apertura y cierre de una sesión de autenticación. Pueden registrar la actividad o proporcionar una "salida limpia" después de que la sesión termina.

- Módulos de Manejo de Passwords: Permiten cambiar el password actual.

La siguiente lista describe cada uno de los módulos PAM y su función:

[10] Interfaz de Aplicación Binaria. La interfaz por la cual una aplicación obtiene acceso al sistema operativo y otros servicios. Debería ser posible correr la misma aplicación compilada en cualquier sistema con la ABI adecuada.

Módulos	Ruta	Descripción
pam_unix	/usr/lib/security/pam_unix.so.1	Proporciona soporte para módulos de autenticación, manejo de contabilidad y manejo de passwords. Cualquiera de los cuatro tipos de definiciones pueden utilizarse con este archivo. Usa passwords de Unix para autenticación.
dial_auth	/usr/lib/security/pam_dial_auth.so.1	Sólo se puede usar para autenticación. Utiliza datos para la autenticación almacenados en /etc/dialups y /etc/d_passwd. Es utilizado principalmente por *login* en modems.
rhosts_auth	/usr/lib/security/pam_rhosts_auth.so.1	Puede usarse para autenticación. Utiliza datos almacenados en ~/.rhosts y /etc/hosts.equiv. Se utiliza principalmente para comandos *rsh* y *rlogin*.

Tabla 3.4 Módulos PAM

Por razones de seguridad estos archivos de módulos deben tener como dueño a root y no deben tener permisos de escritura para el grupo y los demás. Si los archivos no tienen estas restricciones, PAM no cargará los módulos.

Hay varios módulos que no son apropiados para cada servicio. Por ejemplo, el tipo de módulo para manejo de password se puede especificar con el comando *passwd*, pero no hay un tipo de módulo `auth` asociado con este comando ya que *passwd* no tiene nada que ver con autenticación.

Servicio	Demonio o Comando	Tipo de Módulo
`dtlogin`	`/usr/dt/bin/dtlogin`	autenticación, contabilidad, sesión
`ftp`	`/usr/sbin/in.ftpd`	autenticación, contabilidad, sesión
`init`	`/usr/sbin/init`	sesión
`login`	`/usr/bin/login`	autenticación, contabilidad, sesión
`passwd`	`/usr/bin/passwd`	password
`rexd`	`/usr/sbin/rpc.rexd`	autenticación
`rlogin`	`/usr/sbin/in.rlogind`	autenticación, contabilidad, sesión
`rsh`	`/usr/sbin/in.rshd`	autenticación, contabilidad, sesión
`sac`	`/usr/lib/saf/sac`	sesión
`su`	`/usr/bin/su`	autenticación, contabilidad, sesión
`telnet`	`/usr/sbin/in.telnetd`	autenticación, contabilidad, sesión
`ttymon`	`/usr/lib/saf/ttymon`	sesión
`uucp`	`/usr/sbin/in.uucpd`	autenticación, contabilidad, sesión

Tabla 3.5 Nombres de servicios válidos para la configuración de PAM

Característica de Apilado (Stacking)

PAM proporciona un método para autenticar usuarios con múltiples servicios usando apilado, es decir, en el archivo de configuración puede aparecer un mismo servicio con dos formas de autenticarlo, la forma en que valida la autenticación (línea tras línea hasta llegar al último si es que alguna anterior no cumplió con la autenticación requerida) hace alusión a una pila en la que, la primera que cumple basta para terminar la autenticación. Dependiendo de la configuración, se le puede solicitar un password al usuario para cada método de autenticación.

Característica de Mapeo de Passwords

El método de apilamiento puede requerir que el usuario recuerde varios passwords. Con la característica de mapeo de passwords, el password "principal" se usa para descifrar los otros passwords para que el usuario no tenga que recordar o teclear varios. La otra opción es igualar los passwords de cada mecanismo de autenticación, pero incrementa el riesgo de inseguridad.

Acceso a la Red

En los sistemas de cómputo moderno, se ha vuelto un estándar el enlazar las computadoras a una red para poder compartir datos entre los sistemas, lo cual ha traído consigo una manera mas sofisticada y poderosa de cómputo; pero abre también una nueva área con respecto a la seguridad de las computadoras. Por ejemplo, dentro de una red de computadoras, los sistemas individuales tienden a estar configurados de una manera más abierta de lo que deberían para permitir el acceso a la información. También, debido a que mucha gente tiene acceso a la red, hay más probabilidad de permitir accesos no deseados, especialmente a través de "errores de usuario" (elección de password fácilmente adivinable, por ejemplo). Debido a los problemas de seguridad que surgen, Solaris ha proporcionado diversos métodos que ayudan al control del acceso a la red.

inetd

La primera versión de Unix para soportar Internet, BSD 4.2, ponía un servidor dedicado para cada servicio de red. Conforme el número de servicios aumentaba, a mediados de los 80's, los sistemas Unix comenzaron a necesitar más y más programas servidores corriendo en background[11], esperando por conexiones de red. Aunque los servidores estaban esperando, todavía consumían recursos del sistema tales como entradas en las tablas de procesos y espacio en swap[12]. Eventualmente, se creo un único programa, llamado *inetd*, que escucha en varios puertos de red en un momento dado y ejecuta el programa servidor apropiado (basado en TCP o UDP) cuando se recibe una petición de conexión. Este programa se ejecuta al momento de que se inicia el sistema y examina el contenido del archivo */etc/inetd.conf* para determinar qué servicios de red va a manejar.

---

[11] En background significa que el proceso se desliga de una terminal. Esto significa que su entrada estándar ya no es el teclado, sino otros procesos o archivos; lo mismo para su salida.

[12] El área de swap, es un área de disco reservada para mover (swap) procesos que corren en la memoria principal (RAM) a memoria más lenta (disco).

Comandos de Acceso Remoto

Los comandos *rlogin*, *rsh* y *rcp* son característicos del Unix de Berkeley, ahora incluidos en SVR4 y, por lo tanto, en Solaris. Comparten el mismo mecanismo de autenticación basado en los archivos *.rhosts* y */etc/hosts.equiv*. Estos programas corren con setuid de root por lo que requieren una atención minuciosa.

Este mecanismo de autenticación permitirá a los usuarios entrar a la máquina sin teclear su password. Esto puede parecer completamente un hueco de seguridad. Sin embargo, si dos máquinas implícitamente confían la una de la otra entonces se considera mejor permitir a los usuarios moverse libremente entre ellas en vez de estar transmitiendo sus passwords por la red. Los hosts confiables se listan en el archivo */etc/hosts.equiv*.

El daño real viene del archivo *~/.rhosts*. Cada usuario puede poner su propia lista de hosts confiables, poniendo sus nombres en este archivo, pero más que eso, puede también listar a usuarios confiables. Estos usuarios pueden utilizar el comando *rlogin* para entrar con su cuenta sin utilizar password. También la sintaxis del "*.rhosts*" incluye el comodín "+" así como la del *hosts.equiv*. Dos signos mas significan que cualquier usuario desde cualquier máquina puede accesar y asumir la identidad del usuario sin conocer el password.

Cuando se incluye un nombre de host en el archivo */etc/hosts.equiv*, cualquier usuario de esa máquina que tenga el mismo nombre de cuenta de un usuario local puede entrar sin proporcionar su password (excepto root).

Puertos Confiables

Los puertos en el rango de 0 a 1023 son denominados algunas veces "puertos confiables". En Unix, estos puertos están restringidos al superusuario (root); un programa debe ejecutarse como root para poder escuchar en una puerto confiable u originar una conexión desde cualquiera de ellos.

El concepto de puertos confiables intenta evitar que un usuario con privilegios normales obtenga información privilegiada. Por ejemplo, si un usuario pudiera escribir un programa que escuchara el puerto 23, éste programa podría disfrazarse como el programa *telnet*, recibiendo conexiones de usuarios y obteniendo sus passwords.

Seguridad X11

X es un sistema de ventanas popular, basado en red, que permite a muchos programas compartir una única pantalla gráfica. Los programas basados en X despliegan su salida en ventanas, las cuales pueden estar en la misma computadora en la que está corriendo el programa o en cualquier otra en la red.

Cada dispositivo gráfico que corre X es controlado por un programa especial, llamado el servidor X window. Otros programas, llamados los clientes X, se conectan al servidor X window por la red y le dicen qué desplegar. Los clientes populares X son *xterm* (un emulador de terminal) y *xclock* (que despliega un reloj analógico o digital en la pantalla).

Una de las características de los sistemas de ventanas basados en X, como ya se dijo, es permitir a los usuarios correr sus programas en un sistema y desplegar la salida en otro. Esto lleva a muchas implicaciones de seguridad. El sistema de ventanas de Solaris, Open Windows, soporta dos mecanismos de control de acceso: basado en usuario y basado en host. El mecanismo más antiguo basado en hosts se usa por compatibilidad con versiones anteriores de X11.

El mecanismo basado en host es débil porque si se da acceso a otro host entonces todos los usuarios en ese host pueden accesar el servidor X. El nuevo mecanismo basado en usuario o basado en autorización permite explícitamente dar o negar acceso a cualquier Identificador de Red (NetID), pero esta característica requiere que el sistema sea parte de un Sistema de Red de Información (NIS o NIS+).

Las estaciones de trabajo multiusuario son un reto para la seguridad X. En las primeras aplicaciones X, los dispositivos lógicos para el teclado, pantalla y sonido eran leíbles y escribibles para todos; esta disponibilidad causó problemas de seguridad, porque eso significa que cualquiera podía leer el contenido de la pantalla del usuario, del teclado o podía escuchar el micrófono en su oficina. Afortunadamente, es esta versión de Solaris existe el archivo */etc/logindevperm*, que especifica una lista de dispositivos que deben cambiar su dueño a la cuenta que en ese momento este firmada en la estación de trabajo.

Llamadas a Procedimientos Remotos (RPC)

Las RPC permiten que un programa corriendo en una computadora pueda ejecutar una función que está corriendo en otra computadora. Trabaja con arquitectura cliente/servidor pero, a diferencia de los servicios normales de Unix en los que el servidor corre en un puerto confiable predefinido, Sun desarrolló un programa llamado *rpcbind*. Cuando inicia un servidor RPC, dinámicamente obtiene una puerto libre UDP o TCP y posteriormente se registra a sí mismo con *rpcbind*. Cuando un cliente desea comunicarse con un servidor en particular, determina el puerto usado por el servidor contactando al *rpcbind* e inicia la comunicación.

Los programas clientes que se comunican con un servidor RPC necesitan una manera de auntenticarse para que el servidor pueda determinar qué información podrá consultar el cliente y qué funciones le son permitidas. Sin autenticación cualquier cliente en la red que pueda enviar paquetes al servidor podría usar cualquier función.

Hay varias formas de autenticación disponibles para RPC, como se describe en la Tabla 3.6.

Sistema	**Técnica de Autenticación**
AUTH_NONE	No proporciona autenticación.
AUTH_SYS	El servidor confía implícitamente que el usuario es quien dice ser, utilizando el UID y GID para autenticación.
AUTH_DH	Usa una combinación de criptografía de llave secreta y llave pública.
AUTH_KERB	Es una modificación al sistema de RPC para que pueda interactuar con el sistema de autenticación Kerberos. Su única desventaja es que requiere que se instale un servidor de Kerberos.

Tabla 3.6 Tipos de Autenticación

Sistema de Archivos de Red (NFS)

NFS consiste de un conjunto de RPC's que permiten a los clientes manipular archivos y directorios remotos como si fueran locales. La primera versión de NFS nunca se liberó. La segunda versión está ampliamente instalada. En 1993 salió la versión 3 que es la que actualmente está operando.

El protocolo de transporte por default para NFS cambió a TCP desde la versión 2.5 de Solaris. TCP proporciona control de tráfico y restablecimiento de errores. Antes de la versión 2.5, el protocolo de transporte era Datagramas de Usuario (UDP).

En el archivo */etc/dfs/dfstab* se especifica qué directorios se van a compartir. También se puede utilizar directamente el comando *share*. Los directorios listados en este archivo se comparten automáticamente siempre que se inicia la operación del servidor NFS.

NFS tiene la capacidad de permitir que ciertas máquinas monten localmente un sistema de archivos y, además, pueden montarlo como de sólo lectura o lectura y escritura. Una mala configuración puede permitir que un sistema de archivos pueda montarlo cualquier máquina, inclusive, con permisos de lectura y escritura.

SECURE NFS

El uso de NFS en forma segura requiere que todas las computadoras involucradas tengan un nombre de dominio. Un dominio es una entidad administrativa, típicamente consiste de varias computadoras, que juntas constituyen una red mediante el uso de NIS o NIS+. Con este ambiente se puede usar autenticación Diffie-Hellman o Kerberos versión 4 o una combinación de las dos.

NFS utiliza por default la autenticación AUTH_SYS, la cual se encuentra definida en el archivo */etc/nfssec.conf*.

Monitoreo de la Actividad en el Sistema

Auditoría

Un aspecto importante en la seguridad de la computadora es la habilidad para monitorear qué esta pasando y qué ha ocurrido. Es deseable que un administrador conozca todos los aspectos del sistema: la carga normal de operación, quien entra, las horas en las que normalmente entran los usuarios, los accesos a disco, la velocidad de respuesta, etc. Tal vigilancia ayuda al administrador a notar rápidamente cuando esta ocurriendo algo inusual y así poder usar herramientas que Sun le proporciona para auditar tanto al sistema como a los usuarios. La auditoría es muy importante cuando se sospecha de un hueco de seguridad ya que proporciona un método de detectar quien ha utilizado de manera inadecuada los recursos.

La auditoría depende de dos características adicionales: identificación y autenticación. Al momento de ingresar al sistema, se le asigna un identificador único (ID) al proceso del usuario. El ID es heredado por cada proceso que se inicia después del ingreso. Aun cuando un usuario cambie su identidad (mediante *su*), todas sus acciones realizadas son registradas con el mismo ID de auditoría. Esta capacidad se logra sólo al instalar el Módulo Básico de Seguridad que se detalla posteriormente.

Syslog

Syslog es una utilería de propósito general para registro de eventos en el sistema, originalmente desarrollada en la Universidad de California en Berkeley para el programa *sendmail*. Desde entonces, sendmail ha sido portado a varios sistemas basados en System V.

Syslog es configurable, el sistema utiliza un proceso centralizado de registros. Programas individuales que necesitan tener información de registro envían la información a *syslog*. Los mensajes pueden registrarse en varios archivos, dispositivos o computadoras, dependiendo de quien envía el mensaje y su severidad.

El directorio */var/adm* contiene varios archivos de mensajes. Los mensajes más recientes están en */var/adm/messages* (y en *messages.0*) y los más viejos están en *messages.3*. Después de un periodo de tiempo (por lo general cada 10 días) se crea un nuevo archivo *messages*. El archivo *messages.0* es renombrado *messages.1*, *messages.1* es renombrado *messages.2* y *messages.2* es renombrado *messages.3*. El archivo */var/adm/messages.3* actual se borra.

Syslog se configura a través del archivo */etc/syslog.conf*. Se pueden capturar mensajes de error adicionales que son generados por los procesos del sistema modificando */etc/syslog.conf*. Por default, */etc/syslog.conf* direcciona muchos mensajes de procesos del sistema al archivo */var/adm/messages*. Los mensajes de inicio de la máquina, apagado, y caídas se almacenan también aquí.

El archivo */etc/syslog.conf* tiene dos columnas separadas por tabs:

facilidad.nivel	*acción*

- *facilidad.nivel*: Una facilidad es sistema origen del mensaje o condición. Puede ser una lista de facilidades separada por comas. Los valores de las facilidades se listan en la tabla 3.7. Un nivel, indica la severidad o prioridad de la condición que se está registrando. Los niveles de prioridad se listan en la tabla 3.8.

- *acción*: El campo de acción, indica a donde se envía el mensaje.

Origen	Descripción
`kern`	Kernel
`auth`	Autenticación
`daemon`	Todos los demonios
`mail`	Sistema de correo
`lp`	Sistema de impresión
`User`	Procesos de usuarios

Tabla 3.7 Facilidades de origen para mensajes de `syslog.conf`

Prioridad	Descripción
`emerg`	Emergencias del sistema
`alert`	Errores que requieren corrección inmediata
`crit`	Errores críticos
`err`	Otros errores
`info`	Mensajes informacionales
`debug`	Salida usada para depurar
`info`	Mensajes informacionales

Tabla 3.8 Niveles de prioridad para los mensajes de `syslog.conf`

Debido a que el directorio */var/adm* almacena grandes archivos que contienen mensajes, respaldos de caídas del sistema (archivos crash), archivos temporales y otros datos, este directorio puede consumir grandes cantidades de espacio en disco.

Módulo Básico de Seguridad (BSM)

El BSM de Solaris proporciona seguridad adicional, cumple con la definida en el nivel C2 del libro naranja. Las características proporcionadas por el BSM son el subsistema de auditoría y el mecanismo de asignación de dispositivos que proporcionan la característica de "reuso de objetos" característico de dispositivos removibles o asignables. El control de acceso discrecional así como las características de identificación y autenticación son proporcionadas por el sistema Solaris estándar.

A partir de Solaris 2.3, BSM se incluye como parte del disco de distribución del sistema operativo. No se necesita instalar separadamente ya que se puede habilitar o deshabilitar mediante dos scripts simples: */etc/security/bsmconv* y */etc/security/bsmunconv*, respectivamente.

El script *bsmconv* agrega la siguiente línea al archivo */etc/system* para deshabilitar la secuencia de interrupción stop-a:

```
set abort_enable = 0
```

Si se desea continuar con la habilidad de interrumpir la secuencia, se debe comentar esta línea.

La auditoría se habilita iniciando el demonio de auditoría (*auditd*). La existencia de un archivo con el nombre */etc/security/audit_startup* causa que el demonio de auditoría se levante automáticamente cuando el sistema entra al modo multiusuario. Este script que automáticamente configura los eventos y pone las políticas de auditoría se configura al momento de la instalación de BSM.

EVENTOS Y CLASES

Las acciones del sistema que son auditables se definen como "eventos". La mayoría de los eventos son atribuibles a un usuario, sin embargo, algunos eventos no lo son porque ocurren al nivel de interrupción del kernel antes de que dicho usuario se identifique y autentique. Los eventos no atribuibles son auditables también.

Cada evento también pertenece a una "clase" para un manejo más fácil de la configuración de la auditoría. El que se audite o no un evento depende de que el administrador preseleccione una clase que incluya el evento en específico.

En la Tabla 3.9 se muestran las categorías generales de eventos. La tabla 3.10 muestra las clases, su nombre largo, su nombre corto y una descripción; de las 32 posibles clases 18 están definidas e incluyen las dos clases globales *all* y *no*.

Rango de Números	Tipo de Evento
1-2047	Eventos generados por el kernel (llamadas al sistema)
2048-32767	Eventos auditables de programas del sistema operativo a nivel usuario
2048-65535	Disponible para aplicaciones de terceros

Tabla 3.9 Categorías de eventos

Nombre Corto	Nombre Largo	Descripción
no	no_class	Permite apagar un evento seleccionado anteriormente
fr	file_read	Lectura de datos, abierto para lectura, etc.
fw	file_write	Escritura de datos, abierto para escritura, etc.
fa	file_attr_acc	Acceso a los atributos de un objeto mediante las funciones *stat*, *pathconf*, etc.
fm	file_attr_mod	Cambio de los atributos de un objeto mediante el comando *chown*, la función *flock*, etc.
fc	file_creation	Creación de un objeto
fd	file_deletion	Borrado de un objeto
cl	file_close	Llamada a la función de sistema *close*
pc	process	Operaciones de procesos: *fork*, *exec*, *exit*, etc.
nt	network	Eventos de red: *bind*, *connect*, *accept*, etc.
ip	ipc	Operaciones de Comunicación de Inter-Proceso (IPC) de System V
na	non_attrib	Eventos no atribuibles
ad	administrative	Acciones administrativas
lo	login_logout	Eventos de acceso y salida de usuarios al sistema
ap	application	Eventos definidos por la aplicación
io	ioctl	Llamada del sistema *ioctl* para control de dispositivos
ex	exec	Ejecución de programas
ot	other	Misceláneo
all	all	Indica que se desean auditar todas las clases anteriores

Tabla 3.10 Clases de Auditoria

Los registros que genera la auditoría describen las ocurrencias de un evento auditado e incluye información de quien hizo la acción, qué archivos fueron afectados, qué acción se realizó así como dónde y cuándo ocurrió.

Dependiendo del prefijo, una clase de eventos se puede auditar sólo si es exitosa, fallida o ambas. El formato de los prefijos se muestra a continuación.

Prefijo	Definición
None	Auditar tanto eventos fallidos como exitosos
-	Auditar sólo los eventos fallidos
+	Auditar sólo los eventos exitosos

Tabla 3.11 Prefijos usados en la auditoría

FUNCIONALIDAD DEL PROCESO DE AUDITORIA

Al momento de que un usuario entra al sistema, se establecen las siguientes características de auditoría:

- Máscara de preselección del proceso. Se refiere a la combinación de eventos a nivel sistema y eventos específicos de usuario (auditoría de usuarios, ver la sección siguiente).
- Identificador de auditoría (ID). Un proceso adquiere su ID de auditoría cuando el usuario ingresa al sistema y este ID es heredado por cada proceso hijo generado por el proceso inicial. Esto permite reconocer el usuario original aun después de haber utilizado el comando *su*.
- Identificador de sesión (sesión ID). También se hereda a cada proceso hijo y se establece al iniciar una sesión.
- Identificación de terminal. Está formado por el nombre de la máquina y su dirección IP, seguido por un número único que identifica el dispositivo físico en el cual está conectado el usuario.

AUDITORIA DE USUARIOS

La auditoría también tiene la capacidad de registrar eventos de uno o varios usuarios en específico. Si se especifica, las banderas de los usuarios se combinan con las aplicadas al sistema en general.

3.2 Necesidades de Seguridad

El servidor de Nómina va a estar protegido por un firewall, por lo que las amenazas de red no se analizarán aquí sino en el capítulo 5. En esta sección nos ocuparemos de las amenazas que generan tanto los usuarios internos como los externos a nivel sistema operativo.

Análisis de Riesgos

Es siempre una buena idea conocer al enemigo. El mantener registros y auditoría sólo ayudará después de que se haya hecho el daño y, si se tiene suerte, puede llevar a culpar al individuo responsable. Es mejor ubicar los signos antes de que el daño ocurra. Muchas veces es un buen método de prevención, el buscar las herramientas que utilizan los intrusos en los sistemas de archivos. Por "herramientas del intruso", se entiende: directorios, por lo general ocultos, que pueden contener muchas copias de adivinadores de passwords; shells setuid; mulas o caballos de Troya; bibliotecas compartidas, etc. Todas estas amenazas de seguridad son comunes en los sistemas que tienen conexión a internet como es nuestro caso.

Caballos de Troya

Un caballo de Troya es un programa que parece hacer tareas útiles. Sin embargo, ejecutará secretamente comandos que comprometen la seguridad. A menudo verifican por el UID=0 y cambian los permisos en los archivos clave o, tal vez, crean un shell setuid. El siguiente, es un ejemplo de un caballo de Troya que parece ser el comando *ls*. El *ls* real ha sido renombrado a *ls.old*.

```
$ cat /usr/bin/ls
   #!/bin/sh
   if [ `whoami` = root ]
   then
      cp /bin/sh /tmp/.prog
      chmod o+s /tmp/.prog
   fi
   ls.old $*
```

De esta forma, cualquiera que ejecute */tmp/.prog* tendrá usuario efectivo de 0 (root).

Mula de troya

Es una variante del caballo de Troya. Consiste de un programa que se deja corriendo en sesiones vacantes. El programa emula al programa login, imprimiendo un mensaje " login: " en la pantalla. El usuario ignorante de la situación, piensa que esta entrando al sistema, pero en realidad

esta dando su password al programa, el cual puede darle entrada al sistema o sacarlo de sesión, haciendo pensar al usuario que cometió un error al teclear su password.

Virus

Las máquinas Unix no están propensas a un ataque de virus ya que los permisos básicos de archivos lo pueden evitar. Los virus son realmente un fenómeno de PC's. Un virus es un fragmento de programa que ataca a un programa legítimo, el cual trata de transferir copias de sí mismo a otros programas. Puede no dañar otra cosa mas que consumir espacio en disco y CPU, pero puede ser malicioso. Por lo general se transfiere en discos de usuarios que comparten juegos.

Gusanos

Un gusano difiere de un virus porque es un programa completo. Un virus se transporta sobre un programa existente y se transmite pasivamente. Un gusano buscará activamente su proliferación.

El nombre "gusano", fue adoptado para describir el famoso "programa" que causó caos en la internet en 1988. El "gusano de internet" (posiblemente la violación de seguridad más grande de todos los tiempos) fue creado por un estudiante en Cornell. No fue construido con el propósito de hacer daño, pero fue tan exitoso en proliferarse que sobre cargo miles de máquinas Sun y VAX que corrían Unix BSD. A diferencia de los virus ordinarios, una vez iniciado corre continuamente intentando transmitir una versión completa de sí mismo a otras máquinas, donde el ciclo continua.

Probablemente hubiese pasado inadvertido si sólo existiese una sola versión del gusano por máquina. Desgraciadamente, el sistema mantenía varios gusanos que afectaron el desempeño del mismo, llamando la atención del administrador. El creador trató de evitar eso. Si un gusano entraba a una máquina y encontraba otro, el primero terminaría.

Técnicamente el gusano consistió de dos programas, el "arrancador" fue de 99 líneas de C llamado "11.c". Fue compilado y ejecutado en el sistema atacado explotando un bug del programa *sendmail*. Entonces contactaba a la máquina de la que había venido y bajaba el gusano principal. Después de arreglárselas para ocultarse él mismo, buscaba la tabla de ruteo de la máquina para ver a que máquinas se conecta y tratar de diseminar el programa "arrancador" a ellos. Si el enfoque anterior falla, el arrancador trata otro método usando el comando *finger*. Esto involucra enviar al demonio de *finger* tanta información que provoca una cadena desbordada (overflow). A menudo, el arrancador podría copiarse a otro host utilizando el comando *rcp*.

Puertas Traseras

Al paso de los años, se han descubierto varias violaciones de seguridad que han tomado ventaja de la funcionalidad deliberada construida en las aplicaciones. A menudo un programador puede dejar una "ruta secreta" en la aplicación que evita el proceso de verificación normal. Estas llamadas "puertas traseras" a menudo se instalan para propósitos de depuración y se dejan accidentalmente en la liberación del producto.

Algunas otras son creadas a propósito y se disfrazan cuidadosamente en el código fuente para que no sean notadas por los programadores.

Negación de Servicio

En un ataque de negación de servicio un usuario toma tantos recursos compartidos que nadie puede utilizarlos. Este tipo de ataques compromete la disponibilidad de los recursos. Los recursos pueden ser procesos, espacio en disco, porcentaje de CPU, papel de impresora o el tiempo de un asediado administrador de sistemas. El resultado es la degradación o pérdida de un servicio.

Hay dos tipos de negación de servicio. El primer tipo intenta dañar o destruir recursos para no poderlos utilizar. Van desde dañar un disco que paralice el sistema hasta borrar comandos críticos como *cc*, y *ls*.

El segundo tipo de ataque sobrecarga algún servicio del sistema o lo agota (ya sea deliberadamente por un atacante o accidentalmente como resultado de un error de usuario), evitando que otros puedan utilizarlo.

Ataques de Desbordamiento de Buffer

El ataque de desbordamiento de memoria (buffer overflow) ganó notoriedad en 1988 como parte del incidente del gusano de Morris. A pesar de que el reparar una vulnerabilidad de desborde es muy simple, estos ataques continúan hoy en día. La base del problema radica en que esta vulnerabilidad es pasada por alto muy frecuentemente y, además, muchos programas privilegiados corren aun como demonios privilegiados (UID de root). Se desarrollan nuevos programas con más cuidado, pero aún se desarrollan con lenguajes poco seguros como C, donde un simple error puede provocar serias vulnerabilidades.

El éxito de estos ataques también es debido a la naturaleza "desigual" de la que nos protegemos de ellos. El ciclo de vida de una ataque de desbordamiento de buffer es simple: Un usuario (malicioso) encuentra una vulnerabilidad en un programa altamente privilegiado y alguien más implementa un parche para ese "ataque en particular en ese programa privilegado". El arreglo a este ataque intenta resolver el problema en la fuente (el programa vulnerable) en lugar de el destino (el stack que está siendo desbordado).

Este tipo de ataques explotan la falta de verificación de límites en el tamaño de la entrada que se quiere almacenar en el arreglo de buffers. Escribiendo datos fuera del límite del arreglo, el atacante puede hacer cambios arbitrarios al estado del programa que este almacenado al lado de la cadena. La estructura más común para corromper es el de pila o stack, llamado "ataque de ruptura de stack" (stack smashing), que brevemente se describe a continuación. Muchos programas de C tienen vulnerabilidades de desborde de memoria debido a que C carece de una verificación de límites en un arreglo y porque la cultura de un programador de C esta orientada al desempeño del programa que evita verificación de errores en la medida de lo posible.

El ataque de ruptura de stack consiste en alcanzar dos metas dependientes:

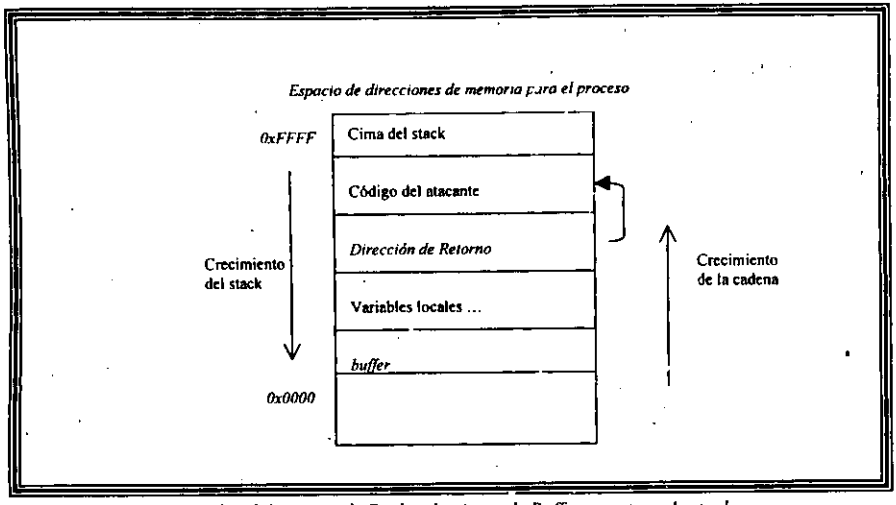

Fig. 5.1 Ataque de Desbordamiento de Buffer y ruptura de stack

- Inyectar código de ataque. El atacante proporciona un cadena de entrada que es realmente ejecutable, código binario nativo a la máquina que se esta atacando. Típicamente este código es simple y hace algo similar a `exec ("sh")` para generar un shell de root.

- Cambiar la dirección de retorno. Hay un espacio de stack para una función actualmente activa arriba del buffer que se esta atacando. El desbordamiento del buffer cambia la dirección de retorno para que apunte al código que el atacante quiere ejecutar. Cuando la función retorna, en vez de brincar de regreso a donde fue llamada, brinca al código del atacante.

Los programas que son atacados usando esta técnica, por lo general, son privilegiados. El código que se inyecta, en la mayoría de la veces, es una secuencia corta de instrucciones que generan un shell bajo el usuario de root. El propósito es dar al atacante un shell con los privilegios de root.

Si la entrada al programa se da desde un proceso local, entonces esta clase de vulnerabilidades pueden permitir a cualquier usuario con una cuenta local convertirse en root. Si la entrada al programa viene de una conexión de red, la vulnerabilidad permitirá a cualquier usuario en cualquier lugar de la red (e inclusive en internet si es que la red tiene conexión a ella) convertirse en root en el host local.

Adivinación de Passwords

Puesto que la única línea de defensa para el acceso a un host es la cuenta del usuario y su correspondiente password, es muy común el tratar de adivinar tales contraseñas. Muchas veces, sin querer, se proporciona información de más acerca de las cuentas de usuarios en el sistema. Tal

información la pueden obtener mediante programas de red como *finger*, o mediante manipulación social. Esta última es la más exitosa, porque la mayoría de las veces los usuarios no están conscientes de las implicaciones que puede tener proporcionar contraseñas por teléfono a personas que se dicen ser administradores o les indican que cambien su password de una forma determinada. A veces, también, la falta de educación en aspectos de seguridad conlleva a la elección de passwords fáciles de adivinar o que nunca se cambian, lo cual aumenta la probabilidad de ser adivinados.

Modificación de los Parámetros de la EEPROM

La posibilidad de este ataque depende de la seguridad física que se tenga en el sitio, ya que requiere que el intruso tenga acceso a la consola conectada directamente al equipo. Este tipo de ataques se refiere a cambiar el dispositivo de arranque de la máquina, borrar su número de serie, modificar dispositivos de entrada, modificar dispositivos de salida, el nombre del comando de arranque, la capacidad de iniciar el sistema operativo en forma automática, entre otros.

Requerimientos de Seguridad

Básicamente la Nómina, a nivel sistema operativo, requiere de servicios de ftp, telnet, NFS, rsh, rlogin, XWindows y correo interno. La necesidad de estos servicios se determinó en base a las necesidades operacionales de la Nómina como son: transferencias de archivos (*ftp*), conexión remota (*telnet*), compartir espacio en disco entre los servidores de desarrollo (NFS), ejecutar respaldos remotos (*rsh*), ejecución del ambiente de ventanas (Xwindows mediante *rlogin*) y envio de correos entre los usuarios de un mismo equipo (correo interno). La información que se guarda en archivos también es confidencial y se comparte con varias dependencias de la Empresa. Está permitido que los operadores de la Nómina compartan información y que puedan escribir en los directorios a los que accedan las dependencias, pero no esta permitido que las dependencias modifiquen tales archivos, sólo los utilizan para transferencias de la información de Nómina a sus respectivas Sucursales.

Se necesita tener disponibilidad de la información con las tolerancias máximas establecidas en el Capítulo 1, un margen de tolerancia de hasta tres días para los módulos de captura en días normales de operación y de un día en cierres de quincena.

Es importante vigilar la integridad de la información para garantizar una consistencia en el comportamiento de los módulos que conforman la Nómina.

La auditoría y monitoreo del servidor es indispensable para conocer la presencia de fallas de servicios, problemas de seguridad y sobre carga del sistema que pudiesen alentar los servicios o, en el peor de los casos, pudiesen atentar contra la disponibilidad de la Nómina.

Se necesitan respaldos históricos de Nómina, ya sea como medio de prevención de desastres o para recuperación de información en caso de errores de usuarios.

Bajo estos requerimientos, se propondrán los lineamientos para la configuración de servicios y prevención de incidentes de seguridad así como actualizaciones del sistema operativo; todo esto para prevenir y evitar la explotación de vulnerabilidades en los servicios que se proporcionan.

3.3 Cuentas de Usuario y Sistema de Archivos

Los sistemas de archivos de Unix controlan la manera en que es almacenada la información de los archivos y directorios en disco. Controla qué usuarios pueden modificar o leer qué objetos y de que forma. El sistema de archivos, por lo tanto, es una de las herramientas básicas para reforzar la seguridad en Unix.

Grupos y Usuarios

Básicamente existen dos grupos en el sistema:

- Usuarios de Nómina
- Usuarios Externos

En el grupo de Nómina se encuentran todos los operadores de la Nómina (incluyendo directivos) y en el grupo de externos, los usuarios de Sucursales de la Empresa que transfieren información de la Nómina.

Cada usuario deberá tener una cuenta asignada, que por ninguna razón puede ser transferible. La rutina de acceso al sistema (login) deberá reconocer contraseñas de hasta ocho caracteres como significativos (el default es seis), lo que significa que los passwords son más largos y por lo tanto más difíciles de adivinar. Esto se logra modificando el archivo */etc/default/passwd* en la línea:

```
PASSLENGTH=8
```

La configuración para los usuarios externos varía de la de los demás usuarios, como se resume en la siguiente tabla:

Variable	Gpo. Nómina	Gpo. Externos	Comentario
umask	007	022	Para Nómina, se aplican restricciones para otros pero no para el grupo, esto porque necesitan compartir archivos entre ellos. Los externos heredan los permisos por default.
shell	/bin/csh	/usr/lib/rsh	Los usuarios de Nómina utilizan csh para el desarrollo de sus actividades. Los externos utilizan shells restringidos, ya que sólo realizan transferencias.
PATH	Los necesarios para realizar su trabajo.	/usr/rbin	Los usuarios externos utilizan sólo el comando *ls* (localizado en */usr/bin*) para listar sus archivos.
Uso de *su*	No	No	El comando *su* sólo lo pueden utilizar los administradores.
Respaldos	Sí	No	Los respaldos los puede realizar sólo el encargado de los respaldo de Nómina, para lo cual, se ha agregado al grupo de *sys*

Tabla 3.12 Variaciones en la configuración de usuarios

Sistema de Archivos de Nómina

El equipo de cómputo de Nómina contiene los siguientes Sistemas de Archivos:

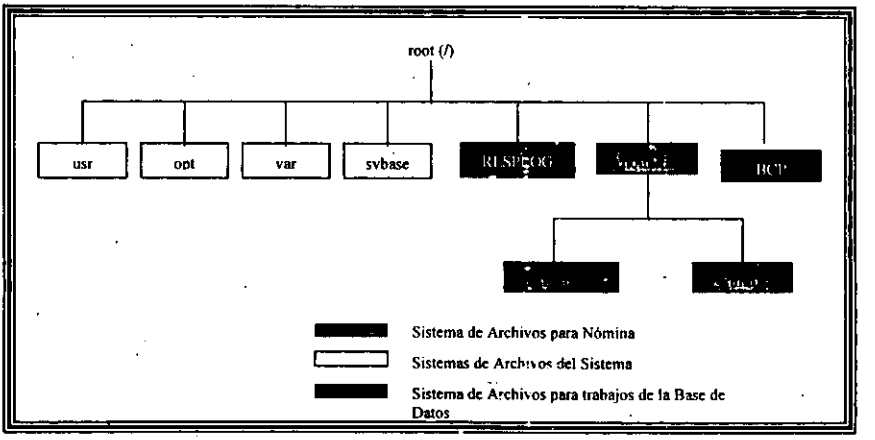

Fig.3.2 Sistemas de Archivos del Servidor de Nómina

Como todo sistema Unix, se necesitan ciertos directorios para el control y operación del sistema y, en el caso de Nómina, se han dedicado tres particiones[13]. El uso de particiones separadas tiene el fin de poder restringir el espacio máximo que un sistema de archivos pueda llegar a ocupar para no perjudicar a otros y dividir los archivos más frecuentemente modificados de los que sufren menos modificaciones (para desarrollar una estrategia de respaldos más óptima).

El directorio */nomina* está dedicado para archivos de poca o nula modificación, contiene archivos binarios y scripts para generación de reportes, así como archivos resultado del procesamiento de la nómina de cada quincena. También contiene los directorios home (*/nomina/home*) de cada usuario y sus archivos temporales o de paso, así como los directorios home de las Sucursales (*/nomina/tmp*) que interactuan con la Nómina; es importante mantener estos dos directorios en dos particiones distintas porque tienden a crecer mucho por ser los directorios de trabajo de los usuarios. En */sybase* se encuentra el software de la base de datos[14]. */RESPLOG* es un sistema de archivos que sirve para mantener los respaldos del log de transacciones de la base de datos (Ver Capítulo 4). Finalmente */BCP* se utiliza para la transferencia de archivos de tablas entre el servidor de producción y los de desarrollo.

Estructura del Directorio /nomina

La estructura de este directorio tiene como propósito distribuir la ejecución de los diferentes procesos de Nómina en directorios privados y públicos dependiendo si la información es local (sólo de utilidad para el proceso o para la persona que lo ejecuta) o es tomada por otras instancias de la Empresa (en el caso de reportes). La siguiente figura muestra la estructura del directorio */nomina*.

[13] En Unix, un disco se divide en ocho "pedazos" y a cada uno se le llama partición. Una partición la conforman un grupo de cilindros apartados para uso de un sistema de archivos.

[14] Sybase utiliza particiones crudas, es decir, sin sistema de archivos, para el almacenamiento de bases de datos. Este directorio sólo contiene el software del DBMS, las bases de datos están en otras particiones.

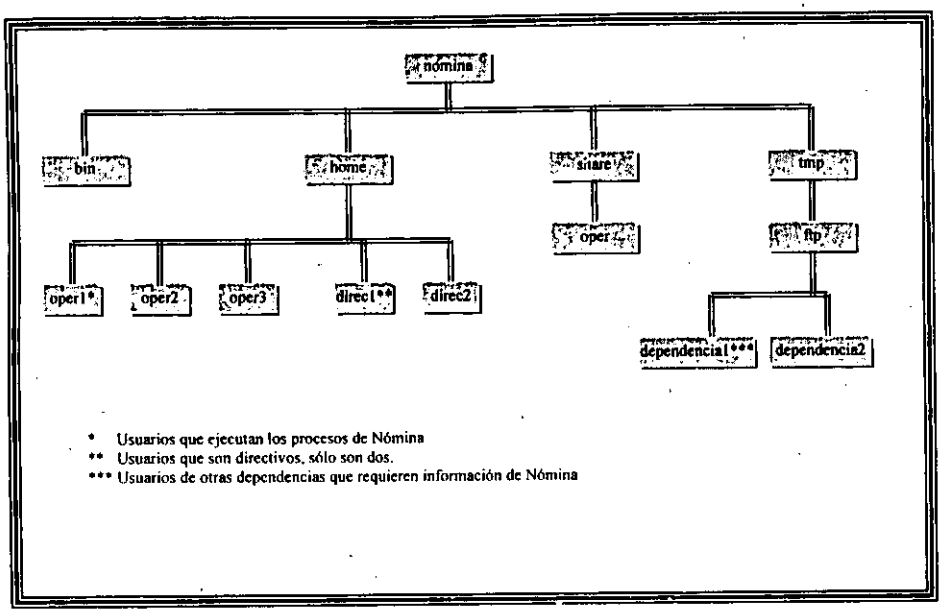

Fig. 3.3 Estructura del Directorio de Nómina

/nomina/bin: Directorio para los ejecutables de nómina y utilerías comunes
/nomina/home: Directorio para trabajo de los usuarios
/nomina/share: Directorio que se mantuvo por compatibilidad con el sistema anterior
/nomina/share/oper: Directorio donde se almacenan los archivos de la Nómina de cada quincena
/nomina/tmp: Directorio que en el sistema actual no tiene ninguna utilidad pero existe por compatibilidad
/nomina/tmp/ftp: Directorio para archivos que se transfieren a otras dependencias

Para establecer los permisos de acceso necesarios para cada directorio de esta estructura, se crearán grupos y a ellos se incorporarán los usuarios del sistema como se explica a continuación.

Permisos en Directorios para Usuarios Directivos

Los directivos pertenecen al grupo de Nómina, este grupo tiene permisos de lectura, por default, en cada directorio. Los directivos tienen un requerimiento importante "necesitan tener acceso a cualquier directorio, sin permitir a todos los miembros del grupo tal privilegio". Los ACL's tienen una característica que permitirá cumplir tal requerimiento: a un directorio se le pueden indicar los permisos que por defecto heredará a los subdirectorios o archivos que sean

creados en él y así sucesivamente. El acceso que necesitan es de lectura y escritura por lo que el ACL en cualquier subdirectorio de */nomina* será:

```
default:user::rwx
default:user:direc1:rwx
default:user:direc2:rwx
default:group::r--
default:mask:rwx
default:other:---
```

Permisos en Directorios para Usuarios Operadores

Los directorios de los operadores de Nómina son accesibles para los directivos; y los miembros del grupo pueden leerlos. Los operadores pueden copiar archivos a los directorios de las dependencias */nomina/tmp*, por lo que necesitan permisos de escritura en tales directorios.

Estos usuarios también están el grupo de Nómina y pueden leer cualquier directorio.

Uno de estos usuarios se encarga de realizar respaldos entre procesos de Nómina, este usuario se deberá integrar al grupo *sys* para que pueda realizarlos.

Permisos en Directorios para Usuarios Externos

Los usuarios externos no tienen permisos mas que de leer su propio directorio (mediante el comando *ls*). Esto se logra configurándoles un shell restringido (en el archivo */etc/passwd*) para que ingresen al sistema (en la sección de "Configuración de Servicios" se explica a detalle como hacerlo). El grupo de Nómina puede escribir en sus directorios.

Espejos de Sistemas de Archivos

La finalidad de crear espejos de sistemas de archivos, es la recuperación inmediata, en caso de una falla de software o de hardware, de los sistemas de archivos críticos para el sistema operativo o para la operación de la Nómina.

Lo ideal sería tener en espejo los diferentes sistemas de archivos para que en caso de una falla en cualquier disco, se pueda seguir operando sin interrupciones; pero esto implica un costo elevado de almacenamiento ya que se tiene que duplicar la información.

Para que el sistema operativo funcione deben existir las particiones */*, */usr*, */var*, *swap* y */opt*. Por otro lado, para que el sistema de Nómina funcione deben existir las particiones */sybase*, */nomina*, */nomina/home* y */nomina/tmp*.

En el caso de las particiones del sistema operativo, es necesario mantenerlas en espejo porque todas son necesarias para el funcionamiento del sistema. Pero el espejo debe ser fuera de línea, debido a que se necesita modificar el archivo que contiene la información de los sistemas de

archivos (/etc/vfstab) para reflejar las nuevas rutas. La forma de implementarlo será utilizando el comando *dd*. Para copiar la información de una partición "/" a su espejo se utiliza:

```
# dd if=/dev/dsk/c0t0d0s0 of=/dev/dsk/c0t1d0s0 bs=100k
# fsck /dev/rdsk/c0t1d0s0
# mount /dev/dsk/c0t1d0s0 /mnt
# cd /mnt/etc
# vi vfstab
```

(Modificar las entradas de los nuevos sistemas de archivos)

```
# umount /mnt
```

Para los demás sistemas de archivos del sistema operativo, se utiliza el mismo procedimiento, excepto la edición de /etc/vfstab.

Para verificar el funcionamiento de la partición de root (/):

```
# init 0
# boot disk1 -s
# sys-unconfig
# boot disk2
```

Si analizamos los sistemas de archivos de Nómina, el más crítico es */nomina/home* porque es el sistema de archivos que más varía por ser el directorio de trabajo de cada usuario, es decir, durante el día se están cambiando un sinnúmero de archivos que, suponiendo que fallara el disco donde se encuentran, se perderían. Por otra parte, */nomina*, */nomina/tmp* y */sybase* son sistemas de archivos que casi no sufren modificaciones, por lo que la información que contienen se puede recuperar del respaldo más reciente. En conclusión, sólo se debe mantener en espejo la información de */nomina/home*. Este espejo sí conviene hacerlo en línea, para que el intercambio de discos, en caso de falla, sea transparente.

Los espejos de los sistemas de archivos deben realizarse en discos distintos de los que actualmente ocupan, como se ilustra en la figura.

Nombre	Ctrl	Tag	Disco	Partición	Tamaño	Sist. Archivos
disk00	c0	t0	d0	s0	256.29 MB	/
				s1	300.59 MB	swap
				s3	500.98 MB	/var
				s4	500.98 MB	swap
				s5	800.51 MB	/opt
				s6	800.51 MB	/usr
				s7	932.34 MB	/var/tmp
disk01	c0	t1	d0	s0	256.29 MB	/
ESPEJO DEL SISTEMA OPERATIVO				s1	300.59 MB	swap
				s3	500.98 MB	/var
				s4	500.98 MB	swap
				s5	800.51 MB	/opt
				s6	800.51 MB	/usr
				s7	932.34 MB	
disk02	c0	t3	d0	s0	99.14 MB	
				s1	100.20 MB	
				s3	500.98 MB	
				s4	1000.90 MB	
				s5	889.10 MB	
				s6	0.0 MB	
ESPEJO DE /nomina/home				s7	1500.82 MB	/nomina/home
disk03	c0	t4	d0	s0	99.14 MB	
				s1	100.20 MB	
				s3	500.98 MB	
				s4	1000.90 MB	/nomina
				s5	889.10 MB	/nomina/tmp
				s6	0.0 MB	
				s7	1500.82 MB	/nomina/home

Fig. 3.4 Espejos del Sistema

3.4 Configuración de Servicios

Solaris es un ambiente de trabajo completo que, al momento de instalación, habilita múltiples servicios como: Protocolo Punto-Punto (PPP), NIS, Administradores de Energía, Correo Electrónico, demonios para Administración de Redes, Servicios para consultas de la Ayuda de Solaris en Web, entre otros, que para nuestro ambiente de Nómina no son útiles. Hay, sin embargo, otros servicios que sí son necesarios, pero que su mala configuración puede resultar en un hueco de seguridad crítico, por ejemplo, exportar un sistema de archivos vía NFS para todo el mundo.

En esta sección se analizará la configuración de los servicios que se necesitan y la deshabilitación de los que no se requieren.

Configuración Requerida

La Nómina es un sistema muy modesto en cuanto a la demanda de servicios. Su ambiente operativo sólo requiere los servicios del shell, Bases de Datos, directorios de trabajo, telnet, ftp, correo electrónico (enviar únicamente), impresión y Xwindows para los operadores de Nómina y ftp para los usuarios externos. La máquina, por otro lado, comparte un sistema de archivos para transferencia de información temporal con otras máquinas de mantenimiento y desarrollo de la misma Nómina mediante NFS.

Para la configuración del servidor se divide el análisis en las siguientes etapas:

Seguridad Física

- Configuración de EEPROM

Servicios

- Configuración de telnet
- Configuración de ftp
- Configuración de correo electrónico
- Configuración de la impresora
- Configuración de Xwindows
- Configuración de NFS
- Eliminación de Servicios no necesarios
- Otras configuraciones importantes

Ambiente de Operadores de Nómina

- Configuración del ambiente del usuario
- Manejo de Cuentas

Ambiente de Usuarios Externos

- Shell restringido
- Manejo de cuentas

Seguridad Física

EEPROM

Es importante el prevenir, aun cuando se tiene seguridad física, el cambio de alguna configuración en el indicador de ok en el sistema (en modo monitor). Para ello se debe poner un password eeprom en modo comando mediante la siguiente instrucción:

```
ok  setenv security-mode=command
passwd = <teclear el password>
ok  printenv
```

Con esto se evita que alguien que no conozca el password modifique alguna configuración del sistema en el modo monitor.

Servicios

Telnet

El telnet no tiene muchas opciones para asegurarlo, por sí solo es un servicio inseguro por la forma en que viaja el password en la red. La confidencialidad de los datos que viajan por la red internet se cubren con el firewall y se estudia en el Capítulo 5.

A nivel sistema operativo lo único que se puede hacer es registrar los intentos de conexión fallidos. Solaris registra los intentos sin éxito de acceso al sistema (después de cinco) en el archivo */var/adm/loginlog*. Este archivo contiene un registro por cada intento y esta compuesto por la cuenta, especificaciones de la terminal y hora. Por default el archivo no existe, así que hay que crearlo con permisos de lectura y escritura para el dueño y el grupo debe ser *sys*.

Es una buena práctica el no dar mayor información acerca del sistema al atacante. Cuando se presenta la pantalla de bienvenida, por default, Solaris muestra la versión de sistema operativo. Este dato lo puede observar cualquiera sin necesidad de que accese al sistema (Fig 3.5).

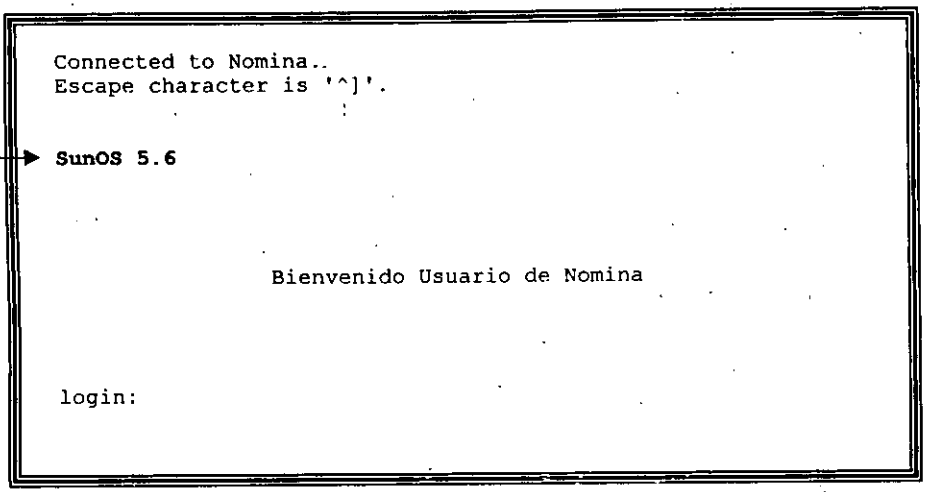

Fig.3.5 Pantalla de Bienvenida de telnet

Para evitar dar a conocer la versión de sistema operativo que se está utilizando se edita el archivo */etc/default/telnetd* y se le agrega la línea:

```
BANNER="          "
```

La pantalla se le presentaría al usuario de la siguiente manera:

```
Connected to Nomina.
Escape character is '^]'.



                Bienvenido Usuario de Nomina


login:
```

Fig.3.6 Pantalla de Bienvenida de telnet configurada

También es necesario asegurarse que la cuenta root sólo pueda conectarse desde la consola. Esto evitará un intento de conectarse directamente desde algún punto en la red. En su lugar, si el administrador requiere usar la cuenta, tendrá que utilizar el comando *su* para tal efecto, lo que permite registrar este hecho en la bitácora correspondiente (Ver "Auditoría" en este mismo Capítulo). Para esto se debe quitar el comentario de la línea `CONSOLE=/dev/console` en el archivo */etc/default/login*.

Ftp

El servicio ftp tiene la misma vulnerabilidad que telnet, viajá en claro el password por la red y la forma de solucionarlo es mediante el firewall.

Al igual que el telnet, el ftp proporciona información al usuario de la versión de sistema operativo al momento de iniciar la sesión (cuando aun no se ingresa al sistema), de la siguiente manera:

```
Connected to Nomina.
220 Nomina FTP server (SunOS 5.6) ready.
Name (Nomina):
```

Fig.3.7 Pantalla de Bienvenida de ftp

En este caso, se crea el archivo */etc/default/ftpd* con la línea:

```
BANNER="BIENVENIDO USUARIO DE NOMINA"
```

La pantalla, entonces, se verá como se muestra en la figura:

```
Connected to Nomina.
220 Nomina FTP server (BIENVENIDO USUARIO DE NOMINA) ready.
Name (Nomina):
```

Fig.3.8 Pantalla de Bienvenida de ftp

También es importante indicar, en el mismo archivo */etc/default/ftpd*, el umask que por default van a tener los usuarios al momento de transferir un archivo ya que siempre se transfieren con umask 022. El umask debe ser de permisos moderados (027).

Como se mencionó en la sección "El Acceso al Sistema" en "Antecedentes", existen cuentas especiales que internamente utiliza el sistema operativo y que ningún usuario debe utilizar para entrar. La cuenta de root también es una cuenta que no debe utilizarse para realizar transferencias de un host a otro por los riesgo que esto implica, ya que, al no tener restricciones de permisos, se pueden sobre escribir archivos no deseados. El archivo que se debe editar es el *ftpusers* que se ubica en */etc* y debe contener la lista de cuentas listadas en la sección antes mencionada, con acceso restringido mediante ftp.

Normalmente, ftp permite realizar una conexión a un usuario que tenga cualquiera de los siguientes shells: Korn shell (ksh), Born shell (sh) o C-shell (csh). Debido a que los usuarios externos tienen un shell restringido, la ruta completa de este shell se debe agregar junto con la de los anteriores en el archivo */etc/shells*, de lo contrario, los usuarios externos no podrán realizar transferencias.

También es muy útil el registrar información más detallada acerca del uso de ftp, esto se logra modificando el archivo */etc/inetd.conf*, agregándole las opciónes "-l –d" al demonio de ftp de la siguiente manera:

```
ftp stream tcp nowait root /usr/sbin/in.ftpd   in.ftpd -l -d
```

Además, es necesario agregar una línea en */etc/syslog.conf* para que registre las actividades del proceso de ftp, de la siguiente manera:

```
daemon.info                    /var/adm/ftpd.log
```

Y por último, crear el archivo */var/adm/ftpd.log* con permisos de lectura únicamente para los demás.

Correo Electrónico

Como se mencionó en párrafos anteriores, los usuarios de Nómina sólo necesitan servicios de correo electrónico para enviar, por lo que no es necesario que esté corriendo el servidor de correo porque éste sólo se necesitaría si se requiriera recibir también. Con el binario de mail o con otra aplicación cliente de Unix es más que suficiente para nuestros propósitos.

Por default, Solaris levanta el demonio sendmail, que es el que se encarga de recibir correo, por lo que es necesario deshabilitarlo de la siguiente manera:

En */etc/rcd.2* renombrar el archivo *S88sendmail* por *NOACTIVOS88sendmail* para que no lo levante el sistema al momento de iniciarse.

Impresión

El servidor de Nómina no tienen una impresora conectada físicamente a alguno de sus puertos, en su lugar, tiene declaradas impresoras remotas las cuales administra el Departamento de Procesamiento de Datos. En cuanto a la seguridad en este servicio desde el host de Nómina no hay nada en especial ya que todos los usuarios pueden mandar a imprimir (excepto los externos ya que tienen un shell restringido y no tienen el comando disponible para mandar a imprimir) y los administradores de la impresora deciden si un trabajo se debe imprimir o no (mediante software propietario).

Xwindows

X mantiene una lista de control de acceso de todos los hosts que son permitidos para hacer peticiones al servidor X. Esta lista de hosts es mantenida mediante el comando *xhost*. Este mecanismo es suficiente (mediante el uso de este comando) para mantener un nivel de seguridad basado en autenticación de la IP del host, ya que sólo lo utilizarán los usuario que estén conectados a la red interna detrás del firewall (Capitulo 5). Ningún usuario externo podrá utilizar Xwindows. Puesto que los únicos que pueden ocupar directamente la consola (y por ende la pantalla gráfica del servidor) son los administradores, deberán tener cuidado de no permitir que todo mundo pueda accesar al servidor X, es decir, no se debe teclear por ninguna circunstancia el comando *xhost* con la opción +.

NFS

El servidor de Nómina comparte con los de desarrollo un Sistema de Archivos de Red que contiene shells para carga de información hacia las bases de datos y archivos con datos de tablas. Puesto que la administración es la misma en los servidores y no se tienen aplicaciones en este sistema de archivos, los únicos requerimientos son:

- Exportar el sistema de archivos a las máquinas adecuadas con permisos de lectura y escritura para tales máquinas.

- En las máquinas cliente montarlo en background para que no interfiera con la operación normal del sistema. Si no se especifica esta opción, se pasma el sistema y todas las sesiones de usuario, esperando que la máquina servidora de NFS exporte el sistema de archivos.

- El dueño del sistema de archivos, tanto en el servidor de producción como en los de desarrollo, debe ser el administrador de la base de datos, ya que está destinado para transferencias de información de la base de datos.

- Asegurarse que tanto en el servidor de producción como en los de desarrollo, la cuenta de administración de la base de datos tenga el mismo identificador de usuario (UID) para que siempre se mapeen correctamente los dueños entre los diferentes servidores.

Estas opciones se configuran en el archivo */etc/dfs/dfstab* en el servidor y también se debe reflejar en el archivo */etc/vfstab* en los clientes como se muestra en el siguiente listado.

```
ultra2 / # cat /etc/dfs/dfstab

        share  -F nfs  -o rw=ultra  /BCP

ultra1 / # cat /etc/vfstab | grep BCP

        ultra2:/BCP  -  /BCP  nfs     -      yes    bg
```

Servicios no Necesarios

Como se dijo en párrafos anteriores, los requerimientos de servicios de la Nómina son muy modestos, pero Solaris por default activa un sinnúmero de ellos y, puesto que no se ocupan, es una buena medida de seguridad el desactivarlos ya que algunos de ellos tienen implicaciones de seguridad serias.

Algunos servicios se encuentran controlados por el super servidor *inetd*, el cual recibe el apelativo de super servidor porque controla la ejecución de otros servicios. Su configuración está basada en el archivo */etc/inetd.conf*. Para desactivar los servicios no requeridos, se agrega un signo "#" al principio de la línea del comando. El archivo sólo deberá permitir los siguientes servicios:

Servicio	Observación
ftp	Se necesita para transferencia de información
telnet	Se ocupa para emular terminales
exec	Se utiliza para respaldos de Unix (ver la sección correspondiente en este capítulo)
rstatd/2-4	Se utiliza para monitorear, en ambiente gráfico, el comportamiento del servidor en cuanto a carga de CPU, accesos a disco y intercambio de contexto. Es de utilidad para los administradores.

Tabla 3.13 Servicios Requeridos para el servidor de Nómina

Además de los procesos controlados por *inetd*, existen algunos otros que se encuentran declarados en los directorios de arranque del sistema operativo, tales como servicios de

administración de energía, servidor de WWW, servidor de ambiente gráfico (dtlogin), correo externo, servidor Kerberos, entre otros.

En el directorio /etc/rc0.d solo existen scripts que dan de baja procesos por lo que no se debe tocar este directorio, lo mismo para rc1.d; en /etc/rcS.d se encuentran los archivos necesarios para que el sistema funcione en modo mono-usuario y sólo se activan los necesarios. Sin embargo en, /etc/rc2.d y /etc/rc3.d se encuentran los scripts que habilitan a múltiples servicios no necesarios. La política que se seguirá es: "*Puesto que el sistema operativo, al momento de inicializarse, sólo ejecuta aquellos scripts que inician con S, se deberán renombrar los scripts de una forma tal que no inicien con alguna de estas letras. No se deberán borrar los scripts porque pueden requerirse en un futuro*". Específicamente se deben deshabilitar los siguientes servicios:

Directorio	Script/archivo	Renombrar/Cambiar a	Observación
/etc/rc2.d	S47asppp	NOACTIVOS47asppp	Script para activar el Protocolo Punto a Punto
	S62skipkey	NOACTIVOS62skipkey	Administrador de llaves para el algoritmo de cifrado SKIP
	S72autoinstall	NOACTIVOS72autoinstall	Script para activar instalación de sistema operativo remota
	S74autofs	NOACTIVOS74autofs	Activa el servicio autofs
	S74xntpd	NOACTIVOS74xntpd	Servicio de Tiempo por Red
	S76nscd	NOACTIVOS76nscd	Cache para Servicio de Nombres (DNS)
	S85power	NOACTIVOS85power	Administrador de Energía
	S88sendmail	NOACTIVOS88sendmail	Correo electrónico
	S91afbinit	NOACTIVOS91afbinit	Configura tarjetas Gráficas
	S91agaconfig	NOACTIVOS91agaconfig	Inicializador de aceleradores gráficos AG-10
	S95networker	NOACTIVOS95networker	Levanta procesos para administración por red
	S96ab2mgr	NOACTIVOS96ab2mgr	Consulta del AnswerBook por red
/etc/rc3.d	S31smc.init	NOACTIVOS31smc.init	Configuración de Sun Media Server
	S76snmpdx	NOACTIVOS76snmpdx	Agente maestro de Solstice Enterprise
	S77dmi	NOACTIVOS77dmi	Proporciona Servicio a Solstice Enterprise
	S99TAS	NOACTIVOS99TAS	Servicios de Web

Tabla 3.14 Servicios inactivos

Otras Configuraciones Importantes

Además de las configuraciones vistas hasta aquí, existen otras que no están relacionadas precisamente con servicios, pero que son importantes para un mayor control de la administración y para una prevención de ataques.

Las prácticas recomendables se refieren a:

CREAR UNA CUENTA DE ADMINISTRACION

No es correcto que los administradores del sistema utilicen la cuenta de root para ingresar al sistema directamente porque, como hay más de uno, es difícil auditar de manera correcta las actividades que uno u otro realiza. Además, como esta cuenta tiene todos los permisos, sólo se debe utilizar para realizar actividades que requieran este privilegio. Esto es con el objetivo de prevenir

accidentes al borrar o modificar, por ejemplo, archivos importantes. En su lugar, cada administrador debe tener una cuenta común y corriente, es decir, sin privilegios especiales, y utilizar el comando *su* cuando requieran realizar alguna tarea de administración. De esta forma hay forma de auditar quien y a que hora utilizó el comando *su*.

DESACTIVAR LA CUENTA DE ADMINISTRACION DE SYBASE

Puesto que los mismos administradores de Unix se encargan de la administración de sybase, esta cuenta debe estar bloqueada (agregando un * al campo de password cifrado de */etc/shadow*), de tal forma que sólo puedan accesar esta cuenta mediante el comando *su*.

ACTUALIZAR LA TABLA /etc/hosts

Este archivo contiene la información de direcciones IP y un nombre o alias qué se le da a un host. Es buena práctica el listar aquí las computadoras personales y los servidores con los que se tiene comunicación. De esta forma, cuando se consulten las conexiones, accesos y bitácoras del sistema, Solaris siempre hace un mapeo de la dirección IP al nombre listado en */etc/hosts*. De esta manera, cuando aparezca una dirección IP en lugar de un nombre de hosts, significa que alguien, fuera de los clientes de Nómina, logró, intentó o esta tratando de conectarse.

EDITAR EL ARCHIVO /etc/init.d/inetinit

Para modificar parámetros de TCP relevantes a la seguridad y a la base de datos. En realidad se tienen que agregar dos parámetros:

```
ndd -set /dev/tcp tcp_keepalive_interval 900000
ndd -set /dev/tcp tcp_ip_abort_interval  10000
```

El primero de ellos se refiere al intervalo de tiempo (en milisegundos) en que el servidor envía un paquete al cliente para saber si sigue activo. Esto es particularmente importante para Sybase porque, algunas veces, los usuarios apagan su máquina de forma inapropiada (con la secuencia de teclas CTRL+ALT+SUPR), lo cual deja sesiones activas en el servidor. Debido a que el valor por defecto es de 7200000 milisegundos (2 hrs.), durante este intervalo el servidor no se da cuenta que ya puede desactivar la sesión y los recursos asociados a ella. Con esta configuración el servidor tarda, a lo más, 15 minutos en dar de baja una sesión inactiva.

El segundo parámetro tiene como objetivo disminuir la posibilidad de un ataque llamado SYN-flooding. Este ataque está catalogado dentro de las Negaciones de Servicios y explota una vulnerabilidad del protocolo TCP/IP. Cuando se inicia una sesión, se realiza un saludo de tres vías: El cliente envía un paquete SYN al servidor, el servidor le retorna un paquete SYN+ACK y espera a que el cliente le envíe un paquete ACK. Cuando el cliente inicia la conversación, el servidor aparta un área de memoria para iniciar la comunicación. Debido a estos dos factores (la espera del retorno del paquete ACK del cliente al servidor y el espacio de memoria reservado para cada inicio de sesión) se puede realizar el ataque que consiste en enviar muchos paquetes (por parte del cliente) SYN, el servidor envía el paquete SYN+ACK pero el cliente no retornar el ACK. El servidor va a esperar un tiempo para desactivar el inicio de sesión y liberar los recursos, este tiempo está controlado por el parámetro `tcp_ip_abort_interval` que por default es de 480000 milisegundos (8 minutos), tiempo suficiente para agotar los recursos de memoria o alentar una máquina. El recomendable es de 10000 milisegundos.

RESTRINGIR EL USO DE COMANDOS

Básicamente hay dos ramas importantes que resolver: los archivos de la base de datos y los archivos del sistema operativo.

En los primeros, el software de Sybase trae permisos en el directorio */sybase/bin* para que todo usuario pueda ejecutar cualquier comando en este directorio (incluyendo el binario para levantar el servidor de bases de datos o el de respaldo), por lo que es necesario quitar permisos de lectura y escritura en todos estos archivos y en todos lo scripts de administración.

En el sistema operativo se tienen que restringir los siguientes comandos:

- **mt**: Este comando se encarga del manejo de la unidad de cinta. Incluye funciones de borrado y rebobinado de cinta. Por ello, se debe restringir su uso quitando permisos de ejecución para los demás. Si alguien necesita realizar respaldos, se debe agregar como miembro del grupo *sys*.

- **ufsdump** y **ufsrestore**: Se utiliza para realizar respaldos de sistema de archivos y para recuperarlos, respectivamente. También se debe restringir su uso ya que puede sobreescribir alguna cinta insertada en la cartuchera o bajar la información que contiene.

- **wall**: Permite enviar mensajes a todos los usuarios del servidor. Sólo los administradores deben tener este privilegio.

- **rwall**: Tiene la misma función que **wall** pero los mensajes se envían a los usuarios de otro servidor. Se deben quitar los permisos para evitar travesuras o ataques de manipulación social.

- **su**: Solo los administradores pueden cambiar de usuario.

ARCHIVO /etc/system

En este archivo se agregan parámetros que rigen el comportamiento del kernel del sistema operativo. Los parámetros importantes, en este caso, son aquellos que nos permiten restringir los recursos que los usuarios pueden tomar del servidor. En la siguiente tabla se exponen los parámetros a declarar y su utilidad.

Parámetro	Utilidad
set maxuprc=100	Máximo número de procesos que un usuario puede tener en el servidor
set maxusers=100	Máximo número de usuarios en el sistema
set noexec_user_stack=1	Para evitar ataques de desbordamiento de pila
set noexec_user_stack_log=1	Para que registre los intentos de ejecución de stack

Tabla 3.15 Parámetros necesarios en /etc/system

Siempre se debe tener una copia de este archivo cuando se modifique, de tal forma que si no levanta el sistema se puede iniciar con el comando *boot –à* el cual pregunta por la ruta donde se encuentra el archivo de respaldo de */etc/system*.

RESTRICCIÓN EN LA EJECUCIÓN DE COMANDOS PERIÓDICOS

La ejecución de programas periódicos mediante *cron* es útil sólo para los administradores, por lo que se debe restringir su uso. Esto se logra agregando las cuentas que lo pueden utilizar en el archivo */etc/cron.d/cron.allow*.

```
# cat /etc/cron.d/cron.allow

root
admon
sybase
```

Al igual que el ftp, las cuentas especiales del sistema deben listarse en el archivo */etc/cron.d/cron.deny*, ya que éstas no deben utilizar esta herramienta.

```
# cat /etc/cron.d/cron.deny

daemon
bin
smtp
nuucp
listen
nobody
noaccess
adm
lp
nobody4
sys
totalnet
uucp
```

AMBIENTE DE USUARIO A NIVEL SERVIDOR

El ambiente de los usuarios de todo el servidor se puede inicializar con el archivo */etc/profile*, en la siguiente tabla se muestran los valores que debe contener:

Opción	Valor	Explicación
umask	027	Permisos para creación de archivos por default -rwxr-x---
stty	erase ^H	Habilita la tecla backspace para borrar
EDITOR	vi	Editor de archivos por default
PATH	/usr/bin	Localización de los programas más utilizados

Tabla 3.16 Ambiente a nivel servidor

Ambiente de root

root necesita una variable PATH restringida para evitar sorpresas, lo más recomendable es que no tenga '.' en el PATH. El '.' indica que el shell debe buscar un programa, incluso, en el directorio donde se encuentre situado root. Esto es muy peligroso ya que se puede pensar estar ejecutando el programa */usr/bin/ls*, pero en realidad se puede estar ejecutando *./ls* que pudiera ser un caballo de Troya. root debe especificar un programa por su ruta absoluta o el shell debe buscarlo en los directorios más comunes.

Ambiente de Operadores de Nómina

Los usuarios de Nómina (que no son externos) utilizan C-shell. La configuración de su ambiente debe incluir parámetros que ayuden a prevenir algún riesgo de seguridad. En la siguiente tabla se listan los parámetros que se deben declarar en cada archivo *.cshrc* en los directorios de los usuarios y en el archivo */etc/skel/local.cshrc* para que automáticamente lo hereden nuevos usuarios:

Opción	Valor	Explicación
PATH	/usr/bin:/nomina/bin:/cobol/bin: /sybase/bin:/usr/openwin/bin:.	Sólo debe contener los directorios estándares para la ejecución de comandos y ambiente gráfico
UMASK	007	Pueden compartir y modificar información entre los operadores pero no con otros
LD_LIBRARY_PATH	/usr/lib:/cobol/coblib	Los directorios donde se encuentran las bibliotecas para ejecución de programas
alias rm	rm –i	Para evitar borrados accidentales de archivos
set directory	/tmp	Para la edición de archivos, algunos demasiado grandes, que no alcanzan en /var/tmp que es el directorio que por default utiliza vi
set history	40	Para mantener la historia de la secuencia de comandos que ha venido tecleando el usuario
ulimit	-c 20480	Indica el tamaño máximo de un archivo core, el valor máximo será de 10 Mb.

Tabla 3.17 Variables de ambiente de operadores de Nómina

Existe un usuario que necesita realizar respaldos, a ese usuario es necesario agregarlo al grupo *sys* para que tenga permisos de respaldar los sistemas de archivos de */nomina*.

Las cuentas de estos usuarios deben tener password y expirar cada sesenta días, el tiempo mínimo para cambio de password debe ser de siete días y el tiempo máximo que puede estar una cuenta inactiva es de sesenta días.

Ambiente de Usuarios Externos

Los usuarios externos sólo requieren la cuenta para realizar ftp, sus passwords no expiran y el shell que ocupan es un shell restringido.

Los comandos que pueden ejecutar se encuentran bajo el directorio */usr/rbin* y sólo incluye el comando *ls*.

root es el dueño de su archivo de configuración de ambiente (*~/.profile*) y sólo debe contener la variable de ambiente PATH=*/usr/rbin*.

Para dar de alta una cuenta, las primeras letras deben indicar la sucursal a la que pertenece, ejemplo: suc1efrain, indica que la cuenta efrain pertenece ala Sucursal1.

3.5 Manejo de la Integridad de la Información

La mayoría de los sistema de cómputo modernos incorporan algún tipo de almacenamiento a largo plazo, por lo general, en forma de archivos (de aplicaciones, de usuario, ejecutables del sistema y bases de datos) almacenados en sistemas de archivos. Por ello es que están expuestos a ataques por las siguientes razones:

- Los intrusos pueden modificar las bases de datos del sistema y los programas que les permitan accesos futuros.

- Los logs del sistema pueden ser borrados para cubrir sus ataques o no permitir su detección en un futuro.

- El comprometer la seguridad del sistema podría llevar a la degradación o negación de servicios.

La integridad de la información sensible es uno de los aspectos de mayor interés para La Empresa. La información sensible son los datos de las bitácoras del sistema, archivos de la historia de los comandos ejecutados por el usuario, las aplicaciones mismas y sus configuraciones, los datos del usuario y las fechas de acceso a los datos.

Prevención

En el sistema operativo existen sistemas de archivos o archivos únicamente que son de sólo lectura (como los comandos del sistema, librerías y software de terceros) y datos que continuamente están cambiando (bitácoras, archivos de configuración y datos de usuarios). Sería ideal un mecanismo para montar un sistema de archivos como de sólo lectura a nivel hardware. Solaris proporciona la opción de montarlos de sólo lectura a nivel sistema operativo, pero esto de nada sirve si se utiliza la interfaz de dispositivo crudo (raw) para modificar el contenido del disco.

Desgraciadamente, el sistema operativo no da ninguna opción para evitar la modificación de archivos que no deberían modificarse. Quizás la única opción sería grabar en CD-ROM la información que no debe cambiar y montarla, sin embargo, el acceso sería muy lento, además, no podría organizarse de una manera adecuada, por ejemplo, nos interesa que no modifiquen los archivos de configuración de */etc*, pero allí también existen archivos de modificación frecuente (como los passwords o pipes de las bitácoras de accesos), lo cual hace difícil el escenario. Lo único que podemos hacer es detectar cuando ocurra una modificación y recurrir a un respaldo confiable para recuperar el archivo modificado.

Detección

En un sistema Unix típico, es un desafío proteger la integridad de los datos del sistema y de los usuarios ya que hay muchas maneras de alterarlos o borrarlos y a menudo algo tan insignificante como el cambio de un bit (como los bits de protección o el UID del usuario) puede conducir a cambios mayores.

El sistema operativo proporciona el programa ASET para la verificación de integridad, sin embargo es poco flexible y no proporciona mayor alcance que la verificación de los archivos del proio sistema. Analizaremos otra herramienta para la verificación de la integridad (Tripwire) y, por supuesto, se dará una propuesta de uso.

Tripwire

Tripwire es una herramienta que auxilia en la administración de Unix a monitorear un grupo designado de archivos y directorios de cualquier cambio. Si se usa para vigilar los cambios en periodos regulares (diario), Tripwire puede notificar archivos corruptos o alterados, para que se tomen las acciones correctivas necesarias.

Una característica importante de este software es que todos los programas necesarios para su funcionamiento los trae integrados, evitando así el posible uso de programas ya comprometidos del sistema operativo; además trae una base de datos de integridad del propio software Tripwire que cuando se ejecuta se "auto-verifica".

En términos simples, Tripwire crea una base de datos con algún identificador único para cada archivo que se quiere monitorear. Mediante la recreación de tal identificador (el cual deberá ser una copia del contenido del archivo completo) y comparándola contra la versión salvada, es posible determinar si un archivo ha sido alterado. Además, mediante la comparación de entradas en la base de datos, es posible determinar si los archivos han sido agregados o borrados del sistema.

Los atributos que sirven para la generación de una firma que identifique de forma única a cada archivo y permita detectar cambios es la siguiente:

Atributo	Significado
p	Permisos y modo del archivo
i	Número de i-nodo
n	Número de ligas
u	Identificador del usuario
g	Identificador del grupo al que pertenece el usuario
s	Tamaño del archivo
a	Fecha y hora de acceso
m	Fecha y hora de modificación
c	Fecha y hora de creación del i-nodo
1	Firma 1
2	Firma 2

Tabla 3.18 Atributos de un archivo que utiliza Tripwire para identificarlo

El archivo de configuración *tw.config* tiene el formato:

```
archivo          atributos_o_plantilla          #comentario
```

Tripwire proporciona plantillas que engloban algunos atributos, tales plantillas son:

Plantilla	Significado	Observación
R	Sólo lectura	Sólo se ignora la fecha y hora de acceso
L	Archivo de log	Ignora cambios a tamaño, fecha y hora de acceso y modificación así como firmas
N	No ignora nada	Incluye todos los atributos
E	Ignora todo	No incluye atributo alguno

Tabla 3.19 Plantillas de Tripwire

Las plantillas o atributos se pueden usar con modificadores, por ejemplo:

Modificador	Significado
+	Indica que incluyan los atributos o plantillas que le siguen
-	Indica que excluyan los atributos o plantillas que le siguen
!subdirectorio	No aplica la plantilla o atributos definidas para un directorio a este subdirectorio
=	Indica que verifique al directorio pero no los archivos que contiene

Tabla 3.20 Modificadores de atributos

El archivo *tw.config*, que es el que guarda la serie de plantillas y modificadores, deberá configurarse de la siguiente manera:

```
#  Primero el directorio "home" de root

=/                      L+ml
/.profile               R-2     # su archivo de configuración

# Unix mismo

/platform               R-2
/kernel                 R-2

# Ahora se verifican algunos archivos y directorios criticos que
# contengan configuraciones del sistema

/dev                    L
/devices                L
=/devices/pseudo        L
/bin                    R-2
/sbin                   R-2

/etc                    R-2
/etc/ftpusers           R-2
/etc/default            R-2
/etc/dfs/dfstab         R-2
/etc/dumpdates          L
```

```
/etc/group              R-2     # los cambios deben ser.pocos
/etc/inet/inetd.conf    R-2
/etc/inet/protocols     R-2
/etc/inet/services      R-2
/etc/init.d             R-2
/etc/motd               L-2
/etc/opt                R-2
/etc/passwd
        .
        .
        .

# Los archivos que tienen el SUID bit prendido ya que son los
# favoritos de los atacantes

/usr/bin/nfsstat        R-2
/usr/bin/passwd         R-2
/usr/bin/ps             R-2
/usr/bin/rcp            R-2
/usr/bin/rsh            R-2
/usr/bin/rdist          R-2
/usr/bin/rlogin         R-2
        .
        .
        .


# Se deben incluir archivos de software de Sybase y cobol

/sybase                 R-2
/sybase/install         L-2
/sybase/errorlog        L-2
=/sybase/init           L-2
/cobol                  R-2

# También los archivos del log de transacciones de la base de
# datos, ya que si se modifican no se pueden aplicar para la
# recuperación de la base de datos

/RESPLOGULTRA           L

# Se debe monitorear la agregación de otro directorio "home"
# de usuarios de nómina internos o externos

=/nomina/home           R-12
=/nomina/tmp/ftp        R-12

# También los directorios de nomina que no deben sufrir
# cambios
# frecuentes

/nomina/bin             R
```

La base de datos de Tripwire, se creo desde el momento de la instalación del sistema operativo en el equipo, de tal forma, que es una base de datos confiable.

En el siguiente listado se muestra una salida de la ejecución de Tripwire.

```
ultra2 /tmp # /usr/local/bin/tripwire/tripwire -interactive
### Warning:    creating ./databases directory!
###
### Phase 1:   Reading configuration file
### Phase 2:   Generating file list
### Phase 3:   Creating file information database
### Phase 4:   Searching for inconsistencies
###
###                    Total files scanned:             23268
###                          Files added:               10
###                          Files deleted:             0
###                          Files changed:             23166
###
###                    After applying rules:
###                          Changes discarded:         23099
###                          Changes remaining:         87
###
added:   -rw-r--r-- root      18 Nov 18 15:28:45 1999 /etc/default/telnetd
---> File: '/etc/default/telnetd'
---> Update entry?  [YN(y)nh?] y
added:   -rw-r----- sybase  7444 Nov 15 10:07:29 1999 /sybase/ULTRA2.201
---> File: '/sybase/ULTRA2.201'
---> Update entry?  [YN(y)nh?] y
changed: drwxr-xr-x root    1024 Nov 22 20:53:23 1999 /
changed: drwxr-xr-x root    3584 Nov 21 03:10:47 1999 /etc
changed: prw------- root       0 Nov 23 10:45:36 1999 /etc/cron.d/FIFO
changed: prw------- root       0 Nov 23 17:19:49 1999 /etc/initpipe
changed: prw------- root       0 Nov 23 17:19:52 1999 /etc/utmppipe
changed: drwxr-xr-x sybase  5120 Nov 22 19:20:33 1999 /sybase
changed: -rw------- sybase  2748 Nov 19 09:36:09 1999 /sybase/.sh_history
changed: -rw------- sybase    15 Nov 19 09:35:14 1999 /sybase/ULTRA.krg
changed: -rwxrwxr-x sybase  1511 Jun  9 10:49:51 1999 /sybase/interfaces
changed: -rw-r--r-- sybase 87961 Nov 23 11:42:49 1999 /sybase/log/ULTRA.log
changed: -rw-r--r-- sybase  7373 Nov 22 21:07:14 1999 /sybase/log/ULTRAB.log
changed: -rw-r----- sybase  7651 Nov 22 19:20:33 1999 /sybase/ULTRA2.cfg
changed: -rw-r----- sybase  7543 Nov 19 09:35:20 1999 /sybase/ULTRA2.bak
### Phase 5:   Generating observed/expected pairs for changed files
###
### Attr        Observed (what it is)         Expected (what it should be)
###=========== ============================= =============================
/
      st_mtime: Mon Nov 22 20:53:23 1999      Thu Nov 11 20:23:41 1999
---> File: '/'
---> Update entry?  [YN(y)nh?] y

/etc
      st_mtime: Sun Nov 21 03:10:47 1999      Sun Nov 14 03:10:46 1999
      st_ctime: Sun Nov 21 03:10:47 1999      Sun Nov 14 03:10:46 1999
---> File: '/etc'
---> Update entry?  [YN(y)nh?] y
```

3.6 Respaldos

Muchas veces no se puede prevenir en que momento sucederá un accidente, se encontrará una falla en algún programa que modifique de manera incorrecta alguna información, ocurra algún desastre natural o habrá un ataque al sistema; pero si se tienen respaldos, se puede comparar el sistema actual con el sistema respaldado y se puede regresar a un estado estable.

Aun cuando se pierda totalmente la computadora, debido al fuego por ejemplo, con un buen grupo de respaldos se puede restaurar la información.

Russell Brand escribió: "Para mí, los datos del usuario son de suprema importancia. Cualquier cosa es reemplazable. Se pueden comprar más discos, más computadoras, más fuentes de poder. Si se pierden los datos, mediante un incidente de seguridad u otra circunstancia, se van".

Los respaldos son uno de los aspectos más críticos en la operación de un sistema. El tener respaldos actualizados, completos y válidos pueden hacer la diferencia entre un incidente menor y una catástrofe.

¿Por qué Hacer Respaldos?

Años atrás, se hacían respaldos diarios debido a que el hardware de la computadora fallaba a menudo sin causas aparentes. Un respaldo era la única protección contra la pérdida de datos.

Hoy en día, las fallas de hardware son aun una buena razón para respaldar los sistemas. En 1990, muchas compañías fabricantes de discos duros dieron dos o tres años de garantía; muchos de estos discos están fallando ahora. Aun hoy con los últimos discos duros, las compañías ofrecen cinco años de garantía, también pueden fallar algún día.

Los respaldos son importantes por otras varias razones:

ERRORES DE USUARIO

Los usuario, generalmente usuarios novatos, borran accidentalmente sus archivos. Realizando respaldos periódicos protege a los usuarios de sus propios errores. Los errores no están limitados a los novatos, más de un experto a sobre escrito accidentalmente una archivo al utilizar un editor o comando de compilador equivocado.

ERRORES DE PERSONAL DE SOPORTE

Algunas veces el personal de soporte comete errores, por ejemplo, un administrador de sistemas borrando cuentas en desuso puede accidentalmente borrar una activa.

FALLAS DE HARDWARE

Las fallas de hardware a menudo destruyen datos en proceso. Si se tiene un respaldo, se puede restaurar en una computadora distinta.

FALLAS DE SOFTWARE

Los programas de aplicación a veces tienen defectos escondidos que destruyen datos bajo circunstancias no bien definidas.

VANDALISMOS E INTRUSIONES

Los crackers de computadoras algunas veces alteran o borran datos. Desgraciadamente, rara vez dejan mensajes indicando si cambiaron alguna información y si lo hicieran, no se puede uno confiar de estos avisos. Si se sufre una intrusión, se pueden comparar los datos de la computadora después de la intrusión con los datos de un respaldo para determinar si algo se cambió.

DESASTRES NATURALES

Algunas veces la lluvia llega a penetrar en los edificios y los "lava". Algunas veces los temblores los dañan. También el fuego es muy efectivo para destruir los lugares donde se guardan los equipos.

OTROS DESASTRES

Algunas veces la naturaleza no tiene nada que ver. Los aviones caen en los edificios, hay fugas de las pipas de gas que causan explosiones, incendios por corto circuito, etc.

Lo que se Debe Respaldar

Hay dos escuelas de pensamiento concerniente a los respaldos de los sistemas:

- Respalda todo lo que es único en tu sistema, incluyendo todos los archivos de usuario, cualquier base de datos del sistema que se pudo haber modificado (tales como */etc/passwd*) y directorios del sistema importantes (como */bin* y */usr/bin*) que son especialmente importantes o que se pudieron haber modificado.

- Respaldar todo porque restaurar un sistema completo es más fácil que restaurar un sistema incompleto y la cinta es barata.

En la Nómina nos inclinamos por la segunda escuela, porque los tiempos de tolerancia son cortos. Se respaldará todo lo que sea necesario para reconstruir el sistema desde cero. En la siguiente tabla se resumen los sistemas de archivos y el porqué se deben respaldar.

Sistema de Archivos	Porque ...
root (/)	Contiene el kernel de Unix, la configuración del sistema (/etc), dispositivos y cobol
/usr	Contiene software de apoyo a la administración, que va variando conforme se instalan nuevas herramientas
/var	Contiene las bitácoras del sistema
/opt	Contiene software propietario de terceros
/nomina	Contiene los archivos para ejecución de nómina y archivos históricos
/nomina/home	Contiene los archivos de trabajo del usuario
/nomina/tmp	Contiene los archivos que se comparten con otras dependencias
/sybase	Contiene el software de la base de datos y su configuración
/RESPLOG	Contiene respaldos de log de transacciones de la base de datos (Ver Capítulo 4)

Tabla 3.21 Sistemas de Archivos que se necesitan respaldar

Tipos de Respaldos

Hay tres tipos básicos de respaldos:

- Respaldo de día cero: Se hace una copia del sistema original. Cuando se instala por primera vez, antes de que la gente comience a usarlo, se deben respaldar todos los programas y archivos en el sistema. Tales respaldos son valiosos después de una intrusión.

- Respaldo completo: Se hace un respaldo en cinta de cada archivo en la computadora. Este método es similar a un respaldo de día cero, con la diferencia de que éste se realiza a intervalos regulares.

- Respaldo incremental: Se hace un respaldo de sólo aquellos datos en el sistema de archivos que han sido modificados después de un evento particular o fecha.

El comando *ufsdump* soporta varios niveles de respaldo incremental (0-9). Cuando se especifica un nivel de respaldo cero se crea un respaldo completo. Del uno al nueve se usan para calendarizar respaldos incrementales, pero no tienen un significado definido. Son sólo un rango de números usados que tienen una relación de respaldos más específicos o más generales. Es decir, si el día lunes se hace un respaldo de nivel 0 (respaldo completo) y el martes uno de nivel 2, éste último sólo respaldará los archivos que se hayan modificado a partir del respaldo de nivel 0. Si el miércoles se realiza un respaldo de nivel 3, sólo se respaldan los archivos modificados a partir del respaldo de nivel 2. Si el jueves se hiciera un respaldo de nivel 1, respaldaría todo lo modificado a partir del respaldo de nivel 0, es decir, lo que se modificó el martes y el miércoles inclusive.

Estrategia de Respaldos

La clave para decidir una buena estrategia de respaldos es entender la importancia y sensibilidad al tiempo de los datos.

Como se mencionó en el Capítulo 1, existe un margen de tolerancia a fallas de hasta tres días para los módulos de captura en días normales de operación y de un día en cierres de quincena. Por lo que, en caso de una pérdida total del equipo, se tiene hasta un día (basándonos en la menor tolerancia) para recuperarlo.

Por otro lado, la Nómina requiere de tres clases de respaldos distintos:

- Respaldos de sistemas de archivos del sistema. Incluyen a todos los sistemas de archivos de Nómina (/nomina, /nomina/home y /nomina/tmp), de la base de datos y del sistema operativo. Para fines prácticos llamaremos a este tipo de respaldos "Normales".

- Respaldos de los archivos de tablas y de logs de la base de datos. Se encuentran en el directorio /RESPLOG y /BCP. A estos les llamaremos respaldos de "Archivos de Base de Datos"

- Respaldos históricos de cierre de quincena de Nómina a los que llamaremos "Quincenales"

Respaldos Normales

Estos sistemas de archivos en total suman un espacio real de 6 Gb y pueden llegar a ocupar hasta 8 Gb. Esto permite realizar respaldos de Unix que alcancen en una sola cinta de 8 Gb en formato de alta densidad.

La actividad diaria de estos archivos se refleja más en los directorios */nomina/home* y */nomina/tmp/ftp/home* de los usuarios de Nómina tanto internos como externos, respectivamente. Los demás directorios sólo se modifican en caso de un cambio en la configuración del servidor (cuentas de usuario, servicios, discos, etc.), en el software de Nómina (en el caso de */nomina/bin*) o en el software de Sybase.

Para este tipo de respaldos se utilizarán tres juegos de cinco cintas cada uno. Una cinta guarda la información de un día y se recicla hasta que se hayan utilizado las cintas de todos los juegos. Los juegos se deben cambiar por cintas nuevas cuando se hayan reciclado máximo diez veces o cuando falle alguna cinta (por defectos de fábrica principalmente). Cabe mencionar que la decisión de tener tres juegos de cintas y reciclarlas es porque la información de respaldos normales se requiere durante una quincena. Al finalizar esta, se hacen respaldos quincenales (que se verán en la sección correspondiente). Todos los respaldos son a nivel 0 (totales).

Para identificarlas, se deben etiquetar de la siguiente manera:

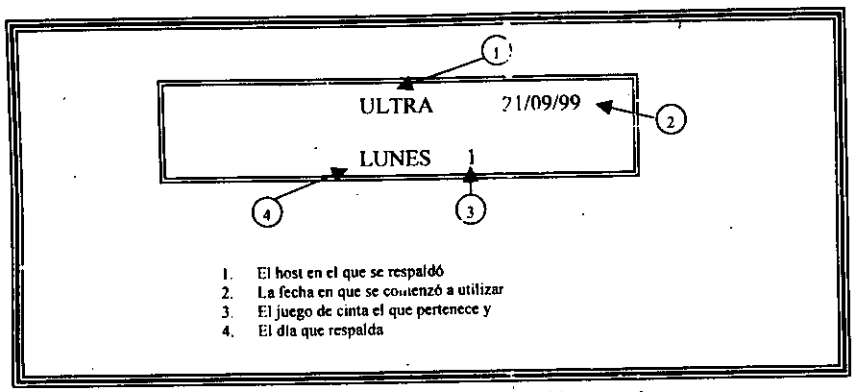

Fig. 3.9 Etiqueta para las cintas con respaldos normales

Respaldos de Archivos de Base de Datos

Estos sistemas de archivos se refieren a información de la base de datos. */BCP* es un archivo temporal de paso que no requiere respaldos. */RESPLOG* requiere respaldos diarios después de que el log de transacciones de la base de datos haya sido respaldado (ver Capítulo 4 en "Respaldos"). Se requiere una cinta aparte para el respaldo de este sistema de archivos y no se debe reciclar, conforme se valla llenando se debe ir guardando y etiquetando de la siguiente forma:

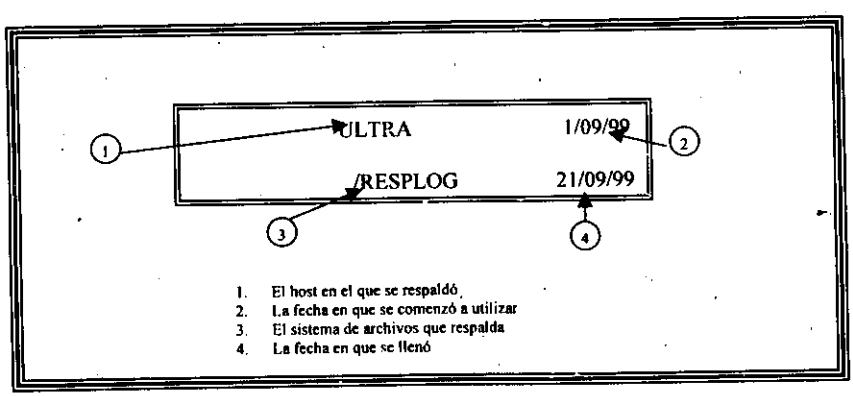

Fig. 3.10 Etiqueta de la cinta de respaldo de /RESPLOG

Se ha determinado no reciclarla porque contiene información del log de la base de datos y que es necesaria mantener en caso de que se necesite recuperar alguna base de datos de quincenas anteriores en alguno de los días que comprende.

Respaldos Quincenales

La historia de los resultados de la Nómina de cada quincena se almacenan en cintas debido a que los archivos ocupan un espacio considerable como para mantenerlos en línea y, además, se necesita la historia de, al menos, cinco años del ambiente completo en el que se corrió la Nómina. De modo que, si cada quincena se generan archivos en promedio de 400 Mb[15] (ya comprimidos con el comando *compress*), en cinco años se necesitarían 48 Gb de espacio en disco (400Mb x 24 Quincenas x 5 años). Por eso, la información de cada quincena se guarda en línea hasta que casi se llena el sistema de archivos */nomina*, momento en el que se borran los históricos más antiguos (porque ya se tienen respaldados) para liberar espacio.

Teniendo en cuenta esto, los respaldos quincenales comprenden todos los sistemas de archivos de Nómina *(/nomina, /nomine/home* y */nomina/tmp)*.

Las cintas destinadas para este propósito no son reciclables, se deben guardar como históricas y etiquetarlas de la siguiente manera:

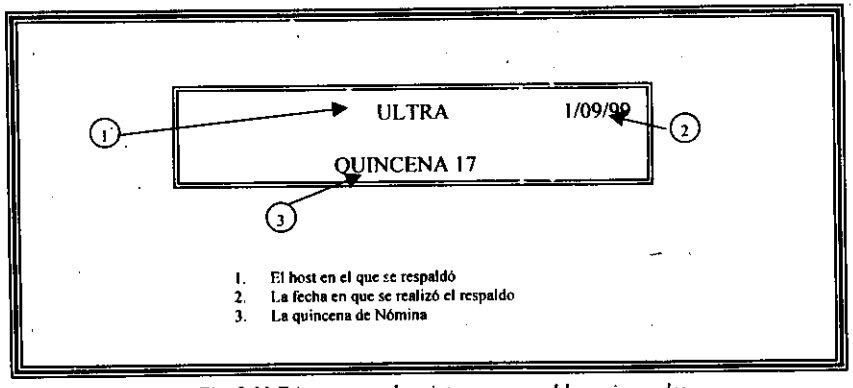

Fig. 3.11 Etiqueta para las cintas con respaldos quincenales

Precauciones

Como se mencionó en las "Características de Seguridad del Sistema Operativo", en los scripts de respaldo se debe utilizar el comando *allocate* y *deallocate* para evitar algún daño accidental o intencional a la información que contienen las cintas.

[15] Incluye los reportes, cheques y depósitos

Las cintas históricas deberán protegerse contra escritura al momento de guardarlas, también se deben proteger cuando se necesite bajar algún respaldo (en cualquier cinta).

Es importante que al terminar un respaldos se verifique que se puede leer correctamente la cinta.

3.7 Auditoría y Bitácoras del Sistema

La auditoría que por default trae configurada el sistema operativo, es suficiente para vigilar el comportamiento del servidor. La configuración por default, mas las líneas que se le fueron agregando en la configuración de servicios, tienen la capacidad para auditar el uso del comando *su*, los intentos de acceso fallidos, reportes de todo tipo de errores del sistema, problemas con el kernel y problemas con los servicios.

Diariamente, al iniciar el día se deben revisar las siguientes bitácoras:

- */var/adm/messages*: Aquí el sistema reporta todos los problemas de hardware, problemas de procesos, cuando se tira o se levanta el servidor y problemas de seguridad. En esta última categoría, registra cuando un usuario hizo más de cinco intentos de entrar al sistema o cuando el usuario quizo cambiar su password y no fue posible.

- */var/adm/sulog*: Reporta los usuarios que usaron el comando *su* para cambiarse a otro usuario y si el intento falló o fue exitoso.

- */var/cron/olog*: Reporta los usuarios que dejaron procesos programados para correrse más tarde, tiempo de duración y comando ejecutado.

- */var/log/syslog*: Indica los correos internos que han sido enviados.

- */var/adm/wtmpx, utmpx, wtmp, utmp*: Registran los usuarios que están conectados y los que se han conectado; duración de la conexión, desde dónde se conectaron y la fecha.

- */var/adm/ftpd.log*: Contiene información más detallada de transferencias de archivos

- */var/adm/lastlog*: Registra la fecha del último acceso exitoso del usuario al sistema

- */var/adm/loginlog*: Registra los intentos fallidos de accesar, en una línea tiene el login, la fecha y la hora de intento.

Las bitácoras */var/adm/messages*, */var/adm/suiog*, */var/cron/olog*, */var/log/syslog*, */var/adm/ftpd.log* y */var/adm/loginlog* se pueden ver desplegándolas con el editor *vi*, por ejemplo. El comando *finger* lee a */var/adm/lastlog*; */var/adm/wtmpx*, *utmpx*, *wtmp* y *utmp* se pueden revisar con el comando *last*.

A algunas bitácoras les da mantenimiento el sistema operativo, tal es el caso de */var/log/syslog* que guarda la historia de hasta ocho semanas, cada semana renombra el archivo con un número consecutivo, de tal forma que en el sistema aparecen los archivos: syslog (el actual), syslog.0 (de una semana atrás), syslog.1 (de dos semanas atrás) y así sucesivamente. Lo mismo pasa para */var/adm/messages* que mantiene históricos de cuatro semanas. Los demás es necesario revisarlos y comprimirlos para que no ocupen mucho espacio y guardarlos el tiempo necesario dependiendo de la información que se haya registrado.

Monitoreo en Tiempo Real del Comportamiento del Servidor

Es importante conocer el desempeño del servidor en todo momento, para diagnosticar una posible anomalía y para familiarizarse con la carga de trabajo típica de los servidores. Para ello debe monitorearse en tiempo real el consumo de CPU, los accesos a disco y el intercambio de contexto de las diferentes aplicaciones. La herramienta que se utiliza es *perfmeter* que es una aplicación gráfica basada en Xwindows. La estación de trabajo que se destine para esto, debe estar conectada a la red interna (protegida por el firewall, ver capítulo 5), para mayor seguridad.

3.8 Otras Tareas Administrativas

Frecuentemente se están encontrando nuevas fallas en el diseño de los programas o en la configuración de servicios, por lo que es importante mantenerse al tanto de tales noticias.

Soporte Técnico

Se tiene contrato de soporte técnico con Sun para la solución de problemas de cualquier tipo. Se tiene también una cuenta de acceso al sitio Web de Sun para bajar parches de sistema operativo que solucionan problemas de mal funcionamiento de programas o cubre huecos de seguridad.

Antes de hacer un cambio a la configuración de algún servicio o instalar algún parche, es necesario contar con un respaldo previo para recuperar el sistema o servicio a su estado anterior a la modificación.

Listas Discusión

Es importante también suscribirse en listas de discusión de problemas en el sistema operativo para estar al tanto de la forma en que se explotan y como evitar su abuso. Una lista recomendable es bugtraq.

Manuales

Es recomendable también leer nuevamente los manuales del sistema operativo, sucede que algunas veces se descubren cosas nuevas o se entienden algunas que en otras ocasiones, por la menor experiencia que se tenía, no se lograron comprender.

Capítulo 4: Seguridad en la Base de Datos

4.1 Características de Seguridad del DBMS[16]

El manejador de bases de datos (DBMS, por sus siglas en inglés Data Base Management System) que se utiliza en la Nómina es Adaptive Server 11.9.2 de Sybase. Este DBMS, funciona bajo una arquitectura cliente-servidor, en la que los clientes p ueden estar en la misma red LAN en la que se encuentra el DBMS o pueden estar en puntos distantes fuera de la LAN. Un mismo DBMS puede ser capaz de soportar múltiples bases de datos en la misma máquina y cada base de datos puede contener cientos o miles de objetos. Los objetos principales que existen en una base de datos son tablas, vistas y procedimientos almacenados y, ligados a las tablas, pueden existir triggers, índices, constraints, reglas, defaults y tipos de datos creados por el usuario. Sólo a las tablas, vistas y procedimientos almacenados se les puede restringir el acceso. El Adaptive Server tiene diversos mecanismos, que permiten proteger a los objetos principales de entradas inapropiadas o no autorizadas y el nivel de seguridad que el DBMS puede ofrecer es C2. Tales mecanismos de seguridad son:

I. Identificación y Autenticación del Usuario

II. División de Roles

III. Control de Acceso Discrecional y

IV. Auditoría

Estas características, se explican más ampliamente en las siguientes secciones.

[16] "Technical Documentation". 1999. Sybase, Inc. Adaptive Server 11.9.2

División de Roles

Los roles soportados por Adaptive Server permiten reforzar y mantener una contabilidad individual. El DBMS proporciona roles del sistema y roles definidos por el usuario.

Roles del Sistema

Los roles del sistema tienen el objetivo de dividir varias tareas relacionadas con la seguridad, administrativas y operacionales. Estos roles vienen predefinidos en el sistema y se pueden asignar a cuentas individuales, lo que permite auditar las acciones de los usuarios. Estos roles son:

- Administrador del Sistema (sa). Realiza tareas administrativas que no están relacionadas con una aplicación en específico. Es quien maneja el almacenamiento en disco, la configuración del servidor, modificación y bloqueo de cuentas de usuario, respaldo y recuperación de la base de datos, diagnóstico de problemas del sistema, concede permisos a los usuarios del servidor, creación de bases de datos y cambio de dueño de las mismas, así como creación de grupos de usuarios cuando es conveniente. El Administrador del Sistema no es necesariamente un individuo; el role puede ser asignado a cualquier número de cuentas individuales.

- Oficial de Seguridad del Sistema (sso), es quien maneja tareas relacionadas con la seguridad en el Adaptive Server, tales como: Creación de cuentas de acceso al servidor, cambio de passwords, manejo del sistema de auditoría y conceder roles de Oficial de Seguridad del Sistema u Operador del Sistema a otros usuarios. Puede entrar a cualquier base de datos pero no tiene permisos especiales sobre los objetos. Una excepción es la base de datos *sybsecurity*, donde sólo un sso puede consultar las tablas de auditoría.

- Operador (oper). Es quien puede respaldar y recuperar bases de datos en todo el servidor.

Además de estos roles, existen dos tipos de dueños: el dueño de la base de datos o dbo y el dueño de objetos.

dbo es el creador de una base de datos o alguien a quien se le da el permiso para crearlas. El sa puede dar el permiso a un usuario para crear bases de datos. Sus tareas son:

- Dar de alta usuarios en la base de datos (pero no en el servidor)

- Dar permisos a los usuarios para crear objetos y ejecutar comandos.

Como se dijo anteriormente, los objetos de una base de datos pueden ser una tabla, índice, vista, default, trigger, regla o constraint. Aquel que crea uno de estos objetos es el dueño del mismo y por ende tiene todos los permisos sobre él. El dueño del objeto debe explícitamente dar permisos sobre el objeto a otros usuarios para que lo puedan accesar. Aún dbo no puede accesar directamente un objeto si no le han dado los permisos necesarios. Sin embargo dbo puede "convertirse" en el dueño del objeto y darse los permisos a sí mismo.

Roles Definidos por el Usuario

sso puede declarar roles para propósitos de seguridad. Estos roles funcionan igual que los grupos de usuarios solo que a nivel servidor. Puede crearse un role para asignarle permisos sobre objetos en varias bases de datos y así evitar la creación de grupos en las bases de datos respectivas.

Identificación y Autenticación del Usuario

A cada usuario del Adaptive Server se le da una cuenta de acceso (login) con un identificador (ID) único. Cuando alguna persona inicia una sesión en el Adaptive Server, debe identificarse y proporcionar un password correcto antes de poder acceder a alguna base de datos. Los logins tienen las siguientes características:

- Nombre de la cuenta, que es única en el servidor.
- Password o contraseña, que puede cambiar el usuario en cualquier momento o algún usuario que tenga asignado el role de sso.
- Base de datos (opcional) en la que se va a ubicar la cuenta una vez que se permite el acceso sin necesidad de ejecutar el comando "*use <base de datos>*". Si no se define, la base de datos que se le asigna es master[17].
- Un lenguaje por defecto (opcional). Especifica el lenguaje en el que se van a desplegar los mensajes al usuario, tales como: español, inglés, alemán, francés, etc.
- El nombre completo del usuario (opcional). Esta característica sirve para propósitos de documentación e identificación de la cuenta.

El password debe ser mínimo de seis caracteres de longitud y se almacena en la tabla *master..syslogins* en una forma cifrada. Además, cuando entran al Adaptive Server desde un cliente, se puede elegir la opción de cifrado de password del lado del cliente para hacerlo ilegible antes de viajar por la red.

Usuarios de Bases de Datos

Es importante aclarar que el login es la cuenta que permite que una persona entre al servidor, pero no basta un login únicamente para poder interactuar con los objetos de una base de datos; es necesario dar de alta un "usuario de base de datos" para que este login pueda manipularlos, de lo contrario no tendrá derecho a intentar "situarse" en ella. El login es único y general. El usuario

[17] Master es una base de datos que contiene tablas para el control de las cuentas de usuario, bases de datos, dispositivos de bases de datos, configuraciones del sistema y los procesos que se están ejecutando, principalmente.

puede llamarse de distinta manera en varias bases de datos pero su ID no cambia porque es el del login. Toda la actividad que el usuario desarrolle se registrará con su ID, lo que permite la realización de auditorías.

El proceso para agregar nuevas cuentas al servidor es:

- Un login con el role de sso debe crear una cuenta en el servidor.
- Un login con el role de sa o el dueño de la base de datos agrega un usuario a la base de datos y, opcionalmente, puede agregar al usuario a un grupo.
- Un login con el role de sa, el dueño de la base de datos o el dueño de objetos da permisos al usuario o al grupo para ejecutar comandos específicos o para utilizar objetos.

Creación de Grupos

Los grupos proporcionan una manera conveniente de dar o quitar permisos a más de un usuario a la vez. Un grupo permite asignar un nombre colectivo a más de un usuario. Cada usuario es miembro del grupo "public" y pueden ser miembros también de un grupo más. "public" es el grupo que existe por default en cualquier base de datos y un usuario puede ser eliminado de cualquier otro grupo, menos del "public".

El Usuario guest

El crear un usuario llamado "guest" en una base de datos permite a cualquier login entrar a ella como tal usuario. Si teclea el comando "*use <base de datos>*" y su nombre no se encuentra en la lista de usuarios dados de alta en esa base de datos, Adaptive Server busca un usuario guest. Si hay alguno, se permite el acceso al login con los permisos que se le hayan otorgado al usuario guest, que por default son los permisos del grupo public.

En la base de datos del sistema (master), existe un usuario guest, por eso es que un usuario que no tiene permisos de entrar a ninguna base de datos, por default es asignado a master con permisos restringidos.

Alias de Usuarios

El mecanismo de alias permite tratar dos o más usuarios como si fueran el mismo, de tal manera que todos tienen los mismos privilegios. Este mecanismo se usa a menudo para que más de un usuario pueda asumir el role de dbo. También permite establecer una identidad de usuario colectiva, dentro de la cual se puede auditar la identidad de cada cuenta individual.

Usuarios Remotos

Es posible que un usuario de otro Adaptive Server pueda ejecutar procedimientos almacenados en el servidor local, siempre y cuando se habilite el acceso remoto. Trabajando con el administrador del otro servidor, se puede permitir, también, que un usuario pueda realizar llamadas a procedimientos remotos (RPC por sus siglas en inglés Remote Procedure Call) del otro servidor al local.

Bloqueo de Cuentas

Al bloquear o borrar una cuenta de Adaptive Server el usuario no podrá entrar al servidor. Sin embargo, bloquear un login es más seguro que borrarlo, ya que el ID asignado a la cuenta no volverá a ser asignado nuevamente.

Control de Acceso Discrecional

Son controles que se usan a la discreción de los dueños de objetos. Son "a discreción" porque el dueño de algún objeto puede decidir permitir el acceso a un objeto o negarlo.

Los usuarios con role de sa, operan fuera de estos controles ya que tienen permisos de acceso a todos los objetos de cualquier base de datos en cualquier momento. La excepción es que sólo los usuarios con el role de sso pueden leer las tablas para la auditoría.

Los comandos de SQL *grant* y *revoke* controlan el sistema de acceso a discreción del Adaptive Server. Se pueden dar varios tipos de permisos a usuarios, grupos y roles con el *grant* y se pueden negar con *revoke*. Con estos comandos se pueden dar permisos a usuarios para manipular objetos en la base de datos (consultar o modificar tablas y vistas o para ejecutar un procedimiento almacenado) y/o para la ejecución de comandos (crear bases de datos, tablas, vistas y/o procedimientos almacenados).

Algunos comandos pueden usarse en cualquier momento por cualquier usuario sin requerir de algún permiso especial, otros sólo pueden usarse por usuarios con cierto status (por ejemplo, sólo por el administrador del sistema con el role de sa) y no son transferibles (tabla 4.1).

La habilidad para asignar permisos para ejecución de comandos o manejo de objetos está determinado por el role que tiene cada usuario. Es posible dar un permiso a un usuario y, a su vez, él puede transferirlo a otros.

Uso de vistas y Procedimientos Almacenados como Mecanismo de Seguridad

Las vistas y los procedimientos almacenados pueden servir como mecanismo de seguridad ya que se puede dar acceso controlado a objetos de bases de datos sin dar el permiso sobre la tabla que contiene la información, es decir, si a un usuario se le da permiso de ejecución sobre un procedimiento almacenado, no es necesario darle permiso a las tablas que éste consulta o modifica. A través de una vista, los usuarios pueden consultar y modificar los datos que pueden ver, el resto de la base de datos no es visible ni accesible.

Cadenas de Dueños

Una vista puede depender de otras vistas y/o tablas. Los procedimientos almacenados pueden depender de otros procedimientos, vistas y/o tablas. Estas dependencias pueden verse como una "cadena de dueños".

Típicamente, el dueño de una vista también posee los objetos subalternos (otras vistas o tablas) y el dueño de un procedimiento almacenado posee otros procedimientos, tablas y vistas referenciados por tal procedimiento.

También, una vista y los objetos en los que se basa, están por lo general en la misma base de datos. Si estos objetos están en una base de datos diferente, un usuario que quiera usar la vista o procedimiento debe ser un usuario válido o debe existir una cuenta guest en esa base de datos.

Cuando un usuario al que se le ha concedido el permiso de ejecución en un procedimiento o vista lo utiliza, Adaptive Server no verifica los permisos de los objetos a los que accesa si:

- Estos objetos y la vista o el procedimiento tienen el mismo dueño y
- El usuario que quiere accesarlos es un usuario válido o guest en cada una de las bases de datos en donde se encuentren estos objetos.

Sin embargo, si no todos los objetos son del mismo dueño, Adaptive Server valida los permisos cuando la cadena de dueños se rompe. En la Fig. 4.1 se ilustra el comportamiento de Adaptive Server.

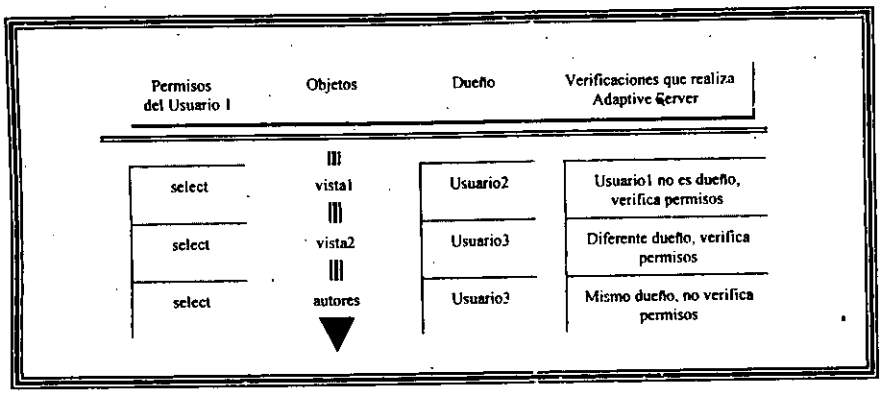

Fig. 4.1 Verificación de permisos para cadenas de dueños

Enunciado	Asignado por defecto A					Se puede dar permiso		
	sa	oper	Dueño de Base de Datos	Dueño de Objeto	Public	SI	No	N/A
alter database	■					(1)		
alter table				■			■	
begin tran					■			■
checkpoint			■				■	
commit tran					■			■
create database	■					■		
create default			■				■	
create index				■				
create procedure			■			■		
create rule			■					
create table			■		(2)	(2)		
create trigger				■			■	
create view			■			■		
dbcc							■	
delete				(3)		■		
disk init	■						■	
disk mirror	■						■	
disk refit	■						■	
disk reinit	■						■	
disk remirror	■						■	
disk unmirror	■						■	
drop				■			■	
dump database		■	■				■	
dump tran		■	■				■	
execute				(4)		■		
grant (en obj)						■		
grant (en cmds)			■			■		
insert				■		■		
kill	■						■	
load database		■	■				■	
load tran		■	■				■	
print					■			■
raiserror					■			■
readtext				■		(5)		
references				■				
revoke (en obj)				■			■	
revoke (en cmds)			■				■	
rollback tran					■			■
save tran					■			■
select				(3)		■		
set					■			■
setuser			■				■	
shutdown	■						■	
truncate table				■			■	
update				(3)		■		
update statistics				■			■	
writetext				■		(6)		

(1) Transferido por ser el dueño de la base de datos
(2) Public puede crear tablas temporales, no requiere permisos
(3) Si es una vista, los permisos son para el dueño por default
(4) Por default para el dueño del procedimiento almacenado
(5) Se transfiere con el permiso de select
(6) Transferido con el permiso de update
No Significa que el uso del comando esta siempre restringido
N/A Significa que el uso del comando nunca esta restringido

Tabla 4.1 Permisos de comandos y objetos

Auditoría

Un elemento principal de un sistema seguro es la contabilidad. Una manera de asegurar la contabilidad es auditando los eventos del sistema. Muchos eventos que ocurren en el Adaptive Server se pueden registrar en una base de datos de auditoría. Cada uno de estos registros contiene la naturaleza del evento, la fecha y hora, el usuario responsable de ello y el éxito o falla del evento. Entre los eventos que pueden ser auditados se encuentran las entradas y salidas al sistema, la iniciación del servidor, conexiones utilizando Llamadas a Procedimientos Remotos (RPC) desde otros servidores, errores del sistema, la ejecución de comandos que requieren roles especiales, el uso de comandos para accesar datos, intentos de accesar a objetos particulares y las acciones de un usuario en particular. El **rastro de la auditoría**, o registro de los eventos de auditoría, permite al Oficial de Seguridad del Sistema, quien maneja la auditoría, reconstruir los eventos que han ocurrido y evaluar su impacto.

Estado de la Seguridad al Momento de Instalar el Servidor

Cuando se instala el Adaptive Server, se configura por default una cuenta llamada **"sa"**, la cual no tiene password y tiene asociados los roles de sa y sso. Esto significa que la cuenta "sa" tiene poder ilimitado por lo que, inmediatamente después de terminada la instalación, se debe cambiar el password.

Es recomendable utilizar la cuenta de "sa" sólo durante la instalación inicial del servidor; después de esto se debe bloquear y dar de alta nuevas cuentas individuales para el/los administradores. No se debe borrar esta cuenta de default ya que se requiere para actualizaciones (upgrades) posteriores del DBMS.

4.2 Necesidades de Seguridad

Análisis de Riesgo

El principal objetivo de seguridad en la base de datos es "restringir a los usuarios a manipular y accesar sólo aquella información que sea necesaria para la realización de su trabajo, mediante el manejo de permisos sobre los objetos de la misma; registrar también los eventos que pudiesen comprometer la seguridad de la base de datos o la disponibilidad de la misma; y proporcionar la mayor disponibilidad posible de la información mediante la verificación de la consistencia de los datos, respaldos y espejos de dispositivos importantes", que en otras palabras es la aplicación del "principio de menor privilegio", auditoría, disponibilidad y consistencia. Lógicamente, lo que se quiere proteger es la información que se almacena en las tablas de la base de datos y su disponibilidad.

En general, las tablas que contienen información de sueldos del empleado, datos personales, prestaciones, categorías, pagos y movimientos (promociones, bajas, licencias, etc.) son confidenciales y por lo tanto, se deben auditar las modificaciones y accesos a las mismas.

Las amenazas a las que se encuentran expuestas las tablas de la base de datos son:

- La consulta de datos confidenciales sin autorización
- La modificación de la información por usuarios no autorizados
- Bugs en el software del DBMS
- La indisponibilidad de ésta debido a problemas de hardware (discos principalmente) y
- La pérdida o modificación de la información por apagones repentinos del equipo

El último punto es un asunto de seguridad física que se minimiza al tener las instalaciones del centro de cómputo bien protegidas, la existencia de un UPS y los mantenimientos preventivos.

Para lograr disminuir las otras amenazas, se diseñó el siguiente procedimiento de seguridad, que se desarrolla en puntos posteriores:

Restricciones de Acceso

Se deben definir grupos de usuarios de acuerdo al modulo de Nómina que manejan, la dependencia a la que pertenecen o el sistema con el que opera. Esto es posible debido a que se comparte información con otras dependencias y cada dependencia o sistema tiene necesidades de

acceso similares. El manejar grupos permita mayor facilidad de administración que si se asignan permisos individuales.

Por parte del sistema de Nómina, se analizarán los requerimientos de información necesarios para cada módulo y de acuerdo a este estudio se creará la tabla de permisos asignados a cada grupo sobre tablas, vistas y procedimientos almacenados.

Además, se deben especificar los DBMS con los que tendrá comunicación el DBMS de Nómina, así como el cifrado de passwords de los clientes que se conecten al mismo.

Disponibilidad

Se analizará la configuración necesaria para proporcionar la mayor disponibilidad posible. Una buena disponibilidad se logra distribuyendo el log de transacciones y los datos de una base de datos en más de un disco. También la utilización de discos espejo permite una alta disponibilidad, por lo tanto se analizarán los dispositivos que requieran espejo. Otra forma de asegurar la disponibilidad es mediante la realización de respaldos, lo cual permite recuperar información en el caso de una falla de hardware.

La distribución de la base de datos en los discos, el diseño de la base de datos, la información sobre los dispositivos, la configuración de las opciones del DBMS y los usuarios dados de alta, deben tenerse en documentos fuera de línea que, en caso de un siniestro o recuperación de la Base de Datos, sea posible re-configurarla como se encontraba antes del incidente.

Consistencia

Es necesario revisar la consistencia de la información durante la operación del sistema y antes de respaldarla para garantizar que se están consultando los datos correctos, para detectar fallas menores y poderlas reparar. Se establecerá el mejor calendario para realizar estas actividades y sobre que bases de datos y objetos, de tal manera que no impacte de manera significativa en el rendimiento del servidor.

También se deberá contar con acceso a la información más reciente de bugs en el software para actualizar el sistema en caso de posibles fallas que afecten el rendimiento o buen funcionamiento del mismo.

Auditoría

Se deberán registrar las acciones que los usuarios realicen y que puedan comprometer la seguridad del sistema. Lo ideal sería registrar todo pero esto conlleva a una pérdida en el rendimiento del servidor, por lo que sólo se registrará lo necesario.

4.3 Usuarios Grupos y Permisos

En el sistema existen dos tipos de usuarios: los que tienen uno o más roles asignados debido a que realizan tareas de administración y los usuarios de Nómina.

Cada usuario debe tener una cuenta en el DBMS sin importar si tiene roles o no. La utilidad de los grupos se debe a que se pueden asignar los mismos privilegios a varios usuarios, lo cual reduce las labores administrativas. En el caso de los encargados de la administración, no se necesitan crear grupos de administradores ya que son dos personas únicamente, por lo que se les asignan los roles necesarios a sus cuentas directamente. Los usuarios de Nómina sí necesitan grupos ya que hay una gran cantidad de ellos. Los grupos definidos en el sistema de Nómina son:

- CAPVAL: En este grupo se encuentran todos los usuarios que utilizan el módulo de Captura y Validación de Movimientos de Personal. Estos usuarios utilizan vistas y procedimientos almacenados,
- PROCMOV: Para los usuarios del módulo de Proceso de Movimientos del Personal,
- PERCEP: Para los usuarios del módulo de Captura y Validación de Percepciones y Deducciones,
- CALCULO: Engloba a los usuarios del módulo de Cálculo,
- REPORTES: Para los usuarios del módulo de Reportes y
- CATALOGOS. Para los usuarios que actualizan la información de los catálogos de la Nómina.

En algunas ocasiones, los directivos de otras sucursales desean consultar información directamente de la Base de Datos. Para esos casos se han creado grupos para cada una de las Sucursales que requiera acceso y se asignan a estos grupos los permisos sobre las tablas que consultan.

Sybase permite restringir las siguientes operaciones:

Permiso	Objeto
select	Tabla, vista, columna
update	Tabla, vista, columna
insert	Tabla, vista
delete	Tabla, vista
references	Tabla, columna
execute	Procedimientos Almacenados

Tabla 4.2 Permisos y objetos en los cuales aplica

Por lo que en las siguientes tablas se resumen los permisos para cada grupo:

TABLA	CAPVAL	PROCMOV	PERIODO	CALCULO	REPORTES	CATALOGOS
ADMR	UDIS	U	U			
CAREAR			S		S	UDIS
CREAS	UDIS		S			UDIS
CATRBCPTO			S	S		UDIS
CBCOS	S	S	S	S		UDIS
CCAM	UDIS					UDIS
CCAPT	UDIS					UDIS
CCATEG	S	S		S	S	UDIS
CCAUS	UDIS				S	UDIS
CCP	S					UDIS
CCPTO		S	S	S	S	UDIS
CCPTOINCO			S			UDIS
CCNDIF		UDIS				UDIS
CDEP	S	S	S		S	UDIS
CEDOCIV	UDIS					UDIS
CEFED	UDIS					UDIS
CERRORCAP			UDIS			UDIS
CEST	UDIS					UDIS
CFUNC	S		S		S	UDIS
CLOT	UDIS					UDIS
CLP	UDIS	S	S	S		UDIS
CMCON		UDIS				UDIS
CMFUN		UDIS				UDIS
CNACS	UDIS					UDIS
CNODS	UDIS					UDIS
CPART	S				S	UDIS
CPER	S	US		SU		UDIS
CPLA	S	UDIS	UIS		S	UDIS
CPRGM	S		S			UDIS
CPZA	UDIS				S	UDIS
CREC	UDIS					UDIS
CREV	UDIS					UDIS
CSITCPTO			S			UDIS
CSUBDEP	S	S	S		S	UDIS
CSUBPROG	S	S	S		S	UDIS
CTAB		S		S		UDIS
CTMPRE	UDIS					UDIS
CTMOVS	UDIS	S				UDIS
CTCPTO			S			UDIS
CTEMPS	UDIS		S	S	S	UDIS
CTTAB		S		S		UDIS
HCPTOPAG				I	S	
HPAG				I	S	
TACUM				IDS		
TASDO				IDS	S	
TCCPTOERR			UIS			
TCMOV	UDIS					
TDEMP	UDIS					
TEESP		S		IDS		
TEMP	S	S	S	S		
TEMPN	UDIS	US	S	SU	S	
TEMOD	UDIS					
THEMP			IDS	S		
TLOT	UDIS					
TMFD	UDIS	UDIS				
TMFH	UDIS	UDIS				
TMOV		UDIS	UDIS	S	S	
TNOM	S	UDIS	S	S	S	
TOFI	IS					
TPSAL				IDS		
TPCAT	UDIS	UDIS				
TPCATHR	UDIS					
TPALIM			UDIS			
TTAB		S	S	S		UDIS
TTABU	S	S			S	UDIS
TTFOL	UDIS					
TTRAN		S		IS		

Nota: U=Update, D=Delete, I=Insert y S=Select

Tabla 4.3 Permisos tabla/grupo/tipo de acceso

SP	CAPVAL	PROCMOV		CALCULO	REPORTES	CATALOGOS
sp_actnomb	E					
sp_acttempl	E					
sp_actinscr	E					
sp_arcxare	E					
sp_arml	E					
sp_bajext	E					
sp_bajxare	E					
sp_bajxopc	E					
sp_blote	E					
sp_cal	E					
sp_chistmfu	E					
sp_chkent	E					
sp_crit	E					
sp_drech	E					
sp_dicthd	E					
sp_flote	E					
sp_qral	E					
sp_qralcau	E					
sp_qralxare	E					
sp_lic	E					
sp_mant	E					
sp_mfdic	E					
sp_mestim	E					
sp_mpass	E		E			
sp_mnom	E		E			
sp_ocdigv			E			
sp_ocatipoe			E			
sp_ovcodprog			E			
sp_ovguard			E			
sp_ovpasig			E			
sp_ovreg			E			
sp_ovtab			E			
sp_pestim	E					
sp_pdicta	E					
sp_presup			E			
sp_pxd			E			
sp_rcentra			E			
sp_rdescpto			E			
sp_rtit			E			
sp_rtot			E			
sp_rcapest			E			
sp_repla			E			
sp_replugpag			E			
sp_reprexcap			E			
sp_reprecxde			E			
sp_reprecxrev			E			
sp_resest	E					
sp_resestrev	E					
sp_restot	E					
sp_resumest	E					
sp_selpasest	E					
sp_semp	E					
sp_serfol	E					
sp_tabgral	E					
sp_tdomemp	E					
sp_tdomnom	E					
sp_temp	E					
sp_tempnom	E					
sp_tierec	E					
sp_tipusu	E		E			
sp_tmfund	E					
sp_tmfunh	E					
sp_totlpag	E					
sp_trec	E					
sp_valfol	E					
Nota:			*E=Execute*			

Tabla 4.4 Permisos procedimiento almacenado/grupo/tipo de acceso

VISTAS	CATVAL	PROCMOV	PERCEP	CALCULO	REPORTES	CATALOGOS
v_A		S				
v_B		S				
v_CLUGPAG		S	S	S	S	
v_P		S				
v_TBASE		S				
v_TSDO		S				
v_acceso_prog	S		S			
v_asigemp	S					
v_cper	S					
v_critica	S					
v_emp	S					
v_empnom	S					
v_folrecc	S					
v_hpza	S					
v_htfu	S					
v_lot_fol	S					
v_lots	S					
v_nolot_fol	S					
v_pzae	S					
v_plaemp	S					
v_pla	S					
v_plafu	S					
v_pzasvac	S					
v_proc	S					
v_rec	S					
v_tgrpousu	S		S			
v_tipsemp	S					
v_trecfu	S					

Nota *S=Select*

Tabla 4.5 Permisos vista/grupo/tipo de acceso

Para implementar de manera más eficiente los permisos, se han codificado los siguientes shells que forman una sentencia SQL a partir de una tabla de tablas/permisos, pasándole como parámetro el nombre del grupo.

```
#!/usr/bin/ksh
#################################################################
#
# Shell para generar comandos SQL para los Grupos de Nomina
# Recibe como parámetro el archivo donde se encuentran los permisos y el
# nombre del grupo.
#
#################################################################

if [ $# -ne 2 ]
then
   echo "
                Uso : permisos.sh archivo grupo
"
   exit
fi

if [ -f $1 ]
then
   :
else
   echo "El archivo especificado no existe, por favor verifiquelo"
   exit
fi

if [ -f $1.sql ]
then
   rm $1.sql
else
   :
fi

echo "use DESATEMP
go" > $1.sql
exec < $1
while read TABLA OP1 OP2 OP3 OP4
do
   op1=""
   op2=""
   op3=""
   op4=""
```

```
   case $OP1 in
      U)op1="update"
        ;;
      D)op1="delete"
        ;;
      I)op1="insert"
        ;;
      S)op1="select"
        ;;
      E)op1="execute"
        ;;
      *)op1=""
        ;;
   esac
   case $OP2 in
      U)op2=",update"
        ;;
      D)op2=",delete"
        ;;
      I)op2=",insert"
        ;;
      S)op2=",select"
        ;;
      *)op2=""
        ;;
   esac
   case $OP3 in
U)op3=",update"
        ;;
      D)op3=",delete"
        ;;
      I)op3=",insert"
        ;;
      S)op3=",select"
        ;;
      *)op3=""
        ;;
   esac
   case $OP4 in
      U)op4=",update"
        ;;
      D)op4=",delete"
        ;;
      I)op4=",insert"
        ;;
      S)op4=",select"
        ;;
      *)op4=""
        ;;
   esac
echo "grant $op1 $op2 $op3 $op4 on $TABLA to $2" >> $1.sql
   echo "go" >> $1.sql
done
```

Cifrado de Passwords

Cuando los clientes de la base de datos se conectan al servidor, el password viaja en claro a través de la red en la configuración por default de un cliente. Los clientes se conectan a través de open clients, que son interfases para comunicarse con el DBMS. Estas interfases pueden ser ct-libraries o db-libraries; las primeras son mas recientes que las segundas e incorporan nuevas características de manejo de SQL pero las dos tienen la capacidad para encriptar passwords.

Los clientes de la nómina se conectan mediante Power Builder (que utiliza ct-libraries) o programas en C (con db-libraries). En el caso de Power Builder, para encriptar el password en la línea **DbParam** de una conexión, se utiliza la sentencia:

```
PWEncrypt="Yes"
```

En el caso de los clientes de C, cuando se establecen los parámetros de la estructura *dblogin*, se utiliza la sentencia:

```
dbsetversion(DBVERSION_100);
DBSETLENCRYPT(LOGINREC, TRUE);
```

Con estas sentencias, en Power Builder y C, se logra un grado más de confidencialidad en la autenticación de los usuarios.

4.4 Disponibilidad y Consistencia de la Base de Datos

El mejor tiempo de preparación para un desastre es antes de que éste pase. Existen varios puntos críticos en la creación y mantenimiento de una base de datos que facilitan su recuperación en caso de problemas. Tales puntos son: su distribución en discos, creación de espejos en los dispositivos importantes[18], verificación de la integridad de los datos y respaldos. Estos puntos se desarrollan en las siguientes secciones para garantizar la disponibilidad mínima de la base de datos dentro de los intervalos de tolerancia establecidos en el Capítulo 1.

Distribución de la Base de Datos

Adaptive Server utiliza transacciones para rastrear los cambios a una base de datos. Las transacciones son unidades de trabajo del DBMS. Una transacción consiste de uno o más comandos SQL que se realizan o fallan como unidad.

Cada SQL que modifica datos es considerado una transacción. Los usuarios pueden definir también transacciones encerrándolas entre un bloque *begin transaction...end transaction*.

Cada base de datos tiene su propio log de transacciones, que automáticamente registra cada transacción ejecutada por un usuario. Este log de transacciones es de "escritura adelantada". Cuando un usuario ejecuta una transacción que modifica datos, Adaptive Server escribe los cambios al log, después de que todos los cambios se han registrado en el log, se escriben a una copia de la página de datos en cache. La página de datos se escribe a disco posteriormente. Si un enunciado de la transacción falla, Adaptive Server deshace todos los cambios hechos. El servidor escribe un registro "end transaction" al log al final de cada transacción, registrando su estado (exitosa o fallida).

Cuando se respalda la información de la base de datos, no se trunca el log de transacciones, éste se trunca sólo cuando se respalda explícitamente. Es necesario respaldar únicamente el log a intervalos regulares para garantizar que siempre habrá espacio suficiente para realizar las transacciones cotidianas. Un respaldo de log sólo se puede llevar a cabo si éste se encuentra en un dispositivo diferente al de datos, por lo que es necesario separar los datos y el log en discos distintos para garantizar su buen funcionamiento y facilitar la recuperación de la base de datos en caso de una falla, como se verá en la sección de respaldos. Cabe mencionar que cuando un log de transacciones se llena, se detiene la operación de la base de datos hasta que se libere espacio, por

[18] Para Adaptive Server, el término dispositivo no necesariamente se refiere a un dispositivo físico completo, se puede referir a una pieza del mismo como puede ser una partición o un archivo que se usa para almacenar base de datos y sus objetos.

ello es importante vigilar su crecimiento ya que, de lo contrario, se atenta contra la disponibilidad de la Nómina.

Teniendo en cuenta lo anterior, la distribución de la base de datos en los dispositivos es la siguiente.

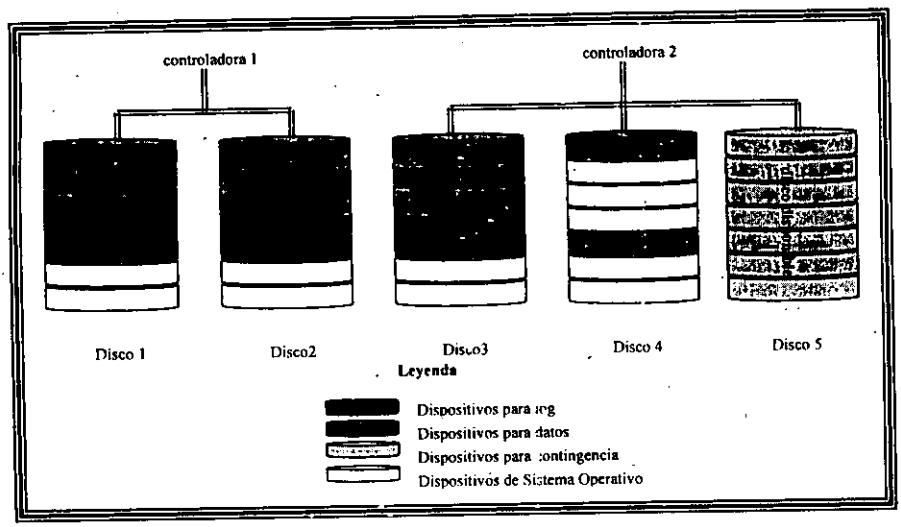

Fig. 4.2 Distribución de la Base de Datos

Esta configuración es suficiente para cubrir las expectativas de disponibilidad, que es de un día máximo en cierre de quincena y de tres días en operación normal, ya que en caso de falla de alguno de los discos, existe uno de respaldo que puede sustituir a cualquiera de los otros. En este disco se tiene creada una base de datos de emergencia que serviría para trabajar temporalmente mientras se restablecen los otros discos.

Esta distribución de la base de datos está más orientada a un buen rendimiento del equipo ya que utiliza dos controladoras, cada una con dos discos, para la base de datos. Esto es porque la disponibilidad requerida es tolerante en tiempos. También con esta distribución se tiene por separado el log y los datos, lo cual permitirá respaldar el log individualmente y garantizar la disponibilidad de espacio en log para su correcto funcionamiento.

Espejos

Los espejos de discos permiten una recuperación de datos sin necesidad de interrumpir el DBMS. Permite la duplicación de un dispositivo de base de datos, es decir, todas las escrituras a un disco se copian a un disco físico separado. Si uno de los dispositivos fallara el otro tiene una copia actualizada de todas las transacciones.

Cuando falla una lectura o escritura a un dispositivo que tiene espejo, Adaptive Server deshabilita el dispositivo dañado y envía mensajes de error. El servidor puede continuar trabajando sin el dispositivo dañado.

Cuando se decide que dispositivo duplicar, se deben sopesar factores como el costo de tener abajo el sistema, posible reducción del desempeño (performance) de la base de datos y el costo de los dispositivos de almacenamiento. Revisando estos puntos ayudará a decidir qué información se debe duplicar: los logs de transacciones, todos los dispositivos en el servidor o sólo ciertos dispositivos.

Los siguientes puntos muestran consideraciones de costos y performance:

- Velocidad de recuperación: Se puede alcanzar una recuperación rápida cuando se les asigna un espejo a la base de datos del sistema que contiene toda la información del servidor (master) y las bases de datos de usuarios (incluyendo los logs de transacciones).

- Espacio de almacenamiento: Una recuperación inmediata requiera alta redundancia (espejos para todas las bases de datos y sus logs), lo cual consume espacio de disco.

- Impacto en el performance: Al asignar un espejo a las bases de datos de usuario se incrementa el tiempo necesario para escribir las transacciones en ambos discos.

Adaptive Server tiene dos tipos de bases de datos: las del sistema (que le sirven para controlar la información que maneja) y las de usuario (que contienen la información de la empresa). El recuperar una base de datos de usuario es mucho más sencillo (si se tienen los respaldos adecuados) que recuperar una base de datos del sistema. La complejidad se debe a que una base de datos del sistema (en específico la base de datos master) contiene toda la información global del DBMS como son: los dispositivos, las bases de datos, las cuentas de usuarios, configuraciones del servidor. roles de cada usuario, lenguajes instalados, orden de datos utilizado, etc., que es información, que al perderla, implica reconstuir todo. Además, cuando se daña esta base de datos, el DBMS no se puede ejecutar, lo cual significa iniciar desde cero la creación de esta base de datos. Todo el procedimiento para recuperarla y poner en línea nuevamente las demás bases de datos es lo que la hace compleja y crítica. Por ello, se creará un espejo para la base de datos master, puesto que las otras bases de datos del sistema se pueden recuperar sin ningún problema a partir de scripts que el mismo software ya incluye

Para la base de datos que contiene la información de Nómina no asignará ningún espejo.

La siguiente figura muestra la distribución de dispositivos del DBMS de Nómina en su totalidad.

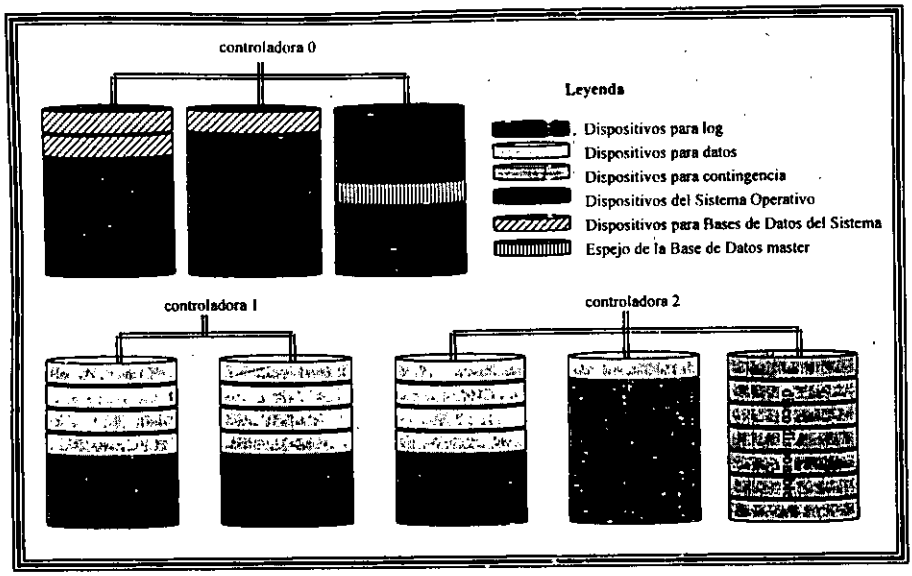

Figura 4.3 Distribución del DBMS de Nómina en los discos

Verificación de Consistencia de Datos y Respaldos

Comandos dbcc

El Verificador de la Consistencia de la Base de Datos (dbcc) proporciona comandos para verificar su consistencia lógica y física. Las funciones de dbcc son:

- Verificación de los enlaces (links) y los punteros de datos tanto a nivel página como a nivel renglón.
- Verificación del alojamiento físico de las páginas.

Los comandos dbcc se deben utilizar como parte del mantenimiento regular de la base de datos, la integridad de sus estructuras internas depende de que tan a menudo se corran estos comandos. Estas corridas pueden detectar errores y corregirlos antes de que se afecte la disponibilidad de la base de datos. También se deben utilizar para determinar la extensión del daño después de que ocurra un error del sistema; **antes de respaldar la base de datos para una mayor confianza de la integridad del respaldo**; y, finalmente, cuando se sospeche que una base de datos esta dañada.

En la siguiente tabla se muestran los comandos para verificar consistencia y sus funciones.

Verificación que realiza	checkstorage	checktable	checkdb	checkalloc	indexalloc	tablealloc	Checkcatalog
Alojamiento de columnas tipo texto	X						
Consistencia de índices		X	X				
Orden alfabético de índices		X	X				
Entradas de Páginas OAM	X	X	X		X	X	
Asignación de páginas	X			X	X	X	
Consistencia de Páginas	X	X	X				
Consistencia de punteros	X	X	X				
Tablas del sistema							X
Cadenas de columnas tipo texto	X	X	X	X			
Columnas tipo texto	X	X	X				

Tabla 4.6 Comandos dbcc y sus funciones

Es importante aclarar que los comandos *checktable* y *checkdb* realizan las mismas funciones, la diferencia es que *checkdb* es recomendable correrlo en el momento en que haya menos actividad en el sistema porque es a nivel base de datos y el *checktable* se ejecuta cuando se presenta un error en momentos en que tiene actividad el DBMS ya que solo se le puede especificar una tabla (y sus correspondientes índices) a la vez. El comando *indexalloc* aun es más específico ya que sólo verifica índices. El comando *checkstorage* requiere una base de datos especial, lo que implica más recursos y esta enfocado para bases de datos en donde hay actividad las veinticuatro horas, por lo que en nuestro caso no nos es de mucha utilidad.

Para la Nómina, se utilizará el siguiente esquema de verificación:

Se deberán ejecutar los comandos dbcc checkdb, dbcc checkalloc y dbcc checkcatalog todas las noches antes de respaldar la base de datos.

Respaldos

Adaptive Server tiene un procedimiento automático de recuperación que protege a los datos de fallas eléctricas y fallas del equipo. Para protegernos de las fallas de discos, se deben hacer respaldos frecuentes de las bases de datos.

En caso de falla de los medios de almacenamiento, se puede recuperar una base de datos si, y solo si, se han estado haciendo respaldos regulares de las bases de datos y sus logs de transacciones. **Nunca se deben usar comandos de respaldo del sistema operativo para respaldar dispositivos de las base de datos**. Al cargar la copia en el Adaptive Server causa corrupción masiva.

Los respaldos de la base de datos se realizan exitosamente aun cuando ésta se encuentre corrupta, por lo que, antes de respaldar la base de datos, se debe verificar su consistencia (con los comandos dbcc), de lo contrario, un respaldo puede ser inservible al recuperarlo.

El comando *dump database* hace una copia de la base de datos completa, incluyendo el log, pero sin truncarlo.

El comando *dump transaction* hace copias del log, proporcionando un registro de cualquier cambio en la base de datos desde el último *dump database* o *dump transaction* ejecutado. Una vez que respalda el log, trunca la parte inactiva. Estos respaldos toman menos tiempo y espacio que un respaldo de base de datos. Como se mencionó anteriormente, sólo se puede respaldar el log si se encuentra en un dispositivo separado.

Cuando el dispositivo de datos falla y la base de datos es inaccesible se utiliza el comando *dump transaction* con la opción *with no_truncate* para obtener una copia actual del log. Esta opción sólo se puede usar si el log esta en un dispositivo separado y la base de datos master es accesible.

Las cintas son preferibles como dispositivos de respaldo, ya que permiten mantener gran cantidad de respaldos de bases de datos y logs de transacciones fuera de línea, sin embargo, los respaldos a disco son mucho más rápidos. Para la Nómina, se utilizarán cintas para los respaldos de bases de datos y discos para los de log de transacciones de acuerdo al esquema que se mostrará más adelante.

Recuperación de Respaldos

Se utiliza el comando *load tran* para recuperar el respaldo hecho con *dump transaction*. La única restricción es que la base de datos en donde se quiere recuperar el respaldo, debe tener al menos el mismo tamaño de la original.

Una vez que se ha restaurado la base de datos, se utiliza el comando *load transaction* para recuperar cada respaldo del log en **el orden en que fueron hechos**. Este proceso reconstruye la base de datos re-ejecutando los cambios registrados en el log de transacciones. Cuando se ha recuperado la secuencia completa de respaldos de log, la base de datos refleja todas las transacciones al momento en que se realizó el último respaldo.

Calendario para Respaldos de Bases de Datos

Es importante tener un buen calendario de respaldos ya que esto determina cuanto trabajo se tiene que realizar cuando se necesite recuperar una base de datos.

Las cintas, con todo y sus ventajas de mantener la información fuera de línea y ser un medio de almacenamiento barato, algunas veces se dañan por defectos de fábrica o por problemas con la unidad de cinta, por lo que también es importante considerar estos factores para minimizar el efecto de pérdida de una cinta, sobre todo porque se almacenan en ellas una gran cantidad de información (aproximadamente 8 GB en formato de alta densidad). Por ejemplo, si en una cinta alcanzan los respaldos de una semana de la base de datos y se llega a dañar, entonces perderemos información de una semana, lo cual puede ser muy perjudicial.

Para el respaldo de la base de datos de la Nómina se utilizarán dos juegos de cintas (uno por semana) y cada juego consta de cinco cintas (una diaria). Cada cinta tendrá un respaldo de un día, es decir, hay una cinta que respalda sólo los días martes, otra los días miércoles, etc. Para el segundo juego de cintas se aplica el mismo criterio (una cinta por día) de tal forma que se alternará una cinta de cada juego. Esto se ilustra en la siguiente figura.

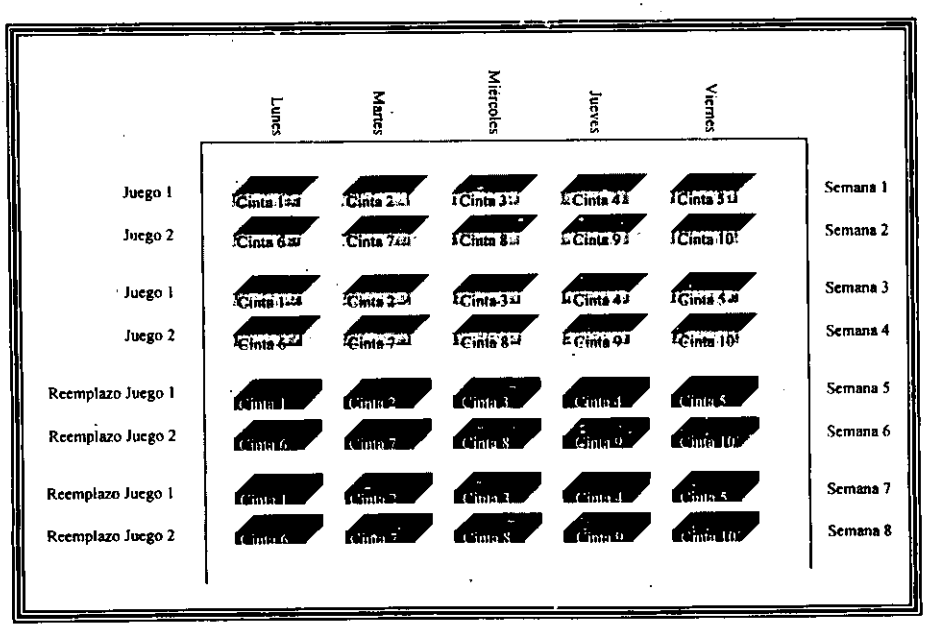

Fig. 4.4 Esquema de Respaldos de Base de Datos para la Nómina

La ventaja de este esquema de respaldos es que, en caso de falla de una cinta, se minimiza la pérdida de datos ya que una cinta contiene información de un solo día y esos días no son de una

semana consecutiva. Por ejemplo, si la cinta 1 se dañara se perdería la información (aparentemente) del día Lunes de la semana 1 y de la semana 3. Se dice que aparentemente se pierde la información porque se puede recuperar aplicando los respaldos de log al respaldo próximo anterior al del día perdido, es decir, si quisiéramos recuperar la información de la cinta 1 de la semana 3, primero se obtiene el respaldo de la cinta 10 y se le aplican los respaldos de log del día Lunes de la primera semana.

Estos respaldos se deberán hacer en las horas en que no hay usuarios que utilizan información en línea, principalmente los módulos de captura, ya que son tardados y alentan un poco el tiempo de respuesta, por lo que es recomendable realizarlos en la noche dejando un trabajo de ejecución que ejecute sistema operativo (mediante *cron*) a las 24:00 hrs.

Es importante identificar las cintas con rubros que especifiquen claramente la información que contienen. Para las cintas en uso, se debe especificar:

- El nombre de la base de datos
- El host en el que se respaldó, esto previniendo un cambio de base de datos a otro host con densidad de la unidad de respaldo distinta al actual (por ejemplo, cambiar de unidad DDS2 a DDS3[19])
- La fecha en que se comenzó a utilizar, para que en el futuro se determine si es necesario rebobinar la cinta para su correcto funcionamiento
- El juego de cinta al que pertenece y
- El día que respalda

Un ejemplo de rótulo de una cinta es el siguiente:

[19] DDS (Digital Data Storage) Almacenamiento de Datos Digital. Formato DAT para la copia de seguridad de datos. DDS es un método de grabación secuencial; debe añadirse datos al final de los datos anteriores.

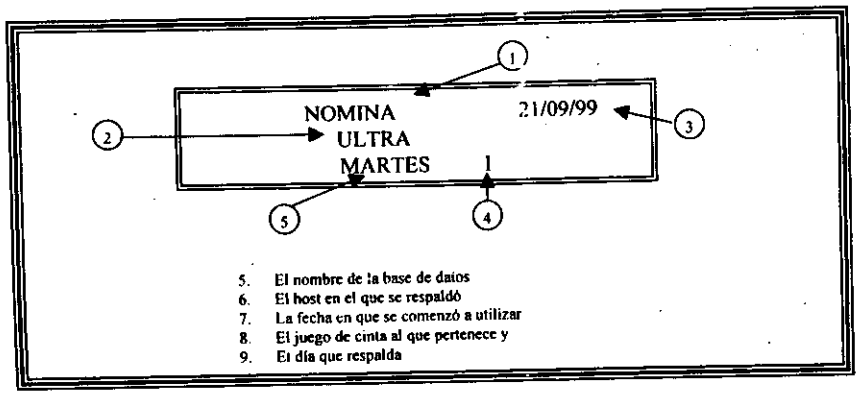

Fig. 4.5 Forma de Identificar una Cinta en Uso

Para las cintas que se van llenando y que se guardarán como históricas (las que contienen la información de la base de datos en cierres de quincena después de ejecutar el Módulo de Cálculo) es necesario protegerlas contra escritura (abriendo la muesca de protección de escritura) y se deben identificar las fechas que se respaldaron en el formato AA/MM/DD.

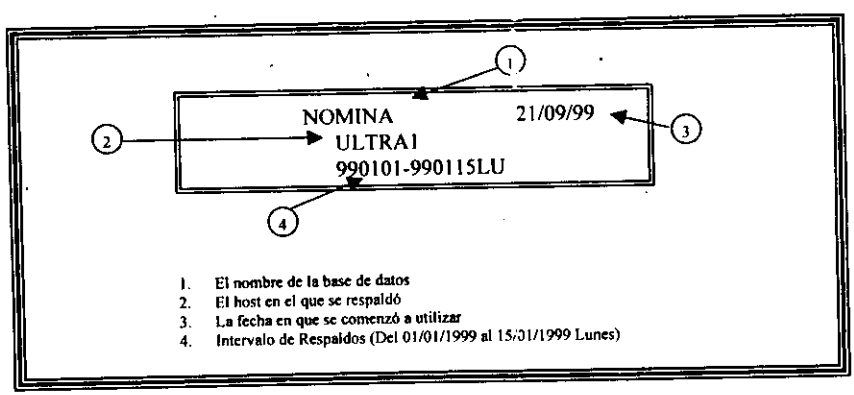

Fig. 4.6 Forma de Identificar una Cinta Histórica

Adaptive Server genera un archivo en cinta por cada base de datos que se respalda y, por default, nombra este archivo tomando las últimas siete letras del nombre de la base de datos, los dos dígitos del año, tres dígitos del día del año (1-366) y el número de segundos desde la media noche en hexadecimal, lo cual da un nombre de archivo poco amigable. Para nombrar a los archivos de cinta se deben tomar el nombre de la base de datos y la fecha en el formato DDMMAA, ejemplo: NOMINA010100 (Respaldo de la Base de datos Nómina del día primero de enero del 2000), lo cual facilita la recuperación del respaldo cuando se indique la opción *with file* del comando *load database*.

Calendario para Respaldos del Log de Transacciones

El respaldo del log de transacciones es mucho más pequeño y rápido que el de la base de datos pero es muy útil en caso de recuperación. Debido a que se realizará en horas pico de trabajo (entre las 12:00 y las 2:00 PM) a diario, es recomendable hacerlos a disco porque son más rápidos y la unidad de cinta puede ser requerida en esas horas para recuperar un respaldo de emergencia o para respaldar algo con urgencia.

El respaldo del log se debe realizar en las horas de más actividad para aumentar la probabilidad de tener toda la información completa cuando se necesite restablecer una base de datos a partir de estos respaldos de log. Los respaldos se deben hacer cada dos horas a partir de las 12:00 hasta las 18:00 hrs. Esto se determinó basándose en la hora en que se comienzan a realizar actualizaciones a la base de datos más frecuentemente.

Como los respaldos son a disco, Adaptive Server crea un archivo por cada respaldo. El nombre de este archivo debe estar formado por el nombre de la base de datos, día, mes y año así como la hora en que se realizó. Este formato es para poder sacar estadísticas de volumen de datos afectados por intervalos de hora. Ejemplo, NOMINA010100-12:00, indica que el respaldo se realizó el 1 de enero del 2000 a las 12.00 hrs.

Estos archivos se pueden comprimir para que ocupen menos espacio y deben mantenerse al menos por una semana para poder recuperar información de la semana inmediata anterior de forma rápida y cuando se requieran respaldos de semanas anteriores se debe recurrir a los respaldos históricos de Unix, donde se respalda el sistema de archivos donde se encuentran los logs. El sistema de archivos de Unix dedicado para tal efecto es /RESPLOG.

4.5 Auditoría

La auditoría es una parte importante de la seguridad, ya que registra la actividad del sistema la cual sirve para detectar intrusiones o mal uso de los recursos.

El Sistema de Auditoría

Un usuario con role de sso puede manejar la auditoría y es el único que puede iniciarla, detenerla, configurar nuevas opciones y procesar los datos generados. Como usuario con el role sso se puede establecer la auditoría para:

- Eventos relevantes para la seguridad a nivel servidor
- Creación, eliminación y modificación de objetos de la base de datos
- Todas las acciones que ejecuta un usuario con algún role
- Dar y negar accesos a la base de datos
- Acciones que involucren importar desde o exportar hacia archivos planos
- Entradas y salidas (login y logout) al DBMS

El sistema de auditoría consiste de:

- La base de datos *sybsecurity* que contiene todas las opciones globales de auditoría y las tablas donde se van registrando los eventos
- La cola de auditoría en memoria a donde se envían los registros antes de escribirlos a la base de datos
- Parámetros de configuración que manejan la auditoría
- Procedimientos almacenados del sistema que se ocupan para la auditoría

Adaptive Server utiliza varias tablas para almacenar los registros de auditoría. En un momento dado sólo una tabla es la actual, esto lo indica el usuario con role sso mediante el comando *sp_configure*. El número recomendado de tablas es dos o más, es posible indicar a la auditoría una sola tabla pero, en este caso, hay una ventana de tiempo durante la cual los registros se pueden perder al momento de vaciar los registros a archivos para su análisis.

El proceso de auditoría se ilustra en la siguiente figura.

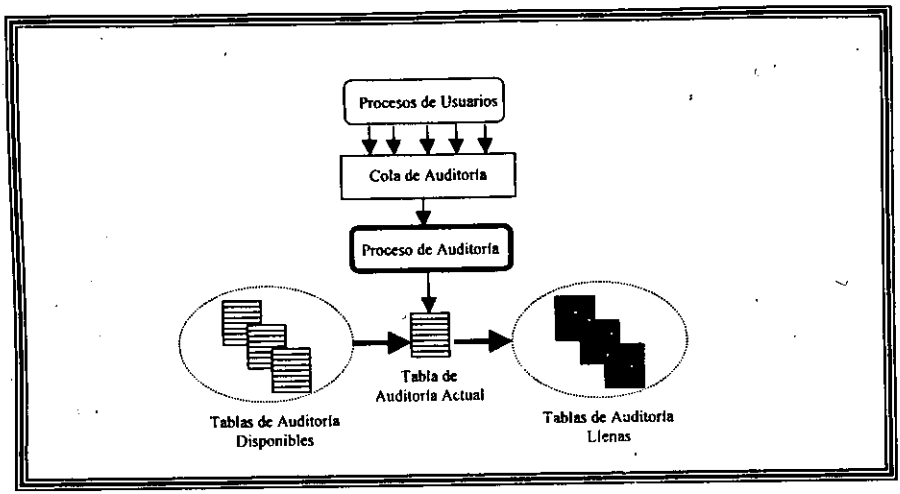

Fig. 4.7 Proceso de Auditoría

Opciones de Auditoría

Las opciones de auditoría de Adaptive Server son muy variadas, para un mejor estudio las vamos a dividir en eventos que generan mucha información por la naturaleza de la Nómina (por el nivel de detalle que ofrecen) y los eventos menos frecuentes (que generan menos información):

Eventos que Generan Mucha Información

- Acciones realizadas por un usuario en particular o por un usuario con algún role
- Todo el texto que un usuario ejecute en una sesión
- Accesos a una base de datos
- Borrado de registros de una tabla
- Ejecución de procedimientos almacenados
- Ejecución de triggers
- Inserción de registros a una tabla o vista
- Intentos de entrar al DBMS (fallidos, exitosos o ambos)

- Fin de sesión de un usuario del DBMS
- Referencias entre tablas de la base de datos
- Ejecución del comando *select*
- Actualización de un registro de una tabla
- Acceso de un usuario a cualquier vista
- Acceso de un usuario a cualquier tabla

Eventos que Generan Menor Información

- La ejecución de comandos *alter table* o *alter database*
- Carga de archivos hacia tablas mediante la utilería *bcp*
- Ejecución de los procedimientos *sp_bindmsg, sp_bindrule y sp_bindefault*
- Creación de objetos
- Ejecución de *dbcc*
- La ejecución de los comandos para manejo de dispositivos
- Eliminación de objetos
- Ejecución de respaldos
- Errores fatales o no fatales
- Acceso a una base de datos vía una función SQL
- Acceso a un objeto de la base de datos vía una función SQL
- Ejecución del comando *grant* para permitir el acceso a objetos
- Recuperación de una base de datos desde un respaldo
- Ejecución del comando *revoke* para negar accesos a objetos
- Ejecución de procedimientos almacenados entre servidores (Remote Procedure Calls)
- Cuando inicia el sistema o cuando se da de baja
- Cuando se utiliza el comando *kill* para terminar una sesión de usuario

- Cuando se pone en línea una base de datos o se modifica un parámetro de configuración del servidor

- Ejecución del comando *truncate table*

Como una primera aproximación, la Nómina debe registrar eventos que primeramente den un indicio de algún problema de seguridad. Si se detecta algo anormal, como por ejemplo demasiados intentos fallidos de conexión o sesiones fuera del horario normal de operación, entonces se deberá utilizar alguna opción de las que generan información detallada acerca de un usuario en particular, ya que por la cantidad de usuarios que se manejan y el número de operaciones que realizan, no es posible registrar las acciones que cada uno ejecuta. Este enfoque se refuerza por el diseño de las tablas de la base de datos de Nómina, ya que todas las tablas que contienen información delicada tienen columnas que registran fecha, hora y usuario que modifica los datos. Aunado a esto, están las restricciones de acceso a tablas, vistas y procedimientos almacenados, por lo que si alguien consulta información de las tablas es porque tiene los permisos correspondientes. Resumiendo, se deben auditar los siguientes eventos:

- Intentos de entrar al DBMS (fallidos, exitosos o ambos) porque proveen la base para detectar un intento fallido frecuente, lo que puede ser indicio de un intento de adivinar passwords.

- La ejecución de comandos *alter table* o *alter database*, porque en un ambiente de producción no es muy frecuente que se alteren los parámetros de una tabla, por lo que si alguien los ejecuta debe ser por una razón muy fuerte. La alteración o "crecimiento" de una base de datos sólo se da cuando hay una insuficiencia de espacio, por lo que su ejecución requiere que se haya hecho un análisis previo de distribución de espacios para asegurar la disponibilidad.

- Creación de objetos. Los objetos de un esquema casi no deben modificarse en un ambiente de producción, su modificación incorrecta puede generar una inconsistencia en los datos obtenidos en los diferentes procesos.

- Carga de archivos hacia tablas mediante la utilería *bcp* porque cuando se hacen estas cargas no es posible respaldar el log de transacciones.

- Errores fatales o no, para predecir o detectar fallas de hardware o software.

- Eliminación de objetos, por las mismas razones que cuando se crean.

- La ejecución de los comandos para manejo de dispositivos. Puesto que involucra eliminación o creación de dispositivos, se debe hacer una análisis de disponibilidad previo.

- Ejecución del comando *revoke* para negar accesos a objetos, porque constituyen el sistema de control de acceso discrecional.

- Ejecución del comando *grant* para permitir el acceso a objetos, por la misma razón anterior.

- Ejecución de procedimientos almacenados entre servidores (Remote Procedure Calls). Para auditar las operaciones que realizan usuarios de otros servidores.

- Cuando inicia el sistema, cuando se da de baja, se utiliza el comando *kill* para terminar una sesión de usuario, se pone en línea una base de datos o se modifica un parámetro de configuración del servidor. Estas operaciones son muy delicadas ya que tienen que ver con la disponibilidad.

- Ejecución del comando *truncate table*. Por su naturaleza (eliminación de información total de una tabla) es de uso restringido al dueño de la base de datos y debe haber una razón muy fuerte para truncar una tabla .

Observaciones de Tablas de Auditoría

Las columnas de las tablas de auditoría contienen los siguientes datos:

Nombre de la Columna	Descripción
event	Tipo de evento que se esta auditando (alter database, alter table, bcp. bind, create, acceso a la base de datos, dbcc, delete, drop, errores, ejecuciones, grant, insert, revoke, inicio del servidor, fin del proceso del servidor, cambios de password, select, setuser, acceso a una tabla, update, acceso a una vista, etc).
eventmod	Más información acerca del evento 0 = No aplica 1 = El evento cumple con los permisos 2 = El evento no cumple con los permisos
spid	Identificador del proceso que generó el registro
eventtime	Fecha y hora del evento
sequence	Secuencia que ocupa el registro dentro de varios registros generados por el mismo evento
suid	Identificador de la cuenta de usuario
dbid	Identificador de la base de datos que afecta
objid	Identificador del objeto que fue afectado
xactid	Identificador de la transacción que contiene evento
loginname	Nombre de la cuenta de usuario
dbname	Nombre de la base de datos
objname	Nombre del objeto
objowner	Dueño del objeto
extrainfo	Información adicional acerca del evento auditado

Tabla 4.7 Descripción de las columnas de la tabla de auditoría

Puesto que estas tablas crecen rápidamente, es importante limitar el crecimiento de éstas para no llenar el espacio de la base de datos, poniéndolas en segmentos[20] separados e implementando thresholds[21] para liberar el espacio (truncándolas); también se debe poner la opción de truncar el log de transacciones a la base de datos para evitar que se llene e influya en el funcionamiento normal del servidor.

[20] Los segmentos son partes de una base de datos que se distribuyen en diferentes dispositivos y sirven para controlar el espacio que puede llegar a ocupar una tabla o índice.

[21] Son procedimiento del sistema que monitorean el crecimiento de un segmento y realizan alguna acción cuando se llega a un límite determinado.

Recomendaciones de Administración

Adaptive Server utiliza un archivo de errores, llamado *errorlog*, donde envía toda la información referente al comportamiento del servidor. Es deseable que el administrador lo revise una vez al día, al menos, para detectar problemas a tiempo.

También es importante llevar un bitácora (libreta con anotaciones), de todos las actividades administrativas que realiza como movimientos de datos entre servidores, creación de índices, problemas del servidor y soluciones, movimientos a la configuración del servidor, etc, anotando siempre la fecha, y de ser posible la hora, en la que ocurrió tal actividad. Estas bitácoras son de gran utilidad para determinar origen de un mal funcionamiento del servidor (por cambios en configuraciones, por ejemplo) ya que no es posible recordar cada tarea realizada a lo largo del tiempo.

Existe una lista de discusión bastante buena, acerca de la administración de servidores Sybase, llamada L-SYBASE. Es bueno que un administrador este subscrito a tales listas porque en ella se exponen problemas que otros administradores han tenido y de los cuales se puede aprender mucho. También se recomienda ingresar frecuentemente al sitio web de Sybase para estar al tanto de nuevos "bugs" y parches para Adaptive Server.

Capítulo 5: Seguridad de Red

5.1 Firewalls

Las entradas forzadas a las computadoras ocurren de diversas maneras porque los sistemas conectados a la internet casi siempre tienen ciertas vulnerabilidades. Muchas veces las empresas instalan firewalls para bloquear intrusos y de esta manera proteger sus redes.

En la construcción de edificios, un firewall (muro contra incendios) está diseñado para evitar que se propague el fuego de una parte a otra. En teoría, un firewall para internet cumple un propósito similar: evita que los peligros de internet se extiendan a la red interna. En la práctica, un firewall para internet se asemeja más bien a la fosa de un castillo medieval que al muro contra incendios de un edificio moderno. Satisface varios propósitos:

- Restringe el acceso a un punto cuidadosamente controlado
- Evita que los atacantes se acerquen a las demás defensas
- Restringe a las personas para que salgan en un punto cuidadosamente controlado

Un firewall se instala con mayor frecuencia en el punto donde la red interna se conecta con el exterior, como se muestra en la fig. 5.1.

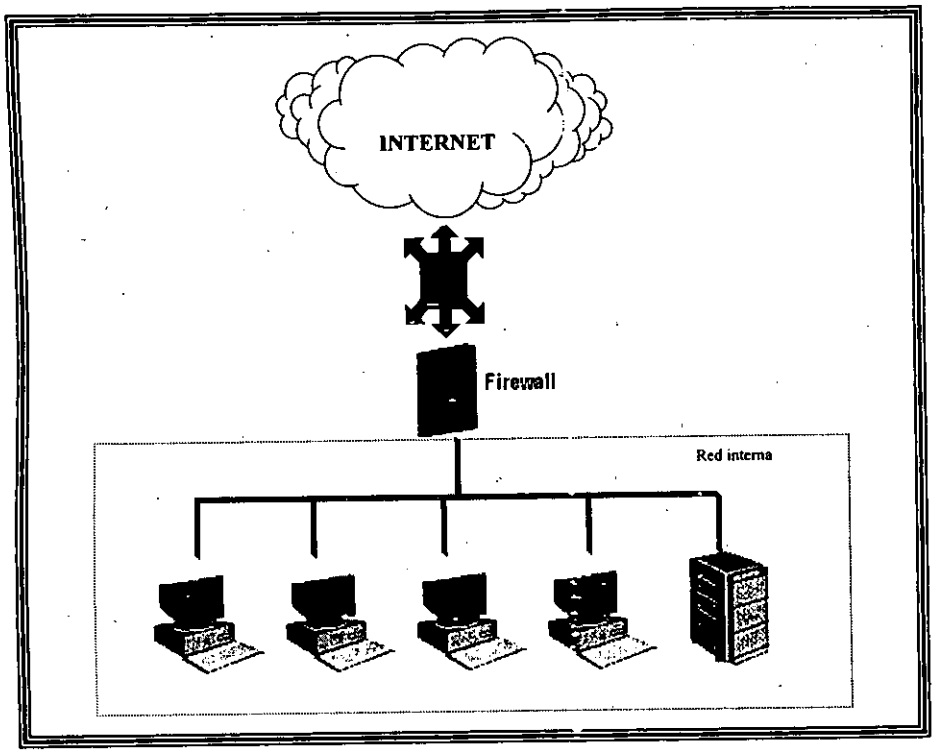

Fig. 5.1 Ubicación del Firewall

Todo el tráfico que viene de la internet o que sale de la red interna pasa por el firewall, por lo tanto, el firewall es un separador, un analizador y por lo general es un conjunto de componentes de hardware (un enrutador, una computadora o cierta combinación de enrutadores, computadoras y redes) con software apropiado.

Un firewall rara vez es un solo objeto físico, aunque algunos de los productos comerciales más nuevos intentan poner todo en la misma caja. Es común que un firewall tenga varias partes y algunas de éstas quizá cumplan otras tareas además de funcionar como parte del firewall.

Se ha comparado el firewall con la fosa de un castillo medieval, y como tal, es vulnerable. No protege contra la gente que ya está dentro; funciona mejor en conjunción con defensas internas y, aunque este llena de cocodrilos, algunas personas lograrán nadar al otro lado.

¿Qué Puede Hacer un Firewall?

Un firewall es como un cuello de botella. Todo el tráfico que entra y sale debe pasar por él. Un firewall da un grado de eficiencia enorme a la seguridad de redes porque le permite concentrar sus medidas de seguridad en este punto de inspección: el punto donde la red se conecta al mundo exterior.

Muchos de los servicios que las persona demandan, por su propia naturaleza son inseguros. Un firewall es como un agente de tránsito para estos servicios. Refuerza las políticas de seguridad del sitio, permitiendo que pasen sólo los servicios "aprobados".

A veces se utiliza el firewall para mantener separadas una sección de la red de otra. Al hacer esto, se evita que los problemas que impactan una sección se extiendan a través de toda la red. Algunas veces se hace esto porque una sección de la red puede ser más confiable que otra; en otros casos, debido a que una red es más sensible que otra. Cualquiera que sea el motivo, la existencia de un firewall limita el daño que un problema de seguridad en la red puede causar a la red en general.

¿Qué no Puede Hacer un Firewall?

Los firewalls ofrecen excelente protección contra las amenazas de la red, pero no son una solución de seguridad total. Pueden evitar que un usuario del sistema envíe información confidencial fuera de la organización a través de su conexión de red; también se podría evitar no teniendo una conexión de red. Pero ese mismo usuario podría copiar los datos en disco, cinta o papel y sacarlos del edificio en su portafolio. Si el atacante está dentro del firewall (si la zorra ya esta en el gallinero), este mecanismo de seguridad no puede hacer nada. Los usuarios internos pueden robar datos, dañar el hardware y el software o modificar los programas de manera sutil sin acercarse al firewall.

Un firewall puede controlar el tránsito que pasa por él pero no puede hacer nada por los flujos de información que no pasen por él.

Un firewall está diseñado para contrarrestar amenazas conocidas y, los bien diseñados al no permitir el flujo de servicios no permitidos (negación preestablecida del servicio), protegen de nuevos servicios que tengan vulnerabilidades. Sin embargo, ningún firewall puede defenderse automáticamente contra cada amenaza nueva que surge en los servicios que controla. Periódicamente se encuentran nuevas formas de explotar vulnerabilidades que antes eran confiables o se descubren ataques que sencillamente antes no se le habían ocurrido a nadie.

Los firewalls no pueden proteger de virus, pero sí pueden relegar esta responsabilidad a otra máquina y sólo para ciertos servicios explícitamente declarados (por ejemplo, ftp o smtp). Hay demasiados tipos de virus e infinidad de formas en que uno de ellos puede ocultarse dentro de los datos. Detectar un virus al azar en los paquetes de datos es muy difícil; requiere de:

1. Reconocer que el paquete es parte de un programa o de un documento

2. Tener una base de datos "siempre" actualizada con los virus más recientes

3. Realizar el reconocimiento del documento para determinar la existencia de un virus

4. Si existe tiene dos caminos a elegir, lo intenta vacunar o lo bloquea. Si lo bloquea no quita el problema de archivos con virus y si lo vacuna puede tardar mucho tiempo.

5. Enviarlo a su destino original

Incluso el primer punto es un reto. La mayor parte de los firewalls son máquinas protectoras con plataformas variadas y con diferentes formatos de archivos ejecutables. Un programa puede ser un archivo ejecutable compilado o un script (por ejemplo un script para algún shell de Unix) y muchas máquinas soportan múltiples archivos ejecutables. Además, la mayoría de los programas están empaquetados para su transporte y con frecuencia también están comprimidos. Los paquetes transferidos por medio de correo electrónico o noticias de Usenet también están codificados a ASCII de diferentes maneras.

La forma más efectiva de solucionar el problema de los virus es a través de programas antivirus instalados en los anfitriones y la educación de usuarios en relación con los peligros y las precauciones que se deben tomar contra ellos.

Los firewalls han evolucionado en la forma de verificar que una conexión es válida o no, siendo selectivos a nivel IP, en sus inicios, hasta ser selectivos en la capa de Aplicación. A los primeros se les llama firewalls de filtrado de paquetes y a los segundos gateways de aplicación. Su evolución se analiza brevemente en la siguiente sección.

Evolución de los Firewalls

Los firewalls han evolucionado a través de una serie de "generaciones". Es posible agruparlos de acuerdo a las necesidades que cubren pero el enfoque más directo es ver dos categorías separadamente.

Filtrado de Paquetes (Primera Generación)

El filtrado de paquetes es el proceso de permitir o negar el paso del tráfico entre dos redes basándose en la información del encabezado de cada paquete de datos. Un dispositivo de filtrado de paquetes utiliza la dirección IP fuente, el destino, el puerto (servicio) y alguna otra información para establecer las reglas para permitir o negar el flujo de tráfico en la red (el protocolo, por ejemplo).

Antes de los firewalls comerciales, la persona encargada de administrar la red comenzaba a crear reglas para deshabilitar cierto tráfico no deseado y los vendedores de ruteadores trabajaron para proporcionar herramientas para satisfacer esta necesidad creciente. El filtrado de paquetes,

en esta etapa, fue llamado "estático" debido a que cualquier método de conexión deseado entre redes internas y externas debía dejarse abierto todo el tiempo.

Las ventajas del filtrado de paquetes son:

- Baja sobrecarga
- Barato o inclusive gratuito
- Bueno para el manejo de tráfico

Debido a que el dispositivo hace poco trabajo fuera de rutear tráfico, la sobrecarga es extremadamente baja, por lo que la velocidad del tráfico es cercana a la del hardware. La habilidad para poner filtros de paquetes es estándar en la mayoría del hardware para conectividad de internet (por lo general ruteadores) y un administrador típico puede hacer algunas incursiones en el control del tráfico a través de estos dispositivos.

Las desventajas del filtrado de paquetes estático son:

- Permite conexiones directas de clientes externos a hosts internos
- Permanentemente deja hoyos abiertos en el perímetro de la red
- Se vuelve inmanejable en ambientes complejos
- Permanece vulnerable a ataques tales como modificación de la dirección fuente ("spoofing), a menos que se configure específicamente para evitar esto
- No ofrece autenticación de usuario

Cuando se permiten conexiones directas entre hosts que se encuentran fuera de nuestra red y los equipos internos, se confía en la seguridad que el administrador tenga en los servicios que proporciona el host (de nuestra red). Esto deja abierta la posibilidad de que el atacante explote las debilidades conocidas del sistema y que, inclusive, pueda utilizar aquellos servicios que no debiera. En pocas palabras, no permite un control tan sofisticado de los servicios.

Filtrado de Paquetes (Segunda Generación)

La desventaja más obvia de los filtros de paquetes estáticos es el conjunto de "puertas" que deben dejarse abiertas en todo momento para permitir el tráfico deseado. Esta desventaja hace a los sitios susceptibles a un gran número de ataques dependiendo de la seguridad de sus hosts en las redes internas. Debido a que la seguridad de los hosts es tratada a menudo con menor prioridad (o inclusive llega a ser muy complejo lograr una buena seguridad por la cantidad de demonios que se ejecutan), estos tipos de ataques fueron y son frecuentemente exitosos.

Para considerar esta situación, se desarrollaron técnicas dinámicas de filtrado de paquetes que abren y cierran "puertas" en el firewall basándose en la información del encabezado del

paquete de datos como se describe arriba. Una vez que una serie de paquetes han pasado por la "puerta" a su destino, el firewall la cierra.

El filtrado de paquetes "stateful" es un mejoramiento al filtrado de paquetes dinámico. Esta tecnología trata de entender algo de los protocolos de niveles más altos y adaptar las reglas de filtrado para ajustarse a sus necesidades específicas (ejemplo, simula conexiones para protocolos sin conexión tales como servicios NFS y RPC). También guarda información del estado y del contexto de la sesión. Esta tecnología puede ser aplicada también al protocolo UDP, poniendo una sesión virtual, dando la ilusión de tener seguridad donde, en realidad, no existe.

El agregar el seguimiento del estado de la sesión al filtrado de paquetes ciertamente puede incrementar la seguridad del filtro básico, pero no aborda el contenido o implicaciones del tráfico que se esta manejando.

Las ventajas del filtrado dinámico de paquetes son:

- Sólo abre hoyos temporalmente en el perímetro de la red
- Baja sobrecarga/Alta velocidad de transferencia
- Soporta casi cualquier servicio

Debido a que se reduce el tiempo en el que se encuentra abierto un hoyo en el perímetro, muchos ataques que funcionaban en el filtrado de paquetes estático ahora son más difíciles o tal vez imposibles. De nuevo, debido a que es muy poco el trabajo hecho fuera del ruteo del tráfico, la sobrecarga es relativamente poca. Por lo tanto, plataformas de hardware similar pueden producir alta velocidad de transferencia cuando usan técnicas de filtro de paquetes dinámico.

Sus desventajas son:

- Permite conexiones IP directas a los hosts internos
- No ofrece autenticación de usuario (si es soportada, se utilizan gateways de aplicación)
- Puede soportar cualquier servicio IP

Mientras que el filtrado de paquetes dinámico reduce las exposiciones, los sistemas externos aun pueden realizar una conexión IP con las máquinas internas. La desventaja principal de cualquier gateway de filtrado de paquetes es que una vez que se permite el acceso a los hosts internos, el atacante tiene acceso directo a cualquier debilidad explotable ya sea en el software o la configuración del host. La habilidad para brincar a otro host interno desde este host dependerá de la seguridad presente en los mismos.

Lo que es comúnmente conocido como "spoofing" (pretender ser una dirección IP confiable como un método para atacar la red), fue una vulnerabilidad bien conocida por muchos años; la mayoría de los filtros de paquetes dinámicos incluyen soluciones para los métodos más comunes de spoofing pero el problema sigue latente en esa confianza que se tiene al sistema externo basada en su dirección IP. Aún si el tráfico entrante es del host correcto, no existen verificaciones para validar que el host está siendo operado por los usuarios autorizados. En otras

palabras, si un atacante ha comprometido el host externo, puede usar al mismo como una puerta para entrar a la red interna.

Además, los firewalls de filtrado de paquetes no soportan el concepto de autenticación fuerte de usuario. El permitir el acceso desde redes no confiables sin una autenticación fuerte es una amenaza seria para la seguridad de una red.

Una de las ventajas de esta técnica sobre los gateways de aplicación (que se verán más adelante) es que se puede permitir cualquier tipo de tráfico. Sin embargo, al no conocer que es capaz de hacer una aplicación (que comandos utiliza, de que manera transmite su información, etc.), no hay manera de estimar la amenaza impuesta por la misma, por lo que a menudo muchas aplicaciones peligrosas se permiten pasar a través de estos filtros.

Gateways de Aplicación (Primera Generación)

Un gateway de aplicación, es un sistema firewall en el cual el servicio es proporcionado por procesos que mantienen el estado y secuencia completa de una conexión TCP. Este tipo de firewalls a menudo redireccionan el tráfico para que la información saliente parezca ser originada por él, en vez de los hosts internos.

Un gateway de aplicación es considerado por los expertos el tipo de firewall más seguro. Todas las conexiones a la red interna pasan por él. Un firewall de este nivel se distingue por el uso de proxies de seguridad (gateways de aplicación) para servicios tales como FTP, TELNET, etc., los cuales evitan el acceso directo a los servidores de la red interna.

Las ventajas de los gateways de aplicación son:

- No permiten una conexión directa entre hosts internos y externos
- Soportan autenticación a nivel usuario
- Analiza los comandos de la aplicación dentro de la porción de datos de los paquetes
- Mantiene bitácoras comprensibles del tráfico y de actividades específicas

La principal ventaja de los firewalls con gateways de aplicación es que no permite conexiones directas bajo ninguna circunstancia.

Los gateways de aplicación son conocidos también como proxies debido a que cuando están ejecutándose en el firewall "permanecen en el hueco", aparentando ser servidores para los clientes y, hacia adentro de la red, simulan ser clientes para los servidores. Es similar a entablar una conversación por teléfono con un abogado y él, en su momento, se comunica con la otra parte. Ya que el contenido y los términos de la conversación son críticamente importantes, se ha decidido que la tercera parte no conozca quién es la que está hablándole al abogado, donde está o inclusive que existe. Cualquier intento de daño por medio de esta conexión será dirigida al abogado, quien específicamente está capacitado para este propósito y está protegido por

estructuras conocidas (en este ejemplo, estructuras legales). Un filtro de paquetes, en este ejemplo, traería a las dos partes a un cuarto y allí los dejaría.

Haciendo referencia al ejemplo del abogado, los proxies realizan una función similar a la del abogado, leyendo el contenido de un documento antes de permitir actuar sobre él. El abogado debe estar íntimamente familiarizado con el significado e implicaciones del contenido del documento en relación con los efectos que conlleva y debe proteger de tomar una acción que pudiera ser peligrosa (Fig. 5.2).

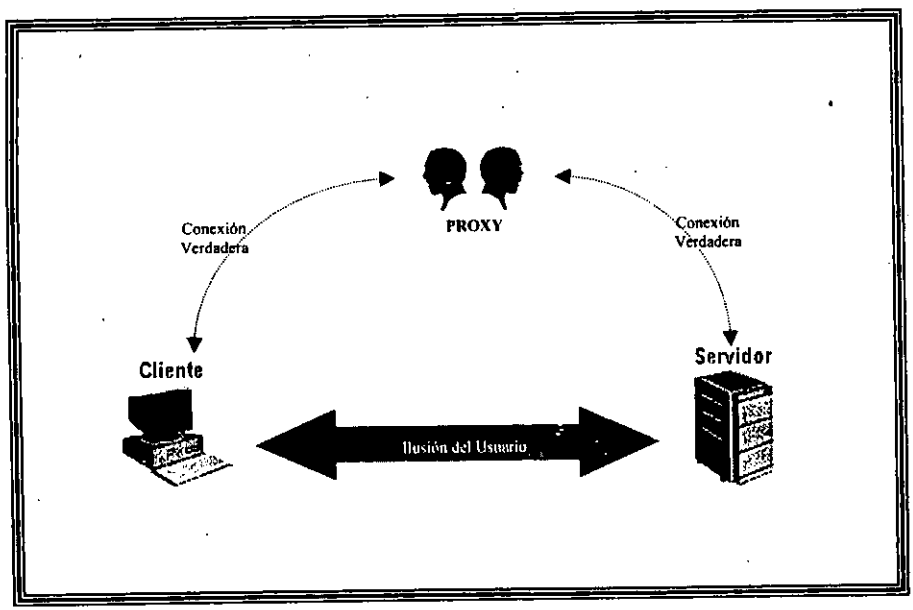

Fig. 5.2 Comunicación mediante proxies

Las desventajas de los gateways de aplicación son:

- Son más lentos que los filtros de paquetes
- Requieren que el cliente interno conozca acerca de la existencia de los gateways
- No soporta cualquier tipo de conexión posible

El primer punto siempre aplica a este tipo de filtros porque un firewall con gateways de aplicación hace más trabajo de seguridad. Si el contenido de cada "documento" es examinado en detalle, el proceso va a tomar siempre más tiempo que simplemente ordenar el correo. Afortunadamente, esta sobrecarga es manejada fácilmente por una plataforma de hardware típica

de un servidor. La velocidad de transferencia de un gateway de aplicación, por lo general, será más alta que la conexión a la red externa.

La segunda desventaja fue la más inconveniente. Para poder hablar con el "abogado" en el firewall, las estaciones de trabajo deben tener versiones especiales del software cliente instalado. (Algunos firewalls permiten "transparencia" en las comunicaciones, lo que significa que las aplicaciones internas no necesitan saber acerca de la existencia de proxies, éstos se analizan en la siguiente sección).

La última desventaja es un factor del nivel de seguridad deseado por la organización que utiliza el firewall. Un ejemplo de esto, es la aparición de una nueva aplicación en uso por la red externa. El enfoque de gateways de aplicación indica que no se debe dejar pasar este tráfico hasta que no se conozca como trabaja y que se puede hacer para evitar que dañe la red interna.

Para subsanar este último punto, los vendedores de gateways de aplicación proporcionan una herramienta para la creación de proxies "genéricos" y pueden permitir alguna forma de filtrado de paquetes. La medida real que se establece a estos vendedores, sin embargo, es qué tan rápido pueden producir un proxy para estas aplicaciones.

Gateways de Aplicación Transparentes (Segunda Generación)

La transparencia es el mayor desarrollo en la segunda generación de gateways de aplicación. Una de las desventajas más grandes de los primeros gateways de aplicación fue que cada estación de trabajo detrás del firewall debía ser configurada para que "estuviese enterada" de la existencia del firewall y debía tener software cliente que estaba diseñado para poder comunicarse con el software proxy. La introducción de transparencia en los gateways de aplicación significa que, en los ambientes modernos de seguridad, la estación de trabajo cliente no tiene que enterarse del firewall o correr software especial para comunicarse con la red externa.

En el mundo de los firewalls con proxies, existen muchos términos que son utilizados con frecuencia y que su entendimiento es fundamental para una buena configuración y administración de los mismos. A continuación se aborda tal terminología.

Terminología para Servidores Proxy

Esta sección describe varios tipos de proxy.

Proxy a Nivel Aplicación en Comparación con Proxy a Nivel Circuito

Un proxy a nivel aplicación, es el que sabe sobre la aplicación específica para la cual esta proporcionando el servicio; comprende e interpreta los comandos en el protocolo de la aplicación. Un proxy a nivel circuito, es el que crea un circuito entre el cliente y el servidor sin interpretar el protocolo de la aplicación. La versión más extrema de un proxy a nivel aplicación es el de *sendmail*, que implementa un protocolo de guardar y enviar, entendiendo los comandos

que fluyen en la comunicación. La versión más extrema de un proxy a nivel circuito, es una de las modernas compuertas proxy híbridas que parecen un proxy para la red externa, pero en realidad es un enrutador para la red interna, es decir, sólo conecta algún cliente externo hacia uno o varios servicios internos y no verifica los comandos ejecutados. Para conocer el destino final, un proxy a nivel circuito basa las decisiones de destino totalmente en la dirección fuente y en los puertos fuente y destino de la conexión.

Proxy Genérico en Comparación con Proxy Dedicado

Aunque "a nivel aplicación" y "a nivel circuito" son términos utilizados a menudo, con mayor frecuencia distinguimos entre servidores proxy "dedicados" y "genéricos". Un servidor proxy dedicado es el que sirve a un solo protocolo; un servidor proxy genérico es el que sirve a varios protocolos. En la práctica, los servidores proxy dedicados son a nivel aplicación; los servidores proxy genéricos son a nivel circuito.

Redes Privadas Virtuales (VPN)

Las VPN extienden la red corporativa hacia oficinas distantes, equipo en casa, vendedores y socios de negocios; pero, en vez de utilizar líneas rentadas caras, utilizan servicios de red IP mundiales, incluyendo la internet. Son independientes de la plataforma, ya que se puede incorporar a cualquier sistema de cómputo que este configurado para utilizar una red IP sin modificarlo, más que para instalar el software remoto.

Los requisitos de seguridad que debe cumplir una VPN son:

- Privacidad de datos mediante cifrado
- Proteger los datos de ser modificados durante su recorrido por la red, usando transformaciones matemáticas llamadas funciones de hash[22] que crean firmas digitales para detectar datos alterados y
- Evitar engaños mediante la autenticación de las partes que se comunican.

El propósito de asegurar la privacidad, se logra manteniendo la información oculta para cualquiera que no sea el destinatario, aun para los que tengan la posibilidad de observar los datos cifrados. Los datos son transformados en una forma que esta cerca de lo imposible de ser leída si no se tiene el conocimiento adecuado. El cifrado y el descifrado generalmente requieren del uso de información secreta conocida como llave.

[22] Una función de hash es una transformación que toma una entrada y retorna una cadena de tamaño fijo. Sus características principales son: la entrada es de cualquier longitud, la salida es de longitud fija, es fácil de calcular para cualquier valor, no se puede obtener el valor original a partir del resultado de aplicar la función y debe ser imposible encontrar dos valores que den el mismo valor al aplicar la función.

Existen dos maneras de utilizar las llaves para cifrar y descifrar mensajes: manejo de llaves secretas y manejo de llaves públicas.

En la criptografía de llave secreta (conocido también como criptografía simétrica), se utiliza una misma llave para cifrar y descifrar un mensaje. Su mayor desventaja es el acuerdo al que deben llegar quien envía y quien recibe para utilizar la misma llave y garantizar que nadie más la llegue a conocer; esto requiere un método con el cual las dos partes se puedan comunicar tal llave sin ser escuchados o interferidos; si se encuentran en lugares físicamente distantes, deben utilizar un teléfono o algún medio de transmisión evitando la revelación de la llave. Debido a que todas las llaves deben permanecer secretas, a menudo tiene dificultades para un manejo seguro, especialmente en sistemas abiertos con un gran número de usuarios. La ventaja de este mecanismo, por otro lado, es que por lo general son más rápidos que los de llave pública.

Para resolver el problema del manejo de llaves, Whitfield Diffie y Martin Hellman introdujeron el concepto de criptografía de llave pública en 1976. En sus sistemas, cada persona tiene un par de llaves: una llamada la llave pública y la otra llave privada. La llave pública se entrega a todas las partes con las que se quiere tener transmisión de datos, mientras que la privada se mantiene en secreto. Con esto se elimina la necesidad de compartir información de llaves secretas entre las partes, ya que todas las comunicaciones involucran llaves públicas. Cualquiera puede enviar un mensaje confidencial, utilizando la llave pública, pero sólo puede ser descifrado con una llave privada la cual es propiedad del destinatario. La llave privada esta matemáticamente ligada a la llave pública, por lo tanto, es posible atacar estos sistemas mediante la derivación de la llave privada a partir de la pública; la defensa contra esto es hacer el problema de derivación difícil, utilizando longitudes de llaves tan grandes como sea posible. La desventaja de este mecanismo es que son más lentos que los de llave privada.

La autenticación es otra parte fundamental en la VPN. Usamos autenticación en nuestra vida diaria cuando firmamos un documento (por ejemplo). La autenticación es el proceso a través del cual uno prueba y verifica cierta información. Algunas veces se desea verificar el origen de un documento, la identidad de quien lo envía (computadora y/o usuario), la hora y la fecha en que fue enviado, etc. Una firma digital permite realizar tales verificaciones. La firma digital de un documento es una pieza de información basada en el documento y en la llave privada de quien lo firma; se crea mediante el uso de una función de hash y cifrando con la llave privada del que envía, aunque existen otros métodos.

Pero aun con este mecanismo, alguien puede publicar llaves haciéndose pasar por otra persona; para ello existen documentos digitales llamados **certificados** que asocian a una persona con una llave pública específica. En su forma más simple, los certificados también contienen una fecha de expiración, el nombre de la autoridad que certifica, un número serial y tal vez otra información.

Una VPN que conecta dos o más redes corporativas es llamada "extranet". La única diferencia entre VPN intra-compañía (intranet) y una VPN inter-compañía (extranet) es la manera que se administra la VPN. En una intranet, toda la red y los recursos VPN son manejados por una misma organización. Con una extranet, no hay una única organización que maneje toda la red y los recursos VPN.

Mediante el cifrado, se proporciona privacidad para todo el tráfico de red permitido entre dos gateways (que pueden ser firewalls). El manejo de las llaves es el aspecto más difícil y crítico de

los sistemas de cifrado. Cualquier solución VPN viable, debe tener un mecanismo de manejo de llaves para negociarlas e intercambiarlas de forma segura.

Los estándares VPN más populares son: Point-to-Point Tunneling Protocol (PPTP), Layer 2 Tunneling Protocol (L2TP) e IP Security (IPSec). Este último define un conjunto de especificaciones para servicios de autenticación, integridad y confidencialidad en el datagrama IP. Las especificaciones de estos tres estándares son un conjunto de RFCs que describen los protocolos utilizados para tunneling [23].

[23] Tunneling se le llama a la encapsulación de un protocolo A dentro de un protocolo B como si fueran datos.

El firewall Gauntlet

El firewall Gauntlet es una combinación de software y hardware; el software corre en una versión "recortada" del sistema operativo Unix. Por recortada se entiende que el sistema operativo en la computadora que aloja al firewall ha sido modificado para que la máquina misma sea menos susceptible a un ataque; esto es a lo que se le llama un "host bastion".

La configuración por defecto de este firewall es permitir casi todos los servicios hacia el exterior a los usuarios de la red protegida (una condición que se puede modificar), mientras bloquea la mayoría de los accesos a la red interna.

Redes Confiables y Redes no Confiables

El firewall Gauntlet debe ser configurado para diferenciar entre los "chicos buenos" y los "chicos malos". El firewall hace esta determinación, usando la información que se le proporcione acerca de las diferentes redes: confiables y no confiables.

La red confiable es la red o redes dentro del perímetro de seguridad (también llamada la red interna). La red confiable es aquella que se esta tratando de proteger, por lo general la red de la empresa. La misma empresa se encarga de administrar las máquinas de esta red y controla las políticas de seguridad.

Cuando se configura el firewall, se le indica explícitamente la red en la que él puede confiar. Después de la configuración inicial, las redes confiables, por lo general, incluyen el firewall mismo y todas las redes dentro del perímetro de seguridad. Otros homónimos para la red confiable son: red corporativa y red interna.

Las redes no confiables (o red externa) son las redes fuera del perímetro de seguridad. Son no confiables porque están fuera del control o conocimiento de la organización; no se tiene control sobre su administración o políticas de seguridad. Son de las que se está tratando de proteger la red. Sin embargo se necesita y se requiere comunicación con ellas aun cuando no son confiables.

Igualmente, cuando se configura el firewall, explícitamente se configuran las redes de las cuales puede aceptar peticiones pero que no confía. Por defecto, después de la configuración inicial, las redes no confiables son todas aquellas que están fuera del perímetro de seguridad.

El firewall maneja las peticiones de las redes confiables de manera diferente a como trata las de las no confiables. Por ejemplo, la configuración por defecto permite peticiones de HTTP provenientes de la red confiable pero no de las redes no confiables. En otras palabras, los usuarios de la organización pueden usar un navegador de Web dentro del perímetro de seguridad,

pero si alguien desde un navegador en una red no confiable trata de accesar la red interna, esa petición de HTTP va a ser rechazada.

Filosofía de Diseño

El Firewall Gauntlet sigue el paradigma:

"Lo que no esta expresamente permitido, esta prohibido"

El firewall debe permitir explícitamente las actividades, ya sea a través de la configuración por defecto que trae el sistema o a través de la configuración que se implemente. Los nuevos servicios no pasan por él a menos que se declaren como permisibles.

Reconociendo que la mayoría de las violaciones de seguridad ocurren mediante una cuenta de usuario comprometida, el Firewall Gauntlet no tiene cuentas de usuario en el sistema operativo. La única cuenta que existe es la del administrador, los usuarios no necesitan entrar a la máquina del firewall para recuperar información del otro lado de la misma.

Combina tecnología de proxies de aplicación, de circuito y de filtrado de paquetes. Nunca hace una conexión directa entre máquinas en lados opuestos.

¿Cómo se Conecta un Firewall Gauntlet?

Cada red (red interna y red externa), está asociada con una tarjeta de red (también llamadas tarjetas interfaz). Cada interfaz tiene una dirección IP separada.

TIPOS DE INTERFASES

Cada interfaz física, en el firewall, se puede identificar como uno de los siguientes tipos:

- Interna: Que será utilizada por las máquinas internas así como por el firewall. Se conecta con los equipos que se desean proteger.
- Externa: La utilizan las redes externas que se necesitan comunicar.
- Servicio: Cualquier servicio de red que se necesite poner dentro del firewall, como servidores http.
- Desconocida: Se puede utilizar este tipo para indefinir una interfaz previamente definida como alguna de los tipos anteriores.

La siguiente figura muestra cómo la computadora que aloja el Firewall esta físicamente conectada a la red externa y a la red interna.

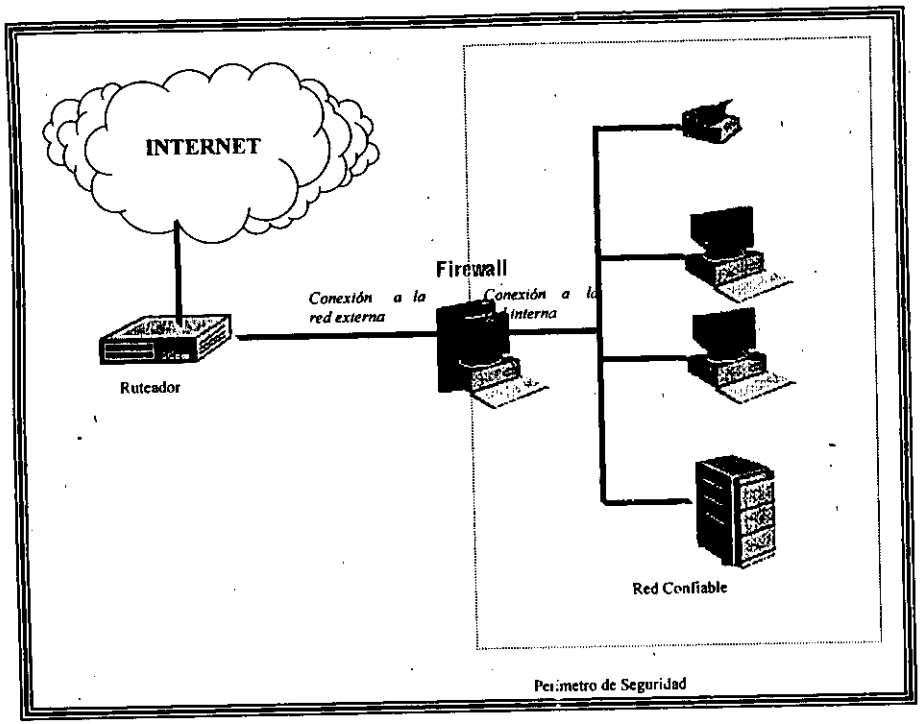

Fig 5.3 Conexión del Firewall Gauntlet

Transparencia

Transparencia significa que, al trabajar, el firewall no es visible para los usuarios. Pueden continuar con el servicio de telnet, por ejemplo, a los sitios clientes sin tener que conectarse explícitamente al firewall.

La configuración por defecto del Firewall Gauntlet implementa transparencia para los usuarios de la red confiable. Esto es acompañado de la creación de ruteadores por default que envían todas las peticiones a las redes no confiables a través del firewall.

En contraste, el firewall no implementa transparencia para peticiones desde redes externas. En esta caso, se requiere que los usuarios fuera del perímetro de seguridad sean avisados que están entrando a la red a través del firewall.

La ventaja del acceso transparente es que no se necesita configurar cada aplicación en los sistemas clientes o aprender nuevos procedimientos para utilizar los servicios soportados.

También se puede implementar el acceso no transparente para los usuarios internos, pero deberán aprender nuevos procedimientos para realizar sus tareas.

Sistema Operativo

El sistema operativo es una versión recortada de UNIX (BSD/OS, HP-UX y Solaris). Como parte del firewall, estos sistemas operativos han sido ajustados para proporcionar soporte sólo a los servicios necesarios para él. Por ejemplo, el sistema operativo del firewall no soporta paquetes de IP Forwarding o paquetes con ruteo de la fuente. Estos servicios cambian la dirección del flujo de paquetes y pueden indicar a la red que burle el firewall. Servicios como NFS, NIS y RPCs están deshabilitados debido a que no es fácil hacerlos seguros.

Los servicios de red no soportados no sólo reportan un error al sitio que los requiere, sino que además, el sistema operativo registra estos intentos de acceso, proporcionando información de la exploración que se trata de hacer a la máquina.

Filtrado de Paquetes

El software del firewall incluye una facilidad para filtrado de paquetes, la cual permite soportar un alto ancho de banda (en otras palabras, velocidad de respuesta) para los servicios que así lo requieran o para las aplicaciones no soportadas. Esta utilería es buena utilizarla cuando se tiene tráfico con menos requerimientos de seguridad.

Esta facilidad checa los paquetes IP basándose en las reglas de filtrado y los procesa de acuerdo a estas reglas. Se pueden detectar paquetes con *spoofing* (paquetes que pretenden ser de una red pero en realidad pertenecen a otra).

Servicios de Seguridad a Nivel Aplicación (Proxies)

El software del Firewall Gauntlet incluye servicios de seguridad por aplicación. Todos los paquetes de datos (y por lo tanto todas las peticiones a la aplicación) van al firewall. Los servicios proxy "relevan" la información de un lado del firewall a otro, evitando que las aplicaciones de redes externas hablen directamente con las aplicaciones internas y viceversa. Los paquetes IP no pasan de un lado a otro, todos los datos pasan a nivel aplicación. Por lo general, cada aplicación habla a través de un proxy que entiende el protocolo que maneja esa aplicación. Actualmente el firewall incluye proxies para los siguientes servicios:

- Servicios de Terminal (Telnet y rlogin)
- Correo Electrónico (SMTP y POP3)
- Servicios de Transferencia de Archivos (FTP)

- Ejecución Remota (Rsh)
- Servicios de Web (HTTP, SHTTP, Info)
- Servicios Gopher (Gopher, Gopher+)
- Servicios X Windows (X11)
- Servicios de Impresión (lp)
- Servicios de SQL (Sybase SQL Server, Microsoft SQL Server, and Oracle, SQL*Net)
- Servicios Multimedia (RealAudio/RealVideo, NetShow, VDOLive, StreamWorks)
- Servicios de Administración de Red (SNMP)

Además, Gauntlet incluye un proxy genérico (también llamado plug proxy). Este proxy sirve para controlar tráfico TCP de un puerto en particular de un equipo en un lado del firewall a un puerto en particular en otro sistema del otro lado del firewall. Al igual que los proxies de servicios específicos, los paquetes IP no pasan de un lado del firewall al otro.

El firewall incluye versiones configuradas de plug proxy para:

- AOL
- Finger
- Compuserve
- LDAP (Manejo de Certificados)
- Lotus Notes
- NetMeeting
- Noticias Usenet (NNTP y News Feed)
- Servicios Web (SSL)
- Whois

También incluye un proxy de circuito para autenticación, funciona igual que el plug proxy pero requiere que los usuarios se autentiquen primero.

Debido a que los proxies utilizan el mismo protocolo que la aplicación que comunican, no es necesario modificar el cliente original o las aplicaciones del servidor.

Todos los proxies son configurables, se pueden permitir o negar peticiones hacia o desde ciertos sitios y redes. También se pueden habilitar o deshabilitar proxies individuales y correr solo los que se necesitan.

Los proxies registran toda la actividad que va o pasa por el firewall. En las bitácoras se pueden recuperar estadísticas de uso o buscar ataques potenciales.

Además, varios proxies soportan sistemas de autenticación fuerte de usuario. Estos sistemas de passwords de una sola vez o seguridad de token, proporcionan seguridad adicional porque los usuarios usan passwords diferentes cada vez que accesan la red.

Registro en Bitácoras

El firewall es capaz de evitar comunicaciones no autorizadas en cualquier dirección y proporciona una bitácora de todas las conexiones a través de él. Los siguientes eventos se registran por default:

- Todos los errores y avisos del kernel del sistema operativo
- Todos los errores y avisos del sistema de archivos
- Los intentos de acceso a los servicios de red, ya sea para servicios soportados o no, y si fueron exitosos o no.
- Todas las conexiones exitosas a la red, registrando la dirección fuente y destino, servicio, hora, hora de desconexión, números de bytes transferidos (si aplica), comandos utilizados (FTP) y URLs consultadas (si aplica).
- Todas las interacciones con el subsistema de autenticación.

El sistema genera un resumen de la actividad del sistema reportando el uso de cada servicio por usuario y el administrador puede generar un reporte de excepciones especificando la información que no está interesado en ver.

Soporte de Cambio de Dirección de Red (NAT)

Los dispositivos que soportan cambio de dirección de red (NAT, Network Address Translation), permiten usar direcciones "privadas"[24] o no registradas en un lado (la red confiable) mientras se conecta por el otro lado a la red externa. Cuando la información que viene de la red con direcciones privadas pasa por el dispositivo NAT, éste cambia la dirección inválida a una dirección válida para poderla usar hacia el exterior. El firewall soporta NAT.

[24] Por privadas se entiende que se están utilizando direcciones IP no validas para ruteo en internet, es decir, se están utilizando IP's privadas especificadas en los RFC 1597.

Gauntlet, por el diseño de firewall de aplicación, cambia todas las direcciones de los paquetes que vienen de la red interna y por la dirección IP de su interfaz externa. Debido a que hay una única conexión del firewall al mundo externo, éste último no tiene conocimiento de las direcciones de la red privada.

Las reglas NAT pueden ser "Dinámicas" o "Estáticas". Las reglas dinámicas permiten a los hosts de la red interna conectarse a los hosts que se encuentran fuera de la red (pero no al revés), mediante el mapeo de una dirección de red local a una dirección global en tiempo de transferencia y borra el mapeo cuando la conexión finaliza; la siguiente ves que se inicie otra conexión del mismo host interno a otro externo, se ocupara otra dirección IP externa distinta a la primera para entablar la comunicación, es decir, el mismo host interno cambia dinámicamente de dirección IP al comunicarse al exterior. Las reglas estáticas anuncian los servidores de la red privada a la red externa pero nunca varían su dirección ya que siempre la cambian a una misma dirección global (Fig. 5.4).

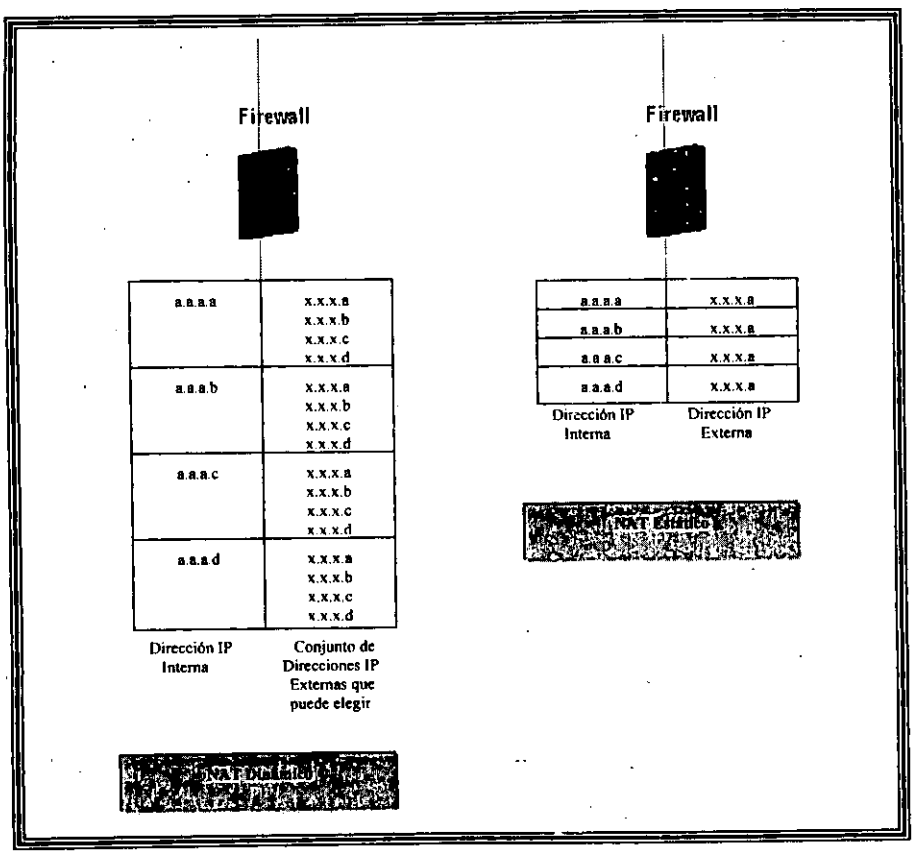

Fig. 5.4 NAT Dinámico y Estático

Autenticación Fuerte de Usuario

Autenticación fuerte significa (1) establecer la validez de la identidad declarada, (2) proporcionar protección contra transacciones fraudulentas estableciendo la validez del individuo. La identificación de un usuario se implementa en las computadoras a través de una cuenta de usuario y un password. El password se mantiene secreto y debe ser difícil de adivinar. En realidad, los passwords algunas veces son débiles (adivinables) y, en el caso de identificar usuarios conectados desde algún punto externo, es posible capturar el password y la cuenta de usuario (ya que por lo general los servicios envían el texto en claro). En consecuencia, mientras parece que la cuenta de usuario y el password constituyen un buen criterio de autenticación, el password es, algunas veces, fácilmente adivinable o capturado. Con la autenticación fuerte de usuario, la autenticación se hace de tal manera que se puede aplicar un alto grado de confianza a la identificación. Esto puede implementarse con passwords de una sola vez o dispositivos de autenticación, aunque estos últimos son más costosos.

Soporte de Redes Privadas Virtuales (VPN)

Las VPN's se implementan utilizando un producto adicional para el Firewall llamado Global Virtual Private Network (GVPN) y Recovery Key, que incorpora cifrado fuerte asegurando la privacidad de la red virtual.

Se puede crear una GVPN entre dos firewalls o entre un firewall y un cliente que cumpla con el estándar ISAKMP/Ipsec[25]. La tecnología RecoveryKey significa que se puede utilizar legalmente el cifrado fuerte internacional y el uso de este tipo de cifrado, como triple DES, asegura la privacidad completa de los datos.

Cuando se crea una GVPN, se pueden elegir cuatro modos:

- IPsec con ISAKMP: Este modo proporciona seguridad IPsec (cifrado fuerte para paquetes IP) y la habilidad de ISAKMP (Internet Security Association and Key Managment Protocol) para negociar dinámicamente las llaves para IPsec, proporcionando la más alta seguridad. Sólo puede ser usado si la otra parte correspondiente de la VPN soporta ISAKMP.

- IPsec con llaves estáticas: Este modo proporciona la seguridad IPsec utilizando llaves estáticas o predefinidas en la red. Esta no es una negociación dinámica de llaves. Este modo es recomendado sólo si la parte correspondiente de la VPN no soporta ISAKMP.

- Cliente ISAKMP: Permite crear una VPN entre el cliente y el firewall.

[25] El Protocolo de Manejo de Llaves y Asociación de Seguridad para Internet (ISAKMP por sus siglas en inglés), define una estructura para el manejo de seguridad y el establecimiento de llaves criptográficas para la internet. Este protocolo permite el uso de IPSec para proporcionar autenticación, integridad y/o confidencialidad de los paquetes IP enviados entre sistemas hosts o firewalls.

- swIPe: Este modo utiliza autenticación secreta estática y precompartida. Es proporcionada para compatibilidad con productos anteriores. Sólo se debe utilizar cuando el cliente no soporta Ipsec.

Puesto que la comunicación se realiza entre firewalls, éstos tienen opciones de confiabilidad dependiendo del tipo de administración (ya que en una misma organización puede haber más de un firewall o pueden ser diferentes organizaciones). Cuando se crea una VPN las opciones de implementación son:

- Una línea privada, que proporciona privacidad sin confiabilidad, es un enlace en el que los paquetes de datos son cifrados pero son manejados por proxies en el firewall. Si los paquetes son robados en el tránsito, el mensaje puede no significar nada para el espía que trata de usarlo porque no podrá ser descifrado propiamente. Además, cuando la información se descifra, el firewall la maneja como si viniese de una fuente no confiable.

- Enlace confiable. Es cifrado (asegurando la privacidad) y tratado como si viniera de una fuente confiable. Un enlace de este tipo sólo es apropiado cuando la otra red en la VPN está en la misma organización; esto es, comparten la misma administración, políticas de seguridad y se apeguen a los mismos procedimientos. Estos tipos de enlaces extienden el concepto de confiable para incluir no sólo los dispositivos dentro de la red interna, sino también para incluir todos los dispositivos de la otra red. En esencia se ha creado una gran red donde todos los dispositivos, no importa su ubicación, están dentro del mismo perímetro de seguridad. Por lo general **no** se recomiendan crear este tipo de enlaces, a menos que la otra red sea parte de la organización.

- Enlace Passthrough. Simplemente permite que los datos que son parte de la VPN (y que no van con destino a este firewall) pasen por él sin ser alterados. Esto sucede cuando en la VPN se ha definido un firewall intermedio.

Llaves Secretas Predefinidas Contra Basadas en Certificados

La autenticación en la VPN puede ser por llave secreta pre-definida, lo cual significa que los administradores que están creando la VPN llegan a un acuerdo para la elección de una llave aleatoria y se comunican esa llave por correo electrónico seguro, teléfono o un disco flexible. La llave puede ser cualquier combinación de letras y números y es deseable que no sea algo fácil de adivinar. La ventaja de esta opción es su facilidad de uso, pero la llave puede ser descubierta con el tiempo.

Se ha desarrollado un método basado en certificados para superar la desventaja del método anterior. El uso de certificados para autenticar significa que cada Firewall en la VPN debe tener un certificado de una entidad única en la que ambos confían.

Para complementar esto, el producto para implementar la VPN viene con un Administrador de Certificados (que se instala en una sola máquina en la organización) que crea, firma y controla los estándares basados en certificados de llave pública para la compañía. Esto permite a la empresa administrar su propia infraestructura en vez de depender de un servicio externo de Autoridad de Certificados.

Para propósitos de la seguridad de la red, se necesitarán crear certificados para todos los firewalls usados en la VPN e instalarlos en los mismos.

Funcionalidad de la VPN

El firewall maneja las VPNs examinando todo el tráfico de salida y cifrando todo el tráfico entre dos hosts que sean marcados como un par cifrado. La secuencia exacta de eventos varía dependiendo de la forma en que se haya configurado: privacidad con confianza o sólo privacidad.

Cuando el firewall esta a punto de enviar un paquete, valida si la fuente y el destino están listados en la tabla de pares cifrados. Si es el caso, el firewall envía el paquete al manejador de swIPe para aplicar el algoritmo de cifrado.

El driver de swIPe utiliza el Estándar de Cifrado de Datos (DES) para cifrarlos usando la llave proporcionada para esta VPN durante la configuración firewall-firewall. El nuevo paquete contiene datos cifrados y un encabezado indicando que tiene un protocolo especial. Finalmente, el firewall envía el paquete por la internet para el firewall de la otra red.

Cuando el firewall remoto lo recibe por la interfaz externa, la capa de IP lo reconoce como un paquete cifrado debido a que es un protocolo especial. Esta información indica que el firewall deberá enviar cualquier paquete de regreso, cifrado con este protocolo especial.

El driver de swIPe descifra los datos usando la misma llave usada para cifrarlo (en caso de llaves secretas). Ahora los datos ya en claro se pasan a la capa de IP y a partir de aquí, el paquete es manejado como cualquier otro.

Si la VPN entre las dos redes usa privacidad sin confianza, la capa de ruteo envía los paquetes al servicio apropiado o proxy. El proxy trata el paquete como si fuera cualquier otro de una red no confiable.

Verificación de Integridad

La base de datos de integridad de Gauntlet es un repositorio ASCII que contiene una colección de sumas de verificación (checksums[26]) criptográficos para los archivos en el sistema. Contiene un checksum para cada archivo e incluye información acerca del dueño, el grupo y modo.

Para verificar la integridad del sistema, se puede crear una base de datos de checksums y compararla con los valores de la existente. Cambios en el checksum de un archivo indican que alguien o algo lo ha modificado de alguna manera.

[26] Es un valor calculado el cual depende del contenido de un bloque de datos. Para realizar verificaciones, este valor vuelve a calcularse y se compara contra el que se tiene almacenado, si son los mismos se asume que no se han realizado modificaciones.

La base de datos no contiene información acerca de archivos que pueden cambiar seguido, tales como los archivos de log.

Funcionalidad del Firewall

La forma de procesar del firewall sigue un conjunto estándar de pasos para cada paquete que recibe:

1. Recepción del paquete
2. Verificación de fuente y destino
3. Tipo de petición
4. Procesamiento de la petición

Recepción del Paquete

La información de ruteo en los hosts externos dirige todas las peticiones hacia la/las interfaces externas del firewall ya que es la única manera de alcanzar los hosts internos. Los equipos en la red interna que quieran comunicarse hacia el exterior, dirigen todos sus paquetes a la interfaz interna del firewall.

Verificación de Fuente y Destino

Una vez que el firewall recibe un paquete, debe decidir que hacer con él. Primero, el kernel del sistema operativo examina el destino y determina si necesita entregarlo localmente. La entrega local incluye los paquetes destinados a los hosts internos. El firewall los absorbe y los redirecciona al proxy apropiado. Si no hay proxies configurados para aceptar tal tipo de paquete, el firewall lo deshecha y registra el acceso fallido.

Tipo de Petición

Ahora que el firewall sabe que el paquete se entrega localmente, verifica su contenido. El sistema operativo examina los datos de configuración en el firewall para determinar si ofrece los servicios en el puerto requerido. Si no, registra el intento como una alerta de amenaza potencial de seguridad y rechaza la petición.

Procesamiento de la Petición

El proxy del servicio procesa la petición como lo haría la aplicación estándar, siguiendo el mismo protocolo. Primero valida la información de configuración. El proxy determina cómo tratar al paquete basándose en la dirección IP fuente y verifica que la petición este permitida para conectarse al host destino (para algunos servicios, los proxies pueden realizar un paso adicional de autenticación del usuario); finalmente el proxy pasa la petición al programa apropiado del otro lado del firewall usando el protocolo estándar para ese servicio.

5.3 Necesidades de Seguridad

Las aplicaciones de la Nómina están desarrolladas sobre una arquitectura Cliente/Servidor en la que los servicios son utilizados por otras dependencias en lugares físicamente distantes. La información tiene que viajar por una red sobre la que no se tiene control (internet) por lo que la posibilidad de ataques es muy alta por las siguientes razones:

- La información que viaja por la red viaja en claro, es decir, si alguien se ubica entre los puntos de comunicación de Nómina y un usuario de una Sucursal puede observar la información que esta viajando, de que servidor proviene y a que cliente va, con lo cual se pierde la **confidencialidad** necesaria de la información.

- Como los servidores tienen contacto directo con los clientes, es posible explotar alguna vulnerabilidad del sistema operativo o dejar fuera de servicio al equipo inundándolo de tráfico. Si el intruso lograra entrar, podría modificar información del equipo (archivos del sistema, programas, el software de Sybase, etc.) con lo que, a pesar de que se puede detectar que hubo modificación, el **tiempo a invertir** para restaurar el sistema puede ser impredecible y, si borra información de bitácoras del sistema, puede ser imposible localizar el origen del acceso. En términos de seguridad se puede llegar a perder la **auditoría**, la **integridad** de la información e inclusive la **disponibilidad** de los servicios.

- Las herramientas del sistema operativo no proporcionan un registro completo de la actividad de los usuarios, ni permiten restringir conexiones externas, a excepción de los servicios controlados por inetd (con tcp-wrappers). Otros servicios no controlados por este demonio son los de Sybase, éste último es particularmente importante. Si alguien entra en Sybase puede borrar tablas (de usuario o de sistema), borrar bases de datos, modificar información, con lo cual se podrían tener **inconsistencias**.

- Actualmente, las bitácoras de los equipos han reportado intentos de conexión fallidos desde otros hosts totalmente ajenos a la Nómina. Esto es una desventaja ya que están teniendo contacto directo con los servidores, lo que permite conocer todas las posibles puertas de entrada y vulnerabilidades de los mismos.

Las necesidades de seguridad, entonces, se obtienen de los puntos anteriores y se solucionan en los siguientes:

- Aplicar el principio de menor privilegio, es decir, los usuarios deben utilizar los servicios necesarios para desarrollar su trabajo de nómina, pero no más.

- Evitar el contacto directo de conexiones externas con hosts internos para evitar la explotación de las debilidades de los equipos.

- Tener un punto único de acceso que permite monitorear el tráfico que viaja hacia los equipos servidores y que este punto de acceso sea difícil de comprometer. Esto permitirá también tener un control preciso sobre políticas de acceso, de servicios y de auditoría.

- Implementar una autenticación de usuario más segura en los servicios de red que lo requieran.

- Cifrar la información que viaje fuera de la red interna.

- Mantener ocultas las direcciones IP de los equipos de la red interna.

Las necesidades de Seguridad se pueden cubrir con el Firewall Gauntlet de Network Associates Inc., el cual va a funcionar como un filtro entre la Internet y los equipos de Nómina. No se intenta proteger toda la red de la casa Matriz, sólo se incluyen los equipos utilizados directamente para la operación de Nómina junto con los servidores (tanto de desarrollo como para producción) así como los servidores de impresión y se excluyen los equipos personales para desarrollo.

Los servicios que se habilitarán en el firewall serán los que se utilicen únicamente para el manejo de la nómina.

Los requisitos para instalar el firewall se listan a continuación y se analizan con más detalle en las secciones siguientes:

- Identificación de Usuarios y Servicios Utilizados: Se analizarán los usuarios que interactúan con la Nómina y se organizarán en grupos. Además, se analizarán los servicios utilizados y se resumirán en servicios utilizados por grupo de usuarios.

- Red Segura y Ubicación del Firewall: Se analizan y esquematizan los puntos de acceso a la red de la casa Matriz, los equipos que se quieren proteger y su justificación.

- VPN: Es un análisis y configuración de la Red Global Segura entre las dependencias que comparten información con la Nómina.

5.4 Servicios de Red

Hay varios servicios estándar que los usuarios utilizan para comunicarse desde o hacia la red de Nómina y que se necesitan para la operación diaria de la misma. Existen razones importantes para utilizar tales servicios; de hecho, sin ellos hay pocas razones para conectarse por red. Pero también existen problemas de seguridad potenciales con cada uno de ellos. Ninguno de estos servicios es, en realidad, seguro; cada uno tiene sus propias debilidades y cada uno ha sido explotado de varias formas por sus propios atacantes. Antes de decidir soportar un servicio se debe evaluar qué tan importante es para los usuarios y si se podrá proteger de sus peligros. Hay varias formas de hacerlo: ejecutar los servicios sólo en ciertas máquinas protegidas, empleando variaciones especialmente seguras de los servicios estándar o, en algunos casos, bloquear los servicios por completo desde o hacia todos los sistemas externos. Es importante conocer un poco más a detalle los servicios que se utilizan en Nómina. La siguiente sección los analiza desde el punto de vista de seguridad.

Servicios y Vulnerabilidades

Correo Electrónico

Es uno de los servicios de redes más populares y básicos. Es de riesgo relativamente bajo pero eso no significa que este libre de riesgos. Falsificar correo electrónico es sencillo (como lo es falsificar el correo postal normal) y las falsificaciones generan dos tipos de ataques: contra la reputación de la organización y de manipulación social (por poner un caso, ataques en los que los usuarios envían correo que se supone viene de un administrador, aconsejándoles a todos que cambien su contraseña de forma específica). Aceptar correo electrónico ocupa tiempo y espacio, lo que puede propiciar ataques de negación de servicio; con una configuración adecuada, sólo se negará el servicio de correo electrónico. En particular con sistemas modernos de correo multimedia, las personas pueden enviar correo electrónico que contenga programas que, si se ejecutan con supervisión insuficiente, pueden resultar en *caballos de Troya*[27].

Aunque la gente se preocupa más sobre el último riesgo mencionado, en la práctica los problemas más comunes son inundaciones inadvertidas (incluyendo cartas cadena) y personas que confían plenamente en la confidencialidad de este servicio envían datos sensibles a través de él.

[27] Es un programa malicioso que rompe la seguridad del sistema y que por lo general reemplaza a un programa normal. Puede instalarse para listar directorios, adivinar passwords, borrar archivos, explotar alguna debilidad del sistema operativo para tener acceso a la cuenta de root, etc.

Con relación a este servicio, el equipo Unix de Nómina no recibe correo, sólo sirve para enviar resultados de Nómina a dependencias externas.

Transferencia de Archivos (FTP)

El correo electrónico transfiere datos de un lugar a otro, pero esta diseñado para archivos pequeños legibles para las personas. Los protocolos para la transferencia de correo electrónico tienen permitido hacer cambios a un mensaje que son aceptables para las personas (por ejemplo, insertar el signo ">" antes de la palabra "from" al principio de una línea, para que quien envía el mensaje no se confunda con una línea de encabezado), pero que no lo son para los programas.

Aunque los sistemas de correo electrónico actuales incluyen algoritmos elaborados para tales problemas, de tal forma que un archivo binario puede dividirse en piezas pequeñas y codificarse en el extremo que envía y volverlo a ensamblar en el que recibe, estos algoritmos son engorrosos y propensos a errores. Además, las personas quizá quieran salir y buscar de manera activa los archivos, en lugar de esperar a que alguien los envíe. Por lo tanto, aun cuando el correo electrónico está disponible es útil tener un método diseñado para transferir archivos al solicitarlos.

FTP es el protocolo estándar para este propósito. En teoría, permitir que los usuarios obtengan archivos no incrementa más el riesgo que permitir el correo electrónico; de hecho, algunos sitios ofrecen servicios que permiten tener acceso a FTP por medio de correo electrónico. En la práctica, sin embargo, las personas realizan más transferencias de archivos cuando FTP está disponible.

La Nómina utiliza mucho este servicio para compartir datos con otras dependencias.

El servicio de FTP debe estar restringido debido a que pueden obtener programas y datos indeseables. Lo que los hace indeseables es que pueden llegar a consumir grandes cantidades de espacio o puede ser software tipo caballos de Troya.

El Protocolo Trivial para la Transferencia de Archivos (TFTP) es un protocolo FTP simplificado que las máquinas diskless[28] utilizan para transferir información. Es en extremo sencillo integrarlo al hardware y, por lo tanto, no soporta ninguna autenticación. No hay razón para proporcionar acceso TFTP fuera de la red; los usuarios de Nómina no transfieren archivos con este protocolo ni existen máquina diskless.

Terminal Remota (telnet) y Ejecución de Comandos

Los programas que proporcionan acceso de terminal remota, permiten que se utilice un sistema remoto como si la máquina fuera una terminal conectada directamente.

[28] Es una forma de configurar una estación de trabajo UNIX que no tiene disco de arranque y busca el sistema operativo en otra máquina

Telnet es el estándar para acceso remoto que imita una terminal (no es una estación de trabajo gráfica) y proporciona acceso sólo a aplicaciones basadas en caracteres.

Telnet se consideró en un tiempo un servicio más o menos seguro porque requiere que los usuarios se autentiquen por ellos mismos. Por desgracia, telnet envía toda su información sin codificar, lo que lo hace muy vulnerable a ataques de espionaje (utilizando programas analizadores de protocolos) y robo. Por esta razón, ahora telnet se considera uno de los servicios más peligrosos cuando se utiliza para entrar a un sitio desde sistemas remotos. Telnet es seguro sólo si la máquina remota y todas las redes entre ella y la máquina local son seguras, lo cual significa que no es seguro a través de internet, donde no podemos identificar con certeza las redes que intervienen y mucho menos confiar en ellas.

Hay varios tipos de esquemas para dar autenticación al iniciar una sesión en un sistema remoto, pero aunque la autenticación protege la contraseña, aun así la sesión puede ser intervenida o robada.

Existen programas, además de telnet, que pueden usarse para tener acceso como terminal remota y ejecución de programas (*rlogin*, *rsh* y *on*). Estos programas se utilizan en un ambiente confiable para permitir que los usuarios tengan acceso remoto sin que deban autenticarse nuevamente. El host al que se conectan confía en que el sistema solicitante ha dado autenticación al usuario en forma correcta. Este modelo es poco seguro ya que no puede confiarse en los hosts que estén fuera de la red local.

Protocolo de Transferencia de Hipertexto (HTTP)

El correo, FTP y Telnet han existido desde los primeros días de internet; en realidad, son extensiones de servicios proporcionados mucho antes de que existiera esa red. El World Wide Web (www) es un concepto nuevo, basado totalmente en internet y, en parte, en servicios existentes y en un protocolo nuevo: el protocolo HTTP.

El Web utiliza tecnología de hipertexto para enlazar una gran cantidad de documentos que pueden incluir texto, imágenes, sonido, video y otros formatos. Se puede "navegar" por estos documentos de cualquier manera (no sólo jerárquicamente) para buscar información. El hipertexto proporciona la posibilidad de ir de un documento a otro sin importar en donde se encuentran almacenados para lo cual se define una "liga" HTTP.

Por desgracia, los servidores Web son difíciles de asegurar. La utilidad del Web se basa, en gran medida, en su flexibilidad, pero ésta dificulta su control. Así como es más fácil transferir y ejecutar el programa correcto utilizando un navegador Web que por medio de FTP, es más fácil transferir y ejecutar un programa peligroso.

Los usuarios de Nómina que se encuentran detrás del firewall podrán utilizar el Web, pero no se deberá consultar información mediante un navegador desde los servidores de Nómina, excepto del que está destinado para ello, que es una máquina que no tiene información importante.

Sybase

El DBMS Adaptive Server de Sybase es el núcleo de todo el sistema de Nómina (desde el punto de vista de servicios). Otros DBMS remotos se comunican con el DBMS local utilizando el protocolo de Flujo de Datos Tabulares (TDS) que utiliza Adaptive Server sobre TCP/IP para viajar por la red. Los DBMS's tienen la capacidad de cifrar los passwords de los usuarios cuando viajan por la red así como la autenticación en la comunicación de los servicios cliente-servidor (como la replicación de datos y la ejecución de comandos entre DBMS's). Quizá su única desventaja es que la información que transfiere es legible por lo que puede ser observada en su tránsito por la red.

Identificación de Usuarios y Servicios Utilizados

La Nómina comparte información con varias dependencias, tal información se distribuye utilizando servicios de red o de forma impresa. Para nuestro estudio nos interesa aquella información que se comparte por medio de servicios de red.

Las dependencias con las que interactua el sistema de Nómina son: Dirección de Recursos Humanos (DRH), Dirección Asuntos del Personal (DAP), Dirección de Planeación (DP) y con la misma Casa Matriz (CM). Cada una comparte información con la Nómina de diferente forma, ya sea proporcionando datos o recibiéndolos. En la siguiente figura se ilustran las dependencias que proporcionan entradas y las que reciben salidas.

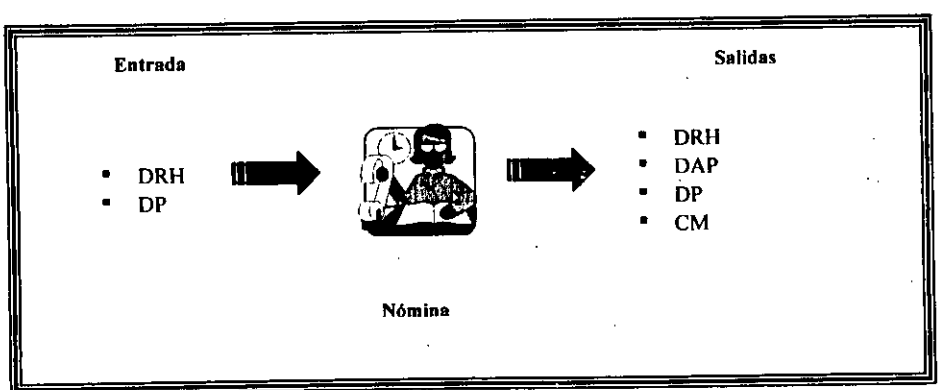

Fig. 5.5 Dependencias con las que se relaciona la Nómina

Las dependencias proporcionan entradas directamente al servidor de producción. Las que obtienen las salidas transfieren o consultan directamente sus datos en el mismo servidor, es decir, todas las dependencias se enlazan al equipo de Nómina.

Grupos de Hosts y Grupos de Servicios

El firewall tiene la facilidad de configurar grupos de hosts para los que se aplican restricciones o se asignan derechos de uso de servicios similares. Para aprovechar esta facilidad, se han definido grupos de hosts, que se determinaron basándose en su ubicación (ya que cada dependencia tiene, por lo general, asignado un segmento de red) y los servicios que utilizan, cumpliendo siempre con el principio de menor privilegio (proporcionar a los usuarios únicamente los servicios permitidos).

Los grupos pueden pertenecer a la red interna o la red externa. Se considera red interna a la red que se quiere proteger, en este caso la parte de red donde se aisla la Nómina y todas las demás redes son externas y no confiables. No confiables significa que para toda petición de servicio proveniente de cualquier red, necesitará autenticarse para poder entrar a los hosts internos.

Grupos de Hosts Externos

INTERNET (UNTRUSTED)

Comprende a todos los hosts que no conocemos, es decir, toda la internet. Este grupo lo trae declarado por default el firewall porque puede suceder que al revisar todas las reglas de acceso no encuentre registrado como válido al host que intenta conectarse; al no encontrarlo lo identifica como perteneciente a esta red y, por ende, le niega el acceso.

NOMADM

Es el grupo de administración, puesto que las conexiones que establezca este grupo tiene que ver con configuración del equipo, cambios de passwords, transferencia de información de la base de datos, configuración del Servidor de Bases de Datos y monitoreo de los equipos, se necesitan restricciones más estrictas como el utilizar passwords de única vez.

DESA

Este grupo engloba a todos los equipos de desarrollo y mantenimiento de la Nueva Nómina, se encuentran en la red local de la CM pero fuera de la red confiable (fig. 5.6). Utilizan servicios de Sybase Adaptive Server, telnet así como ftp y requieren autenticación de una sola vez.

DRH

Aquí se agrupan los usuarios de DRH que se comunican con la nómina. Se encuentran en la red externa (Red de DRH). Este grupo se divide en dos subgrupos.

DRHSYB

Agrupa a los hosts para consultas, captura y actualización de información de empleados.

DRH

Se refiere a los hosts que realizan transferencias de información de la base de datos a archivos planos y que son de utilidad para otras dependencias.

DP

Agrupa a los usuarios que se conectan desde DP, también es una red externa a la que se transfiere información de plazas, resultados de nómina y los movimientos del personal.

CM

Son los usuarios del Sistema de Personal en CM que necesitan conectarse al servidor de producción de la Nómina para la transferencia de movimientos de empleados.

Grupos de Hosts Internos

Los siguientes grupos de hosts internamente no tienen ninguna restricción en el uso de servicios. Sólo se restringen los servicios hacia el exterior.

NOMINA

Aquí se agrupan todas las máquinas cliente (PC's) que se relacionan directamente con la producción de la nómina y la ejecución de procesos locales.

MANTENIMIENTO

Se refiere a la máquina que se utiliza para bajar parches de la internet, software de actualización, para monitoreo interno o para pruebas en general.

SERVPROD

Se refiere al servidor de Producción. No utiliza ningún servicio fuera de la red interna.

SERVDES

Agrupa a los servidores de Desarrollo. No utilizan ningún servicio fuera de la red interna.

La red de la CM, que es donde se encuentra la Nómina, no va a estar completamente protegida. Se definió una parte de la red (en donde se van a localizar los servidores y estaciones de trabajo para la producción de Nómina) como confiable y el resto queda sin protección y es allí donde van a estar los grupos de *NOMADM* y *DESA* (Fig.5.6).

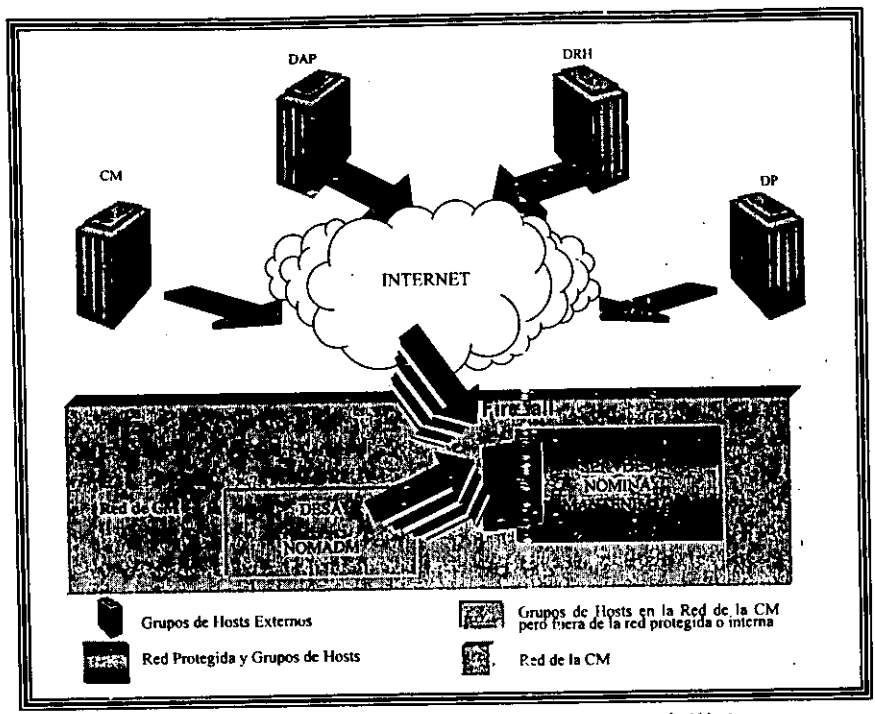

Fig. 5.6 Ubicación de los grupos de usuarios que interactuan con la Nómina

Grupos de Servicios

Un grupo de servicios es simplemente una colección de servicios que se definen como una unidad y se aplican contra un grupo de hosts. Sirven para evitar el estar asignando, al momento de configurar el firewall, un conjunto de servicios individuales para un grupo, se pueden crear grupos de servicios que vayan hacia el mismo destino o que tengan el mismo origen y que faciliten la incorporación de nuevos elementos, restringiéndolos al grupo deseado de hosts.

Se han creado grupos de servicios con el mismo nombre del grupo de hosts, como se explica a continuación:

NOMADM

Engloba servicios de Sybase telnet y ftp hacia la red interna con paswords de única vez (para ftp y telnet).

DESA

Comprende servicios de Sybase, telnet así como ftp y requieren passwords de una sola vez (para ftp y telnet).

DRHSYB

Utilizan sólo servicios de Adaptive Server (Sybase) para consultas, captura y actualización de información de empleados. Puesto que el proxy de Sybase no soporta autenticación, su acceso es transparente.

DRH

Utilizan ftp, telnet y Sybase para la transferencia de resultados de los movimientos del día a archivos de reportes para dependencias como DP. Utilizan passwords de única vez.

DP

Sólo contiene ftp con passwords de una sola vez.

CM

El servicio que soporta es Sybase.

NOMINA

Los servicios que permite hacia el exterior son: http, telnet, ftp y correo electrónico.

MANTENIMIENTO

Engloba servicios hacia el exterior de http, ftp, telnet y SMTP.

En la siguiente figura se puede observar que los servicios que consultan máquinas externas son sólo los necesarios.

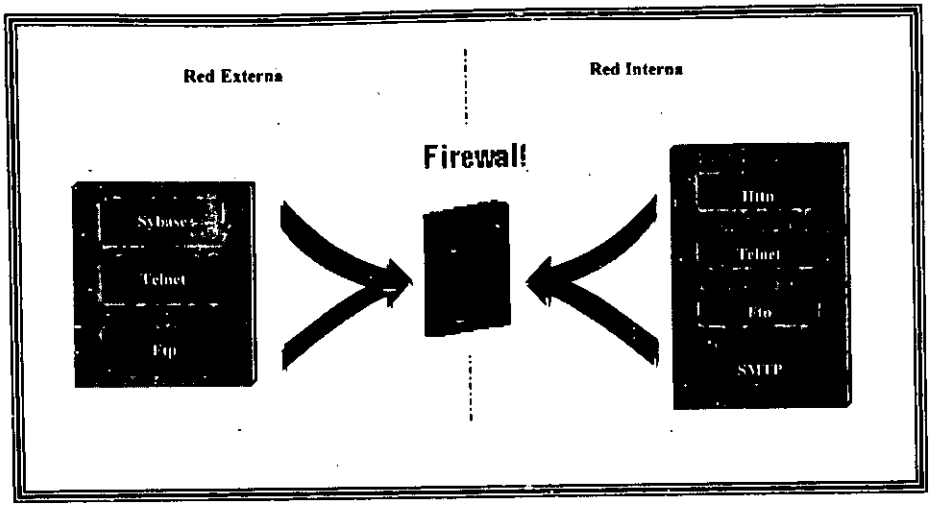

Fig 5.7 Servicios Permitidos de fuera hacia adentro y de dentro hacia fuera en Nómina

Hasta aquí se han analizado los grupos de usuarios externos e internos que conforman el sistema de Nómina así como los servicios que cada uno requiere. En la siguiente tabla se muestran los servicios que cada grupo de hosts utiliza y también se muestran los destinos que cada grupo de hosts puede alcanzar. Por ejemplo, el grupo de hosts DP (en los encabezados de las filas) puede utilizar telnet y ftp para entrar a los servidores de Producción (Grupo ServProd en los encabezados de columnas).

Grupo de Hosts	Ubicació n	Internet	NomAdm	Desa	DRHSyb	DRH	DP	DAP	CM	Manten	Nomina	ServProd	ServDes
Internet	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
NomAdm	Externo	NA	NA	NA	NA	NA	NA	NA	NA	telnet ftp	NA	telnet ftp Sybase	telnet ftp Sybase
Desa	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	telnet ftp Sybase
DRHSyb	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	Sybase	NA
DRH	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	ftp telnet	NA
DP	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	telnet ftp	NA
DAP	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	ftp	NA
CM	Externo	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	Sybase	NA
Manten	Interno	Http Telnet Ftp SMTP	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
Nomina	Interno	Http Telnet Ftp SMTP	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
ServProd	Interno	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
ServDes	Interno	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA

Tabla 1 Política de Seguridad en Servicios que utilizan los Grupos de Hosts

5.5 Ubicación Física del Firewall y Red Segura

El firewall debe estar físicamente seguro, una buena ubicación física es el lugar donde se encuentran localizados los equipos servidores de Nómina ya que cuenta con mayores restricciones para poder llegar a ellos y condiciones ambientales adecuadas. La siguiente es una lista de las características físicas del sitio:

- El acceso a la Sala de Cómputo está restringido
- El acceso al cancel donde se encuentra los servidores también esta restringido
- Existe control de Temperatura
- Existe control de Humedad
- Hay extintores de fuego en caso de incendio
- Se encuentra instalada una alarma contra humo
- La corriente es regulada y con protección de sobrecarga
- La línea de corriente está conectada a un UPS

Estas características hacen al sitio confiable para el buen desempeño de la actividad del firewall.

En cuanto a la red segura, como se dijo en párrafos anteriores, la red de la CM no va a estar totalmente protegida, sólo aquella sección que directamente utiliza los servicios de Nómina en Producción. En el siguiente esquema se observa a detalle la estructura de la red protegida.

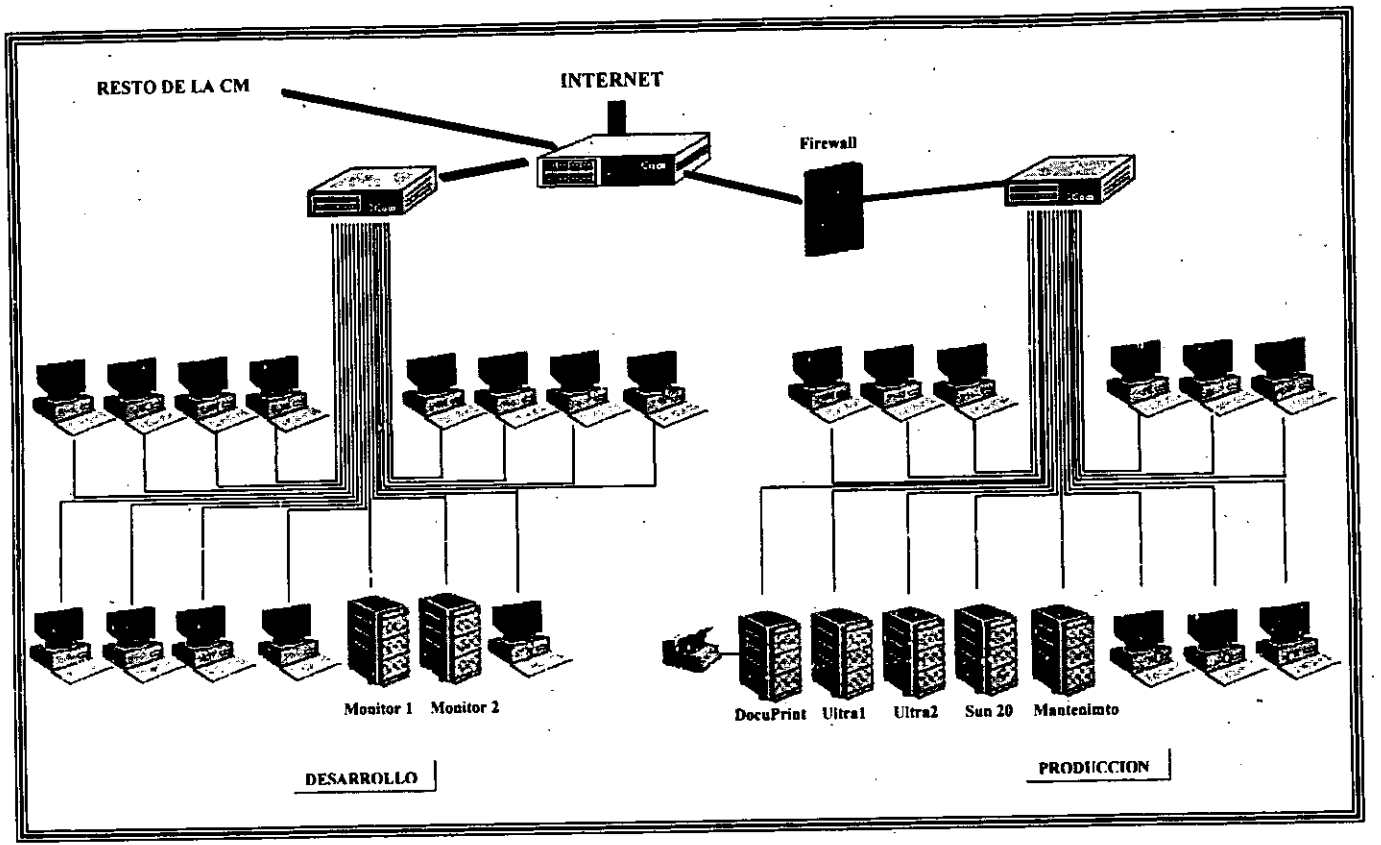

Fig 5.8 Estructura de Red de Nómina

Esquema de la Red Global (VPN)

La comunicación que fluye hacia las dependencias externas debe codificarse para protegerla de ataques pasivos ya que, como se comento en la sección de vulnerabilidades de los servicios, todos ellos envían la información por la red en claro.

En las comunicaciones que habrá con la DRH el cifrado es de firewall a firewall utilizando enlaces privados (no confiables) para seguir manteniendo las restricciones de servicios y hosts en cada una de las redes confiables. En las demás dependencias, se instalará el software para cifrado directamente en las PC's que van a comunicarse con la nómina ya que éstas dependencias no tienen firewalls instalados.

Se utilizará un mecanismo de manejo de llaves públicas con certificados para garantizar una mayor confiabilidad de la información ya que, en este caso, ni siquiera los administradores de los firewalls podrán conocer las llaves que se utilizarán en el intercambio de información (como sucede cuando se utiliza un mecanismo de llaves secretas).

En la siguiente figura se esquematizan las comunicaciones en la VPN.

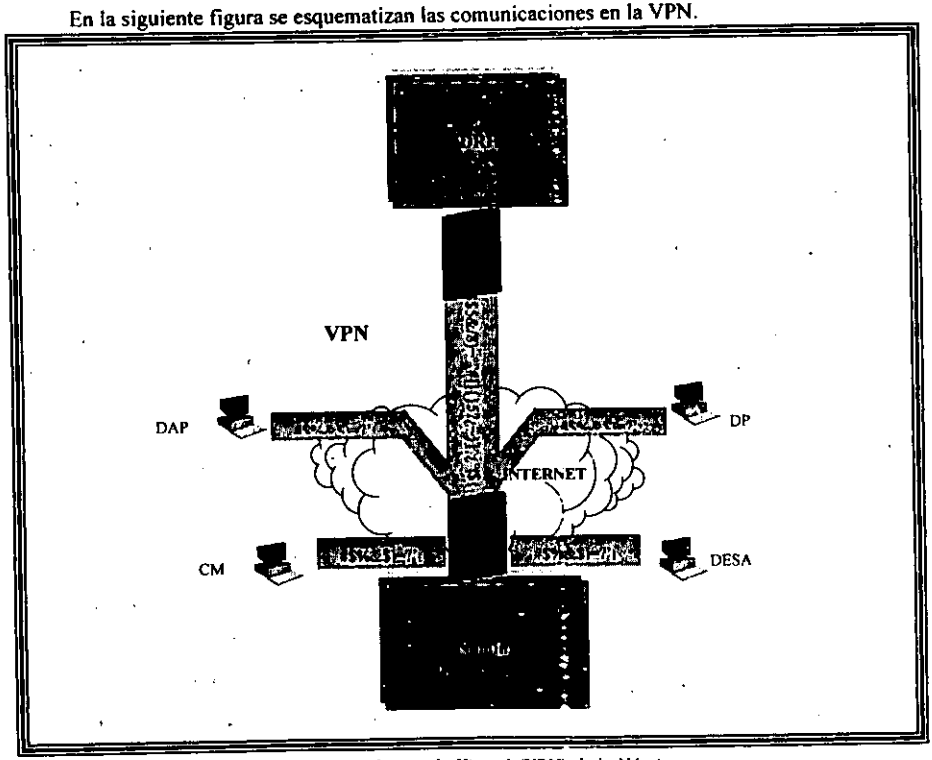

Fig 5.9 Red Privada Virtual (VPN) de la Nómina

5.6 Administración del Firewall

Una vez que el firewall ha sido configurado e instalado, se entra en un ciclo de mantenimiento del mismo. Básicamente se divide en tres grandes grupos: Administración, Monitoreo y Actualizaciones.

Administración

En la administración se necesitan realizar las siguientes tareas:

Respaldos del Firewall y Verificación de Integridad

El propósito de un respaldo es el mantener fuera de línea información que sea difícil de recuperar en el caso de una falla de discos. Para el firewall, la parte sensible a una falla es su configuración, por lo que es necesario que, al momento de instalarlo, se tenga un respaldo total tanto del sistema operativo como del software y configuración del firewall.

Debido a que no se modifica nada a nivel sistema operativo, sino que todo se controla a partir de las configuraciones del firewall, posteriormente será necesario respaldar únicamente la configuración cuando ésta sufra modificaciones.

El respaldo inicial de sistema operativo, software del firewall y configuración deberá hacerse a cinta. Las modificaciones posteriores a la configuración, se pueden respaldar en un disco flexible desde la consola de administración del firewall.

Las bitácoras se deberán respaldar cuando se depuren y mantenerlas fuera de línea como un histórico.

Se debe verificar la integridad del sistema al menos una vez al día para detectar si hay algún indicio de violación de la seguridad o de modificación de archivos sin autorización. Para realizar esta verificación, se deben utilizar dos herramientas: una que verifique los archivos del firewall, la cual viene integrada en la consola de administración del mismo; y otra que verifica la integridad de los archivos del sistema operativo, utilizando tripwire.

Administración de Cuentas

El mantenimiento de las cuentas (agregar nuevas cuentas, quitar viejas, caducar las contraseñas, etc.) es una de las tareas de mantenimiento que se descuidan frecuentemente. En los sistemas firewall es del todo crucial que las nuevas cuentas se agreguen **correctamente**, que las viejas se quiten con oportunidad y que las contraseñas se cambien de modo apropiado. El

procedimiento para agregar, quitar y bloquear cuentas se documenta en las Políticas de Seguridad.

Administración de Espacio en Disco

En los firewalls, las bitácoras son el problema por la demanda de espacio en disco. Es necesario vigilar el crecimiento de la partición asignada a las bitácoras (/var) y respaldarlas para posteriormente depurarlas. La tarea de depurarlas se puede configurar en el firewall, dependiendo de cuanta información se genera es como se debe ir afinando este parámetro. Inicialmente, se puede configurar a un mes y conforme se vaya conociendo el volumen de registros, se irá adecuando al periodo más conveniente.

La configuración de espacio en disco para el firewall es:

/	150 MB
swap	150 MB
/usr	800 MB
/opt	600 MB
/var	**2000 MB**
/space	401 MB

Como se podrá ver, se previene un gran crecimiento de las bitácoras donde se registran los eventos, por lo que se asigna un área de 2 GB para ese propósito.

Mantenimiento Preventivo y Correctivo

Finalmente, un firewall reside sobre un hardware que, como todos los equipos, requiere de un mantenimiento preventivo. Afortunadamente, existen periodos vacacionales en donde la actividad por red es baja o nula y estos periodos se deben aprovechar para darle mantenimiento.

Es recomendable tener un equipo de respaldo con las mismas características y configuración para que, en caso de mantenimientos correctivos o actualizaciones urgentes, supla al firewall y no se tengan o se eviten periodos de tiempo largos fuera de la red.

Monitoreo del Sistema

El monitoreo sirve para indicar varias cosas:

- ¿ Está comprometido el firewall ?
- ¿ Qué clases de ataques se han intentado contra él ?
- ¿ Funciona adecuadamente ?

- ¿ Está proporcionando los servicios que los usuarios necesitan ?

Para responder a estas preguntas se debe conocer cuál es el patrón de uso normal.

¿Qué se Debe Observar?

El firewall registra todos los tipos de eventos que suceden en las conexiones como: hora de conexión, duración de la conexión, intentos exitosos o fallidos de acceso, comandos utilizados, por lo que puede ser abrumador el tratar de revisar toda la información que genera. Para lograr un compromiso práctico, se deben habilitar los eventos de seguridad que no carguen demasiado a la máquina y que no llenen demasiado rápido el disco; luego se deben resumir los registros que se producen.

El mecanismo de monitoreo del firewall permite configurar la frecuencia con la que desea generar un reporte, a que cuenta de correo irá dirigido y qué tipo de mensajes interesa revisar.

Diariamente se deben revisar las bitácoras, ya que entre más rápido estemos enterados de patrones sospechosos de violación o problemas con el firewall, más pronta será la solución. Recordemos que a una máquina puede fallarle un área de disco en cuestión de días o aún peor, un intruso puede violar la seguridad en cuestión de horas, por lo que dependiendo del momento en que se detecte el problema, es el tiempo que se le invertirá en recuperar la información.

En particular se deben registrar los siguientes casos:

- Todos los paquetes rechazados, conexiones negadas e intentos frustados
- Al menos la hora, el protocolo y nombre del usuario de cada conexión exitosa hacia o a través del firewall
- Todos los mensajes de error de cualquier programa proxy o del sistema operativo

Actualizaciones

El último aspecto importante del mantenimiento de los firewalls tiene que ver con estar actualizado. Es obvio que debe mantenerse actualizado el sistema, pero antes de hacerlo, el mismo administrador debe estar actualizado. Todos los días ocurren cosas nuevas, se descubren y explotan nuevos errores, se llevan a cabo ataques nuevos, están disponibles más accesorios, etc. Para ello es bueno subscribirse a listas de discusión y estar siempre involucrado con el fabricante (Network Associates Inc.) en lo que respecta a nuevas actualizaciones y parches. Una lista de discusión recomendable es *greatarticle.com* donde se discuten temas relacionados a todo tipo de firewalls y sus configuraciones y *bugtraq* que es una lista de discusión enfocada a discutir vulnerabilidades y bugs en todo tipo de software. En la UNAM existe una lista de discusión orientada a la seguridad (*gasu*) que también puede ser de ayuda en temas relacionados a

configuración de servicios, reportes de problemas en el software de sistemas operativos y temas de seguridad en general. Finalmente, la lista de discusión orientada totalmente a problemas de Gauntlet es *gauntlet-user*. El administrador debe estar suscrito a esta lista para conocer todo lo relacionado a nuevas vulnerabilidades, actualizaciones, parches, bugs, problemas y soluciones del firewall.

Si el administrador se mantiene actualizado, entonces mantener el firewall actualizado es una labor sencilla. Sólo deben estudiarse los problemas que involucren alguno de los servicios que se están ofreciendo, de allí la importancia de permitir sólo aquellos servicios que sean necesarios. Si después del estudio realizado se determina que es necesario actualizar la versión del firewall, se deberá realizar en un equipo de respaldo (previniendo que tal software tuviera algún bug o cambio en la configuración) y posteriormente en el que se encuentra funcionando.

Conclusiones

Esta propuesta tiene un enfoque práctico que busca aplicar las herramientas en cada nivel de seguridad de la Nómina. Se aprovecharon las características de seguridad que ofrecen el sistema operativo así como el manejador de bases de datos y se propuso la implementación de un firewall para proteger y controlar los servicios de red. En lo que respecta a la seguridad física, se hicieron algunas propuestas para reforzar la seguridad que ya existe en el sitio.

Estas sugerencias permiten tener un mejor control en los servicios que ofrece la Nómina y se "reduce" el riesgo de un ataque en un porcentaje muy alto al limitar al atacante a no tener contacto directo con los servidores y, en su lugar, poner un firewall especializado para recibir ataques como la cara externa de la empresa. Aun cuando un usuario válido intentara utilizar un servicio que no le corresponde, la seguridad del sistema operativo no se lo permite al deshabilitar servicios no necesarios y establecer permisos restringidos en los directorios. También en la base de datos se establecen restricciones para la consulta o modificación de información. Por último, la seguridad Física no permite que cualquier persona tenga contacto físico con el equipo crítico.

A pesar de este esquema de seguridad, es necesario admitir que la Nómina no esta totalmente segura, diariamente se descubren nuevos huecos de seguridad y nuevas formas de entrar a los sistemas. Esto conlleva a un ciclo sin fin en el que se descubre una nueva vulnerabilidad y hay que cubrirla lo más pronto posible. Por eso, al final de los capítulos, se enuncian listas de discusión donde se publican estas nuevas vulnerabilidades para estar al tanto de los problemas a los que se puede enfrentar el administrador y poder buscar los parches correspondientes en los sitios que proporciona el fabricante (Sun, Sybase o NAI).

También se debe resaltar la participación muy importante que tiene un usuario para detectar problemas de seguridad. Se deben concientizar para el buen uso y elección de un password y sobre las prácticas recomendables para mantener la seguridad. Sin su concientización, cualquier esquema de seguridad muy probablemente, será violado por los mismos usuarios internos.

Glosario

Cache: Una sección reservada de la memoria que se utiliza para mejorar el rendimiento. Un cache de disco es una porción reservada de la memoria normal, o memoria adicional en la tarjeta controladora del disco. Cuando el disco es leído, se copia un gran bloque de datos en el cache. Si los requerimientos de datos subsiguientes pueden ser satisfechos por el cache, no se necesita el empleo de un acceso a disco que es más lento. Si el cache es utilizado para escritura, los datos se alinean en memoria y se graban en el disco en bloques más grandes.

Consola: Es la terminal principal del equipo. Normalmente se le llama así al monitor que está físicamente conectado al servidor.

Cracker: El individuo que trata de ganar acceso no autorizado a una computadora. Estos individuos, por lo general, tienen intenciones maliciosas y tienen muchos medios a su disposición para romper la seguridad de un sistema.

DAT: (Digital Audio Tape) Cinta Audio Digital. Tecnología de grabación digital de calidad CD para cintas magnéticas. Unidad DAT de 4mm. de exploración helicoidal que contiene 1.3GB (hasta 2GB con cintas de extensión ampliable cuando se adapta para utilizarlo como almacenamiento de datos).

Demonio: (Daemon). Programa que espera en un segundo plano (background) preparado para actuar en el momento en que aparezca algún acontecimiento. Procede de la mitología griega y significa "espíritu guardián".

Directorio Home: Se dice del directorio donde se posiciona por defecto un usuario de Unix al momento de ingresar al sistema.

EEPROM: (Electrical y Erasable Progammable Read Only Memory) memoria de sólo lectura programable y borrable eléctricamente. Un chip de memoria que retiene su contenido sin potencia. Puede ser borrado, tanto dentro de la computadora como externamente, y usualmente requiere más voltaje para el borrado que él común de +5 voltios usado en los circuitos lógicos. Funciona como RAM no volátil, pero grabar en EEPROM es mucho más lento que grabar en RAM. Las EEPROM son usadas en dispositivos que deben mantener datos al día sin potencia. Por ejemplo, en una terminal de punto de venta que está apagada por la noche. Cuando los precios cambian, la EEPROM pueden actualizarse desde una computadora central durante el día.

EUID: Es el Identificador de Usuarios Efectivo. En Unix un usuario puede tener dos tipos de identificadores: El Efectivo (EUID) y el real (RUID). El primero lo adopta cuando ejecuta un programa que tiene el sticky bit encendido, es decir, se convierte en el usuario dueño del programa mientras este se ejecuta y retorna a su identidad real al término de la ejecución. El RUID es el usuario real, este último nunca cambia.

Foreground/Background: prioritario/no prioritario, preferente/subordinado (de fondo). Prioridad asignada a los programas que corren en un entorno multitarea. Los programas "foreground" tienen mayor prioridad y los programas "background" tienen menor prioridad. A los usuarios en línea se les asigna el foreground y a las actividades de procesamiento en lotes, como largos reordenamientos y actualizaciones, se les asigna el background. Si a las actividades de procesamiento en lotes se les asignara una prioridad mayor, los tiempos de respuesta de la terminal podrían hacerse considerablemente más lentos.

Host: La computadora central o la computadora controladora en un entorno de procesamiento en tiempo compartido o distribuido en red.

Hacker: Persona muy especializada en las particularidades de un sistema que emplea su conocimiento para encontrar vulnerabilidades en los sistemas y buscar una solución. Su acción es benéfica a diferencia de los crackers.

Indice: Es una estructura de almacenamiento, en las bases de datos, para un acceso más rápido a la información.

Intercambio de Contexto: En un ambiente multitarea, ceder el control a otro programa bajo la dirección del sistema operativo. El contexto de un programa es su estado actual.

Internet: (1) Red extensa constituida por una cantidad de redes menores. (2) Red nacional orientada a la investigación que engloba más de tres redes gubernamentales y académicas en 40 países.

Interfaz: Una conexión e interacción entre hardware, software y usuario. Las interfases de hardware son los conectores, zócalos y cables que transportan las señales eléctricas en un orden prescrito. Las interfases de software son los lenguajes, códigos y mensajes que utilizan los programas para comunicarse unos con otros, tal como entre un programa de aplicación y el sistema operativo. Las interfases de usuario son los teclados, ratones, diálogos, lenguajes de comando y menúes empleados para la comunicación entre el usuario y la computadora. El diseño y construcción de interfases constituye una parte principal del trabajo de los ingenieros, programadores y consultores. Los usuarios "dialogan" con el software. El software "dialoga" con otro hardware, así como con otro software. El hardware "dialoga" con otro hardware y todo este "diálogo" no es más que el uso de interfases. Deben ser diseñadas, desarrolladas, probadas y rediseñadas, y con cada encarnación nace una nueva especificación que puede convertirse en un estándar, de hecho o regulado.

Kerberos: Sistema de seguridad desarrollado en MIT que autentica a los usuarios. No ofrece autorización de los servicios técnicos ni de las bases de datos; establece identidad en la entrada al sistema, lo cual se utiliza en toda la sesión.

LAN: (Local Area Network) Red de Area Local. Red de comunicaciones que sirve a usuarios dentro de un área geográficamente limitada.

Modo Multiusuario: Es el modo al que se lleva un sistema Unix para que sea compartido por dos o más usuarios. Existen otros modos como el monousuario donde únicamente el

administrador del sistema puede entrar; monitor, donde se ejecuta el programa monitor para iniciar la máquina; nivel 2, donde no existen servicios de compartición de archivos (NFS), etc.

Parche: Un arreglo temporal o rápido a un programa. Demasiados parches en un programa lo hacen difícil de mantener.

Pipe: Espacio compartido que acepta la salida de un programa para la entrada en otro. En Unix, la orden pipe es una línea vertical (|). La sentencia dir|sort dirige la salida de la lista de directorios a la utilidad de ordenación.

Procedimiento Almacenado: Es un conjunto de sentencias SQL que se almacenan en el servidor de bases de datos. Tienen la ventaja de minimizar el tráfico en la red, ya que por ésta sólo tiene que viajar el nombre del procedimiento y sus parámetros y no la secuencia de instrucciones que lo componen.

Puerto: Canal lógico de comunicaciones utilizado por los servicios en Unix. Es una manera de que el cliente sepa a que servicio quiere conectarse en el servidor, quien tiene múltiples servicios habilitados.

SQL: (Structured Query Language) Lenguaje de Consulta Estructurado. Lenguaje utilizado para interrogar y procesar datos en una base de datos relacional. Desarrollado originalmente por IBM para sus macrocomputadoras, han habido muchas implementaciones creadas para aplicaciones de base de datos en mini y microcomputadoras. Las órdenes (mandatos) de SQL se pueden utilizar para trabajar interactivamente con una base de datos, o pueden incluirse en un lenguaje de programación para servir de interfaz a una base de datos. La implementación de SQL que utiliza Adaptive Server se llama T-SQL.

Swap: El proceso de swap se refiere al reemplazo de un segmento de un programa en la memoria por otro, y su restablecimiento a su estado original cuando se requiera. En los sistemas de memoria virtual, se denomina "paging".

TCP/IP: (Transmission Control Protocol/Internet Protocol) Protocolo de control de transmisiones/protocolo Internet. Conjunto de protocolos de comunicaciones desarrollado por la Defense Advanced Research Projects Agency (DARPA - Agencia de proyectos de investigación avanzada de defensa) para intercomunicar sistemas diferentes. Se ejecuta en un gran número de computadoras VAX y basadas en UNIX, y es utilizado por muchos fabricantes de hardware, desde los de computadoras personales hasta los de macrocomputadoras. Es empleado por numerosas corporaciones y por casi todas las universidades y organizaciones federales.

Trigger: Es similar a un procedimiento almacenado, pero con la diferencia de que se ejecuta automáticamente al momento de modificar la columna de una tabla a la cual se encuentra ligado.

UDP: (User Datagram Protocol) Protocolo de Datagrama para Usuario. Protocolo TCP/IP que permite que una aplicación envíe un mensaje a una o varias aplicaciones ejecutándose en la máquina destino. La aplicación es responsable de un envío confiable.

Vista: Una vista es un conjunto de datos limitado que se le proporciona a un usuarios. Estos datos se obtienen de otras tablas o vistas. No ocupa espacio, el servidor sólo guarda la definición de la sentencia SQL que contiene la vista.

Bibliografía

Chapman Brent. *1997. Construya Firewalls Para Internet. O'Reilly & Associates, Inc. México. Primera Edición en Español. p. 503.*

"Definición de Requerimiento de La Nómina". 1995. La Empresa.p. 150.

Freedman Alan. *1993. Diccionario de Computación. McGraw Hill. México. Primera Edición en español. México. p. 934*

Garfinkel Simson. *1996. Practical Unix & Internet Security. O'Reilly & Associates Inc. USA. Segunda Edición. p. 1004.*

"Gauntlet Administration Guide". 1999. Network Associates, Inc. p. 357.

"Gauntlet Users Guide". 1999. Network Associates, Inc. p. 130.

Siyan Karanjit. *1997. Firewalls Y La Seguridad En Internet. Prentice-Hall Hispanoamericana.México. Segunda Edición. p. 585*

"Solaris 2.x System Administration". 1997. Sun Educational Services.USA. Revisión F.1. p. 489

"Technical Documentation".1999. Sybase, Inc. Adaptive Server 11.9.2

ftp://ftp.unm.edu/ethics

http://cs.purdue.edu

http://docs.sun.com:80/ab2/coll.47.4/TRANSITION

http://sunsolve.sun.com

http://www.alw.nih.gov/Security/security-prog.html

http://www.cert.org

http://www.ietf.org

http://www.internic.net/rfc

http://www.nai.com

http://www.radium.ncsc.mil/tpep/library/rainbow/5200.28-STD.html

http://www.sun.com

http://www.sun.com/smcc/solaris-migration/docs/whitepapers.html

http://www.sun.com/software/white-papers/wp-security

http://www.sybase.com

http://www.tis.com/support

http://wzv.win.tue.nl