UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CONTADURÍA Y ADMINISTRACIÓN

SISTEMA UNIVERSIDAD ABIERTA Y EDUCACIÓN A DISTANCIA

AUTORAS. M. I. LOURDES YOLANDA FLORES SALGADO

M. A. REYNA ELIZABETH CABALLERO CRUZ

ADMINISTRACIÓN DE UNIX		Clave: 2048
Plan: 2005		Créditos: 8
Licenciatura: Informática		Semestre: 5º
Área: Redes y Telecomunicaciones		Hrs. Asesoría: 2
Requisitos: Ninguno		Hrs. por semana: 4
Tipo de asignatura:	Obligatoria ()	Optativa (x)

Objetivo general de la asignatura

Al finalizar el curso, el alumno conocerá el trabajo del administrador del sistema operativo UNIX, definirá las principales rutinas y subrutinas del sistema operativo y controlará, administrará y dirigirá las operaciones principales para el manejo del mismo.

Temario oficial (horas sugeridas 64)

	Horas
1. Introducción	2
2. Políticas de uso y administración de los equipos	4
3. Utilerías básicas del sistema operativo UNIX para administradores	6
4. Alta y baja del sistema	4
5. Mantenimiento claves de usuarios	4
6. Instalación y mantenimiento de dispositivos	6
7. Sistema de archivos	6
8. Respaldos	6
9. Configuración de la red	6
10. Administración de la memoria virtual	6
11. Monitoreo del desempeño del sistema	6
12. Seguridad	4
13. Instalación de software	4

Introducción

Actualmente las organizaciones dependen cada vez más de los sistemas de información para realizar sus operaciones de forma eficiente, por ello es necesario un "Administrador de Sistemas" quien será el responsable de configurar, mantener y actualizar un sistema, para que pueda ser utilizado por los usuarios. Por ello el administrador debe lidiar tanto con los aspectos técnicos en materia de cómputo, como con aspectos administrativos e incluso con el comportamiento y el trato con los usuarios.

Para mantener el adecuado funcionamiento del sistema, el administrador requiere tener conocimientos generales en áreas muy diversas, desde sistemas operativos, tecnologías de redes, programación, conocimientos de hardware, bases de datos, etc., hasta conocimientos básicos sobre el área de trabajo de la empresa, como economía, derecho, física, química, matemáticas, etc.

Las técnicas y procedimientos utilizados para administrar un sistema se basan principalmente en un conjunto de reglas y principios empíricos “bien conocidos (*well known*)” que han sido considerados a través de los años como los “más adecuados”. En muchos casos, los administradores de sistemas no cuentan con una preparación formal que les permita desempeñarse óptimamente dentro de su organización.

Existe cada día una creciente necesidad en las empresas de contar con Administradores de Sistemas que tengan una preparación formal que les proporcione los fundamentos teóricos, que les permitan no sólo mantener los equipos en condiciones aceptables de operación sino analizar, sintonizar y mejorar el desempeño de los mismos.

La Facultad de Contaduría y Administración ha tenido la visión de incluir como opción al desarrollo curricular de sus alumnos de la Licenciatura en Informática, la materia optativa “Administración en UNIX”, misma que ha permitido darles una visión de la preparación formal que se requiere para desempeñarse como administradores de sistemas. De hecho algunos de estos alumnos egresados de Informática se desempeñan actualmente como administradores de sistemas UNIX en diversas dependencias de la UNAM y en algunas organizaciones privadas.
Un aspecto importante por considerar sobre este Apunte es el siguiente:

UNIX™ es una marca registrada, por lo que no todos los sistemas operativos parecidos o con características de UNIX se pueden llamar UNIX™, a éstos se les

conoce como “Unix like”, sin embargo, para efectos didácticos se utilizaráindistintamente la palabra UNIX para referirnos tanto a sistemas operativos UNIX™ como a sistemas operativos “Unix like”. Son tantos que no sería posible abarcarlos en tan solo un curso.

Si se revisa la historia de este sistema operativo y las diversas ramificaciones que ha tenido, se encontraría que básicamente, en la actualidad, la mayoría de los sistemas toman características de las dos principales ramas de desarrollo de los sistemas operativos UNIX: UNIX™ System V, y *Berkeley Software Distribution*(BSD). Así, para dar un aspecto teórico más general se aborda cada tema desde ambos puntos de vista. El de System V (UNIX™) y el de BSD. Solo cuando sea importante mencionar las diferencias en cada sistema operativo se mencionarán de manera específica.

Los **temas1** y **2** son introductorios; tienen como objetivo ofrecer una visión general sobre las competencias y funciones de un administrador de sistemas dentro de una organización.

El **tema3** permite establecer las utilerías básicas del sistema operativo que un administrador de sistemas UNIX debe conocer para el desempeño de sus actividades diarias.

El **tema4** dará al alumno una visión de las actividades que se realizan para levantar o dar de baja un sistema, este tema está dividido en dos partes, primero se explica el procedimiento para sistemas operativos basados en BSD y luego para sistemas operativos basados en System V.

El **tema5** involucra los procedimientos para dar de alta claves de usuario y los procedimientos para dar de baja o cancelar estas claves.

El **tema6** incluye dos subtemas, impresoras y terminales; ambos siguen la misma estructura explicativa diferenciada donde primero se da una visión general del tema y luego se mencionan las diferencias entre BSD y System V.

El **tema7** incluye todo lo referente a la administración de dispositivos de almacenamiento. Es una idea general de cómo manejar estos dispositivos en los diversos sistemas operativos. No incluye aspectos avanzados de sistemas de archivos.

El **tema8** trata de las características y consideraciones que debe tener un respaldo de información y el manejo de las unidades de cinta para respaldos.

El **tema9** es sobre la configuración de las interfaces y manejo de servicios dentro del sistema.

El **tema10** solo hace referencia al concepto general de memoria virtual y los procedimientos de configuración y monitoreo de la misma tanto en BSD como en System V.

El **tema11** consiste en dar una visión de las herramientas del sistema operativo (intrínsecas al mismo) que permiten monitorear su desempeño, esto es cómo puede un administrador de sistemas monitorear un sistema sin requerir herramientas externas.

El **tema12** menciona los principales aspectos que un administrador de sistemas debe considerar para mantener un sistema lo más seguro posible.

Finalmente, el **tema13**trata sobre las consideraciones a tomar en cuenta para instalar un software, ya sea el propio sistema operativo o cualquier aplicación que se requiera dentro de la organización.

TEMA 1. INTRODUCCIÓN

Objetivo particular

El alumno reconocerá la importancia de la administración de sistemas y sus competencias, a través de una visión general de este tema para desempeñar dicha profesión.

Temario detallado (2 horas)

1.1. Administración de sistemas
1.2. Perfil del administrador de sistemas (Competencias del administrador de sistemas)
1.3. Actividades del administrador de sistemas(Consideraciones iniciales que debe tomar en cuenta un administrador de sistemas)

Introducción

Este tema contiene información introductoria sobre el tema Administración de Sistemas. Aún cuando las actividades generales pueden variar dependiendo del sistema operativo y del sitio de trabajo, se busca dar una visión general de las actividades que involucra esta profesión y los requisitos que deben cubrir aquellos que se dediquen a la misma.

1.1. Administración de sistemas

La Administración de Sistemas se encarga de la operación de la tecnologia orientada al cómputo.

Como se mencionó en la Introducción, el Administrador de Sistemas es la persona responsable de configurar, mantener y actualizar un sistema, para que pueda ser utilizado por los usuarios.

Cuando un sistema informático funciona con un sistema operativo UNIX, el administrador se convierte en un administrador de sistemas UNIX.

Bajo el esquema del sistema operativo UNIX, esto implica que el administrador de sistemas será en sí mismo el “superusuario” del sistema, esto es el usuario privilegiado por excelencia, teniendo acceso a todos los recursos del sistema y permisos y privilegios suficientes que le permitan configurar, mantener y actualizar dicho sistema.

1.2. Perfil del administrador de sistemas (competencias del administrador de sistemas)

La competencia se define como la “Pericia, aptitud o idoneidad para hacer algo”. Constituye un conjunto de atributos que una persona posee y le permiten desarrollarse adecuadamente en un determinado ámbito. Se puede decir que es una combinación entre la personalidad, aptitudes y conocimientos.

- **Aptitudes y personalidad**
  - **Trabajar bajo presión**. Se refiere a la capacidad de trabajar y resolver de forma satisfactoria las actividades desarrolladas, aún ante circunstancias totalmente adversas.
  - **Habilidad analítica**. Es la capacidad que tiene una persona para realizar un análisis lógico: identificar los problemas, reconocer la información significativa, buscar y coordinar los datos relevantes que le ayuden a su posible solución.
  - **Capacidad para aprender**. Está asociada a la asimilación de nueva información y su eficaz aplicación.
  - **Innovador**. Es la capacidad que le permite crear conocimiento a partir de la improvisación, experimentación, creatividad y el contacto directo.
  - **Autonomía**. Capaz de tomar decisiones. Supone actuar proactivamente cuando ocurren desviaciones o dificultades sin esperar a consultar a toda la línea jerárquica.
  - **Espíritude colaboración**. Hace referencia a la capacidad de confiar en los demás como sistemas de apoyo informales, dicha confianza proviene de compartir los rigores del entrenamiento y de la valorización de los saberes de los demás.
  - **Trabajo en equipo**. Supone facilidad para la relación interpersonal y capacidad para comprender la repercusión de las propias acciones en el éxito de las acciones del equipo.

- **Responsable**. Esta competencia se relaciona al compromiso con que las personas realizan las tareas encomendadas.

- **Valores**
  - **Temple**. Serenidad y dominio en toda circunstancia. Implica seguir adelante en medio de circunstancias adversas.
  - **Respetuoso**. Implica ser considerado, atento y educado en todo momento.
  - **Imparcial**. Implica obrar con base en la objetividad, sin favoritismos de ninguna especie.
  - **Tolerante**. Respetar las ideas, creencias o prácticas de los demás cuando son diferentes o contrarias a las propias.
  - **Paciente**. Implica tolerar las contrariedades y adversidades.
  - **Íntegro y honesto, confiable**. La integridad implica congruencia entre pensamiento y acción, implica autenticidad. Una persona honesta valora la sinceridad lo que la hace confiable. Implica que tendrá un comportamiento adecuado, correcto y justo, no hará mal uso de los recursos que le son confiados. Admitirá sus errores, si es el caso, y tratará de corregirlos.

- **Conocimientos**
  - Técnicas de programación.
  - Dominio de al menos un lenguaje de programación.
  - Funcionamiento del sistema operativo.
  - Técnicas de administración del sistema operativo.
  - Conocimientos básicos de hardware y mantenimiento de dispositivos.

1.3. Actividades del administrador de sistemas (Consideraciones iniciales que debe tomar en cuenta un administrador de sistemas)

Planeación de las actividades

Es muy importante que las actividades de administración sean planeadas, es decir,el administrador de sistemas debe trazar un programa de acción para cada actividad que realice.

Dado que el administrador es el responsable del funcionamiento del sistema y debido a que cuenta con los privilegios de superusuario que le permiten realizar cambios en su configuración, por ello cuando se este trabajando con privilegios de administrador hay que poner especial atencion para no cometer errores que puede tener consecuencias en el funcionamiento del sistema. El administrador debe definir el procedimiento a realizar en cada una de sus actividades, considerando las posibles consecuencias (positivas o negativas) para el funcionamiento del sistema.

Registro de las actividades

Las actividades realizadas por el administrador del sistema deben ser registradas con el fin de dejar constancia de los cambios en la configuración de cualquier elemento del sistema. Estos registros pueden ser en formato electrónico, dentro o fuera del sistema, o en papel. Se recomienda el uso de al menos dos métodos para registro ya que en caso de algún incidente habrá respaldo del mismo.

Guardar copias de seguridad

Antes de realizar cualquier modificación en el sistema por mínima que sea, es importante respaldar los archivos originales. A veces los administradores hacen cambios que a primera vista parecen no tener ninguna repercusión pero en ocasiones no son adecuadas provocando una operación inusual en el sistema.

Conocimiento de la documentación del sistema

Una característica de los sistemas operativos UNIX es su falta de estandarización, si bien existen elementos comunes entre todos cada empresa o grupo de desarrollo maneja características propias.Como es por ejemplo los programas de instalación de cada version de UNIX.

Es común que el administrador de sistemas UNIX se vea en la necesidad de buscar información documental, investigar e incluso solicitar apoyo a otros administradores de sistemas para realizar sus actividades.

A continuación se mencionan los principales medios donde un administrador debe comenzar cuando requiera buscar información documental.

- Documentación en línea o en formato electrónico
    - Manuales del sistema (man)
    - Libros
    - Revistas especializadas
    - Manuales
        - Administración
        - Usuario
        - Aplicaciones
    - Páginas web de organizaciones reconocidas
    - Páginas web de grupos de usuarios
    - Blogs
    - Listas de discusión
- Documentación Impresa
    - Libros
    - Revistas especializadas
    - Manuales
        - Administración
        - Usuario

        - Aplicaciones

- RFC (*Request For Comments*)
    - Son notas de trabajo de la comunidad de desarrollo e investigación en Internet. Un documento puede relacionarse con cualquier tópico de las comunicaciones en cómputo o de las especificaciones de un estándar.
    - La mayoría de los RFC son descripciones de los protocolos de red o servicios y a menudo dan descripciones detalladas de los procedimientos y formatos de trabajo de comités técnicos o talleres de trabajo. Son de dominio público a menos que se especifique lo contrario.

Conocimiento del hardware del sistema

Es muy importante que el administrador conozca las características físicas de los equipos a su cargo, así como de cualquier dispositivo periférico conectado a ellos. (CPU, número de cores, velocidad, Memoria, Almacenamiento secundario, Unidades internas, externas, etc.).

Junto con las características físicas también debe conocer la ubicación física y los procedimientos de encendido y apagado de los mismos.

En muchos casos los administradores de sistemas realizan sus actividades en forma remota y no en sitio. Es importante, en caso de cualquier eventualidad, saber en dónde se ubican físicamente los equipos. En sitios donde la configuración de los sistemas es complejo los procedimientos de encendido y apagado son fundamentales para asegurar la integridad de la información contenida en ellos.

Bibliografía básica del tema 1

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Nemeth, Evi; Snyder, Garth; Hein, Trent R. (2007). *Linux administration handbook.* (2nd ed.) Stoughton, Massachusetts: Pearson Education. [Vista previa]

Bibliografía complementaria

Burguess, M. (2000). *Principles of Network and System Administration.*Chichester, Sussex: John Wiley & Sons.

Sitios de Internet

(Todos los sitios del apunte, recuperados o consultados, funcionan al 10/01/12)

Darmohray, Tina. (Ed.). (2001). *Job Descriptions for System Administrators*, Short Topics in System Administration #8, The System Administrators Guild (SAGE), disponible en línea: http://www.sage.org/field/jobs-descriptions.html

Actividades de aprendizaje

A.1.1. Elabora un mapa mental que incluya todas las competencias (valores y aptitudes) del administrador de sistemas.

A.1.2. Elabora un cuadro sinóptico de los conocimientos requeridos para desempeñar la profesión de Administrador de sistemas.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿Qué es la administración de sistemas?
2. ¿Quién es el superusuario?
3. ¿Cuáles son las aptitudes que debe poseer un administrador de sistemas?
4. ¿Cuales son los valores que debe poseer un administrador de sistemas?
5. ¿Qué se entiende por competencia de “Autonomía”?
6. ¿Qué se entiende por “Tolerancia”?
7. ¿Qué son los RFC?
8. ¿Cuáles son los principales conocimientos que debe poseer un administrador de sistemas?
9. Mencione al menos tres tipos de documentos electrónicos que debe conocer un administrador de sistemas.
10. ¿Por qué se deben planear las actividades del administrador de sistemas?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. ¿Qué es la administración de sistemas?
   a) Es la rama de la ingeniería que administra la operación de un sistema de cómputo, y que involucra a los usuarios
   b) Es la rama de la ingeniería que se encarga de configurar, mantener y actualizar un sistema de cómputo
   c) Es la rama de la ingeniería que se encarga de la administración operacional de un sistema de cómputo sin involucrar a sus usuarios
   d) Es la rama de la ingeniería que se encarga de administrar los usuarios de un sistema

2. ¿Cuáles son las responsabilidades del administrador de sistemas?
   a) Instalar, mantener y reparar un sistema para ser utilizado por los usuarios
   b) Configurar, mantener y actualizar un sistema para ser utilizado por los usuarios
   c) Configurar, mantener y actualizar un sistema sin importar la actividad de los usuarios
   d) Configurar, mantener y actualizar un sistema para ser utilizado para darservicios

3. Explica el concepto de autónomo dentro de las Competencias del Administrador de Sistemas:
   a) Capaz de tomar decisiones. Actuar proactivamente cuando ocurren desviaciones sin esperar a consultar a toda la línea jerárquica
   b) Capaz de tomar decisiones basado en las ordenes que recibe
   c) Capaz de trabajar por su cuenta para resolver los problemas
   d) Capaz de decidir por su cuenta que se debe hacer sin consultar a nadie

4. Valor que implica serenidad y dominio en toda circunstancia:
   a) Tolerancia
   b) Temple
   c) Paciente
   d) Integridad

5. ¿Cuál de las siguientes opciones **no** corresponde a un conocimiento que debe tener el Administrador de Sistemas?:
   a) Fundamentos de electricidad
   b) Técnicas de programación
   c) Dominio de al menos un lenguaje de programación
   d) Técnicas de administración del sistema operativo

6. ¿Cuál de las siguientes opciones **no** es una consideración que debe tomar en cuenta el administrador de sistemas?
   a) Registro de las actividades
   b) Realizar copias de seguridad
   c) Conocer la estructura de su organización
   d) Conocer el hardware de su sistema

7. Son competencias de aptitud y personalidad.
   a) Trabajo en equipo, espíritu de colaboración, paciencia
   b) Honestidad, trabajo en equipo, espíritu de colaboración
   c) Paciencia, Honestidad y capacidad de aprender
   d) Trabajo en equipo, capacidad de aprender, espíritu de colaboración

8. Son notas de trabajo de la comunidad de desarrollo e investigación en Internet:
   a) Manuales del sistema
   b) Blogs especializados
   c) RFC
   d) Páginas web de grupos de usuarios

9. Medios de información documental utilizados por el administrador del sistema.
   a) Manuales de usuario y desarrollo
   b) Manuales en línea y correo electrónico
   c) Manuales de aplicaciones y desarrollo
   d) Manuales de usuario y administración

10. Actividad que se recomienda realizar antes de cualquier modificación en el sistema.
   a) Planear las actividades
   b) Comunicar a los usuarios
   c) Registrar las actividades
   d) Respaldar

TEMA 2. POLÍTICAS DE USO Y ADMINISTRACIÓN DE LOS EQUIPOS

Objetivo particular

El alumno reconocerá el proceso y la importancia del establecimiento de políticas de la administración de sistemas y su relación con los principios básicos de ésta.

Temario detallado (4 horas)

2.1. Consideraciones
2.2. Procedimientos para establecer políticas
2.3. Principios de administración de sistemas
2.4. Mecanismos de comunicación con el usuario

Introducción

Debido a la creciente utilización de las tecnologías de la información, las organizaciones dependen cada vez más de sus sistemas informáticos y con ello de una adecuada administración para los mismos, esto obliga a las entidades a crear medidas de emergencia y establecer políticas y procedimientos que les permitan contrarrestar cualquier posible eventualidad o problema que pudiera presentarse y que afecte sus operaciones.

Las políticas son directrices, que tienen por objeto establecer los lineamientos en los cuales deben conducirse los usuarios en el uso de recursos computacionales, de manera que permita asegurar la protección y la integridad de los datos de los sistemas, redes, instalaciones de cómputo y procedimientos manuales.

2.1. Consideraciones

En las políticas se representa la filosofía de una organización y el pensamiento estratégico de la alta dirección.

El documento de políticas debe contener:

- Políticas claras y concisas.
- Definición de responsabilidades de los recursos informáticos aplicado a todos los niveles de la organización que abarcan las metas y las directrices generales.
- Acciones necesarias para asegurar que los empleados o usuarios afectados por una política específica reciban una explicación clara completa de la política y entiendan cuál es su propósito.
- Definición de los requerimientos de seguridad de los sistemas que abarca el alcance de la política.
- Establecimiento de las violaciones y sanciones por no cumplir con las políticas.
- Identificación de las responsabilidades de los usuarios con respecto a la información a la que tienen acceso.

Asimismo, las políticas deben seguir un proceso de actualización periódica sujeto a los cambios organizacionales relevantes, como son: el aumento de personal, cambios en la infraestructura computacional, alta rotación de personal, desarrollo de nuevos servicios, regionalización de la empresa, cambios o diversificación del área de negocios, etc.

2.2. Procedimientos para establecer políticas

En la formulación de políticas se debe considerar que los sistemas son vulnerables a una diversidad de amenazas y atentados por parte de: personas, desastres naturales, fallas en las instalaciones, etc.

Estos aspectos originan que los directivos de las organizaciones reconozcan la necesidad de establecer políticas, normas y directrices.

Con la identificación de riesgos se puede definir las políticas orientadas a la estrategia y objetivos de la organización, en ellas se debe cubrir con los siguientes aspectos:

- Alcance de las políticas, identificando sistemas y personal sobre la cual aplicará.
- Objetivos de la política y descripción de los elementos involucrados.
- Determinación de las sanciones que deben aplicarse por el no cumplimiento de la política.
- Responsabilidades por cada sistema, servicio y recursos informáticos a todos los niveles de la organización.
- Identificación de requerimientos de seguridad física y lógica.
- Planeación formalizada para administración de los equipos. (Bayuk, 1996).
- Gestión de los medios necesarios para administrar correctamente los sistemas.
- Monitoreo periódico de los procedimientos y operaciones de la empresa, de forma tal, que ante cambios, las políticas puedan actualizarse oportunamente.
- Detallar explícita y concretamente el alcance de las políticas con el propósito de evitar situaciones de tensión al momento de establecer los mecanismos de seguridad que respondan a las políticas trazadas.

La naturaleza de las políticas radicará en el tipo de organización, es decir, cada organización creará sus políticas basadas en sus propios objetivos.

En el caso de la administración de sistemas, los administradores generalmente crean o influyen en la definición de políticas, tales como: apertura de cuentas, horas de mantenimiento, responsabilidad de los respaldos, borrado de archivos temporales, cuotas de disco, seguridad del sistema, etc.

2.3.Principios de administración de sistemas

Para que un administrador pueda mantener funcionando adecuadamente un sistema, debe analizar el comportamiento del mismo. Esto implica que debe determinar parámetros que le permitan conocer cuándo el sistema se comporta en forma adecuada y cuándo no.

Burgess (2000) define 3 principios fundamentales de la administración de sistemas.

Cimentación: La administración de sistemas comienza con una política que permita decidir el comportamiento deseado en relación con el comportamiento que se puede mantener.

Previsibilidad: La administración de sistemas busca trabajar con un sistema previsible. La previsibilidad es la base de la confiabilidad y por ende de la seguridad.

Escalabilidad: Un sistema escalable es aquel que crece en conformidad a la política establecida, es decir, mantiene la función de previsibilidad aún si incrementa su tamaño.

Una política en sí misma es una declaración formal de propósitos codificada tanto como sea posible, dentro de un plan de administración e involucra un esquema de respuestas ante posibles eventos. La política determina un punto de equilibrio en el sistema.

Como los sistemas cómputo-humanos (sistema + usuarios) no son determinísticos, las políticas permiten minimizar la parte no predecible en su comportamiento.

> **Interdependencia**. Mientras más dependiente sea un servicio de otro, más vulnerable es. Disminuir las dependencias incrementa la previsibilidad y confiabilidad de un sistema.
> **Separación de datos**. Los datos deben ser separados del sistema operativo en una estructura diferente de directorios. Esto facilitará las labores de mantenimiento, reinstalación y actualización del sistema. (Burgess, 2000).

En este sentido se deberán definir los aspectos deseados en el comportamiento del sistema.

- Recursos disponibles
- Servicios disponibles

Como se mencionó anteriormente, el administrador de sistemas debe establecer las políticas de uso y administración de los equipos a su cargo con el fin de mantener un cierto orden.

La administración de sistemas no sólo involucra computadoras e individuos sino comunidades. Existe una comunidad local de usuarios que trabajan en sistemas multiusuario; una comunidad de sistemas multiusuario formando una red local de sistemas de cómputo y finalmente una comunidad global formada por sistemas locales y redes de todo el mundo.

Principio de simplicidad: Los usuarios tolerarán las reglas siempre y cuando éstas sean fáciles de entender.

El establecimiento de límites al comportamiento de los usuarios y políticas que los regulen, permite conseguir el equilibrio en interés de la comunidad.

Los principios de administración de sistemas relacionados con el establecimiento de políticas de uso son:

> **Principio de libertad**. Las cuotas, límites y restricciones tienden a antagonizar con los usuarios. Los usuarios tienen en alta estima el concepto de libertad personal. Las restricciones deben ser minimizadas en la medida de lo posible para evitar una reacción desfavorable de los usuarios.
> **Principio de autoridad**. El administrador del sistema debe mantener el control y vigilar cualquier posible abuso en el mismo.
> **Principio de hostigamiento**: El abuso en el uso de un recurso público puede ser considerado como hostigamiento por aquellos usuarios que lo comparten.
> **Principio de fallas humanas**: No todas las fallas pueden ser evitadas. No importa las precauciones que se tomen, la ocurrencia de fallas es inevitable. Sin embargo las fallas deben ser previsibles. Se debe contar con listas de verificación y líneas de acción que permitan proteger al resto del sistema. Las fallas humanas se pueden minimizar si las políticas y procedimientos de administración se establecen y verifican conforme a un sistema de aseguramiento de la calidad. (Burgess, 2000)

2.4. Mecanismos de comunicación con el usuario

Es importante que el usuario conozca las políticas de administración del sistema, así como las situaciones por la que está pasando el equipo, con ello se persigue el mejor desempeño en el otorgamiento del servicio.

El administrador en UNIX, puede hacer uso de las siguientes alternativas para comunicarse con los usuarios:

	Tipo de archivo	**Acción**
`/etc/issue`	archivo de texto	Contiene un mensaje o la identificación del sistema, aparece antes de la entrada al sistema.
`/etc/motd`	Archivo de texto	Contiene un mensaje que aparece cuando se ingreso al sistema.
mail	instrucción	Envía correos electrónicos.
news	instrucción	Se utilizan para comunicar tópicos más largos que los usuarios deben leer, sin sobre cargar el mensaje inicial. /usr/news/* /var/news/*
Write	instrucción	Envía un mensaje a otro usuario.
Wall	instrucción	Envía un mensaje a todos los usuarios conectados en el sistema.

Bibliografía básica del tema 2

Bayuk, Jennifer. (1996).*Security through process management*. En: National Information Systems Security, Conference Proceedings, octubre 22-25. Baltimore. [Versión electrónica], disponible en línea: http://csrc.nist.gov/nissc/1996/papers/NISSC96/paper015/bayuk.pdf

Burguess, M. (2000). *Principles of Network and System Administration.*Chichester, Sussex:John Wiley & Sons.

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Nemeth, Evi; Snyder, Garth; Hein, Trent R. (2007). *Linux administration handbook.* (2nd ed.) Stoughton, Massachusetts: Pearson Education. [Vista previa]

Powers S; Peek J; O'Reilly T and Loukides M. (2003). *Unix Power Tools*(3rd ed.)Sebastopol, CA: O'Reilly and Associates.

Bibliografía complementaria

Dijker, B. (Ed.) (1996) *A Guide to Developing Computing Policy Documents*, Short Topics in System Administration #2, The System Administrators Guild, USENIX Association.

Sitios de Internet

UNICOS/mp System Administration (2006) *Communicating System Information to Users*. Cray Inc. disponible en línea: http://docs.cray.com/books/S-2311-23/html-S-2311-23/fixednaw1jsda38.html)

Actividades de aprendizaje

A.2.1. Elabora un mapa mental en el que describa los principales elementos que se deben tomar en cuenta al elaborar un documento de políticas de administración de sistemas.

A.2.2. Elabora un cuadro sinóptico en el cual clasifique los principios de administración de sistemas.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿En qué consiste una política de administración de sistemas?
2. ¿Qué importancia tiene para un administrador de sistemas la existencia de un documento de políticas?
3. Enumera los elementos que debe contener un documento de políticas.
4. ¿Cuáles son los tres principios fundamentales de la administración de sistemas?
5. ¿En qué consiste el principio de Simplicidad?
6. Explica la relación entre el principio de Libertad y el de Hostigamiento.
7. ¿En qué consiste el principio de Fallas Humanas?
8. ¿Cuáles son los principales mecanismos de comunicación con los usuarios?
9. Explica la relación entre los mecanismos de comunicación con los usuarios y el establecimiento de políticas de administración.
10. ¿Cuál es la función del archivo `/etc/motd`?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. ¿Cuál de las siguientes opciones es una característica de las políticas del sistema?
    a) Tienen procedimientos que permiten contrarrestar cualquier eventualidad o problema que pudiera presentarse.
    b) Deben ser preparadas por los usuarios para utilizar el sistema.
    c) Cada usuario define las políticas en la forma que mejor le corresponda para tener mejor uso del equipo.
    d) Son muy importantes pero prescindibles.

2. ¿Qué debe definirse en las políticas?
    a) Su alcance, identificando sistemas y personal sobre el cual aplicará.
    b) Deben ser claras y concisas.
    c) La tarea del administrador de sistemas.
    d) Los nombres de usuario que estarán activos en él.

3. ¿Cuál es el concepto de cimentación?
    a) La administración de sistemas comienza con una política que permite definir el comportamiento deseado en relación con el comportamiento que se puede obtener.
    b) La administración de sistemas comienza con una política que permite definir el comportamiento que se va a obtener.
    c) La administración de sistemas comienza con una política que permite definir el comportamiento que deberán tener los usuarios durante su uso.
    d) La administración de sistemas comienza con una política que permite definir el comportamiento que deberán tener los administradores.

4. ¿Qué es el principio de escalabilidad?
   a) Denota un sistema que es capaz de crecer conforme a lo estipulado en las políticas sin dejar de ser previsible.
   b) Denota un sistema que es capaz de crecer a pesar de no conocer su comportamiento al hacerlo.
   c) Denota un sistema que para ser capaz de crecer debe de detenerse.
   d) Denota que el sistema no debe crecer para no perder su previsibilidad.

5. ¿Por qué se debe evitar la interdependencia? -Porque el sistema:
   a) consume más recursos si éstas existen.
   b) trabaja más lento ante ellas.
   c) es más vulnerable si éstas existen.
   d) no puede trabajar sólo.

6. Explica el principio de libertad.
   a) El usuario debe poder hacer lo que desee con su cuenta.
   b) El usuario debe poder hacer lo que desee con el equipo.
   c) Las restricciones impuestas al usuario deben ser las menores posibles.
   d) Las restricciones impuestas al usuario deben ser independientes a las de otros.

7. ¿Qué mecanismo de comunicación utilizarías para mostrar un mensaje a todos los usuarios que están conectados?
   a) Mail
   b) Wall
   c) Motd
   d) Write

8. Se utiliza como mecanismo de comunicación con los usuarios, sin sobrecargar el mensaje inicial del sistema.
   a) Issue
   b) Motd
   c) News
   d) Write

TEMA3. UTILERÍAS BÁSICAS DEL SISTEMA OPERATIVO UNIX PARA ADMINISTRADORES

Objetivo particular

El alumno reconocerá los elementos, marcas de tiempo y comandos del sistema operativo UNIX, así como una visión general del mismo, que le permiten al administrador de sistemas desempeñar su profesión.

Temario detallado(6 horas)

3.1. El sistema operativo UNIX
3.2. Marcas de tiempo
3.3. Find
3.4. Cron

Introducción

Como se mencionó en el tema 1 es necesario que el administrador de sistemas conozca el sistema operativo utilizado por los equipos. Para el caso de la Administración en UNIX, se requiere que el administrador tenga conocimientos más allá de un simple usuario, es decir, que tenga un profundo conocimiento del sistema operativo UNIX.

Para asegurar que el alumno que estudia esta materia tenga al menos, los conocimientos mínimos necesarios para un mejor aprovechamiento del curso, este tema tiene como objetivo dar al alumno una breve referencia sobre los comandos

y utilerías básicas del sistema operativo UNIX requeridos por un administrador de sistemas.

3.1. El sistema operativo UNIX

UNIX es un sistema operativo desarrollado como un proyecto de investigación en los Laboratorios Bell (filial de AT&T) a finales de la década de los 60 y principios de los 70. La finalidad era desarrollar un sistema operativo que proporcionara a los usuarios, un ambiente de trabajo “simple, poderoso y elegante”.

El sistema operativo UNIX utiliza un enfoque de diseño basado en la solución de problemas a través de la interconexión de herramientas simples. Muchas herramientas de UNIX están diseñadas para hacer una sola función y ninguna otra; la potencia del sistema operativo se obtiene a través de las interacciones entre los programas.

El sistema operativo UNIX se compone de 3 elementos fundamentales:

- *Kernel o núcleo del sistema*

  Es la parte medular de UNIX. Es la capa más interna del sistema operativo y es la que interactúa directamente con el hardware (parte física). Proporciona el sistema de archivos, los mecanismos de planificación de la CPU, la funcionalidad de gestión de memoria, etc. (Silberschatz, A., Galvin, P., Gagne, G., 2006)

- *Shell*

  Es la capa externa del sistema operativo UNIX, actúa como interfaz entre el usuario y el resto del sistema. (Sánchez, S. 1999) Es un intérprete de comandos y un lenguaje de programación. Permite

modificar en las características con que se ejecutan los programas en UNIX.

- *Sistema de archivos*

  Es la estructura que organiza los datos para manejar la información contenida en un dispositivo de almacenamiento secundario. En UNIX se basa en un modelo de árbol invertido y recursivo de tipo jerárquico donde los archivos están bajo directorios y los directorios bajo otros directorios.

  Como UNIX utiliza un formato consistente para los archivos, el flujo de bytes (Byte Stream), permite leer o escribir a un dispositivo, por lo que en UNIX todo se considera como un archivo. Así en realidad cuando estamos haciendo referencia al sistema de archivos de UNIX, implícitamente estamos haciendo referencia al sistema de Entrada/Salida de UNIX.

Para efectos didácticos y facilitar el aprendizaje y entendimiento del sistema operativo UNIX, se añade un cuarto elemento importante:

- *Sistema de control de procesos*

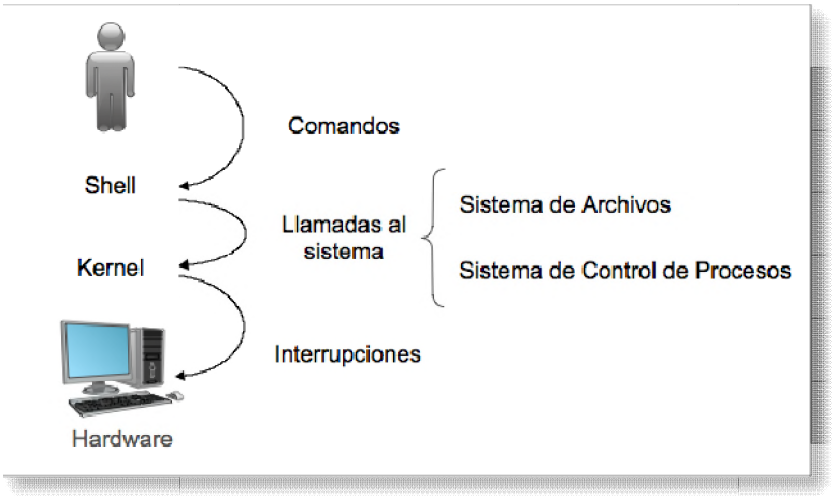
  Un proceso es un programa en ejecución. Para que un proceso se ejecute requiere acceder a la CPU, así mismo necesita una porción de memoria y acceso al sistema de archivos.

  Si bien el control de procesos es una actividad intrínseca al kernel, es tal la importancia que tiene esta función que la hemos separado para su mejor entendimiento.

  El sistema de control de procesos incluye todos aquellos aspectos asociados con los procesos que afectan directamente su ejecución, tales como la creación y eliminación de procesos, su acceso a los

recursos del sistema (CPU, Memoria, E/S), monitoreo del estado de los procesos, etc.

Una vez que hemos visto cada una de las partes que componen el sistema operativo UNIX, es importante mencionar cómo se relacionan. Si bien puede haber variaciones, en esencia, dichas relaciones se dan de la siguiente forma:



Un *usuario* del sistema operativo UNIX ejecutará *comandos* o programas que serán interpretados por el *Shell* quien pasará las instrucciones al kernel. Tanto el Shell como los programas acceden al kernel en forma de *llamadas al sistema,* dependiendo del tipo de llamada, el kernel proporcionará un servicio ya sea accediendo al *Sistema de Archivos* o al *Sistema de Control de Procesos*. Como el kernel es el núcleo del sistema operativo, será lo que se comunique directamente con los dispositivos a través de *Interrupciones* al *hardware* y el manejo de excepciones.

UNIX básico

Debido a que este curso es de Administración de UNIX, se requiere que el alumno tenga un manejo previo del mismo. Esto es, que el alumno conozca los comandos básicos que le permitan:

- Entrar y salir de sesión
- Navegar en el sistema de archivos
- Manipular archivos y directorios
    - Crear, eliminar, modificar, listar, acceder
    - Rutas absolutas y relativas
    - Permisos
    - Metacaracteres
- Manipular procesos
    - Listar, eliminar
    - Procesos en primero y segundo plano
- Acceder a los manuales del sistema (man)

Además de los comandos básicos del sistema operativo UNIX, un administrador de sistemas debe tener conocimientos previos sobre:

- Filtros y expresiones regulares
- Redireccionamientos de Entrada/Salida
- Marcas de tiempo
- Programación en Bourne Shell, AWK y PERL.

3.2. Marcas de tiempo

Una marca de tiempo es un registro que indica la fecha y hora en que ocurrió un evento específico.

Para cada archivo en el sistema, UNIX maneja 3 marcas de tiempo.

- *mtime* Modification Time.
  - Fecha de última modificación del archivo.
    Se altera cuando el contenido del archivo cambia.
    creat, mknod, pipe, utime, write.
    `$ ls –l`
    `$ stat -m`
- *atime* Access time.
  - Fecha de último acceso del archivo.
    Se modifica cuando vemos el contenido de nuestro archivo.
    creat, mknod, pipe, utime, read
    `$ ls –lu`
    `$ stat -a`
- *ctime* Creation Time.
  - Fecha de última modificación del inodo
    Esta marca se modifica en sí cuando la estructura o algún atributo del archivo cambia.
    Chmod, chown, creat, link, mknod, pipe, unlink, utime, write.
    `$ ls –lc`
    `$ stat -c`

3.3. Find

Es uno de los comandos más importantes y útiles de Unix. Permite localizar archivos que cumplan con ciertas condiciones, y actuar sobre ellos de diversas formas.

Sintaxis

```
find rutas operadores
```

donde

rutas

Especifica el(los) directorio(s) donde comenzará la búsqueda. Siempre se realiza una búsqueda recursiva.

Operadores de localización de archivos

`-name nombre`	Encuentra los archivos que tengan el nombre especificado. Si se utilizan metacaracteres es necesario encerrarlo entre comillas.
`-type c`	Encuentra los archivos del tipo especificado.
`-user nombre`	Encuentra los archivos que pertenezcan al usuario especificado. Se puede especificar por nombre o por número.
`-group nombre`	Encuentra los archivos que pertenezcan al grupo especificado.
`-nouser`	Encuentra los archivos cuyo dueño no existe en el sistema.
`-perm modo`	Encuentra los archivos que coinciden exactamente con el modo. `-modo` Encuentra los que al menos coincidan con ese modo.

`-inum n`	Encuentra los archivos que coincidan con ese número de inodo.
`-size n`	Encuentra los archivos del tamaño especificado, donde n significa “exactamente n”, +n significa “mayor que n” y `-n` “menor que n”.
`-atime n`	Encuentra los archivos que han sido accedidos hace n días (también se puede usar +n y `-n`).
`-mtime n`	Encuentra los archivos cuyo contenido fue modificado hace n días.
`-ctime n`	Encuentra los archivos cuyo i-nodo fue modificado hace n días.
`-newer archivo`	Encuentra todos los archivos que han sido modificados más recientemente que el archivo especificado.

Operadores de acción

`-print`	Imprime en la salida estándar la ruta de cada archivo encontrado.
`-exec comando`	Ejecuta el comando especificado cada vez que se encuentra un archivo. La cadena `{}` en comando será reemplazada por el nombre del archivo. Siempre y cuando esté rodeada por espacio en blanco. Comando debe terminar con `\;`
`-ok comando`	Igual que -exec, pero pregunta antes de ejecutar cada comando.

Operadores relacionales

`op1 -a op2`	Encuentra los archivos que cumplan con los dos operadores. Este operador no es necesario, basta con poner los operadores uno tras otro.
`op1 -o op2`	Encuentra los archivos que cumplan con cualquiera de

	los dos operadores.
`!operador`	Encuentra los archivos que **no** cumplan con el operador.
`\(expr\)`	Permite agrupar operadores para que sean evaluados antes que el resto.

Ejemplos

```
find . -name “*.o” -exec rm -f {} \;
find ~ ~yflores /home/info/egrli -name proyecto.doc -print
find /tmp -atime +15 -print -exec rm -f {} \;
find . \! -type d -print
```

3.4. Cron

Es una utilería que permite la ejecución periódica de tareas. Es atendida por un demonio o programa que se ejecuta de forma continua llamado crond, que revisa las tareas que hay que ejecutar y ejecuta las que sea necesarias.

Cualquier cosa que los comandos ejecutados por cron produzcan en la salida o el error estándar será enviada por correo electrónico al dueño de la tarea correspondiente.

crontab

Las tablas de cron o archivos crontab de los usuarios se pueden encontrar, dependiendo del sistema operativo, en alguna de las siguientes ubicaciones:

```
/var/spool/cron
/var/spool/cron/crontabs
/var/cron/tabs
```

En el caso específico del cron de BSD, adicionalmente existe una tabla cron del sistema que se ubica en: `/etc/crontab`

Formato del *archivo crontab* (tabla de cron)

`minutos horas días meses dias_semana comando`	Crontab de usuarios SV y BSD
`minutos horas días meses dias_semana usuario comando`	Crontab del sistema BSD Solo puede ser modificada por root.

donde

Campo	**Significado**	**Rango de valores**
`minutos`	Minutos después de la hora	`0-59`
`horas`	Horas del día	`0-23` `(0 = media noche)`
`días`	Días del mes	`1-31`
`meses`	Meses del año	`1-12`
`días_semana`	Días de la semana	`1-7 (1=Lunes) BSD` `0-6 (0=Domingo) SV`

Cada campo puede ser un solo número, un par de números separados por un guión (indicando un rango), una lista separada por comas de números y rangos, o un asterisco que representa todos los valores válidos de ese campo.

crontab

El comando crontab permite manipular las tablas de cron de los usuarios.

Sintaxis:

```
crontab [ -u usuario ] archivo
crontab [ -u usuario ] { -l | -r | -e }
```

Opciones:

`-u usuario`	Actúa sobre el archivo crontab del usuario especificado. Solamente root puede utilizar esta opción.
`archivo`	Reemplaza el archivo crontab del usuario por el archivo.
`-l`	Lista el archivo crontab del usuario
`-r`	Borra el archivo crontab del usuario
`-e`	Edita el archivo de crontab del usuario. Utiliza el editor definido por omisión en el sistema o en su caso el especificado en la variable de ambiente EDITOR del Shell.

Ejemplos:

```
5 0 1 * * sh /scripts/moth_log
```

Ejecuta moth_log a las 0:05 hrs. del primer día de cada mes.

```
15 1 * * 1 /usr/security/cops -v
```

Ejecuta el comando cops a la 1:15 am de todos los lunes.

Bibliografía básica del tema 3

Flores, Y. (2006). *Introducción a UNIX.* Plan de Becarios en Supercómputo, DGSCA, Universidad Nacional Autónoma de México.

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) O'Reilly Media: Sebastopol, CA.

Powers S; Peek J; O'Reilly T and Loukides M. (2003). *Unix Power Tools*(3rd ed.)Sebastopol, CA:O'Reilly and Associates.

Sánchez, S. (1999). *UNIX y Linux guía práctica*. México: Alfaomega/Ra-Ma.

Sarwar, S. A., Koretsky, R., Sarwar, S. M. (2002). *El libro de UNIX*. Madrid: Addison Wesley.

Bibliografía complementaria

Silberschatz, A., Galvin, P., Gagne, G. (2006) *Fundamentos de Sistemas Operativos*. (7ª ed.). Madrid: McGraw-Hill/Interamericana.

Zamboni, D. (1993). *Notas de utilerías de UNIX*. Plan de Becarios en Supercómputo, DGSCA, Universidad Nacional Autónoma de México.

Sitios de Internet

Yolinux.com (s.f.)Linux Information Portal, *Linux System Administration and Configuration*: http://www.yolinux.com/TUTORIALS/LinuxTutorialSysAdmin.html.

Actividades de aprendizaje

A.3.1. Elabora un esquema de los principales elementos que componen el sistema operativo UNIX.

A.3.2. Elabora un cuadro sinóptico donde expliques las marcas de tiempo de UNIX.

A.3.3. Elabora un resumen de la sintaxis principal de las utilerías cron y find.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿Qué es el kernel y para qué sirve?
2. ¿Qué es un proceso?
3. ¿Qué se entiende por sistema de archivos?
4. ¿Qué es una marca de tiempo?
5. ¿Cuál es la utilidad del comando find?
6. ¿Para qué sirve el comando exec en find?
7. ¿Cuántos tipos de operadores maneja find y cuáles son?
8. ¿Cuál es la diferencia entre los operadores de acción exec y ok en find?
9. Explica el funcionamiento de la utilería cron.
10. Menciona las diferencias entre la implementación de cron de BSD y la de SV.

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. ¿En dónde fue desarrollado el sistema operativo UNIX?
    a) En la universidad de Berkeley
    b) En el MIT
    c) En los laboratorios Bell
    d) En las oficinas de General Electric

2. ¿Cuáles son los 3 elementos del sistema de archivos UNIX?
    a) Sistema de Archivos, Shell, Kernel
    b) Kernel, sistema de archivos, arreglo de discos
    c) Sistema de archivos, red, Memoria de Acceso Aleatorio
    d) Kernel, Shell, Procesador

3. ¿Para qué sirve un sistema de control de procesos?
    a) Para monitorear el uso de CPU de los procesos y controlar su acceso.
    b) Para monitorear el uso de Memoria.
    c) Para monitorear las entradas y salidas.
    d) Todas las anteriores.

4. ¿Cuál de las siguientes no es una marca de tiempo?
    a) Atime
    b) Mtime
    c) Ptime
    d) Ctime

5. ¿Con qué comando veo la marca de tiempo modificación de un archivo?
   a) ls –l
   b) ls –lc
   c) ls –lu
   d) ls -i

6. ¿Cuál de los siguientes no es un operador de find?
   a) name
   b) type
   c) modified
   d) perm

7. ¿Con qué comando muestro los archivos del usuario yoli que no hayan sido leídos en los últimos quince días?
   a) find . –name yoli –atime -15
   b) find . –name yoli –atime +15
   c) find . –user yoli –atime -15
   d) find . –user yoli –atime +15

8. ¿Cómo muestro las entradas de cron que ya están en ejecución?
   a) contrab –s
   b) crontab –ls
   c) crontab –l
   d) crontab

9. ¿Cómo agrego una entrada de crontab?
   a) crontab –a <archivo>
   b) crontab –file <archivo>
   c) crontab –add <archivo>
   d) crontab <archivo>

10. ¿Cuál de las siguientes entradas de cron es incorrecta?
   a) `*  *  *  *  *  find`
   b) `1  3  *  *  7  sh  script`
   c) `*  *  *  *  8  1`
   d) `5  0  1  *  *  ls -l`

TEMA 4. ALTA Y BAJA DEL SISTEMA

Objetivo particular

El alumno reconocerá las fases que involucra el proceso de alta y baja del sistema operativo Unix. Además identificará los conceptos y elementos básicos comprendidos en el alta y baja de sistema, utilizando los comandos o archivos de configuración.

Temario detallado (4 horas)

4.1. Alta del sistema
4.2. Niveles de operación del sistema
4.3. Proceso init
4.4. Archivos de Inicio
4.5. Baja del sistema
4.6. Sincronización memoria-disco
4.7. Comandos *Halt*, *Reboot* y *Shutdown*

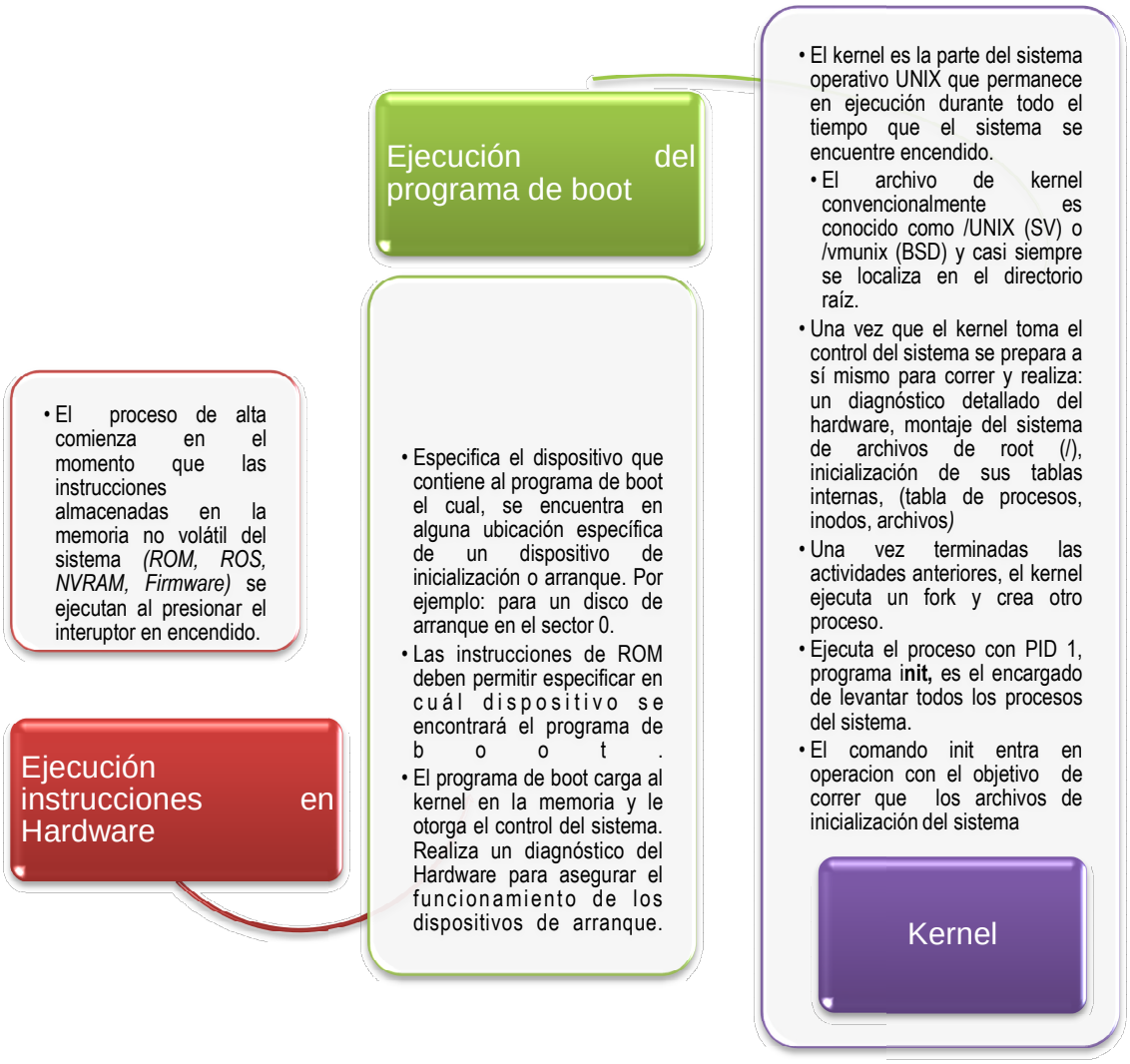
Introducción

El sistema operativo UNIX es un sistema compuesto por multiples elementos, encender o apagar un sistema UNIX es más complicado que solo oprimir un botón. Alta del sistema o *Bootstrapping* es el término utilizado para ''Dar de alta un sistema'' o ''levantar el sistema'', proceso compuesto por varios pasos, que van desde la ejecución de instrucciones de hardware y termina con la operación del sistema operativo. En el sistema UNIX se pueden determinar los programas que se ejecutarán en el momento del encendido e indicará el modo de arranque del sistema, así como cuántos y qué usuarios pueden conectarse al sistema, esto es definido por los niveles del sistema indicados por las versiones del sistema operativo Unix que son: Sistema V (*System V*) introducido por AT&T y BSD(Berkeley *Software Distribution*) creado en la Universidad de Berkeley.

4.1. Alta del sistema

Alta del sistema (*Bootstrapping*). Término utilizado para ''Dar de alta un sistema'' o ''levantar el sistema''. El nombre viene del hecho de que una computadora requiere de su sistema operativo para realizar cualquier cosa, sin embargo, los servicios de éste no están disponibles si el sistema no está dado de alta, de forma que la computadora debe por sí misma ser capaz de ''ponerse las botas''. Comúnmente el término ''bootstrapping'' es abreviado como ''booting''.

El proceso de alta del sistema, consiste en los siguientes pasos:



4.2. Niveles de operación del sistema

En el sistema UNIX se pueden determinar los programas que se ejecutarán en el momento del encendido e indicará el modo de arranque del sistema, así como cuántos y qué usuarios pueden conectarse al sistema.

Las versiones del sistema operativo Unix son: Sistema V (*System V*) introducido por AT&T y BSD (*Berkeley Software Distribution*) creado en la Universidad de Berkeley. Estas versiones de UNIX las adoptaron varios fabricantes generando diferentes distribuciones como lo son: *Solaris, Irix, HP-UX, AIX*, etc. Por ello es importante identificar a qué vertiente pertenece el Unix con cual se esté trabajando, por ejemplo Solaris es el sistema operativo de Sun Microsystems, basado en Sistema V.

Niveles del Sistema en BSD

En general un sistema puede estar en 1 de 3 estados:

Monitor (nivel más bajo)
Monousuario o *single-user* (la máquina está en *stand alone*)
Multiusuario o *multi-user* (la máquina ya tiene usuarios)

Niveles del sistema en SV

0	Monitor, Shutdown o Powerdown La máquina está lista para ser apagada.
1	Estado Administrativo Es el modo de Mantenimiento.
Ss	Single User Modo monousuario.

2	*Multi User* sin red Estado normal de un sistema cuando no está conectado a una red.
3	*Multi User* con red Estado normal de un sistema conectado a una red.
4	Definible por el usuario.
5	*Firmware*. Estado en que el control lo tiene un programa en ROM.
6	*Reboot*. Se utiliza para cambiar de nivel.

Para identificar el nivel en que se encuentra en ese momento el sistema, las instrucciones son: `who –r o runlevel`.

4.3. Proceso init

Es importante diferenciar el comando `init`, del proceso `init`. El proceso `init`, cuyo identificador de proceso es el 1 (PID 1), es el más importante ya que es el encargado de levantar todos los procesos del sistema.

La instrucción (o comando) `init` ejecuta los archivos de inicialización del sistema, su papel primario es crear procesos a partir del archivo /etc/inittab en donde se describen qué procesos se inician en la carga y durante la operación. También se puede cambiar de nivel, para ello se utiliza la instrucción `init` y el identificador del nivel, por ejemplo:

# init 0	Lleva al sistema al nivel más bajo y deja lista la máquina para ser apagada.

4.4. Archivos de Inicio

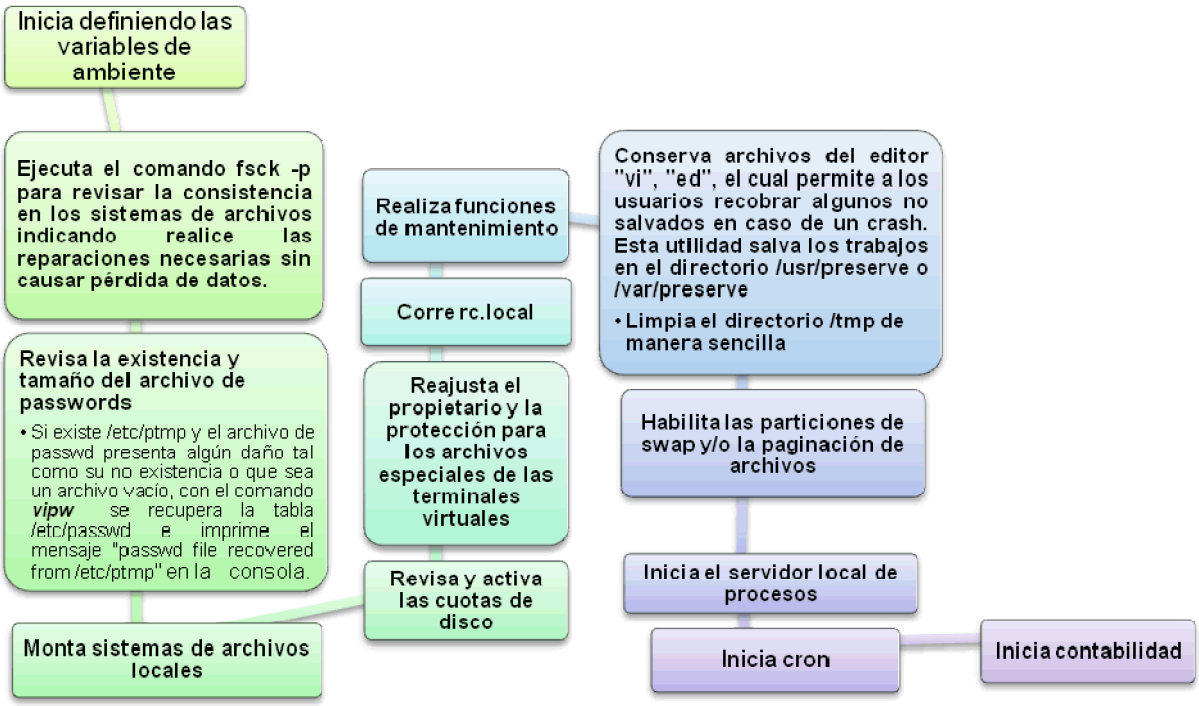
Ejecutan todas las actividades necesarias para que el sistema funcione en un nivel determinado (monousuario, multiusuario).

Es muy importante verificar qué procesos se activan y cualquier modificación que se haga de ellos.

Archivos de Inicio en BSD

/etc/rc /etc/rc.local	Algunos sistemas usan archivos adicionales de inicialización, como `/etc/rc.boot` y `/etc/rc.single`.
/etc/rc.boot	Se encarga de colocar el *hostname* y revisar los sistemas de archivos.
/etc/rc.single	El sistema está iniciando en modo monousuario, el control pasa al intérprete de comandos de *single-user*.

Descripción del proceso de inicialización con el archivo `/etc/rc`



Archivo de inicio `/etc/rc.local`

Coloca el nombre del sistema y su dirección en red, activa la comunicación con otros hosts en la red local. La inmensa mayoría de los rc.local inician varios servicios de red y sus respectivos demonios.

Ejemplos de demonios de red iniciados por `/etc/rc.local`

Demonio(s)	Propósito
routed	Nombre de un hostname dinámico remoto que provee ruteo de datos por TCP/IP

timed	Maneja la sincronización entre los diferentes relojes del sistema en la red local. Si es invocado con la bandera "-M" este sistema puede actuar como reloj maestro, de aquí todos los demonios tomarán el tiempo correcto para sincronizarse a sí mismos. Al menos un sistema en la red debe tener la bandera "-M", de otra manera "timed" será ineficaz.
sendmail	El demonio mail es responsable del ruteo para el correo local y a través de la red.
nfsd	Permite hacer disponible el sistema de archivos local para sistemas remotos vía NFS.
ypbind	Los demonios NIS implementan el servicio de base de datos distribuidas NIS, permitiendo compartir un grupo de archivos de configuración.

Archivos de Inicio en Sistema V

/etc/inittab	Indica las acciones que ejecutará el programa init.
/etc/rc[0-6] /etc/rc.sysinit	Programa que ejecuta acciones durante el inicio o término del funcionamiento del sistema.
/etc/rc[0-6].d	Directorio que almacena los archivos de inicialización de los servicios que serán habilitados según el nivel.
/etc/init.d	Directorio que almacena los vínculos a los programas de inicio que serán ejecutados.

Archivo de inicio**`/etc/inittab`**

Este archivo contiene las acciones que va a tomar el comando `init` para llevar al sistema a un estado en particular.

El formato de este archivo es

cc: estados : acción : comando

Donde:

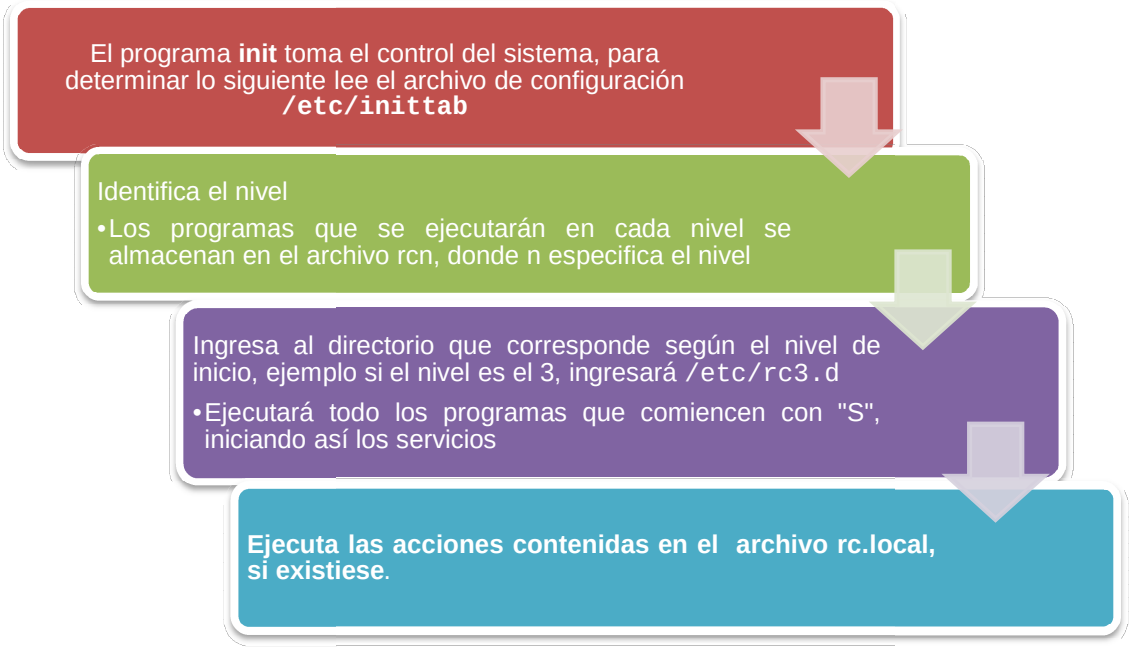
`cc` = etiqueta de 2 caracteres que identifica esa línea.

`estados` = contiene los nombres de los runlevels para los cuales se va a aplicar esa línea. Si está en blanco, aplica para todos.

`acción` = de qué modo se va a procesar esa línea.

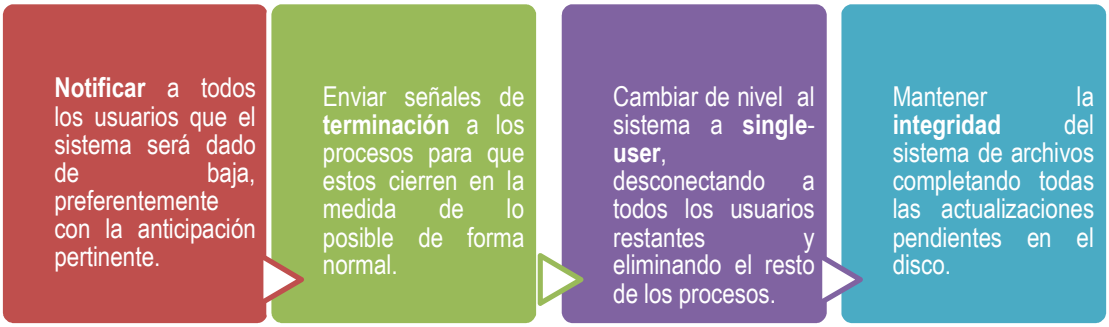
`# al principio` = no procesa la línea, se utiliza para señalar particularidades del sistema.

Descripción del proceso de inicialización con el archivo `/etc/inittab`



4.5. Baja del sistema

En el proceso de baja del sistema se deben considerar los siguientes aspectos:



Para asegurarse de que los pasos anteriores se cumplan UNIX provee el comando *shutdown*, el cual imprime una serie de mensajes a todos los usuarios que están conectados, advirtiendo sobre el cierre del sistema; después de ser enviado el último de estos mensajes, finaliza todas las conexiones de usuarios en el sistema, poniendo la consola en el nivel que se haya definido por defecto para el sistema. Generalmente modo single-user.

4.6. Sincronización memoria-disco

Una de las partes más importantes del proceso de cierre es la sincronización en los discos. El comando `sync` finaliza todas las operaciones en el disco y copia las actualizaciones del superbloque, garantizando que el sistema puede apagarse sin dañar los archivos.

El comando se puede ejecutar manualmente si es necesario, aunque el prompt de UNIX regrese inmediatamente el comando `sync` no desarrolla inmediatamente las escrituras requeridas en disco, por tal motivos se recomienda ejecutarlo hasta tres veces y esperar unos segundos para que la actividad del disco cese antes de hacer algo.

sync

sync

sync

4.7. Comandos Halt, Reboot **y** Shutdown

Debido a que sistema está operando en diversas tareas es importante terminar los procesos de forma adecuada para evitar la pérdida de información y/o inconsistencias en el sistema.

Halt

Lleva al sistema a un punto donde la alimentación se puede suprimir con seguridad, este proceso involucra la detención total del CPU.

Reboot

Detiene el sistema e inmediatamente reinicia la operación del sistema.

Shutdown

Existen dos variaciones del comando shutdown, estas son:

Shutdown en BSD

shutdown tiempo mensaje

El mensaje se manda a los usuarios y se registra en las bitácoras del sistema.

Tiempo	
+m	Minutos
h:m	Cierra el sistema en "h" horas y "m" minutos en

	relación con un reloj de 24 horas
yymmddhhmm	Hora específica
now	Inmediatamente

Opciones	
-h	Halt
-r	Reboot
-k	Simulado, deshabilita las conexiones.
/etc/nologin	Cuando este archivo existe, no permite entrar en sesión.

El nivel por defecto es single user.

Shutdown en SV

shutdown-gn -il [-y]

-gn	Periodo de gracia en segundos
-il	Nivel al que se quiere llevar al sistema
-y	Para que shutdown no nos pida confirmación de nada

Bibliografía básica del tema 4

Frisch, Æleen. (2002) *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media. [Vista previa]

Maxwell, S. (2002).*UNIX System Administration. A Beginner's Guide*. EUA: McGraw-Hill. [Vista previa]

Nemeth, E., Hein, Trent R., Snyder, G. & Whaley, B. (2010). *UNIX and Linux system administration handbook*. (4th ed.) Boston, MA: Prentice Hall. [Vista previa]

Bibliografía complementaria

Shah, S., Soyinka, W. (2007) *Manual de Administración de LINUX.*(4ª ed.) México: McGraw Hill.

Sitios de Internet

Menéndez de Llano Rozas, R. (2007). *AdministraciónBásica de un Sistema UNIX-Linux.*Sitio del Departamento de electrónica y computadores de la Universidad Abierta de Cataluña OpenCourseWare. Disponible en línea: http://ocw.unican.es/ensenanzas-tecnicas/prueba.

Suppi Boldrito, R., Jorba Esteve, J. (2007). *Administración Avanzada de GNU/Linux.* Sitio de la Universidad Abierta de Cataluña OpenCourseWare. Disponible en Internet: http://materials.cv.uoc.edu/cdocent/7FWNKASR7N3XHVRF138B.pdf

Actividades de aprendizaje

A.4.1. Elabora un esquema de los principales elementos que componen el proceso de alta y baja del sistema operativo Unix.

A.4.2. Elabora un cuadro sinóptico donde expliques el funcionamiento de los comandos de alta y baja del sistema operativo Unix.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. Explica el concepto general de los niveles del sistema.
2. ¿En qué consiste el modo “single user”?
3. ¿Qué es el “programa de boot”, cargador o bootloader?
4. En un sistema UNIX basado en Sistema V, ¿qué ocurre si se elimina el archivo /unix?
5. Explica la funcionalidad del proceso init e indica qué ocurre en un sistema UNIX si se elimina dicho proceso.
6. Si se realiza un cambio a la tabla de inittab, ¿se requiere forzosamente reiniciar el sistema para que dicho cambio tome efecto? Justifica tu respuesta.
7. ¿Cuáles son los archivos de inicio principales de BSD?
8. ¿Para qué sirve el comando sync?
9. ¿Cuál es la diferencia entre shutdown, halt y reboot?
10. En UNIX Sistema V, el directorio de inicio /etc/init.d ¿es forzosamente requerido? Justifica tu respuesta.

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. ¿Qué nivel de sistema representa multiusuario?
   a) 4
   b) 2
   c) 1
   d) 0

2. Ejecuta el alto del sistema e inmediatamente reinicia la operación del sistema:
   a) sync
   b) halt
   c) reboot
   d) init

3. ¿Qué opción se utiliza con el comando shutdown para reiniciar un sistema?
   a) –h
   b) –i6
   c) –b
   d) –i0

4. Ejecuta los archivos de inicialización del sistema:
   a) Bootloader
   b) Init
   c) Unix
   d) Shell

5. El nivel normal de operación de un sistema Unix es:
   a) Monitor
   b) Cliente-Servidor
   c) Monousuario
   d) Multiusuario

6. Son archivos o directorios de inicio de Sistema V:
   a) rc y rc.local
   b) inittab e init.d
   c) inittab y rc.local
   d) rc e init.d

7. Se utiliza para verificar la integridad de los sistemas de archivos:
   a) sync
   b) fsck
   c) chkdisk
   d) verify

8. Implica detener la actividad del procesador:
   a) Powerdown
   b) Halt
   c) Monitor
   d) Sync

9. Sirve para identificar de operación del sistema:
   a) telinit
   b) init
   c) who –r
   d) which

10. Nombre del kernel en BSD:
    a) unix
    b) vmlinuz
    c) vmunix
    d) vlinuz

TEMA 5. MANTENIMIENTO, CLAVES DE USUARIOS

Objetivo particular

El alumno reconocerá las fases que involucra el proceso de alta y baja del sistema operativo Unix. Además, identificará los conceptos y elementos básicos comprendidos en el alta y baja de sistema, utilizando los comandos y/o archivos de configuración.

Temario detallado (4 horas)

5.1. Usuarios y Grupos
5.2. Procedimiento para añadir una clave de usuario en el sistema
5.3. Procesos automáticos para dar de alta usuarios
5.4. Baja de usuarios
5.5. Herramientas automáticas para dar de baja cuentas de usuario

Introducción

Un administrador de sistemas debe lidiar tanto con los aspectos operacionales del sistema como con los usuarios del mismo.

Para que un usuario tenga acceso al sistema, el administrador debe otorgarle una cuenta. Esta cuenta debe ser única e intransferible de forma que cada persona que utilice el sistema pueda distinguirse de los demás usuarios.

UNIX proporciona a los usuarios mecanismos que les permiten compartir información, para ello cada usuario debe pertenecer al menos a un grupo de

usuarios. Dependiendo de su cuenta de acceso al sistema y del grupo o grupos a que pertenece, el usuario tendrá ciertos privilegios dentro del sistema.

5.1. Usuarios y grupos

Las tablas principales utilizadas en los sistemas operativos UNIX para definir usuarios y grupos respectivamente son `/etc/passwd` y `/etc/group`.

Usuarios

`/etc/passwd`

Es el archivo donde se configuran los datos de las cuentas de usuario en el sistema.

Se considera una tabla de datos donde cada línea en el archivo corresponde al registro de un usuario en el sistema. Cada registro está compuesto por 7 campos de datos separados por el carácter “dos puntos” (:).

Estructura del archivo `/etc/passwd`

```
login:passwd-cifrado:UID:GID:GECOS:HOME:shell
```

Donde:

`login`	Nombre de la cuenta o clave del usuario.
`password-cifrado`	Password cifrado con un algoritmo criptográfico comúnmente basado en DES (*Data Encryption Standard*) o MD5.

En la mayoría de los sistemas UNIX actuales, el password cifrado ya no se encuentra en esta tabla y ha sido trasladado a la tabla `/etc/shadow` que se explicará más adelante, sin embargo el campo se mantiene por compatibilidad.

`UID`	Identificador del usuario. Mientras que el login o clave de usuario sirve como identificación para un usuario, el sistema en realidad reconoce a este usuario por medio de su número de identificación, el UID.

El sistema trae pre-configurados en su tabla `/etc/passwd` algunos usuarios. El usuario más importante en un sistema UNIX es aquel que cuenta con el UID = 0. En todos los sistemas su login es "`root`". El sistema identifica a dicho usuario como el "superusuario" concediéndole todos los privilegios de "Administrador del Sistema".

Normalmente se acostumbra utilizar números pequeños o inferiores a 100 para cuentas privilegiadas, predefinidas por el sistema o cuentas de aplicaciones, y números a partir de 100 para las cuentas de usuario.

Existen algunos usuarios especiales que aparecen en la mayoría de los casos, estos usuarios generalmente tienen UID entre 1 y 5:

bin	Dueño de los programas ejecutables.
daemon	Dueño de los demonios
sys	Dueño de los archivos del sistema.
adm	Se encarga de los archivos de contabilidad y algunas bitácoras del sistema.

Otros usuarios especiales que se puede encontrar en el sistema son:

operator	Realiza labores de administración, tiene privilegios especiales pero no de root.
nobody	Es la cuenta con menos privilegios en el sistema. Se utiliza para servicios tales como NFS o Web. Este usuario es el usuario con el UID más alto en el sistema (65534) o un número negativo como -2.
GID	Identificador de grupo. Coloca al usuario entre cierta clase de usuarios. (Véase `/etc/group`).
GECOS	Su nombre deriva del campo de información utilizado en el sistema operativo GECOS (*General Electric Comprehensive Operating System*). Sirve para guardar la información de contacto de un usuario tal como su nombre, oficina, teléfono, etc., datos que van separados por una coma (,), aunque en la mayoría de los casos simplemente se pone el nombre del usuario.
HOME	Directorio HOGAR. Es el directorio que se le asigna al usuario. En este directorio van a estar sus archivos. Cuando entra en sesión el sistema lo ubica ahí.
Shell	Intérprete de comandos. Corresponde al intérprete asignado por omisión al usuario. En algunos sistemas, para que un shell se considere válido, debe existir en la lista `/etc/shells`.
chsh	Es el comando de BSD que permite cambiar el shell a otro que esté definido en el archivo `/etc/shells`.

En la actualidad, la mayoría de los sistemas UNIX utilizan además del archivo `/etc/passwd`, una tabla adicional llamada `/etc/shadow`.

`/etc/shadow`

Archivo de acceso restringido que almacena las contraseñas (passwords). Permite además manejar otros elementos relacionados con las cuentas de usuario.

Estructura del archivo `/etc/shadow`

`login:passwd:lastchg:min:max:warn:inactive:expire:flag`

Donde:

`Lastchg`	Número de días transcurridos desde 01-01-1970 y la fecha en que el password fue modificado por última vez.
`Min`	Mínimo número de días que deben transcurrir entre cambios de password.
`Max`	Máximo número de días dentro de los cuales un password es válido.
`Warn`	Indica con cuántos días de anticipación se avisará al usuario que su password va a expirar.
`Inactive`	Número de días de inactividad permitidos para el usuario.
`Expire`	Fecha de expiración de la cuenta.
`Flag`	Campo no utilizado. Reservada para usos futuros.

`# pwconv`

Instala y actualiza el `/etc/shadow` con información del `/etc/passwd`

El único usuario que puede escribir en `/etc/shadow` es `root`. Cuando se cambia el passwd como usuarios, el sistema permite adoptar la identidad de `root` momentáneamente para escribir en ese archivo.

Grupos

Los grupos son el mecanismo que utiliza UNIX para compartir datos y privilegios de acceso

`/etc/group`

Es el archivo donde se definen los grupos del sistema. También sirve para asignar grupos secundarios a los usuarios.

Estructura del archivo `/etc/group`

```
nombregpo:password_gpo:GID:lista_de_usuarios
```

Donde:

`nombregpo`	Nombre del grupo.
`password_gpo`	Contraseña cifrada de grupo. Permite a los usuarios acceder a los privilegios de grupo mediante el uso del comando `newgrp`. En la mayoría de los casos este campo no se usa por lo que permanece en blanco.
`GID`	Identificador de grupo. Coloca al usuario entre cierta clase de usuarios (grupo).

Mientras que el `nombregpo` sirve como identificación para un grupo, el sistema en realidad reconoce a este grupo por medio de su número de identificación, el GID.

El sistema trae pre-configurados en su tabla `/etc/group` algunos grupos.

El grupo al que pertenece el usuario root es normalmente el grupo `root`, GID=0. En BSD este grupo suele llamarse *wheel*.

Se acostumbra utilizar números pequeños o inferiores a 100 para grupos privilegiados, predefinidos por el sistema o grupos de aplicaciones, y números a partir de 100 para los grupos de usuarios.

Existen algunos grupos que aparecen en la mayoría de los casos y generalmente tienen GID entre 1 y 5, éstos son:

`bin`	Grupo para los programas ejecutables.
`daemon`	Grupo para los demonios.
`sys`	Grupo para los archivos del sistema.
`adm`	Grupo para los archivos de contabilidad y algunas bitácoras del sistema.
`Lp`	Grupo para el sistema de impresión.
`kmem`	Grupo para los el manejo de memoria.

Otros grupos especiales que se pueden encontrar en el sistema son:

`Operator`	Grupo para labores de operación del sistema.
`nobody, nogroup o noaccess`	Son los grupos con menos privilegios en el sistema. Se utilizan para servicios tales como NFS o Web. Estos grupos son el GID más alto en el sistema (65534).
`lista_de_usuarios`	Permite especificar usuarios que pertenecen al grupo como grupo secundario.

Asignación de grupos

Existen dos formas en que el administrador puede asignar grupo al usuario:

- Implícitamente: Sólo en `/etc/passwd`
- Explícitamente: `/etc/group`

El grupo primario es aquel definido de forma implícita en el archivo `/etc/passwd`. No es necesario definirlo en el `/etc/group`.

Además de su grupo primario, un usuario puede pertenecer a otro grupo, dicho grupo debe asignarse en forma explícita y se considerará un grupo secundario.

Consideraciones

En BSD un usuario pertenece simultáneamente a todos los grupos asignados.

En el caso de SV, un usuario puede pertenecer a un solo grupo a la vez. Cuando el usuario entra en sesión, pertenece a su grupo primario, si desea cambiar sus credenciales de grupo para adquirir privilegios de su grupo secundario, el usuario deberá utilizar el comando newgrp.

Comandos para manejo de grupos

`id`	Muestra las credenciales de identificación de usuario.
En BSD	`gid` ==grupo primario.
En SV	`gid` == grupo al que pertenece en ese momento el usuario.
`Groups`	Muestra la lista de grupos de un usuario.
`Newgrp`	Permite al usuario ingresar a un nuevo grupo. (Sólo con SV).

Administración

`groupadd`	añadir
`groupdel`	borrar
`groupmod`	modificar

5.2. Procedimiento para añadir una clave de usuario en el sistema

1. Asignar un nombre (login)

El login debe ser asignado de acuerdo con las políticas del sitio. Junto con el login se debe asignar un UID y un grupo (GID). El UID debe ser único para cada usuario.

2. Modificar las tablas correspondientes

Si el grupo es nuevo, deberá editarse la tabla `/etc/group` para darlo de alta.

Editar las tablas `/etc/passwd`, `/etc/shadow`, `/etc/group` (en caso de asignación de grupo nuevo o grupos secundarios).

En el caso de BSD no existe `/etc/shadow`, en su lugar existe un archivo denominado `master.passwd`. A partir de este archivo se genera el archivo `/etc/passwd`, el cual debe ser compilado utilizando `pwd_mkdb`.

En algunos sistemas puede existir:

`/etc/passwd.dir`

`/etc/passwd.pag`

En este caso significa que la tabla debe compilarse utilizando el comando `mkpasswd`.

3. Asignar un password

4. Crear su directorio HOME

5. Poner archivos de inicio en su directorio HOME

.login .profile .cshrc

Una forma de que los nuevos usuarios del sistema tengan la misma configuración inicial, es utilizando un directorio denominado skel (esqueleto), que

continiene los archivos y/o directorios que se copian cuando se genera una cuenta.

Los esqueletos de estos archivos usualmente se encuentran en:

```
/etc/skel
/usr/skel
```

6. Cambiar el dueño y el grupo a los correspondientes para el usuario.

```
# chown -Rh login directorio_home_del_usuario
# chgrp -Rh grupo directorio_home_del_usuario
```

7. Dar de alta la clave en cualquier otro sistema requerido para el usuario.

- mail
- tablas de impresión
- cuotas

8. Probar la clave

5.3. Procesos automáticos para dar de alta usuarios

SV	passmgmt, useradd
SOLARIS	passmgmt, useradd
AIX	useradd, mkuser, SMIT
HP-UX	useradd, smh, sam
BSD	Adduser
OpenBSD	user, useradd, adduser
Linux	useradd

5.4. Baja de usuarios

Existen diversas razones por las cuales se requiere dar de baja una cuenta de usuario en el sistema. Los principales motivos son:

- Cuenta temporal. (Se abrió para un proyecto determinado o por un periodo específico de tiempo y ha concluido su validez).
- Transferencia de la cuenta a otra máquina (migración).
- El usuario dejo de trabajar en la empresa o institución, o cambió su adscripción.
- Por violación a las reglas del sistema (abuso, problema de seguridad, etc.).

Cuando *una cuenta se da de baja*, ésta puede ser:

1) Borrada permanentemente.
2) Deshabilitada.
3) Deshabilitada y luego borrada.

Procedimiento para *deshabilitar una cuenta de usuario*

1) Poner un asterisco (*) en el campo de passwd-cifrado. Esto será realizado en la tabla correspondiente de acuerdo con la configuración del sistema.
2) Verificar que el usuario no tenga activado un mecanismo de confianza para el acceso a su cuenta (`.rhosts, .shosts, .ssh/authorized_keys`).
3) Cambiar el shell de usuario. Se recomienda poner un shell no válido. El más común es `/bin/false`.

Borrar una cuenta

Consideraciones. De acuerdo con las políticas del sitio, se debe comunicar al usuario que se dará de baja su cuenta. Preguntar al usuario si ya realizó respaldos, y en caso contrario, darle un límite de tiempo para ello.

Pasos a seguir

1) Eliminar cualquier proceso del usuario en el sistema.
2) Revisar que no haya dejado trabajos en ejecución automática o calendarizada (`at, cron`).
3) Borrar las líneas correspondientes de las tablas de configuración apropiadas (`/etc/passwd`, `/etc/shadow`).
4) Eliminar cualquier definición de la clave de usuario en los grupos secundarios o si el usuario tiene un grupo personal, eliminarlo (`/etc/group`).
5) En aquellos sistemas que lo requieran, recompilar la tabla de usuarios.
6) Dar de baja la cuenta de cualquier otro servicio (mail, alias, impresión, cuotas).
7) Borrar su directorio HOME.
8) Realizar una búsqueda de cualquier otro archivo perteneciente al usuario en el sistema y borrarlo.

5.5. Herramientas automáticas para dar de baja cuentas de usuario

SV	passwdmgmt -d <usuario>, userdel <usuario>
SOLARIS	Userdel
AIX	userdel, SMIT
HP-UX	userdel, smh, sam
OpenBSD	Rmuser
Linux	userdel

Nota: En todos los casos en los que se borren usuarios con las herramientas, verificar después de utilizarlas.

Bibliografía básica del tema 5

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Nemeth, E., Hein, Trent R., Snyder, G. & Whaley, B. (2010). *UNIX and Linux system administration handbook*. (4th ed.) Boston, MA: Prentice Hall. [Vista previa]

Bibliografía complementaria

Maxwell, S. (2002).*UNIX System Administration. A Beginner's Guide*. EUA: McGraw-Hill.

Smith, R. W. (2009). *LPIC-1 Linux Professional Institute Certification. Study Guide.* (2ª ed.) Indianápolis, IN: Wiley Publishing, Inc

Pons, Nicolás. (2009). "Permisos de acceso a usuarios". *LINUX, Principios básicos del uso del sistema*, Barcelona, Eni, cap. 7, pp. 145-150. Disponible en línea: http://www.editions-eni.fr/_Download/1f6a0d7b-226d-43fb-a909-ba7354beb8f2/Linux_(Extracto-del-Libro).pdf, consultado el 29/07/11.

Sitios de Internet

González Barbone, V. A. (2002). “Superusuario y usuarios especiales”. *Curso de Administración UNIX*. Instituto de Ingeniería Eléctrica, Facultad de Ingeniería, Universidad de la República, Uruguay. Disponible en línea: http://iie.fing.edu.uy/ense/asign/admunix/superusu.htm

Jorba Esteve, J. (2010). “Administración Local. Administración de sistemas GNU/Linux”. *OpenCourseWare.* Universidad Abierta de Cataluña. Disponible en línea: http://materials.cv.uoc.edu/continguts/PID_00157329/web/main/materias/PID_00157328-5.pdf

Actividades de aprendizaje

A.5.1. Elabora un mapa mental donde describas el procedimiento de alta, baja y modificación de las claves de usuarios.

A.5.2. Elabora un cuadro sinóptico en el que especifiques las consideraciones que el administrador de sistemas toma en cuenta al crear, eliminar y modificar una cuenta de usuario.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. Explica la diferencia entre nombre de usuario (login) e identificador de usuario (UID).
2. ¿En qué archivo se define el grupo primario de un usuario?
3. ¿Qué significa y para qué sirve el campo GECOS en la tabla `/etc/passwd`?
4. ¿Cuál es la utilidad del archivo `/etc/shadow`?
5. ¿Cuáles son los archivos de inicio que se colocan en el directorio HOGAR (HOME) de los usuarios y en dónde se encuentran dichos archivos?
6. Explica la relación entre los conceptos grupo primario y grupo secundario, y el funcionamiento del comando newgrp en Sistema V.
7. ¿Qué es el directorio HOGAR?
8. ¿Cuáles son las formas en que el administrador puede asignar un grupo al usuario?
9. ¿Para qué se utiliza el usuario nobody?
10. Menciona 2 formas diferentes para desactivar una clave de usuario.

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. ¿Cuál de los siguientes campos no es requerido obligatoriamente para abrir una cuenta de usuario en el sistema UNIX?
   a) Login
   b) UID
   c) GID
   d) HOME

2. ¿Cuál de los siguientes campos no se encuentra en el `/etc/passwd`?
   a) Login
   b) Password cifrado
   c) HOME
   d) Nombre de grupo

3. Programa que Instala y actualiza el archivo `/etc/shadow` con información `/etc/passwd:`
   a) psswd
   b) passmgmt
   c) pwconv
   d) vipw

4. Son archivos que almacenan la definición usuarios y grupos en UNIX:
   a) `/etc/passwd` y `/etc/group`
   b) `/etc/shells` y `/etc/group`
   c) `/etc/passwd` y `/etc/skel`
   d) `/etc/passwd` y `/etc/shells`

5. Herramienta automática para creación de cuentas en Sistema V:
   a) adduser
   b) passmgmt
   c) mkuser
   d) admintool

6. Permite cambiar el grupo asociado a un archivo:
   a) newgrp
   b) groupmod
   c) vigrp
   d) chgrp

7. Grupo con privilegios para administrador del sistema en Unix:
   a) sys
   b) Wheel
   c) bin
   d) adm

8. ¿Cuáles son los permisos por omisión del archivo `/etc/shadow`?
   a) 644
   b) 640
   c) 444
   d) 400

9. Carácter comúnmente utilizado para desactivar una cuenta:
   a) #
   b) @
   c) *
   d) ~

10. Shell utilizado por los sistemas UNIX para desactivar una cuenta:
    a) /usr/bin/denylogin
    b) /bin/false
    c) /etc/nologin
    d) /var/empty

TEMA 6. INSTALACIÓN Y MANTENIMIENTO DE DISPOSITIVOS

Objetivo particular

El alumno identificará las actividades y consideraciones involucradas en la puesta en marcha y el mantenimiento de dispositivos, así como también reconocerá las herramientas para uso éstos.

Temario detallado (6 horas)

6.1. Impresoras
6.2. Terminales

Introducción

La utilización de cualquier sistema genera la necesidad de habilitar dispositivos nuevos, ya que cuando fue instalado el sistema no se encontraban, por ello se deben configurar en un momento posterior. Para ello el sistema operativo Unix posee un robusto sistema para la configuración de terminales e impresoras.

Dentro del sistema operativo Unix se encuentran los programas de los dispositivos, es decir, los controladores que permiten el funcionamiento del dispositivo, si éstos no se encuentran, el dispositivo no funcionará y si lo hace lo hará de forma inadecuada. Para solucionarlo se puede hacer de forma modular que es utilizando el controlador para la versión de Unix que se esté utilizando o reconfigurar el kernel.

El acceso a los dispositivos es a través de archivos especiales, que convencionalmente se encuentran en el directorio /dev (y sus subdirectorios). El kernel transforma las operaciones sobre estos archivos especiales en llamadas a los controladores.

6.1. Usuarios y grupos

El sistema de impresión en UNIX tiene como funciones:

- Registrar los requerimientos de impresión por parte de los usuarios.
- Registrar el trabajo de las impresoras.
- Arrancar los programas encargados de imprimir.
- Filtrar los archivos de los usuarios (si es necesario) para que se impriman adecuadamente.
- Llevar un seguimiento del estado de los trabajos.
- Alertar sobre posibles problemas de impresión.

Para realizar estas funciones el sistema de impresión se compone de lo siguiente:

1. Comandos de usuario
El usuario los utiliza para imprimir, borrar o ver sus trabajos (interactúa con el sistema de impresión).

2. Demonio de impresión
Es el proceso que proporciona el servicio de impresión. Atiende las peticiones de los usuarios.

3. Comandos administrativos
Comandos que permiten decirle al demonio que ejecute ciertas acciones o manejar las colas de impresión.

4. Colas de impresión (directorios de spool)

Es el lugar donde se van a realizar los registros temporales de los archivos que se van a imprimir.

Sistema de impresión en BSD

1. Comandos de usuario: lpr, lpq, lprm

`lpr`	Coloca el archivo en la cola apropiada, copia el archivo al directorio de spool. `% lpr -Pimpresora archivo`
`lpq`	Sirve para ver que trabajos se están imprimiendo (da el estado de la cola de impresión). `% lpq -Pimpresora`
`lprm`	Borrar trabajos de la cola de impresión. `% lprm -Pimpresora num_job`

2. Demonio: `lpd`

`lpd` (LP daemon) Lee los archivos de spool y los manda a la impresora.

`/usr/lib/lpd`

`/usr/sbin/lpd` se ejecuta desde `/etc/rc.local`

3. Comando de administración: `lpc`

`lpc` Herramienta administrativa (*line printer control*). Es un intérprete de comandos que permite manipular el sistema de impresión.

`lpc>` comando

Comandos de `lpc`

`status`	Permite ver el estado de los trabajos encola
`abort`	Aborta el demonio y desactiva la impresión
`stop`	Deshabilita el demonio de impresión (primero se vacía la cola de trabajos pendientes)
`start`	Habilita el demonio de impresión
`disable`	Desactiva la recepción de trabajos
`enable`	Activa la recepción de trabajos
`down`	Deshabilita el demonio de impresión y desactiva la recepción de trabajos
`up`	Habilita el demonio de impresión y activa la recepción de trabajos
`topq`	Pone los trabajos al principio de la cola de impresión

4. Directorio de spool: `/usr/spool/lpd`

Es responsabilidad del administrador crear el directorio de spool que pertenece a:

usuario	grupo	permisos
daemon	daemon	755

En algunos sistemas puede variar el usuario y el grupo.

Archivo de configuración de impresoras

`/etc/printcap`

Base de datos donde se definen todas las impresoras dadas de alta en el sistema.

Está conformado de la siguiente forma:

`nombre1/nombre2|` ... : `\campo:campo`:

Ejemplo:

```
splash:\
        :lp=/dev/lp:\
        :mx=#0:\
        :sd=/usr/spool/lpd/splash:
```

Cada impresora puede tener varios nombres, la que se llame lp será considerada como la impresora por defecto, siempre y cuando lp sea el primer nombre en la lista de nombres. Después de modificar el archivo printcap hay que reiniciar el demonio lpd para que los cambios surtan efecto.

Procedimiento para añadir una impresora local

1) Conectarla físicamente
2) Verificar que lpd este activado en `/etc/rc.local`
3) Añadir un registro en `/etc/printcap`
4) Crear el directorio de spool (daemon daemon 755)
5) Crear el archivo de contabilidad o filtro (si corresponde)
6) Activar la cola de impresión
7) Probarla

Procedimiento para añadir una impresora remota

En el caso de impresoras remotas, se trabaja en modo cliente-servidor, es decir, el cliente es aquella máquina que enviará trabajos a impresión y el servidor es aquella que tiene conectada la impresora.

Es común que algunas impresoras tengan interfaces de red, en este caso la propia impresora será considerada el servidor.

Configuración del cliente

Dar de alta la impresora en `/etc/printcap` indicando la dirección IP del servidor y en su caso, el nombre de la impresora tal como se conoce en el propio servidor. También se debe indicar un directorio de spool.

Ejemplo:

```
splash:\
:lp=:\
        :mx=#0:\
:rm=192.168.115.19:\
        :rp=xerox2:\
        :sd=/usr/spool/lpd/splash:
```

Nota: En este ejemplo la impresora remota se llama xerox2 y está conectada al sistema remoto con IP 192.168.115.19.

Configuración del servidor

Para que un cliente sea aceptado en un servidor de impresión BSD, se debe dar de alta en el archivo `/etc/hosts.lpd`. Algunos sistemas requieren que el host sea conocido, es decir que esté dado de alta en `/etc/hosts`.

Sistema de impresión en System V

1. Comandos de usuario: lp, lpstat, cancel

`lp`	Coloca el archivo en la cola apropiada, copia el archivo al directorio de spool. `% lp -d impresora archivo`

`lpstat`	Sirve para ver qué trabajos se están imprimiendo (da el estado de la cola de impresión).
	`%lpstat`
	`%lpstat -pimpresora`
	`%lpstat -d`
	`%lpstat -t`

`cancel`	Borrar trabajos de la cola de impresión. `%cancel num_job`

2. Demonio: lpsched

`lpsched` (LP Scheduler) Lee los archivos de spool y los manda a la impresora.

`/usr/lib/lpsched`

Se habilita desde `/etc/rc2.d/S??lp`

3. Comandos de administración

`lpshut` Desactiva el servicio de impresión.

`disable` Deshabilita el demonio de impresión.

`enable` Habilita el demonio de impresión

`reject` Desactiva la recepción de trabajos

`accept` Activa la recepción de trabajos

`lpmove` Mueve los requerimientos a otro destino.

```
# lpmove jobimpresora
# lpmove impresora1 impresora2
```

`Lpusers` Cambia la prioridad en la cola.

4. Directorio de spool: /usr/spool/lp

Clases de impresoras

El System V permite agrupar impresoras de un mismo tipo. Una clase es una agrupación. Las impresoras pueden pertenecer a la clase por omisión o a una clase definida por el administrador del sistema.

El comando que permite manipular las impresoras/clases es lpadmin.

```
# lpadmin -p impresora -c clase    (añadir)
# lpadmin -x <impresora o clase>   (borrar)
# lpadmin -p impresora -r clase    (quitar)
```

Procedimiento para añadir una impresora local

1. Conectar físicamente
2. Asegurarse que exista una liga de /etc/init.d/lp a un directorio de inicio del sistema.

```
     /etc/rc2.d/S??lp
/etc/rc0.d/K??lp
```

3. Matar al demonio lpsched

```
% lpshut
```

4. Añadir impresora con

```
% ladmin -p impresora
```

5. Levantar al demonio

```
# lpsched
```

6. Habilitar la impresora y la cola de impresión

```
# accept
# enable
```

7. Probar

Procedimiento para añadir una impresora remota

Configuración del servidor

Para configurar el servidor se utiliza el comando `lpsystem`. Se debe indicar si el servidor de impresión es de tipo bsd o sv.

lpsystem

Inserta una línea en el archivo

`/etc/lp/Systems o /etc/printers.conf`

```
# lpsystem -t bsd host        (para un cliente BSD)
# lpsystem -t s5 host         (para un cliente SV)
```

Configuración del cliente

Para configurar el cliente, la impresora se dará de alta con lpadmin, indicando el tipo de servidor al que se enviarán los trabajos de impresión.

```
# lpadmin -p imp1 -T s5  -s host   (para un servidor BSD)
# lpadmin -p imp1 -T bsd -s host   (para un servidor SV)
```

6.2. Terminales

Las terminales son en esencia el modo utilizado para escribir o mostrar datos en un sistema de computación.

El kernel de UNIX proporciona el mecanismo para acceder a terminales seriales típicamente conocidas como tty palabra que es una abreviatura de *TeleTYpe writer*.

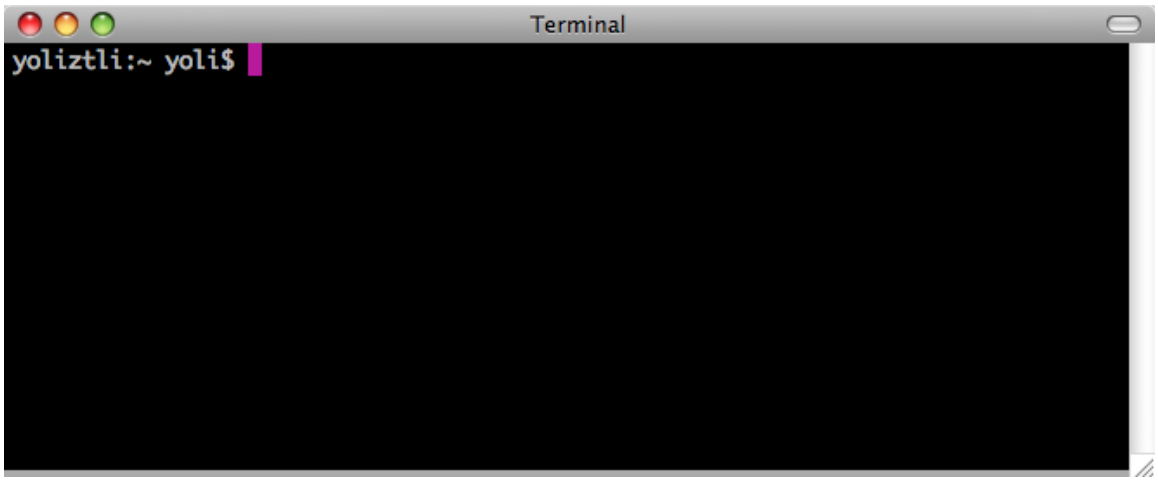
A principios de los años 70 la “terminal” era una teleimpresora de salida secuencial. Estas máquinas TTY no tenían ningún tipo de video. Los usuarios mecanografiaban los datos y éstos aparecían impresos. Ese fue el origen de la terminal de texto y la línea de comandos.

Con el paso del tiempo, los ttys originales fueron reemplazados por monitores. Y en la actualidad hay sistemas que cuentan con pantallas con capacidad de video.

La palabra terminal, sin embargo, se sigue utilizando para hacer referencia a la operación del sistema en modo texto.

Se puede decir que las terminales son la forma de acceder al sistema sin utilizar la interfaz gráfica del mismo, permiten realizar todo tipo de tareas en formato de texto.

Cuando se ingresa a una terminal el sistema muestra el símbolo del sistema o prompt el cual indica que el sistema está listo para recibir órdenes.



Comúnmente se conoce a la terminal como consola, sin embargo, es importante no confundir ambos términos ya que corresponden a dispositivos diferentes, mismos que se mencionan a continuación:

Dispositivos Especiales

Consola del sistema

`/dev/console`

Representa la consola del sistema. Aquí se muestran los errores del sistema. Cuando se usa como una terminal regular, `/dev/console` hace referencia a ese dispositivo terminal. Cuando una sesión de sistema de ventanas está siendo ejecutada, `/dev/console` puede convertirse en una de estas ventanas.

Terminal

`/dev/tty`

El archivo especial `/dev/tty` tiene un propósito especial. Es un sinónimo para cada controlador de proceso TTY. Puede usarse para asegurar que la salida se dirige hacia la terminal, independientemente de cualquier redirección de E/S.

Terminales físicas

Son aquellas que están conectadas directamente al sistema mediante un cable.

Como todo dispositivo en UNIX, los puertos seriales se manejan a través de archivos especiales localizados en el directorio /dev

Los nombres de los archivos especiales varían según el sistema operativo, siendo los más comunes:

```
/dev/ttySn      BSD
/dev/ttyn       SV
```

donde "n" es un dígito correspondiente al número de la línea serial (comenzando con 0).

Pseudoterminales

Se refiere a aquellas terminales que permiten manejar conexiones indirectas establecidas mediante:
Un manejador de ventanas
A través de la red

Para cada pseudoterminal UNIX utiliza dos archivos especiales:

Pseudoterminal maestra o controladora
El kernel la usa para controlar las operaciones de esa terminal.

```
/dev/pty[p-s]n
/dev/ptcn
/dev/ptc/n  (en SV4)
```

Ejemplo: `/dev/ptyq5`

Pseudoterminal esclava

Es la terminal en la que está trabajando el usuario, está controlada por la pseudoterminal maestra.

`/dev/tty[p-s]n`

`/dev/pts/n  (en SV4)`

Ejemplo: `/dev/ttyq5`

Las dos partes trabajan en pares, teniendo el mismo número de dispositivo n.

El usuario solo ve el dispositivo de la pseudoterminal esclava.

La siguiente es una lista de los nombres de los dispositivos especiales tanto para terminales físicas como virtuales:

Versión UNIX	Terminal Física o Línea Serial	Pseudo Terminal Maestra	Pseudo Terminal Esclava
HP-UX 10	`/dev/tty0p0`	`/dev/ptyp0` `/dev/ptym/ptyp0`	`/dev/ttyp0` `/dev/pty/ttyp0`
Solaris	`/dev/term/a` `/dev/ttya`	`/dev/ptyp0`	`/dev/ttyp0` `/dev/pts/0`
AIX	`/dev/tty0`	`/dev/ptc`	`/dev/pts/0`
Linux	`/dev/ttyS0`	`/devptmx` `/dev/ptyp0`	`/dev/pts/0` `/dev/ttyp0`
BSD	`/dev/ttyS0` `/dev/ttyd0`	`/devptmx` `/dev/ptyp0`	`/dev/pts/0` `/dev/ttyp0`

Bajo Solaris, los pseudodispositivos son todas las ligas en el directorio `/devices/pseudo.`

TTY

El comando tty muestra cuál archivo especial está siendo usado para cada sesión registrada. Por ejemplo:

```
$ tty
/dev/pts/0
```

Bibliografía básica del tema 6

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Nemeth, E., Hein, Trent R., Snyder, G. & Whaley, B. (2010). *UNIX and Linux system administration handbook*. (4th ed.) Boston, MA: Prentice Hall. [Vista previa]

Bibliografía complementaria

Strang, J., Mui, L., O´Reilly, T. (1992). *Termcap & Terminfo*. Sebastopol, CA: O'Reilly & Associates, Inc. [Vista previa]

Sitios de Internet

Jorba Esteve, J. (2010). "Administración Local. Administración de sistemas GNU/Linux". *OpenCourseWare.* Universidad Abierta de Cataluña. Disponible en línea: http://materials.cv.uoc.edu/continguts/PID_00157329/web/main/materias/PID_00157328-5.pdf.

Meléndez, L. (2004). "Funcionamiento de las terminales en UNIX". Área de Sistemas y Comunicaciones. Centro de Cálculo Científico. Universidad de Córdoba, España. Disponible en línea: http://www.uco.es/ccc/sistemas/doc_ccc/terminales.html.

Actividades de aprendizaje

A.6.1. Elabora un esquema de los principales elementos que componen el sistema de impresión del sistema operativo Unix.

A.6.2. Elabora un mapa mental donde describas el procedimiento para agregar una impresora en SV y BSD.

A.6.3. Elabora un cuadro sinóptico donde expliques los tipos de terminales en el sistema operativo Unix.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿Cuáles son los principales componentes del sistema de impresión en UNIX?
2. Antes de poner en marcha un sistema de impresión, se debe eliminar cualquier trabajo que esté encolado. ¿Qué instrucción se emplea para ello?
3. ¿Cómo se configura una impresora remota en Sistema V?
4. ¿Cómo se configura la impresora por omisión en BSD?
5. ¿Cuál es la diferencia entre una pseudoterminal y una terminal virtual?
6. Explica la diferencia entre una terminal y una consola.
7. ¿Qué es una terminal física?
8. ¿Cuál es la funcionalidad del comando `tty`?
9. ¿Cuál es la instrucción que detiene la recepción de trabajos en Sistema V?
10. ¿Cuál es la instrucción que detiene la impresión de trabajos en BSD?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. En un sistema BSD, los usuarios no pueden enviar impresiones desde su máquina local. ¿Qué archivo se debe modificar para corregir dicho problema?
   a) `/etc/hosts`
   b) `etc/hosts.allow`
   c) `/etc/host.lpd`
   d) `etc/hosts.deny`

2. Comando que permite administrar el sistema de impresión en BSD:
   a) `lpsched`
   b) `lpadmin`
   c) `lpc`
   d) `lpdaemon`

3. Archivo de configuración de impresoras en BSD:
   a) `/etc/printcap`
   b) `/etc/printers.conf`
   c) `/etc/init.d/lp`
   d) `/etc/lp/Systems`

4. Permite añadir una impresora en SV:
   a) `lpc –P impresora`
   b) `lpadmin –p impresora`
   c) `enable impresora`
   d) `vi /etc/printcap`

5. Comando que permite cambiar la prioridad de un trabajo, poniéndolo al principio de la cola de impresión en BSD:
   a) `topq`
   b) `lpuser`
   c) `toplp`
   d) `lpadmin`

6. Archivo que se utiliza como controlador de cada proceso TTY:
   a) `/etc/ttys`
   b) `/dev/console`
   c) `/dev/tty`
   d) `/usr/bin/tty`

7. Corresponde a un nombre para una pseudo terminal esclava en Linux:
   a) `/dev/tty0`
   b) `/dev/ptyp0`
   c) `/dev/ttyp0`
   d) `/dev/ptc/0`

8. Muestra el archivo especial asociado a la sesión del usuario:
   a) `stty`
   b) `TERM`
   c) `xterm`
   d) `tty`

9. Símbolo del sistema, indica que está listo para recibir órdenes:
    a) Consola
    b) Shell
    c) Prompt
    d) Terminal

10. Dispositivo donde se muestran por omisión los errores del sistema:
    a) `/dev/tty`
    b) `/dev/pst/0`
    c) `/dev/console`
    d) `/dev/null`

TEMA 7. SISTEMA DE ARCHIVOS

Objetivo particular

Al término del tema el alumno podrá:
Reconocer la importancia del sistema de archivos de Unix, así como los principales elementos que lo componen y su relación con los dispositivos. Además será capaz de añadir,configurar y monitorear sistemas de archivos locales y en red.

Temario detallado (6 horas)

7.1. Archivos y discos duros
7.2. Formato físico
7.3. Particiones
7.4. Archivos de dispositivo
7.5. Creación de sistemas de archivos
7.6. Montaje de sistemas de archivos
7.7. Monitoreo
7.8. Cuotas
7.9. NFS (*Network File System*)

Introducción

La administración del sistema de archivos (*File System*) es una de las actividades más importantes de un administrador de sistemas debido a que la información reside en los sistemas de archivos.

El administrador es el encargado de:

- Poner los sistemas de archivos disponibles para los usuarios.
- Monitorear y administrar el espacio en los dispositivos de almacenamiento secundario.
- Asegurar la integridad de los sistemas de archivos.
- Asegurar la confidencialidad de la información.
- Verificar y corregir fallas en los sistemas de archivos.
- Configurar y conectar nuevos dispositivos de almacenamiento secundario cuando se requiera.

7.1. Archivos y discos duros

Un archivo es “una colección de información relacionada con un nombre”. (Silberschatz, Galvin y Gagne, 2005, p. 353) En UNIX, el concepto de archivo es muy importante, ya que todo en este sistema se considera como un archivo.

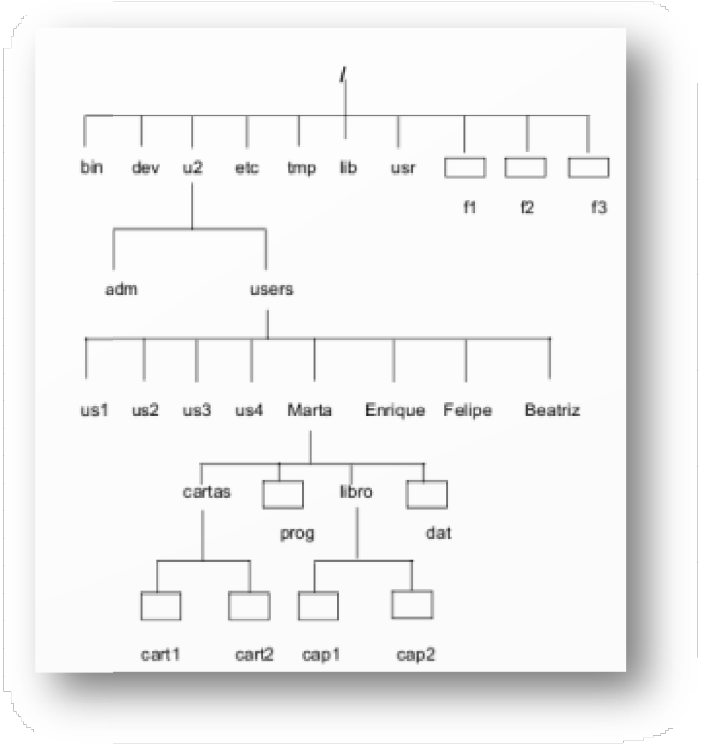
“El sistema no impone estructura alguna a los archivos, ni asigna significado a su contenido; el significado de los bytes depende únicamente de los programas que interpreta el archivo”. (Kernighan y Pike, 1987, p. 43)

El sistema operativo maneja los archivos como simples flujos de bytes, lo cual le permite tener una interfaz única para el manejo de la información.

En el caso de los dispositivos, los datos que fluyen en interconexiones no son más que una secuencia de bytes, por lo que desde el punto de vista del sistema pueden ser considerados como archivos.

Los directorios son “archivos que permiten localizar otros archivos dentro de la estructura del sistema de archivos”. (Sánchez, 1999, p. 32)

El sistema de archivos es la estructura de tipo jerárquico basada en un modelo arborescente y recursivo, en donde los nodos pueden ser tanto archivos como directorios, y estos últimos pueden contener a su vez directorios o subdirectorios.

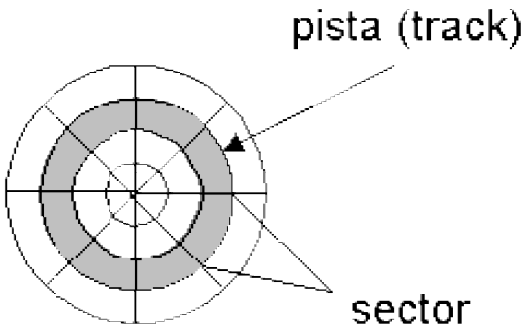


El sistema de archivos organiza la información en un dispositivo de almacenamiento secundario.

El disco duro es la forma más común de almacenamiento secundario, consiste en un:

> Soporte de almacenamiento de información de acceso directo, es decir, se puede acceder a una determinada información sin necesidad de pasar por toda la información anterior. Un plato es una base metálica sobre la que hay una capa de material magnetizable en la que se registra la información en puntos sobre pistas concéntricas divididas en sectores, y estas a su vez en bloques. (Alcalde, García y Peñuelas, 1988, p. 97)

Cada superficie, o plato, está dividida en pistas concéntricas. Una *pista* (*track*) es una porción del disco que pasa bajo una cabeza durante una rotación del disco. Cada una de ellas está dividida en segmentos llamados *sectores*, los cuales son la unidad mínima de información que puede leer o escribir un disco duro. Generalmente, cada sector almacena 512 bytes. Asimismo, un **cilindro** está compuesto de un conjunto de pistas descritas por todas las cabezas (en platos separados).



7.2. Formato físico

Antes de poder utilizar un disco duro, este debe tener un formato que permita al sistema operativo reconocer su geometría.

El *formateo* consiste en colocar los códigos en la película magnética sobre la superficie del disco, que divide la superficie en sectores, bloques, pistas y cilindros.

En la mayoría de los casos, el formato ya viene realizado por el fabricante; sin embargo, puede requerirse formatear un disco duro por algún motivo en particular.

El proceso de formato de un disco también identifica bloques-defectuosos o imperfecciones en el medio que resultan en áreas que no pueden ser utilizadas confiablemente para leer o escribir.

El programa que sirve para formatear en la mayoría de los sistemas UNIX es:

```
# format
```

En el caso de Linux y BSD, el comando es:

```
# fdisk
```

7.3. Particiones

> Cada disco duro constituye una unidad física distinta. Sin embargo, los sistemas operativos no trabajan con unidades físicas directamente, sino con unidades lógicas. Dentro de una misma unidad física de disco duro puede haber varias unidades lógicas. Cada una de estas unidades lógicas constituye una partición del disco duro. (Barajas, 2001)

Se puede considerar que una partición corresponde a un conjunto de cilindros asignados a una unidad lógica.

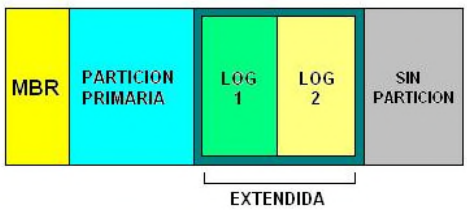
Un disco duro al menos debe tener una partición, pues es ahí donde residirán los sistemas de archivos.

Tradicionalmente, los sistemas escriben las particiones en un registro denominado "etiqueta" (*label*) del disco, la que se almacena en los primeros bloques del disco. El contenido de esta varía entre sistemas, pero usualmente contiene, además de la tabla de particiones, información que permite localizar el kernel al momento del inicio del sistema. Esta etiqueta en Windows es conocida como Master Boot Record (MBR).

En arquitecturas x86, un disco duro solo puede tener cuatro particiones primarias.

Las *particiones primarias* son aquellas que se pueden direccionar directamente desde el MBR.

Si un sistema requiere un esquema de particiones superior a cuatro, debe utilizar una partición extendida y particiones lógicas.



La *partición extendida* es primaria especial, utilizada para crear una o más particiones lógicas sobre ella. Esto significa que la partición extendida no tiene razón de ser por sí sola, sino que tiene que existir al menos una partición lógica creada sobre ella. La idea principal de la partición extendida es romper la limitación del MBR sobre la cantidad de particiones que puede tener un disco duro.

Las *particiones lógicas* se crean sobre una partición extendida. Este tipo de particiones no son direccionables directamente desde el MBR, por lo que sistemas operativos como Windows impiden su utilización para el sistema.

El tamaño de las particiones nunca debe exceder la capacidad del disco duro.

¿Porqué crear particiones?

Existen diversos motivos por los cuales un administrador de sistemas puede decidir particionar un disco duro. En el caso de UNIX, se deben crear al menos dos particiones:

Root	Partición que debe contener al menos todo lo necesario para que el sistema funcione adecuadamente en modo monousuario.
Swap	Partición que sirve como memoria virtual cuando no hay suficiente memoria física. Cada sistema UNIX debe tener al menos una partición de *swap*.

Además de lo anterior, un buen esquema de particiones tiene las siguientes ventajas:

- Facilita las labores de respaldo.
- Hace más accesible la restauración del sistema.
- Permite manejar políticas diferentes para cada FS.

- Previene problemas graves por el llenado de particiones, ya que no se llenan todas al mismo tiempo.
- Puede mejorar el rendimiento del sistema.
- En el caso de presentarse algún daño en alguna de las particiones, aísla el sistema, previniendo que lo afecte en su totalidad.

Particiones en BSD

El esquema de particiones *BSD Style* implica que se debe asignar un espacio en disco a BSD, el cual es conocido como *slice,* puede verse como una partición extendida, es decir, es una partición primaria especial que se utiliza para crear una o más particiones de BSD.

En BSD las particiones se crean sobre el *slice*.

BSD utiliza letras para nombrar a las particiones:

- En sistemas de ocho particiones utiliza de la letra **a**,a la letra **h** (a-h).
- En sistemas de 16 particiones utiliza de la letra **a**,a la letra **p** (a-p).

La letra **c** está reservada para la partición *overlap*, es decir, se reconoce como todo el disco. Esta partición no puede ser borrada ni modificada.

Por convención, para las particiones *root* y *swap* utiliza las letras **a** y **b** respectivamente.

Particiones en SV

SV utiliza números para nombrar las particiones.

Por convención, para las particiones *root* y *swap* se utilizan los números `0` y `1`, respectivamente. Sin embargo, en Linux las particiones se numeran a partir del`1`.

En algunos sistemas existen particiones reservadas para ciertos usos, por lo que se debe verificar la documentación de estos. Por ejemplo, Solaris utiliza la partición 2 como *overlap*, es decir, es la partición asociada a todo el disco.

Entre las utilerías para el manejo de particiones se encuentran las siguientes:

BSD	fdisk (*slices*) disklabel (*partitions*)
SV	format fdisk prtvtoc (listar el contenido de la tabla de particiones)
Linux	fdisk parted

7.4. Archivos de dispositivo

Aun cuando un disco ha sido formateado y dividido en particiones, no está listo para almacenar archivos. Antes de eso, se debe crear explícitamente la estructura del sistema de archivos.

Como el sistema operativo UNIX trata cada partición como dispositivos lógicos totalmente independientes, se debe tener un archivo especial asociado a cada dispositivo sobre el cual se pueda construir el sistema de archivos.

Dispositivos de E/S

Los dispositivos de E/S se pueden dividir a grandes rasgos en dos categorías: dispositivos por bloques y dispositivos por caracteres. (Tanenbaum, 1998, p.154)

- **Dispositivos de bloque**

En este contexto, un bloque representa la unidad con la que el kernel realiza las operaciones de E/S. (Silberschatz, Galvin y Gagne, 2005, p. 703) El tamaño de este varía dependiendo del sistema operativo, teniendo como mínimo 512 bytes. La información se almacena en bloques y las transferencias son de un bloque cada vez.

- **Dispositivos de carácter**

Transfieren los datos como flujos de bytes, no poseen estructura de bloques. (Stallings, 1997, p. 424) El kernel no realiza casi ningún preprocesamiento de las solicitudes de lectura o escritura de archivo realizadas en un dispositivo de caracteres, simplemente pasa la solicitud al dispositivo en cuestión y deja que el dispositivo la trate. (Silberschatz, Galvin y Gagne, 2005, p. 703)

Los dispositivos son manejados por los controladores, cada uno de estos maneja un tipo de dispositivo o, cuando más, una clase de dispositivos similares.

El nombre interno de un archivo de dispositivo consta de su tipo, carácter (c) o bloques (b), y un par de números, llamados mayor (*major*) y menor (*minor*).

El número mayor codifica el tipo de dispositivo, mientras que el número menor distingue casos diferentes del dispositivo. (Kernighan y Pike, 1984, p. 71)

Los nombres de los archivos especiales asociados a disco, varían dependiendo del UNIX, pero básicamente son:

Sistema operativo	Dispositivo de bloque	Dispositivo de carácter
Solaris	`/dev/dsk/c?t?d?s?`	`/dev/rdsk/c?t?d?s?`
HP-UX	`/dev/disk/disk?` `/dev/disk/c?t?d?`	`/dev/rdisk/disk?` `/dev/c?t?d?`
AIX	`/dev/hdisk?`	`/dev/rhdisk?`
Linux	`/dev/hdl?` `/dev/sdl?`	-

BSD	`/dev/wd?l`
	`/dev/sd?l`
	`/dev/ad?l`
	`/dev/da?s?l`
		`/dev/rwd?l`
		`/dev/rsd?l`

MacOSX	`/dev/disk?s?`	`/dev/rdisk?s?`

En la mayoría de los sistemas, el proceso de *boot* crea automáticamente los archivos especiales apropiados cuando detecta que existe nuevo *hardware*.Sin embargo, si esto no sucediera, se deben crear los archivos.

mknod

Comando que permite crear archivos especiales de dispositivo.

```
# mknod name [c|b] mayor menor
```

Donde:

`Name`	**Nombre del archivo especial**
`[c|b]`	Implica el tipo de archivo: **c** Dispositivo de carácter (dispositivo crudo, *raw*) Las entradas y salidas se manejan uno a uno **b** Dispositivo de bloque (dispositivo cocinado, *cooked*) **Las entradas y salidas se manejan a través de un búfer**
`Mayor`	**Identifica el manejador (*driver*), utilizado por el kernel, para comunicarse con el dispositivo**
`Menor`	**Identifica la ubicación del dispositivo**

En Solaris, los dispositivos se encuentran bajo del directorio `/devices`, para configurarlos debe utilizarse la herramienta `drvconfig`. Sin embargo, el sistema casi siempre crea los dispositivos necesarios simplemente reiniciándose con la opción -r

```
ok> boot -r
```

En algunos sistemas basados en BSD, el archivo script `/dev/MAKEDEV` contiene instrucciones para la creación de archivos especiales a partir de parámetros conocidos.

7.5. Creación de sistemas de archivos

Una vez que se conocen los dispositivos especiales sobre los cuales se construirá el sistema de archivos,se procederá a su creación.

El formateo del sistema de archivos consiste en crear la estructura básica del sistema de archivos de UNIX.

El sistema de archivos de UNIX se compone de cuatro elementos principales:

Bloque de *boot*	Está localizado al principio del archivo, generalmente no es utilizado por el FS. Se deja para el procedimiento de *bootstrapping*. En una partición de arranque, contiene las instrucciones necesarias para inicializar el sistema operativo.
Superbloque	Almacena toda la información de las direcciones del disco. Describe el estado del FS, tales como tamaño, bloques en uso, libres, etc.
Tabla de i-nodos	i-nodo es la estructura que contiene la información de identificación de cada archivo en el sistema, junto con datos esenciales, tales como su longitud, sus marcas de tiempo y las direcciones de localización del archivo en el superbloque. La tabla de i-nodos es la estructura de datos que almacena toda la lista de definiciones de archivo (i-nodos). Cada uno de ellos está numerado, de tal forma que la combinación del nombre del dispositivo y su número en esta lista sirve para identificar en forma única un archivo en particular.
Área de datos	Esta área se usa para almacenar el contenido de los archivos.

Para crear la estructura del sistema de archivos, UNIX proporciona de manera genérica el comando `mkfs` o `newfs`.

mkfs

Comando que se usa para instalar un sistema de archivos dentro de una partición de disco, se debe crear toda su estructura.

```
# mkfs     nombre_del_archivo_especial
```

7.6. Montaje de sistemas de archivos

Para que un sistema de archivos pueda utilizarse en UNIX, debe ser reconocido por el sistema operativo.

La raíz o sistema de archivos de *root*, como se le conoce, es el punto principal de donde se desprende toda la estructura del sistema de archivos de UNIX. Siempre está almacenada en un dispositivo; sin embargo, no es necesario que la totalidad del sistema jerárquico resida en el mismo dispositivo.

Cuando el administrador de sistemas decide crear más particiones o anexar nuevos discos al sistema, generalmente dispondrá de nuevos sistemas de archivos. Para que estos sean asequibles, el administrador deberá anexarlos en un punto de la estructura jerárquica de UNIX, al cual se le conoce como punto de montaje.

El *proceso de montaje* implica crear referencias del punto de montaje al lugar donde se encuentra la raíz del sistema de archivos a montar, de forma que los procesos puedan utilizar el nuevo sistema de archivos.

Para montar un FS se requiere de dos argumentos:

1. El nombre del directorio existente en la estructura jerárquica de UNIX (punto de montaje).
2. El nombre del archivo especial, cuyo volumen de almacenamiento asociado tiene la estructura de un sistema de archivos independiente, conteniendo su propio directorio jerárquico.

mount

Comando que permite montar un FS.

```
# mount   dispositivo punto_de_montaje
```

Ejemplo

```
# mount /dev/dsk/dks0d1s6 /usr
```

umount

Elimina las referencias en el directorio del primer sistema de archivos al sistema de archivos que se va a desmontar. Una vez que el FS ha sido desmontado, no puede ser accedido por los procesos de UNIX.

Tipos de sistemas de archivos

Si bien, el sistema de archivos en UNIX está formado por su estructura básica de bloque de *boot,* superbloque, tabla de i-nodos y área de datos,las implementaciones de dicha estructura son muy variadas. Cada desarrollador maneja su propia implementación, e incluso, dentro de un mismo sistema operativo se tiene la capacidad de manejar diversas implementaciones al sistema de archivos.

Las implementaciones más comunes son:

Sistema operativo	Sistema de archivos
Solaris	UFS
HP-UX	HFS, VxFS
AIX	JFS, JFS2
Linux	ext2, ext3, ext4, ReiserFS, xfs
OpenBSD	Ffs
MacOS	HFS+

Archivos de configuración

Para que un sistema de archivos sea montado durante el proceso de alta del sistema, se debe definir en los archivos de configuración del sistema.

UNIX	Archivo
BSD	**`/etc/fstab`**
SV	**`/etc/vfstab`**

BSD

`/etc/fstab`

Formato del archivo:

```
archivo_especial punto_de_montaje tipoopcionesfrecdump passno
```

Donde:

`archivo_especial`	Archivo de bloques asociado al dispositivo
`punto_de_montaje`	Directorio donde se montará el FS
`Tipo`	Tipo de sistema de archivos en el dispositivo
`Opciones`	Parámetros que determinan características del montaje
`Frecdump`	Niveles de respaldo para el sistema de archivos Determina la política de respaldos
`Passno`	Orden en el que se van a verificar los FS al iniciar el sistema

Ejemplo

```
/dev/root /xfsrw,raw=/dev/rroot 0 0
/dev/dsk/dks0d1s7 /usr/people xfs rw,quota,raw=/dev/rdsk/dks0d1s7 0
0
/dev/dsk/dks0d3s7 /usr/local xfs rw 0 0
```

SV

`/etc/vfstab`

Formato del archivo:

```
archivo_especial archivo_crudo punto_de_montaje tipo
fsck_pass automount opciones
```

Donde:

archivo_especial	Archivo de bloques asociado al dispositivo
archivo_crudo	Archivo de carácter asociado al dispositivo
punto_de_montaje	Directorio donde se montará el FS
Tipo	Tipo de sistema de archivos en el dispositivo
Fsckpass	Orden en el que se van a verificar los FS al iniciar el sistema
automount	Controla si el sistema de archivos será automontado al ser accedido
Opciones	Parámetros que determinan características del montaje

Ejemplo*:*

```
/dev/dsk/c0t0d0s1       -       -       swap    -       no      -
/dev/dsk/c0t0d0s0       /dev/rdsk/c0t0d0s0      /       ufs     1       no      -
/dev/dsk/c0t0d0s5       /dev/rdsk/c0t0d0s5      /usr    ufs     1       no      -
/dev/dsk/c0t0d0s3       /dev/rdsk/c0t0d0s3      /opt    ufs     2       yes     -
/dev/dsk/c0t0d0s6       /dev/rdsk/c0t0d0s6      /usr/local      ufs     2       yes
-
/dev/dsk/c0t0d0s4       /dev/rdsk/c0t0d0s4      /usr/people     ufs     2       yes
-
swap    -       /tmp    tmpfs   -       yes     -
/dev/dsk/c0t2d0s3       /dev/rdsk/c0t2d0s3      /var    ufs     1       yes     -
/dev/dsk/c0t2d0s4       /dev/rdsk/c0t2d0s4      /inet   ufs     1       yes     -
```

7.7. Monitoreo

El administrador del sistema debe realizar actividades de gestión de los recursos del sistema. En el caso de los sistemas de archivos, esto incluye:

- Mantener la integridad del sistema de archivos.
- Verificar el uso y disponibilidad de espacio en los dispositivos de almacenamiento secundario.
- Evitar que los usuarios hagan mal uso del sistema.

Problemas de sincronización memoria-disco

La falta de sincronización memoria-disco, durante el proceso de baja del sistema en los sistemas de archivos montados, puede ocasionar errores de integridad en el sistema de archivos. Los cuales pueden ser:

- Bloques que pertenecen a varios archivos.
- Bloques en uso, marcados como libres.
- Bloques libres, marcados como en uso.
- Conteo incorrecto de ligas.
- Inconsistencia de datos en la tabla de los FS.
- Inconsistencia entre i-nodos y bloques referenciados.
- Archivos perdidos.

UNIX proporciona el comando `fsck`, que permite verificar la integridad de los sistemas de archivos. Se recomienda utilizarlo en modo de mantenimiento o monousuario y que los FS que vayan a ser verificados se encuentren desmontados.

fsck

```
# fsck [opción] dispositivo
```

`-n`	no a las preguntas
`-y`	sí a las preguntas
`-q`	modo silencioso.

Si `fsck` encuentra archivos o cadenas perdidos y no asociados a ningún archivo específico, los ubicará en un directorio llamado `lost+found`, ubicado en el punto más alto del sistema de archivos y el cual únicamente debe tener permisos para *root* (700).

Monitoreo del uso del FS

Existen tres herramientas básicas, proporcionadas por el sistema, que permiten al administrador monitorear el uso del sistema de archivos.

du

Comando que permite saber cuánto espacio ocupa un archivo o directorio. Por omisión, en casi todos los UNIX reporta el resultado en bloques de 512bytes.

`-a`	**Cuánto ocupa en bloques un archivo (*default*)**
`-k`	**Cuánto ocupa en Kbytes un archivo**
`-s`	**Cuánto ocupa en total el directorio**

df

Reporta uso de espacio en sistemas de archivos. Al igual que du, por omisión, en casi todos los UNIX reporta el resultado en bloques de 512bytes.

```
-k   Reporta el resultado en kbytes
```

El formato de salida cambia en algunos sistemas operativos. Pero, en la mayoría los datos son los siguientes:

FileSystem	Nombre del FS
Available	Número de bloques disponibles (tamaño del FS)
Used	Número de bloques utilizados
Free	Número de bloques libres
%	Porcentaje en que el FS está siendo utilizado
Mounted on	Punto de montaje del FS

find

Permite localizar archivos que cumplan con ciertas condiciones y actuar sobre ellos de diversas formas.

Ejemplos de find:

```
find . -perm -002
```

Busca archivos que tengan permiso de escritura para otros a partir del directorio actual.

```
find /tmpu -atime +7
```

Busca archivos cuya fecha de último acceso sea superior a siete días, a partir del directorio /tmpu.

```
find / -nouser
```

Busca desde raíz, archivos cuyo dueño no exista en /etc/passwd.

7.8. Cuotas

Insuficiencia en el espacio de almacenamiento

Cuando un sistema UNIX tiene problemas de espacio en disco, el administrador debe optar por alguna de las siguientes medidas:

- Adquirir un nuevo disco.
- Revisar bitácoras.
- Borrar archivos viejos.
- Ejercer presión a los usuarios para que depuren.
- Implantar cuotas.
- Importar disco remoto.
- Implementar o instalar algún mecanismo de migración de archivos.

Cuotas de disco

Mecanismo que permite limitar:

- Número de bloques que un usuario puede utilizar en un FS.
- Número de archivos que puede crear.

Existen dos clases de límites:

Límite suave (*soft*)	Límite duro (*hard)*
Menor que el duro, se puede rebasar, el sistema avisará al usuario que ha desbordado su límite, pero le permitirá seguir trabajando por un tiempo.	Límite que el usuario NO puede rebasar, una vez que se alcanza el sistema, ya no permite hacer nada al usuario.

Las cuotas se definen y configuran por sistemas de archivos, es decir, cada FS puede manejarse con políticas diferentes de cuotas.

Creación de cuotas

1. Modificar los archivos de configuración (`fstabo vfstab`), indicando la palabra *quota* en las opciones y en su caso el dispositivo tipo crudo del FS.

2. Crear un archivo llamado *quotas* en el directorio raíz del FS. Este archivo debe pertenecer a `root` y tener permisos 600.

3. Establecer las cuotas que tendrán los usuarios mediante la utilización del comando `edquota`.

`# edquota usuario`- Permite establecer cuotas para un usuario en particular

`# edquota –p usuario lista_de_usuarios` -Permite copiar las cuotas de un usuario prototipo a los usuarios especificados.

4. Activar el sistema de cuotas:

`quotaon`	activa el sistema de cuotas.
`quotaoff`	desactiva el sistema de cuotas.

5. Verificar las cuotas de todos los usuarios:

```
quotacheck –v FS
```

6. Generar un reporte de cómo están configuradas las cuotas.

```
repquota –V FS
```

Monitoreo de cuotas

Un usuario puede conocer su cuota de disco utilizando el comando:`$ quota –v`

El superusuario puede conocer el uso de espacio que un usuario está haciendo de un sistema de archivos con cuotas, añadiendo el *login* del usuario.`# quota –v usuario`

7.9. NFS (Network File System)

Servicio creado por Sun Microsystems:

- Funciona por medio de RPC (Remote Procedure Call).
- Permite que un disco conectado físicamente a una máquina pueda usarse por otra como si fuera o estuviera conectado físicamente a ella.
  - -El tiempo de respuesta es más grande.
  - -El manejo de los permisos es diferente.
  - -Funciona con base en el modelo cliente-servidor.4.
  - -El kernel debe tener activada la opción NFS.

NFS utiliza tres demonios:

`Nfsd`	**Atiende las peticiones de archivos**
`Mountd`	**Atiende las peticiones de montaje**
`Portmap`	Realiza un mapeo de puertos Ayuda a establecer la comunicación entre el cliente y el servidor

Los demonios se arrancan en:

`/etc/rc.local` BSD

`/etc/init.d/nfs` SV

Montar un FS a través de NFS

Servidor	Cliente
Define qué directorios va a exportar y qué máquinas van a poder montar	**Define los FS que va a montar vía NFS**

Configuración del servidor

→BSD

Dependiendo de la versión de NFS que se utilice, el archivo `/etc/exports` utilizará la siguiente sintaxis:

```
/etc/exports
directorio     opciones
```

Ejemplo:

```
/appl          -access=host1:host2,ro
/usr/people    -root=host1,rw
```

En este caso el sistema estará exportando el sistema de archivos `/appl`, con opciones de solo lectura a las máquinas `host1` y `host2`; mientras que el sistema de archivos `/usr/people` estará siendo exportado con privilegios de `root`, en modo lectura y escritura a la máquina `host1`.

El formato del `/etc/exports` puede variar, quedando de la siguiente forma:

```
/etc/exports
FS   cliente(opciones)
```

Ejemplo:

```
/admsuper  172.20.1.95(rw)
```

Donde el servidor está exportando el sistema de archivos `/admsuper` a la máquina con dirección IP `172.20.1.95,` en modo lectura/escritura.

Nota: es recomendable exportar con los mínimos privilegios necesarios.

→SV

```
/etc/dfs/dfstab
```

Este archivo contiene una serie de comandos `share` (se pueden ejecutar desde la línea de comandos). Se utiliza para exportar algún dispositivo al momento del inicio del sistema, ya que en él se especifican los sistemas de archivos a exportar.

La sintaxis básica es:`share –F FSType –o opciones punto_de_montaje`

Ejemplo:

```
share -F nfs -o ro:saka:batari:chokol /opt
```

El servidor estaría exportando el sistema de archivos /opt en modo solo lectura a las máquinas saka, batari y chokol.

Montar FS a través de NFS. Configuración del cliente

El cliente debe montar el FS, indicando a través del comando `mount` que el tipo de sistema de archivos es NFS.

```
# mount  -t nfs servidor:FS  punto_de_montaje
```

Opciones más comunes:

`Bg`	**(Para que mande en segundo plano y el sistema no se congele)**
`Retry`	**(Cuántas veces va a reintentar, *retry*=número)**
`Timeo`	(*Timeout timeo*=segundos)
`Retrans`	(Núm. de retransmisiones que va a pedir)

También se puede montar editando los archivos `/etc/fstab` o `/etc/vfstab`.

Bibliografía básica del tema 7

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Nemeth, E., Hein, Trent R., Snyder, G. & Whaley, B. (2010). *UNIX and Linux system administration handbook*. (4th ed.) Boston, MA: Prentice Hall. [Vista previa]

Bibliografía complementaria

Alcalde, Eduardo; García, Miguel y Peñuelas,Salvador. (1988). *Informática básica*. México: McGraw-Hill.

Kernighan, B. y Pike, R. (1987). *El entorno de programación UNIX*. México: Prentice Hall.

Sánchez, S. (1999).*UNIX y Linux guía práctica*. México: Alfaomega/Ra-Ma.

Silberschatz, A., Galvin, P. y Gagne, G. (2006). *Fundamentos de sistemas operativos*. (7ª ed.) Madrid: McGraw-Hill.

Stallings, W. (1997). *Sistemas operativos*. (2ª ed.) Madrid: Prentice Hall.

Tanenbaum, A. (1998).*Sistemas Operativos. Diseño e implementación*. (2ª ed.), México: Prentice Hall.

Sitios de Internet

Suppi Boldrito, R. y Jorba Esteve, J.,*Administración Avanzada de GNU/Linux.*Universidad Abierta de Cataluña, disponible en línea: http://materials.cv.uoc.edu/cdocent/7FWNKASR7N3XHVRF138B.pdf, Consultado: 01/07/2011

Actividades de aprendizaje

A.7.1. En un equipo con sistema operativo Linux instalado, realiza el procedimiento necesario para:

- Conectar un nuevo disco duro, crear dos particiones y ponerlas a disposición de los usuarios en los directorios /nuevo1 y /nuevo2.
- Montar el sistema de archivos /appl del servidor 192.168.0.100 en la máquina local.

A.7.2. Realiza un cuadro comparativo del manejo de dispositivos en Linux, Solaris y OpenBSD, incluyendo:

- Tabla de particiones.
- Nombres de dispositivos.
- Comandos importantes relacionados al manejo de sistemas de archivos.
- Comandos de monitoreo de sistemas de archivos.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿Qué es un sistema de archivos?
2. ¿Cómo se llama a la división lógica de un disco duro?
3. ¿Para qué sirve el comando `mount`?
4. ¿Cuál es la diferencia entre los comandos `du` y `df`?
5. ¿Quién es el encargado de determinar y establecer las cuotas?
6. ¿Qué aspectos limitan las cuotas?
7. ¿Cuántas particiones primarias se pueden crear comúnmente en un disco duro?
8. ¿Cuál es la diferencia entre un dispositivo de bloque y uno de carácter?
9. ¿Cuál es la función del número mayor (major) en archivos asociados a dispositivos?
10. ¿Para qué sirve el comando `mknod`?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. ¿Qué es un archivo?
   a) Es un espacio de memoria que reserva el usuario.
   b) Es la información del usuario en la memoria del sistema.
   c) Es información referenciada por un nombre.
   d) Es un espacio en los discos de almacenamiento.

2. El comando dusirve para:
   a) Saber cuánto espacio ocupa un archivo o directorio.
   b) Saber cuánto espacio tengo disponible en mi sistema de archivos.
   c) Saber cuántos discos tengo disponibles.
   d) Ver cuántas particiones tienen mis discos duros.

3. El comando `find` sirve para localizar:
   a) Los archivos que cumplan ciertas condiciones.
   b) El espacio que ocupa un archivo o directorio en el sistema.
   c) Los últimos archivos a los que accedió el sistema.
   d) Patrones en archivos.

4. ¿Con qué comando puedo encontrar en el sistema todos los archivos sin dueño?
   a) find / -nouser
   b) find * -nouser
   c) find /sys -userno
   d) find / -owner no

5. Las cuotas en el sistema de archivos permiten limitar el:
   a) Número de bloques y archivos.
   b) Uso de la memoria virtual.
   c) Uso del procesador.
   d) Acceso a archivos y directorios.

6. ¿Qué es NFS?
   a) Un sistema de archivos.
   b) Un protocolo para compartir datos en la red.
   c) Un sistema de archivos conectado a Internet.
   d) Un servicio para compartir sistemas de archivos en red.

7. ¿Qué archivo utiliza NFS para configurar los sistemas de archivos a compartir?
   a) /etc/nfs
   b) /etc/exports
   c) /etc/exportfs
   d) /etc/sysconfig/nfs

8. Comando que permite revisar las cuotas (como usuario):
   a) quota -v
   b) quota -l
   c) quota -r
   d) quota -u usuario

9. Comando que permite ver la tabla de particiones de un disco duro en BSD:
   a) prtvtoc
   b) parted
   c) disklabel
   d) partitionlist

10. Comando que permite crear un archivo especial de bloque en UNIX:
    a) mkfile
    b) dd if=/dev/MAKEFILE of=/dev/blk
    c) mknod
    d) mkfs

TEMA 8. RESPALDOS

Objetivo particular

Al término del tema, el alumno podrá:
Identificar la importancia de los respaldos yla forma en que estos se realizan.

Temario detallado (6 horas)

8.1. Unidades de cinta
8.2. Políticas de respaldo
8.3. Herramientas de respaldo

Introducción

El elemento principal para el desarrollo de las organizaciones es la información, a través de ella se toman las decisiones que posibilitan éxitos en las organizaciones. La información debe ser rápida, veraz, oportuna y suficiente. Por ello es importante protegerla, lo cual se logra realizando una copia de la información, denominada *respaldo*, este proceso es de suma importancia, ya que permite recuperar información que haya sido dañada por errores de programación, fallas del *hardware*, intrusos, robo o destrucción de los equipos, desastres naturales, etc.

Un principio crucial, al decidir realizar copias de seguridad, es la identificación de las aplicaciones e informaciones críticas, ya que con base en esto se decide la frecuencia con la que se realizarán las copias de seguridad, los archivos que se copiarán, así como los medios donde se almacenarán las copias.

Un *respaldo* es la copia de los archivos, directorios y programas almacenados en el sistema, con la finalidad de proteger la información, la cual se realiza en un lugar diferente en donde los datos se encuentran almacenados. En Unix se utilizan cintas.

Tipos de respaldos

- **Respaldo de día cero**. Es el tipo de respaldo que se hace al sistema con su configuración original, tal como se encuentra inmediatamente después de una instalación de sistema y antes de realizar cualquier cambio en él.

- **Respaldos completos**. Consiste en hacer un respaldo completo del sistema, es decir, de todos los archivos que hay en él, incluyendo los ocultos, los del sistema, áreas del sistema o temporales, archivos con líneas cruzadas o ligas, etc.

- **Respaldos parciales**. Como alternativa a una copia de seguridad completa, el usuario puede seleccionar los archivos y directorios que desea copiar, el *software* los leerá y escribirá de uno en uno. Esto permite la restauración rápida de un archivo o grupo de archivos.

▪**Respaldos incrementales**. Otro de los tipos de copias de seguridad a considerar es la incremental. Consiste en realizar exclusivamente copia de seguridad de los archivos que han sido modificados desde la última copia de seguridad. La idea consiste en que las copias sucesivas de todos los archivos de datos contendrán probablemente archivos de los que ya se ha hecho respaldo, lo cual hace más lento el proceso de copia. Se pueden realizar respaldos incrementales que se apliquen exclusivamente a los archivos modificados o incluidos desde la última copia de seguridad.

8.1. Unidades de cinta

Las cintas son accedidas por dispositivos tipo crudo (*raw*). Para el manejo de cintas UNIX utiliza básicamente dos dispositivos:

- Dispositivo *rewind* (rebobina y retensiona las cintas)
- Dispositivo *norewind* (no rebobina las cintas)

Los nombres de los archivos especiales asociados a estos dispositivos varían dependiendo del UNIX, pero básicamente son:

Unix	Rewind	No rewind
SunOS	/dev/rst?	/dev/nrst?
Solaris	/dev/rmt/?	/dev/rmt/?n
Digital UNIX	/dev/rmt/?	/dev/nrmt?
AIX	/dev/rmt?	/dev/rmt?.1
IRIX	/dev/rmt/tps?d? /dev/tape	/dev/rmt/tps?d?nr /dev/nrtape
Linux	/dev/st?	/dev/nst?

Nota:cuando se manejen múltiples respaldos en una sola cinta, es muy importante estar seguro del funcionamiento de los dispositivos *rewind* y no *rewind*, ya que varía en cada UNIX.

mt

Permite controlar los dispositivos magnéticos, enviando diferentes comandos a los dispositivos. Si no especifica el dispositivo, utilizará el que haya definido en la variable de entorno TAPE. La sintaxis básica del comando es:

```
mt [-f dispositivo_de_cinta] comando [cuenta]
```

`mt` se utiliza para dar comandos a las unidades de cintas magnéticas. Por omisión utiliza el dispositivo *norewind*.

Los comandos más utilizados son:

Comando	Acción
status	Da el estado de la cinta
Fsf	Recorre un archivo de cinta hacia delante (*forward*)
Bsf	Recorre un archivo de cinta hacia atrás (*backward*)
Rewind	Rebobina la cinta

El funcionamiento de `mt` varía para cada sistema operativo, ya que en algunos sistemas comienza la cuenta desde el archivo actual y en otros no, por tanto el administrador deberá verificar primero este funcionamiento antes de realizar cualquier respaldo.

Ejemplos

```
$ mt status
        Controller: SCSI
```

```
            Device: ARCHIVE: Python 01931-XXX5.63
            Status: 0x20262
            Drive type: DAT
            Media: READY, writable, at BOT
    $ mt fsf 1
```

Algunos códigos importantes que reporta `mt` son:

`BOT`	Begin of Tape
`EOT`	End of Tape
`EOD`	End of Data
`Block`	Si es diferente de 0, la cinta está recorrida
`File number=`	Número de archivo de la cinta

8.2. Políticas de respaldo

La formulación de políticas se establece con una estrategia basada en el tiempo y esfuerzo expresado por las modificaciones a los archivos, el tiempo y esfuerzo, lo cual esrepresentado por la copia de seguridad de los archivos y el valor del contenido de los archivos. Cada organización formulará sus políticas de acuerdo con su operación, por ello no se pueden homogenizar. Por ejemplo, un banco tiene la política de respaldar todas las noches todas las operaciones realizadas y durante el día realizar respaldos periódicos. En cambio, un centro de cómputo respaldará la información de sus usuarios cada lunes o cada mes, el tiempo será definido por la operación del sistema y los medios para realizarlos, también es posible que exista una política que omita esta acción.

8.3. Herramientas de respaldo

Dump

Utilería de respaldo propia de BSD.

Permite realizar respaldos de sistemas de archivos, ya sea completo o incrementables. Maneja nueve de respaldo incremental, lleva un registro de cuándo se hizo el último respaldo de cada sistema de archivos y en qué nivel. Si se solicita un respaldo incremental, se respaldan todos los archivos que hayan sido modificados desde la última fecha en que se realizó un respaldo de un nivel menor.

Maneja el archivo de configuración `/etc/dumpdates`

```
<archivo especial><nivel><fecha><hora>
   /dev/rz0c     2       sun feb 26   12:09:45  1995
```

Si es la primera vez que se va a respaldar, debe crearse el archivo de configuración.

```
# touch    /etc/dumpdates
```

La sintaxis básica de `dump` es:

```
dump      [opciones]          FS
```

Opciones:

Niveles	0 al 9 (el 0 es un respaldo completo)
u	actualiza el archivo /etc/dumpdates
s	tamaño cinta en pies 9 tracks 2300
d	densidad (BPI) 9 tracks 6250 BPI
c	capacidad en megabytes 2m = 2gigabytes 200 = 200 megabytes
w	wait (únicamente imprime, no hace nada)
f	dispositivo destino

Ejemplo:

Respaldo completo nivel 0, actualizará el archivo dumpdates, el tamaño de la cinta es de 2300 pies, la cinta se encuentra ubicada en `/dev/rmt1`, su densidad es de 6250 bpi y el FS a respaldar es `/dev/dsk/rz0c`.

```
# dump 0usfd 2300 /dev/rmt1 6250 /dev/desk/rz0c
     dump (/inv to tape): Dates of this level 0 dump:Wed
Feb 22 07:08:29 1995
     dump (/inv to tape): Dumping /inv
                  "           : to tape
                  "          : mapping
                  "          : estimated 249248 sectors en
408 volume (s)
```

En los respaldos realizados con dump, para recuperar la información, se utilizarestore

Sintaxis

```
# restore opciones  argumentos     [archivos y directorios]
```

Opciones:

r	lee y restaura la cinta completa
R	selecciona una cinta de un respaldo multivolumen
x	Extrae
f	dispositivo de entrada
h	no recursivo
v	*Verbose*
i	interactivo

EJEMPLO

```
# cd /usr/users
# restore xf /dev/rmt1   yoli otro/tmp/dos
```

Backup

Utilería de respaldo propia de SV.

Sintaxis
 # backup [opciones] archivo(s)

Opciones**:**

`-c`	Respaldo completo
`-p`	Respaldo incremental
`-f`	Lista archivos
`-u user`	Respalda archivos bajo $HOME de user
`-d`	Dispositivo
`-t`	Indica que el dispositivo es cinta (*default floppy*)

EJEMPLO
#backup -c -t -d /dev/rmt/0 / **respaldo completo en cinta de root**

`/etc/bkup/bkreg.tab` Registro de respaldos

Comandos de administración de *backup*

`bkstatus`	Ver el estado de un respaldo (*active, suspend, pending, failed, waiting, completed*)
`bkhistory`	Ver el estado de un respaldo ejecutado con anterioridad

dd

Comando que permite copiar de dispositivo a dispositivo

Sintaxis
 # dd [opcion=valor]

Opciones:

if	*Input file*
of	*Output file*
ibs	Tamaño de bloque de entrada
obs	Tamaño de bloque de salida
cbs	Tamaño de bloque de conversión
fskip	Salta archivos
count	Número de bloques a transferir
conv	Tipo de conversión ascii EBCDIC ® ASCII swab invierte cada par de bytes

EJEMPLO
 # dd if=/dev/rmt0 of=/tmp/root/respaldo ibs=20 obs=20 conv=swab **$ dd if=archivo1 of=archivo2**
Copiar una cinta completa **$ if=/dev/tape of=cinta cbs=20b** **$ if=cinta of=/dev/tape cbs=20b**

Cpio

Comando que permite empaquetar archivos en un contenedor cpio, es decir, almacena otros archivos en uno solo. El contenedor puede ser un disco, otro archivo, una cinta o un entubamiento. Los archivos pueden ser ordinarios, dispositivos o FS completos.

Sintaxis

```
# cpio -o [llaves]
# cpio -i [llaves] [patrones]
# cpio -p [llaves] [directorios]
```

Opciones:

o	Genera un contenedor en la salida estándar
i	Toma archivos en un contenedor de la entrada estándar
P	Lee de la entrada estándar los nombres de los archivos que hay que copiar y los copia en el directorio especificado.
m	Conserva los atributos de los archivos
t	Crea una tabla de contenidos
v	Modo detallado, en la salida estándar aparecerá cada uno de los pasos que la instrucción ejecuta, conocido como *verbose*
d	Especificar un directorio

EJEMPLO:

```
# find /var | cpio –o > /dev/tape
# find . –name “*.txt” | cpio –o /tmp/textos.cpio
# cpio –i < /dev/st0
# find . | cpio –pd /tmp/nuevo
```

Tar

Comando que permite empaquetar archivos en un contenedor `tar`, es decir, almacena otros archivos en uno solo. El contenedor puede ser otro archivo, una cinta o un entubamiento [*pipe*]. Los archivos deben ser ordinarios, FS completos.

Su nombre significa *Tape Archiver*, pero uno de sus usos más comunes es el manejo de archivos de disco.

Normalmente se combina con algún programa de compresión/descompresión de datos (compress, gzip, etc.) para disminuir el espacio ocupado por los archivos.

Se utiliza para distribuir programas de dominio público, pues permite incluir todos los archivos necesarios en un solo "archivo de distribución".

SINTAXIS
`tar llave [ argumentos ] archivo | directorios […]`

Llave. Debe contener alguna de las siguientes funciones:

`c`	Crear un respaldo
`x`	Extraer archivos de un respaldo
`t`	Listar los archivos de un respaldo

Opciones más utilizadas:

`v`	Proporcionar información de lo que se está haciendo (*verbose*)
`f file`	Utilizar el archivo específico en vez de la unidad de cinta de default para hacer el respaldo. Se usa para hacer respaldos en un archivo en disco o en dispositivos alternos. Se puede utilizar para especificar la entrada o salida estándar

EJEMPLOS

```
$ tar c .
Crear un respaldo en la cinta de default del directorio actual
y todos sus subdirectorios.

$ tar cvf backup.tar ./usr ./etc ./bin
Crear el archivo backup.tar que contenga los directorios usr,
etc y bin del directorio actual.

$ tar cvf  - . | gzip > ../respaldo.tar.gz
Crea un respaldo del directorio actual en la salida estándar,
que es pasado al gzip para que lo comprima, y el resultado es
puesto en el archivo reslapaldo.tar.gz

$ tar cvf - . | ( cd /tmp/dup; tar xvf -)
$ tar cvf - . | tar xvCf /tmp/dup -
Son formas equivalentes para crear un duplicado perfecto del
directorio actual en el directorio /tmp/dup
```

Nota: se recomienda no usar rutas absolutas.

La instrucción`tar` no puede cambiar la ruta de un archivo al momento de recuperarlo. Por ejemplo, si un respaldo se realizó con el siguiente comando:

```
$ tar cv /home/pepe
```

Al momento de recuperarlo, `tar` va recuperar los archivos con exactamente la misma ruta, de manera que si `/home/pepe`ya existía, sus archivos serán reemplazados por los del respaldo.

La solución es utilizar rutas relativas al momento de crear un respaldo. Por ejemplo, en vez del comando anterior, se puede ejecutar:

```
$ cd  /home/pepe
$ tar cv ./pepe
```

Con lo cual el respaldo se podrá recuperar debajo del directorio actual, sin importar cuál sea.

Respaldos remotos

Para realizar respaldos en máquinas que carecen de unidad de cinta, debe combinarse `tar` con algún comando de red como `rsh` o `ssh`, así como el comando `dd`, el cual permite especificar el dispositivo al que se escribirá en la máquina remota.

```
# tar cvf - * | ssh –l usuario maquina dd of=/dev/tape
                                obs=20b
# ssh –l usuario maquina dd if=/dev/tape ibs=20b | tar xvf -
```

Bibliografía básica del tema 8

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Garfinkel, Simson; Spafford, Gene & Shwartz, Alan. (2003). *Practical Unix and Internet Security*. (3rd ed.) O'Reilly Media: Sebastopol, CA. [Vista previa]

Nemeth, Evi; Snyder, Garth; Hein, Trent R. (2007). *Linux administration handbook.* (2nd ed.)Stoughton, Massachusetts: Pearson Education. [Vista previa]

Sitios de Internet

Yolinux.com (s.f.)Linux Information Portal, *Linux System Administration and Configuration*: http://www.yolinux.com/TUTORIALS/LinuxTutorialSysAdmin.html

Actividades de aprendizaje

A.8.1. Elabora un esquema de respaldo que almacene únicamente los archivos que se hayan modificado o creado en las últimas 3 horas. Utilizando dos herramientas de respaldo.

A.8.2. Elabora un mapa mental donde describas el procedimiento para respaldar los archivos de configuración, aplicaciones y la información de usuarios en un servidor con sistema operativo Irix.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. Explica la importancia de realizar respaldos.
2. ¿Que utilerías permiten crear respaldos incrementales?
3. ¿Cuándo es conveniente usar el comando `dd`?
4. ¿Cuáles son los criterios para formular políticas de respaldo?
5. ¿Cuál es el uso de la utilería `tar`?
6. ¿Cuáles son las ventajas del comando `cpio`?
7. ¿Cuál es la opción que se utiliza para especificar el tamaño del bloque de entrada, con el comando `dd`?
8. ¿Qué comando de respaldo es propio del SV?
9. ¿Cuáles son las razones que motivan la realización de respaldos remotos?
10. ¿Para qué sirve el comando `mt`?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. Es el respaldo que se realiza con la configuración inicial del sistema.
   a) Respaldo parcial.
   b) Respaldo incremental.
   c) Respaldo completo.
   d) Respaldo de día cero.

2. La sintaxis del comando `tar` es:
   a) tar llave archivo directorio
   b) tar archivo directorio
   c) tar llave [argumentos] archivo [directorio]
   d) tar archivo [argumentos] llave [directorio]

3. El comando `cpio` es un:
   a) Empaquetador de archivos
   b) Empaquetador y compresor de archivos
   c) Compresor de archivos
   d) Ninguna de las anteriores

4. Comando que se emplea para recuperar la información respaldada con `dump`:
   a) recover
   b) restore
   c) dd
   d) undump

5. Es el respaldo que se realiza en todo el sistema:
   a) Respaldo parcial
   b) Respaldo incremental
   c) Respaldo completo
   d) Respaldo de día cero

6. Son los tipos de respaldo que permite el comando `dump`:
   a) Día cero y parciales
   b) Completos e incrementales
   c) Incrementales y parciales
   d) Parciales y completos

7. Son códigos de la utilería `mt`:
   a) BOT, EOT, EOF
   b) BOF, EOT, EOD
   c) BOT, EOF, EOD
   d) BOT, EOT, EOD

8. Es el respaldo que se realiza solo a archivos/directorios seleccionados:
   a) Respaldo parcial
   b) Respaldo incremental
   c) Respaldo completo
   d) Respaldo de día cero

9. Es un tipo de dispositivo en la unidad de cinta del sistema operativo Unix:
   a) fsf
   b) rewrite
   c) rewind
   d) status

10. En este tipo de respaldo solo se copian los archivos nuevos o que hayan sido modificados desde el último respaldo:
    a) Respaldo parcial
    b) Respaldo incremental
    c) Respaldo completo
    d) Respaldo de día cero

TEMA 9. CONFIGURACIÓN DE LA RED

Objetivo particular

Al término del tema, el alumno podrá:
Configurar los servicios de red de un equipo con sistema operativo UNIX.

Temario detallado (6 horas)

9.1. Archivos básicos de configuración de la red
9.2. Instrucciones básicas de configuración de la red

Introducción

Actualmente el término *Internet* está incorporado en la sociedad y se ha convertido en una herramienta que permite comunicarnos con equipos ubicados en cualquier parte del mundo. Muchos de estos se encuentran bajo el ambiente operativo Unix, acondicionados adecuadamente para que se pueda acceder a ellos desde diferentes lugares, por lo que es importante identificar los elementos que logran una configuración adecuada.

9.1. Archivos básicos de configuración de la red

Los equipos conectados en red poseen un nombre que los identifica en Internet, esto se denomina `hostname`, para no perder el nombre del equipo se almacena en el siguiente archivo de configuración:

Archivo	Descripción
/etc/hostname	**Permite especificar el nombre de la máquina en la mayoría de los sistemas operativos "UNIX Like". Su nombre puede variar entre hostname y HOSTNAME.**

Algunos sistemas utilizan nombres o archivos diferentes para esta funcionalidad. Por ejemplo:

Archivo	VERSIÓN
/etc/hostname.interface	**Solaris**
/etc/rc.config.d/netconf	**HP-UX**
/etc/sysconfig/network **/etc/sysconfig/network-scripts/ifcfg-ethX**	**Linux**

En UNIX, el archivo `/etc/hosts` describe la relación del nombre del equipo y la dirección numérica. Se utiliza durante el arranque o cuando no haya servidores de nombres habilitados.

/etc/hosts	- Principal archivo de configuración de red. - Contiene las direcciones de todos los hosts conocidos. - Incluye a localhost y la dirección IP del host local
EJEMPLO # Dirección IP Nombre DNS Alias # la línea de localhost es obligatoria. 127.0.0.1 localhost 132.248.168.85 coronel.dgsca.unam.mx coronel	
En Linux, la IP se especifica en: /etc/sysconfig/network-scripts/ifcfg-eth0 IPADDR=132.248.168.85	

El archivo/etc/defaultdomain, encontrado en la mayoría de los UNIX,contiene una línea con el nombre de dominio completo al que pertenece la red del equipo que se está configurando.

/etc/defaultdomain	- Permite especificar el dominio del sistema.
EJEMPLO fciencias.unam.mx	
Nota: en sistemas que no manejan este archivo, el dominio se obtiene del archivo /etc/hosts.	

El *ruteador*o *gateway* (en su caso), conecta en los diferentes segmentos de red o redes enteras,casi siempre ocupa la dirección 254 del segmento de red.Si el sistema no cuenta con un archivo específico para asignar el ruteador, se puede levantar en los archivos de inicio rc's.

/etc/defaultrouter -Contiene la dirección del ruteador.
En Linux: /etc/sysconfig/network o /etc/sysconfig/network-scripts/ifcfg-eth0 GATEWAY=ruteador

Los servidores de nombres le indican a su dominio dónde buscar los registros, contiene una lista de dominios y las direcciones de los servidores.

/etc/resolv.conf - Especifica por dónde comenzará el sistema a buscar un nombre canónico.Este archivo existe en todos los UNIX.
FORMATO DEL ARCHIVO: domain dominio search rutas de búsqueda ; comentario nameserver servidor_de_nombre_DNS Nota: los parámetros domain y search son excluyentes.
EJEMPLO nameserver 10.3.1.211 nameserver 10.3.1.221

La máscara de red indica al dispositivo de red por dónde enviará los paquetes, ya sea por la red local o una red externa.

/etc/netmask - Especifica la máscara de red.
En Linux: /etc/sysconfig/network-scripts/ifcfg-eth0 NETMASK=máscara

El archivo /etc/nsswitch.conf permite especificar el orden que el sistema seguirá cuando existan peticiones para ingresar a los diferentes servidores.

/etc/nsswitch.conf - Proporciona el orden de búsqueda
EJEMPLO hosts: files, dns Con este orden, primero buscará en el archivo de configuración. /etc/hosts, y después consultará los DNS habilitados en elarchivo resolv.conf

Un servidor es un equipo preparado para proporcionar un servicio, como lo son servidores de correo electrónico. Los sistemas que se encuentran en este ambiente de trabajo poseen un archivo en donde se habilitan los servicios que el sistema proveerá.

Es importante configurarlo adecuadamente, ya que si esto no ocurre, se ocasionarán problemas de seguridad. Actualmente los ataques a los equipos son constantes, por ello se deben cerrar puertos que no se utilicen.

/etc/services - Define los puertos habilitados
FORMATO <servicios> <#puerto/<protocolos> <parámetros> ftp.data 20/tcp ftp 21/tcp ssh 22/udp (user datagram protocol) ssh 22/tcp (transfer control protocol) telnet 23/tcp httpd 80/tcp
Puertos que generalmente no se utilizan y se consideran inseguros: courier ingreslock uucp
Si no se utiliza un servicio de red como NFS o NIS se deben cerrar los servicios rpc.* nis nfs Nota: cuando se deshabilite un servicio hay que eliminar al demonio y cerrar el puerto asociado a ese servicio.

El programa `inetd` tiene el objetivo de ejecutar los programas (demonios) que proporcionan un servicio en el sistema.

/etc/inetd.conf - Contiene los demonios de red que son manejados a través de inetd.
FORMATO <servicio> <tipo socket> <protocolo> <delay> <programa> <argumentos> telnet steam tcp wail /usr/etc/telnetd telnetd <socket> Tipo de archivo que permite hacer la conexión. <delay> que mientras esté atendiendo un servicio no puede correr otro o viceversa. Wait Nowait
EJEMPLO talkd telnetd ftpd fingerd Nota: en la mayoría de los sistemas operativos UNIX se utiliza la convención de que los nombres de los demonios terminen con d.
Linux (algunas versiones) xinetd.conf Archivo /etc/xinetd.d/ Directorio

9.2. Instrucciones básicas de configuración de la red

Es posible modificar los parámetros de red sin editar los archivos. Esta acción se recomienda cuando se trabajará con ese segmento de red por corto tiempo. Si el equipo se apaga o reinicia, se deberá volver a configurar la red.

Lo primero que se debe hacer es identificar la interfaz de red, los nombres más comunes se presentan a continuación:

Unix	Interfaz
IRIX	eco, et0
Linux	eth0
SunOS	le0, ie0, ec0
Solaris	le0
Aix	en0, et0

Para conocer información sobre las interfaces de red se puede utilizar la instrucción `ifconfig –a.`

```
% ifconfig -a
ec0:
flags=c43<UP,BROADCAST,RUNNING,FILTMULTI,MULTICAST>
        inet    132.248.168.94    netmask    0xffffff00
broadcast 132.248.168.255
lo0: flags=1849<UP,LOOPBACK,RUNNING,MULTICAST,CKSUM>
inet 127.0.0.1 netmask 0xff000000
```

Nota: en sistemas donde ifconfig no soporta la opción –a, puede utilizarse el comando netstat -i.

ifconfig Comando que permite configurar las interfaces de red.
FORMATO # ifconfig interfaz [dirección [parámetros]] up Habilita la interfaz de red. down Deshabilita la interfaz de red. inet IP Permite especificar la dirección IP. En algunos sistemas inetaddr. netmask mask Máscara de la red (tipo de red. broadcast Mensajes públicos de reconocimiento.
EJEMPLO # ifconfig ec0 down # ifconfig ec0 inet 132.248.168.94 netmask 0xffffff00 broadcast 132.248.168.255 # ifconfig ec0 up

Para agregar el ruteador desde la línea de comandos:

route Comando que permite configurar y mostrar las tablas de ruteo.
FORMATO route add default ruteador 1 route add default gw ruteador
EJEMPLO route add default gw 192.168.61.254

Bibliografía básica del tema 9

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Nemeth, E., Hein, Trent R., Snyder, G. & Whaley, B. (2010). *UNIX and Linux system administration handbook*. (4th ed.) Boston, MA: Prentice Hall. [Vista previa]

Bibliografía complementaria

Nemeth, Evi; Snyder, Garth; Hein, Trent R. (2007). *Linux administration handbook.* (2nd ed.)Stoughton, Massachusetts: Pearson Education. [Vista previa]

Tanenbaum, Andrew S. (2003). *Redes de computadoras*. (4a ed.) México: Pearson Educación. [Vista previa]

Sitios de Internet

Jorba Esteve, J. (2010). "Administración Local. Administración de sistemas GNU/Linux". *OpenCourseWare.* Universidad Abierta de Cataluña. Disponible en línea: http://materials.cv.uoc.edu/continguts/PID_00157329/web/main/materias/PID_00157328-5.pdf

Actividades de aprendizaje

A.9.1. Elabora un esquema de los principales elementos que componen el sistema de red del sistema operativo Unix.

A.9.2. Elabora un mapa mental donde describas el procedimiento para configurar la red en un sistema operativo Irix, con los siguientes datos:

```
IP: 132.248.2.89
DNS: 132.248.202.4
Ruteador: 132.248.2.254
```

Cuestionario de autoevaluación

Responde el siguiente cuestionario.

1. ¿Cuál es la instrucción que me permite conocer mi dirección IP?
2. ¿Cuáles son los servicios que se deben deshabilitar si el servidor no los va utilizar?
3. ¿En qué archivo de configuración se indican las rutas de búsqueda de los servidores de nombres?
4. ¿Cuál es archivo de configuración en donde se indican los nombres cortos para las máquinas de uso frecuente?
5. ¿Cuál es el comando que inicia el servicio de red del sistema?
6. ¿Cuál es la instrucción que permite revisar mi tabla de ruteo?
7. ¿Cuál es la función de la instrucción *ifconfig*?
8. ¿Cuáles son los nombres que identifican la interfaz de red en las diferentes versiones de sistema operativo Unix?
9. ¿Qué función tiene del ruteador?
10. ¿Cuál es el formato que utiliza el archivo /etc/inetd.conf?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. Es el formato del archivo /etc/hosts:
   a) IP nombre
   b) IP nombre alias
   c) IP dominio nombre
   d) IP alias nombre

2. Archivo que especifica el dominio del sistema:
   a) /etc/domain
   b) /etc/defaultdomain
   c) /etc/maindomain
   d) /etc/domains

3. Permite deshabilitar la tarjeta de red:
   a) ipconfig down eth0
   b) ifconfig down eth0
   c) ipconfig eth0 down
   d) ifconfig eth0 down

4. Especifica los servidores de nombres:
   a) /etc/resolvconf
   b) /etc/resolv
   c) /etc/nameservers
   d) /etc/resolv.conf

5. Este archivo contiene puertos habilitados:
   a) /etc/ports
   b) /etc/ports.conf
   c) /etc/services
   d) /etc/services.conf

6. Contiene los demonios de red que son manejados a través de inetd:
   a) /etc/inetd
   b) /etc/inetd.conf
   c) /etc/netd
   d) /etc/netd.conf

7. Con este comando se agrega un ruteador:
   a) route add
   b) add route
   c) route_add
   d) add_route

8. Se puede dar de baja un ruteador:
   a) route delete
   b) delete route
   c) route_delete
   d) delete _route

9. Sirve para obtener información de la red:
   a) netinfo
   b) netstat
   c) ifconfig
   d) ipconfig

10. Este parámetro no se encuentra en el archivo de configuración `/etc/resolv.conf`.
    a) domain
    b) nameserver
    c) search
    d) find

TEMA 10. ADMINISTRACIÓN DE LA MEMORIA VIRTUAL

Objetivo particular

El alumno reconocerá el funcionamiento de la memoria virtual (swap) del sistema operativo Unix en relación con la memoria principal de dicho sistema, e identificará los principales archivos de configuración para añadir o eliminar swap, así como los procedimientos a seguir para la administración y monitoreo del mismo.

Temario detallado (6 horas)

10.1. Conceptos básicos
10.2. Swap
10.3. Procedimiento para añadir swap
10.4. Procedimiento para eliminar swap
10.5. Monitoreo del swap

Introducción

La memoria es un recurso del sistema que debe ser administrado eficientemente. Es importante recordar que la memoria principal (RAM) de un sistema es finita.

En un sistema multitarea, donde varias tareas se ejecutan en forma concurrente, éstas comparten los recursos. En un sistema multiusuario, dichos recursos no sólo son compartidos entre las diferentes tareas, sino entre los diferentes usuarios.

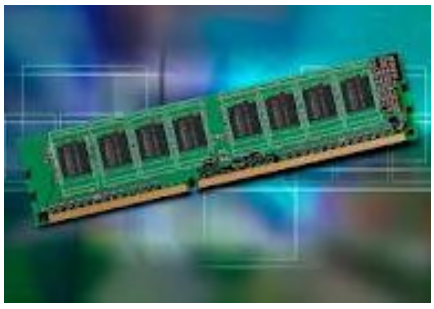
Si bien, el propio diseño del sistema operativo ayuda a gestionar la memoria proporcionando mecanismos adecuados para ello, corresponde al administrador del sistema configurar adecuadamente el mismo.

En este tema se presentarán los conceptos generales que debe tener en cuenta un administrador de sistemas para gestionar adecuadamente la memoria en un sistema UNIX; se abordarán, en forma general, los procedimientos de configuración del área de intercambio (swap) tanto en SV como BSD y se presentarán las principales herramientas que existen para su administración y monitoreo.

10.1.Conceptos básicos

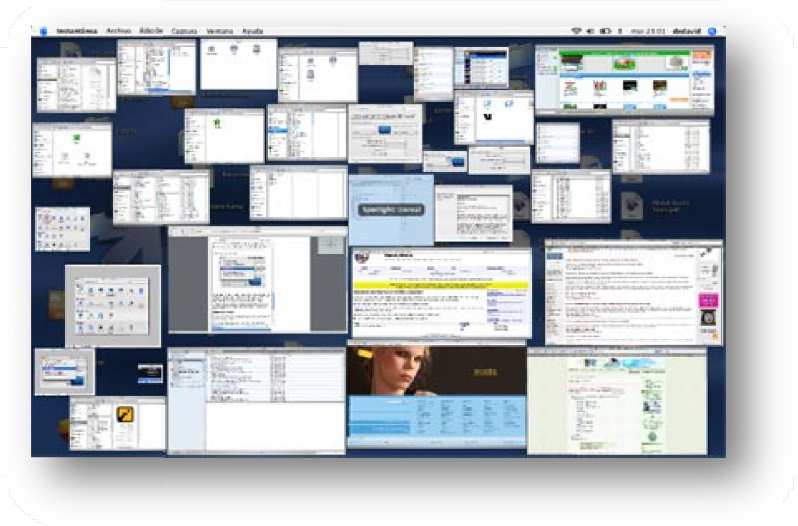
Memoria principal

La memoria principal es uno de los componentes más importante de un sistema de cómputo. Está constituida por un conjunto de celdas, referenciables por medio de una dirección lineal, capaces de retener información por un tiempo determinado.



Para que un programa se ejecute, su código y sus datos necesitan estar cargados en memoria.

En un sistema multitarea, la memoria se reparte entre los diferentes procesos ejecutados en el sistema. Asimismo, las rutinas del sistema operativo también residen en memoria, por lo que se pueden dar situaciones en las que la memoria principal no tenga capacidad suficiente para todos los procesos en ejecución. El sistema operativo es el encargado de gestionar la memoria de un sistema aprovechando eficientemente el espacio disponible y evitando los conflictos de uso.



Existen varios esquemas que utilizan los sistemas operativos con el fin de administrar la memoria. En este tema se mencionarán únicamente, y de forma muy general, los mecanismos de Intercambio y Paginación.

Intercambio (Swapping)

El esquema de intercambio implica la utilización de la memoria secundaria del sistema como un área de intercambio.

El área de intercambio (swap) se comporta como una extensión de la memoria principal. Cuando los procesos no caben en la memoria principal, aquellos que no están activos o aquellos con más baja prioridad, son expulsados de la misma, copiando su imagen al área de swap (swap out). Una vez liberado espacio en la

memoria principal se pueden crear nuevos procesos o se pueden intercambiar los que están en swap hacia la memoria principal (swap in).

Paginación (Paging)

En la paginación, el espacio de direcciones lógicas de un proceso se divide en unidades llamadas **páginas**. Las unidades correspondientes en la memoria física se denominan **cuadros de página**. Las páginas y los cuadros de página siempre son del mismo tamaño.

El esquema de paginación utiliza la memoria secundaria como un área de intercambio, sin embargo las transferencias entre la memoria y el disco, siempre están en unidades de página. Se mantienen en la memoria principal las páginas que están usando los procesos activos. Si un programa direcciona una página que no está en memoria principal, se produce una falla de página. El sistema operativo toma un cuadro de página con poco uso y vuelve a escribir su contenido en el disco. Después captura la página recién referida en el cuadro de página que acaba de liberarse.

Memoria Virtual

Mecanismo de administración de la memoria. Permite la ejecución de procesos cargados parcialmente en la memoria principal del sistema. Utiliza la unidad de almacenamiento secundario como área de intercambio. La idea es mantener en memoria principal sólo los fragmentos de cada proceso que se estén utilizando.

> En un estado estable, prácticamente toda la memoria principal estará ocupada con fragmentos de procesos, por lo que el procesador y el sistema operativo tendrán acceso directo a la mayor cantidad de procesos posible. Así pues, cuando el sistema operativo traiga a memoria un fragmento, deberá expulsar otro.(Stallings, 1997, p. 285)

Las principales **ventajas** de la utilización de memoria virtual son, que:

- Permite trabajar con programas de mayor tamaño que la memoria física.
- Permite tener más programas cargados a la vez.
- Reduce el uso de E/S para la el intercambio. No se intercambian procesos completos, sólo fragmentos de éstos.

La **desventaja** principal del uso de memoria virtual es que los dispositivos de almacenamiento secundario, utilizados como áreas de intercambio, son más lentos que la memoria física del sistema, por lo que cualquier acceso al sistema de E/S para recuperar un dato o parte de algún proceso tendrá un costo en tiempo y, por ende, en el rendimiento del sistema.

10.2. Swap

El **swap** es el área de intercambio manejada por el sistema operativo UNIX. Su funcionamiento depende estrictamente del sistema operativo. Algunos manejan **intercambio** (swapping), otros **paginación** (paging), y la gran mayoría una técnica denominada **paginación por demanda** que combina tanto intercambio como paginación.

Durante la instalación del sistema operativo UNIX se puede configurar una partición de swap. El tamaño de la partición de swap depende de las características y el uso que se quiera dar al sistema. En equipos con poca memoria (menos de 4Gb) se recomienda que el swap sea al menos 2 veces del tamaño de la memoria RAM. En equipos con memoria promedio (entre 4 y 8Gb), se recomienda que el swap sea al menos del tamaño de la memoria física. Para equipos que tienen gran cantidad de memoria, el swap puede no ser utilizado, por lo que aparentemente puede ser innecesario, aún así se recomienda siempre tener un área de swap configurada en el sistema (al menos la mitad de la memoria física), especialmente si dicho sistema es utilizado para correr procesos demandantes en memoria como bases de datos, servidores de archivos, algunos servicios de internet, correo electrónico, desarrollo y compilación de aplicaciones, etc. El swap también es necesario para activar las opciones de hibernación, comúnmente utilizadas en laptops y notebooks.

Tipos de swap

UNIX soporta dos tipos de swap:

1.	**Partición física (device swap)** Una partición ordinaria de disco se asocia como área de intercambio. Dependiendo del sistema operativo, se identifica como tipo swap, raw, sw o linux swap. 
2.	**Archivo dedicado (file system swap)** El swap se configura en un archivo regular que se encuentra en un sistema de archivos. Este archivo regular se utiliza como área de intercambio. Los archivos de swap son muy útiles si se necesita expandir el área de swap y no es posible asignar espacio en el disco duro para crear una partición mayor.

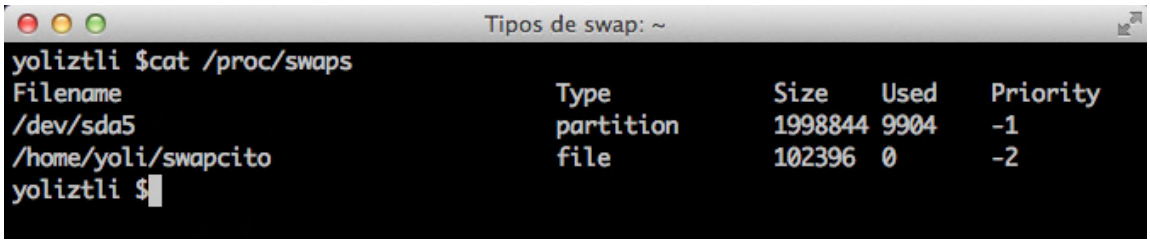
El archivo regular debe crearse de la siguiente forma:

En BSD

```
dd if=/dev/zero bs=1g count=tamañoengigas of=archivo_swap
```

En SV

```
mkfile tamaño[k|b|m|g] archivo_swap
```



10.3. Procedimiento para añadir swap

Añadir swap en un sistema operativo UNIX se realiza siguiendo los pasos que se enlistan a continuación:

1. Seleccionar el dispositivo.Partición física o Archivo dedicado (previamente creado).

2. En algunos casos se debe configurar el dispositivo como swap. Ejemplo:

UNIX	Comando
Linux	`mkswap {swapfile|swapdevice}`
FreeBSD	`vnconfig -c swapdevice swapfile`

3. Activar el swap.
   Una vez que se ha creado y elegido el dispositivo, y si es el caso configurado, se debe activar el área de intercambio.

BSD (Incluyendo Linux)	SV
`swapon {swapfile|swapdevice}`	`swap -a {swapfile|swapdevice}`
En OpenBSD `swapctl -a {swapfile|swapdevice}`	

4. Si se desea dejar el swap configurado permanentemente, se deben modificar los archivos correspondientes con las opciones que se indican a continuación:

BSD	SV
/etc/fstab `{swapfile|swapdevice} swap swap defaults 0 0`	/etv/vfstab `/appl/swap - - swap - no -`

Nota: *En algunos casos las opciones pueden variar dependiendo del sistema operativo*.

10.4. Procedimiento para eliminar swap

Si por alguna causa se desea eliminar algún área de swap, se debe verificar, primero, que ésta no esté en uso (*tema 10.5*), es decir 100% libre.

Una vez que se verifica que dicha área no está siendo utilizada, se puede proceder a eliminarla:

1. Desactivar el swap.

BSD (Incluyendo Linux)	**SV**
`swapoff {swapfile|swapdevice}`	`swap -d {swapfile|swapdevice}`
En OpenBSD `swapctl -d {swapfile|swapdevice}`	

2. Para el caso de BSD, si el área de swap fue configurada con `vnconfig`, se deberá desconfigurar.

```
vnconfig -u swapdevice
```

3. Si el área de swap fue dada de alta en los archivos de configuración (/etc/fstab para BSD o /etc/vfstab en SV), eliminar la línea correspondiente.

4. Si el dispositivo utilizado fue un archivo dedicado (filesystem swap), eliminar el archivo.

```
rm swapfile
```

Si el dispositivo fue una partición física se puede reutilizar.

10.5. Monitoreo del swap

Como se mencionó anteriormente, el sistema operativo es el encargado de gestionar la memoria del sistema, sin embargo el administrador es quien configura las áreas de intercambio.

Es importante que el administrador conozca en todo momento cómo está funcionando el sistema con el fin de determinar si la configuración es adecuada. En algunos casos puede suceder que el swap no se utilice, en otros puede ser insuficiente. Será tarea del administrador decidir cualquier cambio o extensión a la memoria del sistema, ya sea física o virtual.

Dependiendo del sistema operativo, existen diversos comandos para monitorear el swap. Los más comunes se listan a continuación:

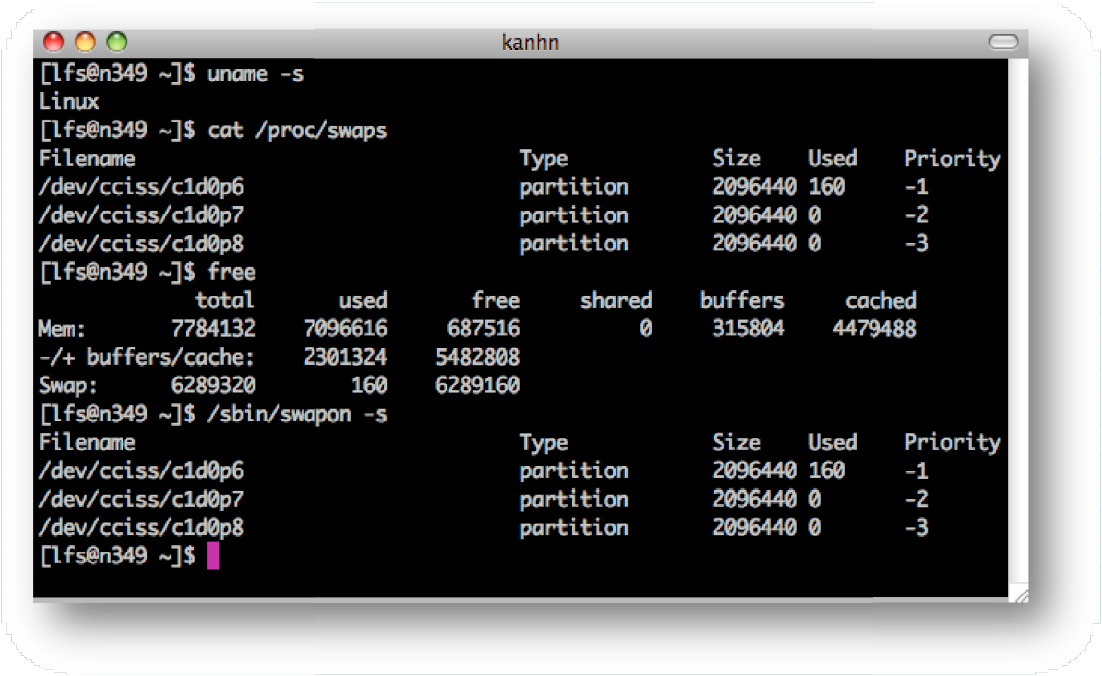
BSD

```
pstat –s
```

En **Linux,** los dispositivos de swap y su uso se pueden ver en tres formas:

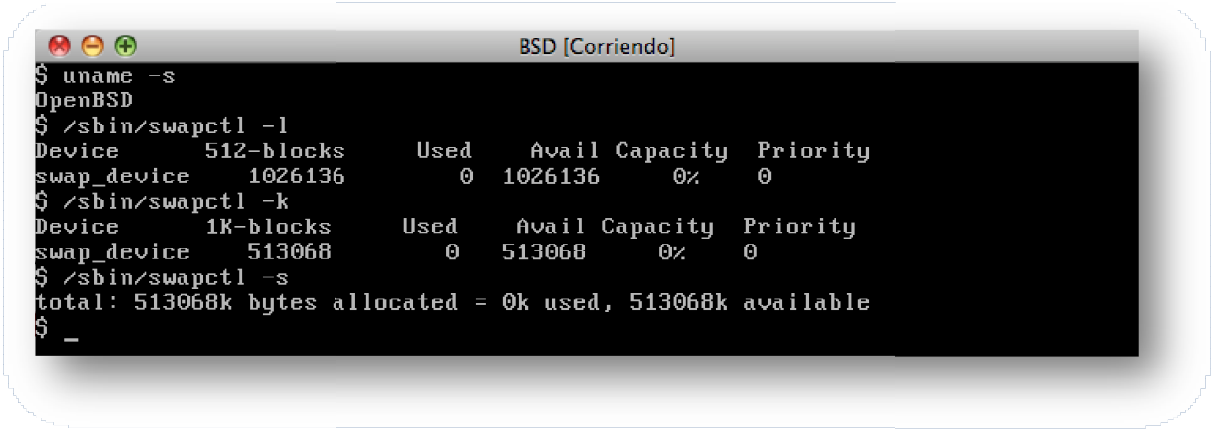
```
cat /proc/swaps
```

```
free
swapon -s
```



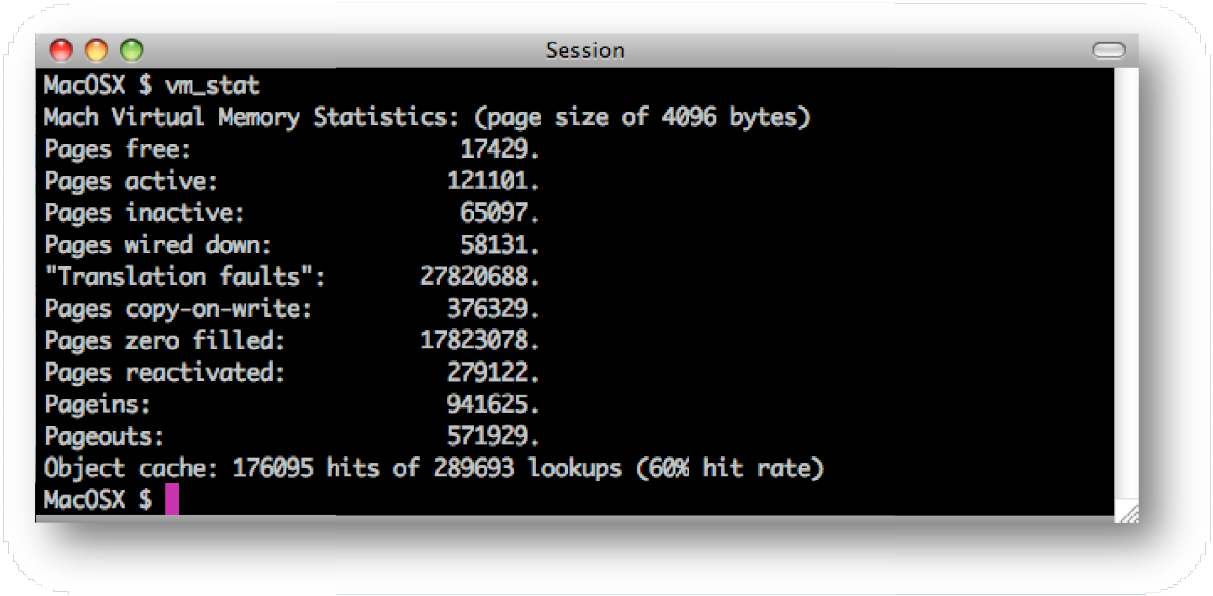
En **OpenBSD**:

`swapctl -l`	Lista uno por uno los dispositivos de swap
`swapctl -k`	Lista con tamaños en Kilobytes en vez de bloques de 512 bytes
`swapctl -s`	Lista el total



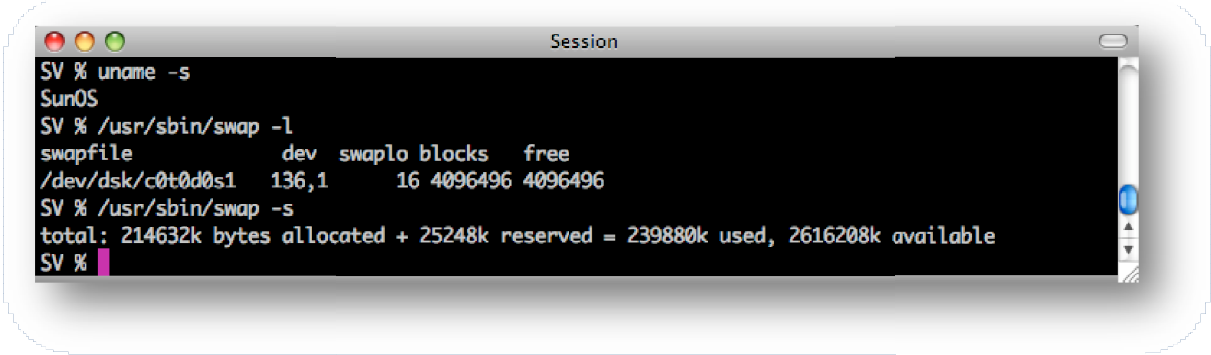
En **MacOSX**:

`vm_stat`	Proporciona estadísticas de uso de la memoria virtual



SV

`swap -l`	Lista los dispositivos de swap uno por uno
`swap -s`	Lista el total



Bibliografía básica del tema 10

Flores, Y.; Caballero, R. (2006). *Técnicas básicas de Administración del Sistema Operativo UNIX*. Plan de Becarios en Supercómputo, DGSCA, Universidad Nacional Autónoma de México.

Frisch, A. (2002). *Essential System Administration*. (3ª ed.) Sebastopol, CA: O'Reilly Media.

Maxwell, S. (2002). *UNIX System Administration*. *A Beginner's Guide.*EUA: McGraw-Hill.

Bibliografía complementaria

Stallings, W. (1997).*Sistemas Operativos*. (2ª ed.) Madrid: Prentice Hall.

Sitios de Internet

Memoria física y virtual.(2005). En *Red Hat Enterprise Linux 4: Introducción a la administración de sistemas*. Red Hat, Inc. Disponible en línea: http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-isa-es-4/ch-memory.html, consultado el 06/01/12.

Actividades de aprendizaje

A.10.1. Elabora un cuadro sinóptico de los tipos de swap y los comandos principales de configuración del mismo.

A.10.2. Describe, con tus propias palabras, en un párrafo de no más de 20 líneas, ¿qué es el swap y cómo funciona?

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿Qué es la memoria virtual?
2. ¿Qué es el swap?
3. ¿Cuál es la diferencia entre memoria virtual y la memoria RAM?
4. ¿En qué consiste el proceso de paginación?
5. Menciona los principales criterios para determinar el tamaño del área de swap en un sistema UNIX.
6. Describe los tipos de swap de Unix y cómo se relacionan.
7. ¿Con qué comandos puedo manipular el swap en OpenBSD?
8. ¿Cuál es el tamaño recomendado para el swap?
9. ¿Con qué comando se puede verificar cuánta memoria virtual disponible queda en el sistema?
10. ¿Cuál es la función del comando `swapon`?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. Un tipo válido de swap es:
   a) virtual
   b) dinámico
   c) de carga
   d) físico

2. De las siguientes afirmaciones sobre memoria virtual, selecciona la que más se acerque a una definición correcta:
   a) Es de gran capacidad y gran velocidad.
   b) Es de de poca capacidad pero gran velocidad.
   c) Es de gran capacidad pero poca velocidad.
   d) Es de poca velocidad y poca capacidad.

3. El comando que permite agregar memoria virtual en OpenBSD, es:
   a) `swapctl –a`
   b) `swapctl –add`
   c) `swapctl --configure –add`
   d) `swapctl --configure –a`

4. Este tipo de memoria se caracteriza por ser de menor velocidad:
   a) RAM
   b) Caché
   c) Swap
   d) ROM

5. La importancia del swap radica en que permite:
   a) aumentar la velocidad de nuestro sistema.
   b) ejecutar una gran cantidad de aplicaciones sin que el sistema se vuelva torpe.
   c) asignar más memoria de que la que tenemos físicamente disponible a los programas.
   d) almacenar más aplicaciones en el sistema.

6. Es el comando que permite dar de baja swap en Solaris (recuerda que Solaris es un sistema operativo apegado a SV):
   a) `swap –d`
   b) `swap –delete`
   c) `swapctl --configure –delete`
   d) `swapctl --configure –d`

7. ¿Para qué sirve el comando `mkswap` en Linux?
   a) Crea un archivo para poder ser montado como swap.
   b) Crea una partición de swap y la agrega a la tabla de particiones.
   c) Inicializa un dispositivo de swap.
   d) Es un comando no válido en Linux.

8. En un sistema de 512 megabytes de memoria RAM, ¿cuál es el tamaño recomendado de swap?
   a) 512 megabytes
   b) 1 gigabyte
   c) 2 gigabytes
   d) 5 gigabytes

9. Un usuario requiere correr un programa que utiliza más memoria de la que se tiene disponible en un sistema UNIX. El área de cómputo no cuenta con presupuesto para comprar más memoria RAM. ¿Qué opción tiene el administrador para dar servicio a dicho usuario sin afectar a los demás usuarios del sistema?
   a) Crear un archivo y añadirlo como dispositivo de swap.
   b) Reparticionar el disco, aumentando el tamaño de la partición de swap.
   c) Modificar los privilegios del usuario
   d) Pedir al usuario que modifique su código para que no ocupe tantos recursos del sistema.

10. El comando que permite activar el proceso de intercambio en un área de swap, en un sistema UNIX es:
   a) `swap`
   b) `swapctl`
   c) `swapon`
   d) `swap –a`

TEMA 11. MONITOREO DEL DESEMPEÑO DEL SISTEMA

Objetivo particular

El alumno reconocerá la importancia del monitoreo del desempeño de un sistema Unix, así como la utilidad del uso del registro de bitácoras (`syslog`) y su configuración. A partir de ello, será capaz de monitorear la actividad de un sistema Unix, empleando las principales herramientas.

Temario detallado (6 horas)

11.1. Localización y solución de problemas
11.2. Bitácoras
11.3. Herramientas de monitoreo
11.4. Desempeño del sistema

Introducción

Actualmente, las empresas y organizaciones están dependiendo cada vez más de sus recursos informáticos y de los servicios que éstos les proporcionan para realizar sus operaciones diarias. Debido a ello, la inversión en infraestructura informática va siendo mayor, pues resulta vital su buen funcionamiento.

En este caso, el **Administrador de Sistemas** será el responsable de mantener el adecuado funcionamiento del sistema, y es quien determinará los parámetros que le permitan conocer cuándo el sistema se comporta en forma adecuada y cuándo no.

Asimismo, el administrador se encargará de supervisar continuamente los recursos y servicios, y comparará su funcionamiento contra la política establecida. También será el encargado de detectar cualquier posible falla en el sistema, a través de la supervisión de los registros o bitácoras del mismo.

En este tema se abordarán los principales elementos que le permitirán al administrador, monitorear su sistema. Se revisará, en primer lugar, el syslog o sistema de bitácoras centralizado de UNIX, y posteriormente se mencionarán las principales herramientas de monitoreo y recomendaciones para determinar el adecuado comportamiento del sistema y mejorar el desempeño del mismo.

11.1 Localización y solución de problemas

Realizar el monitoreo del desempeño de un sistema es una tarea compleja, ya que los sistemas están compuestos por diversos recursos que interactúan constantemente y se deben considerar las interrelaciones entre los distintos componentes del mismo, no sólo los componentes individuales.

Al proceso de localizar y corregir errores en el sistema se le conoce como *Troubleshooting*. Esto es, cuando el sistema no actúa como se espera, se analizará para determinar las causas que generan este comportamiento anómalo y, de este modo, realizar las modificaciones necesarias que conlleven al sistema.

Llevar a cabo este proceso (*troubleshooting*), implica que el administrador debe:

- Determinar el problema (¿qué sucede?).
- Detectar la causa del problema (¿por qué sucede?)
- Realizar modificaciones al sistema (acciones correctivas)

En el tema **1** vimos que es muy importante para un administrador documentar y registrar las actividades realizadas, a fin de dejar constancia de los cambios que se hagan en cualquier elemento del sistema, por lo que se registrarán actividades tales como:

- Actualizaciones
- Cambios en la configuración
- Instalación de aplicaciones
- Solución de problemas

De la misma forma, debe conocer y revisar constantemente las bitácoras del sistema, ya que éstas le serán muy útiles al momento de determinar las causas de un problema. El sistema de bitácoras de UNIX permite registrar tanto la actividad del sistema como los errores del mismo.

11.2. Bitácoras

La localización de las bitácoras en el sistema operativo UNIX puede variar de acuerdo con la versión y configuración local del mismo. En forma casi genérica, suelen encontrarse bajo el directorio `/var/log`, siendo las más comunes:

Archivo	Bitácora
`pacct`	Contabilidad de procesos. Graba todos los comandos ejecutados en el sistema. No se activa por omisión ya que puede consumir muchos recursos en el sistema.
`lastlog`	Registra los últimos accesos.
`utmp`	Cada usuario conectado en el sistema.
`wtmp`	Registro de conexiones de usuarios.
`messages`	Bitácora principal del sistema.

Syslog

El **sistema centralizado de bitácoras** de UNIX, conocido como `syslog`, permite registrar mensajes de actividad del sistema. Cada mensaje consta de 3 partes fundamentales:

1. Nombre del programa, demonio o sistema que genera el mensaje	**2**. Nivel de urgencia del mensaje	**3**. Texto del mensaje

Categoría del mensaje

Los mensajes se agrupan en diferentes categorías, dependiendo del programa que los genera. Las más comunes son:

Categoría	Mensajes generados por:
`kern`	Mensajes del kernel
`user`	Procesos de usuarios
`auth`	Sistema de autorización y acceso
`cron`	Demonio de cron
`daemon`	Demonios del sistema

Categoría	Mensajes generados por:
`lpr`	Sistema de impresión
`mail`	Sistema de correo electrónico
`security`	Subsistema de seguridad
`local0 a local7`	A definir por el administrador del sistema

Prioridades

La prioridad de un mensaje se determina de acuerdo con su categoría y su nivel de urgencia. Los mensajes se agrupan en niveles de acuerdo con la importancia de los mismos.

Nivel	Significado
`emerg`	Error muy grave. Caída inminente del sistema
`alert`	Condición que debe ser corregida de inmediato
`crit`	Condición crítica
`err`	Error ordinario
`warning`	Avisos de advertencia.
`notice`	No es un error pero debe ser revisado
`info`	Mensaje informativo
`debug`	Incluye todo tipo de mensajes

Adicionalmente, para la configuración de syslog, se utiliza el siguiente nivel:

Nivel	Significado
`none`	No incluir mensajes

Demonio de syslog

El demonio `syslogd` recibe y procesa los mensajes del sistema de acuerdo con la configuración especificada en el archivo:

```
/etc/syslog.conf
```

Archivo de configuración del sistema de bitácora centralizada de UNIX

Los mensajes se envían al destino correspondiente dependiendo de su prioridad (categoría.nivel).

El administrador indica si los mensajes van a un archivo, a la terminal, o serán procesados por algún otro demonio de syslog.

El formato de este archivo es:

```
selector     accion
```

Selector

El selector es la lista de especificaciones de prioridad, es decir, la unión de una categoría y un nivel de prioridad.

```
categoria[,categoria].nivel[;categoria.nivel]
```

Ejemplo:

```
*.debug;mail.none
```

* Registra todos los mensajes, excepto aquellos generados por el correo electrónico.

Acción

Especifica qué hacer con el mensaje. Generalmente la acción indica el nombre de un archivo donde se registrará el mensaje, pero se puede indicar también el nombre de un usuario a quién se avisará en caso de estar conectado, o el nombre de un host para que sea procesado por un demonio de syslog remoto.

Ejemplo

```
*.emerg                         *
*.alert                         root
mail.*                          /var/log/mail.log
```

* Todos los mensajes graves serán enviados a la terminal de todos los usuarios conectados en el sistema.
* Los mensajes de alerta serán enviados a la terminal de root.
* Los mensajes generados por el correo electrónico se registrarán en el archivo`/var/log/mail.liog`

Ejemplo de un archivo de bitácora de UNIX:

Es importante que el administrador tome en cuenta que las bitácoras tienden a crecer y ocupar espacio en disco. Puede utilizar la utilería cron para ciclarlas. Asimismo, las bitácoras se deben incluir en el programa de respaldos del sistema.

11.3. Herramientas de monitoreo

Cuando se habla de desempeño de un sistema, existen tres recursos fundamentales: el CPU, la memoria y el subsistema de Entrada/Salida. Los

administradores deben tener pleno conocimiento del comportamiento de dichos recursos, ya que del correcto ajuste en su interacción dependerá un desempeño eficiente.

El sistema operativo UNIX proporciona herramientas que permiten al administrador realizar un monitoreo adecuado de su equipo. A continuación se presentan las más comunes.

Monitoreo de la CPU

Este recurso implica la disponibilidad de ciclos de procesador. El kernel distribuye los ciclos entre los procesos de acuerdo con el nivel de prioridad y número de ciclos requeridos para ejecutarse.

Carga del sistema y estado de procesos
uptime • Comando útil para ver la carga del sistema. • Muestra el tiempo actual del sistema, el tiempo que la máquina ha estado levantada, y nos muestra el porcentaje de procesos en la cola del scheduler o planificador de procesos, en los últimos 1, 5 y 15 minutos. • El sistema se considera cargado cuando reporta un número superior a 2 veces el número de procesadores de la máquina.
ps Comando que sirve para ver el estado de los procesos. Proporciona información sobre: • Usuario • Tiempos de CPU • Memoria calculada • Memoria reservada • Prioridad de los procesos • Modificador de prioridad (nice) • Estado de los procesos • Comandos

w

Muestra qué usuarios están en el sistema y da una idea general de qué están haciendo. Los campos desplegados por omisión son:

- Clave de usuario
- Tty
- Host desde el que está conectado el usuario
- Tiempo que lleva en sesión
- Tiempo de espera de la terminal
- Tiempos de CPU

La primera línea que muestra la salida de este comando es la salida del comando `uptime`.

Monitoreo de la memoria

La memoria se entiende como la disposición de espacio para almacenamiento temporal de información la cual se caracteriza por ser de rápido acceso. Es importante la relación de la memoria con el área de swap. En circunstancias de saturación de la memoria, el sistema hará uso del área de intercambio para simular memoria virtual, la cual, al acceder a un sistema de almacenamiento secundario, será más lenta.

Estado de la memoria virtual
vmstat (BSD) - Reporta estadísticas de la memoria virtual. - Reporta procesos tanto en la cola de ejecución como en la cola de espera. r Procesos en la cola de ejecución. b Procesos en esperando por el administrador de memoria virtual para paginar parte del proceso a memoria real.

`avm`	Tamaño de la memoria virtual activa.
`fre`	Tamaño real de la memoria libre.
`us`	Porcentaje de tiempo de CPU operado en modo usuario.
`sy`	Porcentaje de tiempo de CPU operado en modo kernel.
`id`	Porcentaje de tiempo de CPU inactivo, sin procesos ni E/S pendientes.

- Reportan uso de la memoria y el swap (ver tema10)
- Reporta uso del área de swap (ver tema 10)

free, /proc/meminfo y /proc/swaps (linux)

swap (SV)

Monitoreo de Entrada/Salida

La cantidad de información o de datos que se puede enviar para las operaciones de entrada y salida debe ser compartidaentre todos los programas en ejecución (incluyendo el kernel).

En el subsistema de Entrada Salida interactúan condiciones físicas y lógicas tales como: velocidad en los procesos de escritura y lectura, disposición en los medios internos de comunicación, atención del kernel a las peticiones sobre servicios, etc. Es importante tener datos sobre la tasa de transferencias en los dispositivos, procesos en espera por servicio de E/S y estadísticas de la red, para determinar el desempeño del sistema.

Subsistema de Entrada/Salida	
iostat	
Monitorea las actividades de entrada/salida. También da información sobre el uso de la CPU.	
`%tm`	Porcentaje. Uso de ancho de banda del dispositivo.

`Kbps`	Cantidad de datos leídos y escritos en Kbytes por segundo para el drive
`tps`	Las transferencias (peticiones de E/S) por segundo hechas al disco.
`Kb_read`	El número de Kbytes leídos desde un drive
`Kb_wrtn`	El número de Kbytes escritos en el drive

Red
netstat Permite conocer datos relacionados a las conexiones activas de red tales como: • Protocolo en uso • Tablas de ruteo • Estadísticas de las interfaces • Estado de las conexiones

11.4. Desempeño del sistema

El desempeño de un sistema depende de la eficiencia con que los recursos son utilizados por los diversos procesos que corren en él.

Los parámetros para medir el desempeño de un sistema varían de acuerdo con las necesidades del sitio donde cumple sus funciones. Aspectos tales como tipos de usuarios que hacen uso del equipo, requerimientos para acceder a los recursos, aplicaciones utilizadas, volumen de información que se maneja, horas de alto uso, etc., influyen sobre su comportamiento.

Para todo administrador es importante conocer el comportamiento “normal” del sistema ante situaciones comunes de trabajo. Características como un sistema lento, trabajos que tardan mucho en terminar, sistemas saturados, etc., pueden ser indicativos de cambios necesarios en la configuración o arquitectura de un sistema.

Es importante que el administrador tome en cuenta todos los elementos involucrados antes de realizar algún cambio ya que el cambio en un componente del sistema puede afectar a otro. También se deben tomar en cuenta las necesidades de todos los usuarios. Las quejas de los usuarios pueden ser un indicador de problemas, al igual que los cambios repentinos en el comportamiento del mismo.

Actividad de la CPU

A grandes rasgos, la cantidad de tiempo requerido para ejecutar un proceso se descompone en:

Tiempo	Significado
Real	Tiempo total que tarda un código en ejecutarse.
Usuario	Tiempo dedicado a la ejecución de un código en “estado de usuario”.
Sistema	Tiempo dedicado a la ejecución de un código en “estado sistema”. Incluye tiempo gastado en ejecutar llamadas al sistema y operaciones de E/S.

El tiempo que la CPU permanece sin utilizarse se denomina *idle.*

Se recomienda ajustar el uso de la CPU o actualizarlo en ciertos casos:

Indicador		Significado
%idle < 5 %user →100	Por largos periodos de tiempo	La CPU está al máximo
%sys > 25		Puede indicar un cuello de botella en operaciones E/S o UNIX invierte mucho tiempo en su propio manejo.

Dependiendo del caso, el administrador debe:

- Identificar programas específicos que sobrecargan la CPU.
- Identificar programas específicos que están realizando muchas llamadas al sistema.

Se puede dar una situación en la que muchos procesos en segundo plano se ejecutan con una prioridad alta y consumen mucho tiempo de la CPU o un proceso esté fuera de control. Si el tiempo de respuesta fuera inaceptable, el administrador puede disminuir la prioridad de algunos procesos y terminar los procesos no deseados. Si aún así el nivel de carga del procesador se mantiene alto, se puede intentar distribuir las cargas de trabajo en diferentes horarios (alto y bajo uso). Si no fuesen suficientes estas opciones, se debe pensar en un crecimiento del sistema.

Actividad de la memoria

Los problemas de memoria se presentan cuando los requerimientos de la misma, por parte de los procesos, exceden la capacidad de la memoria principal disponible en el sistema. Para manejar la falta de memoria, el sistema inicia el proceso de paginación, es decir, mueve porciones de los procesos activos a disco en la medida que es reclamada la memoria.

Para prevenir la paginación, se debe aumentar físicamente la memoria o disminuir la cantidad de memoria requerida por los procesos. Si no es posible realizar

alguna de las dos cosas y, con el fin de que el sistema no se colapse por falta de memoria, se puede incrementar el tamaño de la memoria virtual.

Actividad del subsistema de E/S

Los dispositivos de almacenamiento secundario y la red afectan el desempeño general del sistema.

Se puede presentar un problema de E/S cuando:

- Pocos procesos demandan intensivamente operaciones de E/S.
- Demasiados procesos demandan operaciones de E/S.
- La CPU permanece inactiva, en espera de que terminen las solicitudes de E/S.

La configuración del sistema debe realizarse con base en las características del mismo. Se recomienda colocar la información a la que se obtenga acceso con más frecuencia en los discos más rápidos y distribuir la carga de trabajo de los discos y controladoras en forma equitativa. En los servidores de archivos, se deben destinar los discos más rápidos a los sistemas de archivos compartidos y al espacio de intercambio.

Los problemas de E/S para la red son de dos formas: la red se sobrecarga o pierde integridad de los datos. Cuando una red está sobrecargada, la cantidad de datos que necesita transferir es mayor a la capacidad de la misma. Estos problemas normalmente se solucionan cambiando la configuración de la red. En el caso de los errores de red, éstos deben ser escasos en relación con la cantidad de paquetes transferidos. Un índice alto de errores en un solo servidor puede indicar que existe un problema físico con el equipo. El uso excesivo de la red incrementará el número de colisiones. Es importante que el administrador verifique esta condición pues puede ser percibida como mal desempeño del sistema.

Bibliografía básica del tema 11

Flores, Y., (2006). *Tópicos selectos de Administración de Supercómputo*. Plan de Becarios en Cómputo de Alto Rendimiento, DGSCA, Universidad Nacional Autónoma de México.

Loukides, Michael Kosta; Loukides, Mike. (1990). *System Performance Tuning*.Sebastopol, CA: O'Reilly Media.

Musumeci, Gian-Paolo D.; Loukides, Mike. (2002). *System Performance Tuning*. (2nd ed.) Sebastopol, CA:O'Reilly Media. [vista previa]

Bibliografía complementaria

HP. (2008). *Guía del administrador de sistemas HP-UX: Tareas de administración rutinarias. HP-UX 11i versión 3*, disponible en línea: http://bizsupport2.austin.hp.com/bc/docs/support/SupportManual/c01976185/c01976185.pdf, consultado el 10/01/12

Pruett, C., Strickland, K. Vetter, S. (2001). *IBM eServer Certification Study Guide - pSeries AIX System Administration.*Disponible en línea:http://www.redbooks.ibm.com/abstracts/sg246191.html?Open , consultado el 10/01/12.

Sitios de Internet

Red Hat, Inc. (2005). Capítulo 2. Resource Monitoring, en *Red Hat Enterprise Linux 4: Introducción a la administración de sistemas.*Disponible en línea: http://docs.redhat.com/docs/es-ES/Red_Hat_Enterprise_Linux/4/html/Introduction_to_System_Administration_Guide/ch-resource.html, consulado el 10/01/12.

Giorgio Ingargiola. (s.f.) *Some useful UNIX System Commands and Tools.*Temple University, Philadelphia, PA. Disponible en línea: http://www.cis.temple.edu/~ingargio/cis307/readings/system-commands.html, consulado el 10/01/12.

Actividades de aprendizaje

A.11.1. Elabora un mapa conceptual donde expliques el funcionamiento del sistema centralizado de bitácoras de UNIX.

A.11.2. Elabora un cuadro sinóptico en el cual agrupes, por recurso (CPU, memoria, disco, red), las diversas herramientas de monitoreo que revisamos en este temay donde describas para qué se utiliza cada una de ellas.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿Qué es una bitácora en UNIX?
2. ¿Cuáles son las bitácoras más importantes? Explica cada una de ellas.
3. Describe qué es el sistema *syslog*.
4. ¿Para qué sirve el comando *uptime*?
5. ¿Para qué sirve el comando *ps*?
6. ¿Qué comando empleamos para conocer los usuarios que están conectados en el sistema?
7. Describe cuando menos tres elementos reportados por el comando *vmstat*.
8. ¿A qué se denomina tiempo "*idle*"?
9. Enumera tres ventajas de conocer los comandos para el monitoreo del equipo.
10. ¿Qué acciones tomarías para prevenir un problema de E/S con base en el monitoreo y los parámetros de desempeño del sistema?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. El comando `uptime` se puede definir como un comando:
   a) útil para ver la carga del sistema.
   b) útil para ver los procesos del sistema.
   c) para reportar el número de procesadores de la máquina.
   d) que muestra el estado de la memoria.

2. La sentencia que mejor describe al comando `ps`, es el siguiente:
   a) Es un comando que sirve para ver el estado de los procesos en ejecución.
   b) Es un comando útil para ver el estado de la memoria virtual
   c) Es un comando para manipular el sistema de procesos del sistema.
   d) Es un comando el sistema de impresión de UNIX.

3. El término *Troubleshooting*a qué hace referencia:
   a) Al procedimiento para localizar y corregir errores en un sistema.
   b) A un proceso del sistema para localizar y corregir errores en el mismo.
   c) Al registro de los errores del sistema en sus correspondientes bitácoras.
   d) Al proceso de revisar las bitácoras del sistema para ver que errores hubo.

4. El directorio del sistema operativo Unix donde se localizan comúnmente las bitácoras es:
   a) /etc/log
   b) /etc/var
   c) /usr/local/log
   d) /var/log

5. Elementos genéricos que conforman un mensaje registrado por `syslog`:
   a) nombre del programa, demonio o sistema, nivel de urgencia, texto del mensaje.
   b) código de error, nombre del demonio o sistema, texto del mensaje.
   c) numero de error, nivel de urgencia, texto del mensaje.
   d) únicamente cuenta con el texto del mensaje.

6. Son categorías válidas para `syslog`:
   a) kern, lpr, mail
   b) kern, log, var
   c) kern, etc, var
   d) kern, mesg, local0 a 7

7. Es el archivo de configuración de syslog:
   a) /etc/syslog.conf
   b) syslogd
   c) /var/log/syslog.conf
   d) /etc/syslog.conf.ini

8. Es importante llevar el monitoreo del desempeño de un equipo porque:
   a) Es un requisito del sistema.
   b) Se necesitan entregar reportes a los niveles superiores de mando.
   c) Mantiene al equipo en condiciones de operación correctas y se pueden prever y resolver problemas
   d) Se puede hacer que funcione más rápido el sistema.

9. Es la bitácora principal de sistema:
   a) messages
   b) secure
   c) pacct
   d) wtmp

10. Registra a los usuarios que se han conectado al sistema:
   a) utmp
   b) lastlog
   c) wtmp
   d) secure

TEMA 12. SEGURIDAD

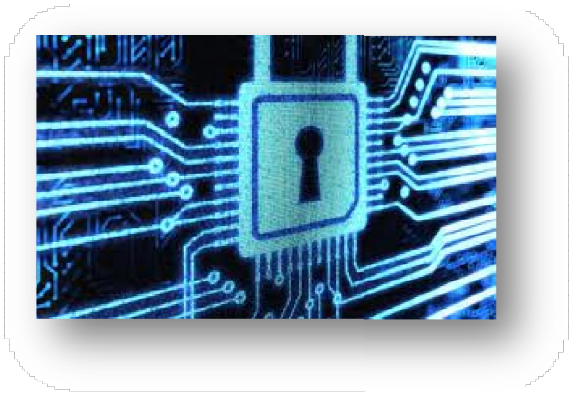
Objetivo particular

El alumno implantará mecanismos de seguridad básicos en un sistema operativo Unix, y reconocerá la relevancia de ésta para el buen desempeño del equipo.

Temario detallado (4 horas)

12.1. Importancia de la seguridad
12.2. Políticas de seguridad
12.3. Herramientas básicas de seguridad

Introducción



La seguridad en los sistemas se ha convertido en una disciplina dirigidaaproteger la integridad y la privacidad de la información almacenada en un sistema informático. El arribo del intercambio electrónico a través de Internet ha generado una exposición latente al riesgo, en donde las empresas se enfrentan diariamente adiversas amenazas, ya sea en los servicios de la red, por saturación del ancho de banda o alteracióndel funcionamiento del servidor, entre otros.

En la actualidad, el administrador de sistemas es el encargado de establecerlas políticas y reglas técnicas destinadas a prevenir, proteger y resguardar la información y recursos contenidos en el sistema.

12.1. Importancia de la seguridad

En términos generales, podemos definirla **seguridad** como el proceso de mantener cualquier objeto exento de todo daño, peligro y riesgo.

En nuestros días, las amenazas a los sistemas operativos, en nuestro caso Unix, han ido en aumento,lo que ha promovido que el tema de la seguridad en sistemas ocupe un lugar importante en todas las organizaciones (y también de manera particular), ya que se persigue que los servidores que contienen toda la información crítica, no sean objeto de daños o alteraciones por circunstancias o factores externos. Así pues, la finalidad de la seguridad espropiciar que los datos resguardados presenten:

- Disponibilidad
- Integridad
- Confidencialidad y
- Cumplimiento de las leyes, regulaciones y estándares aplicables.

Estos objetivos no son suficientes ante los ataques, por ello a menudo están sujetos a otros esquemas con el objetivo de salvaguardar el funcionamiento de los sistemas.

Las fallas de seguridad en los sistemas pueden llegar a ser costosas para las organizaciones, y pueden ir desde la intrusión sin autorización al sistema hasta la pérdida del mismo, por ello en diferentes organizaciones existen las figuras de **Administrador de sistemas** y **Administrador de seguridad**. La función de éste último consiste en proteger al sistema de las amenazas alas que se encuentra expuesto; su trabajo deberá ser coordinado con el administrador de sistemas y los privilegios que tenga en el sistema dependerán de cada organización y/o

administrador. Pueden coexistir estas dos figuras en una misma instancia, pero el administrador de seguridad tendrá ciertas restricciones para hacer modificaciones al sistema, ya que no conoce a fondo el funcionamiento del sistema y la interacción que tienen los usuarios con éste, así se podrá evitar que éste pueda realizar algún cambio que no sea adecuado sin el conocimiento previo del responsable del sistema.

Debido a las características de diseño del sistema operativo Unix, unsolo cambio en un archivo de configuración o programa en el sistema puede llegar a comprometer la seguridad total del sistema operativo en su conjunto. Por ello el monitoreo y la revisión constante de archivoses necesario para mantener un sistema en funcionamiento de forma segura;la pérdida de información o los daños permanentes que pueda sufrir el sistema son los principales riesgos que hay que prever.

Entre las amenazas que se pueden presentar, están:

Amenazas físicas

- **Robo de la unidad de almacenamiento o equipo de cómputo** provocado por personas que tienen un interés en esa información.
- **Vandalismo**. Ladrones o personas que no les interesan la información.
- **Catástrofes naturales**. Cuando los fenómenos naturales, como la lluvia, terremotos, huracanes o el viento, superan un límite de normalidad.

Amenazas lógicas

- **Ingresos al sistema utilizando el protocolo de red**. Ocasionadas por personas que realizan modificaciones sin autorizacion y que tiene el interés en el equipo o la información.

- **Vandalismo electrónico**. Este se caracteriza por alterar la información del sistema, pero no tienen interés en dicho sistema. Lo realizan solo por molestar.

12.2. Políticas de seguridad

Se debe entender y evaluar los riesgos de la seguridad para desarrollar políticas que establezcan con claridad las normas y procedimientos que se deben seguir.La política y la conciencia de seguridad deben ser impulsadas desde los niveles más altos, hacia toda la organización, alcanzando a los usuarios para generar una conciencia amplia de seguridad. La alta gerencia en la organización debe considerar la seguridad como un elemento importante, y cumplir con todas las reglas y regulaciones, al igual que los demás.

El administrador de sistemas debe proveer seguridad al sistema, ésta será variada, puesto que se encuentra sujeta al servicio o servicios que prestan el servidor y la organización que lo alberga ya que es diferente la seguridad de un sistema con servicio de web a uno con servicio de aplicaciones. Pero existen aspectos básicos de seguridad informática que un buen administrador debe buscar, como:

- Protección de la información de ser leída o copiada por cualquier persona que no ha sido explícitamente autorizado por el dueño de esa información.

CONFIDENCIALIDAD

- Protección de la información (incluidos los programas) a que sean eliminados o alterados de ninguna manera sin el permiso del propietario de dicha información.

INTEGRIDAD DE DATOS

- Garantizar la disponibilidad de servicios y recursos para el correcto funcionamiento del sistema.

DISPONIBILIDAD

- Asegurarse de que el sistema se comporte según lo esperado. Si el software o el hardware de repente empieza a comportarse de forma diferente, especialmente después de una actualización o corrección de un error, un desastre puede ocurrir.

CONSISTENCIA

- Regular el acceso al sistema. Evitar que personas desconocidas, no autorizadas y cuentas de sistema ingresen.

CONTROL

- Etapas que requieren tiempo y gastos considerables para la reconstrucción y reinstalación del sistema ocasionandos por eventos no controlables.

RECUPERACIÓN

- Se realiza cuando usuarios no autorizados cometen actos dolosos. En estos casos, es necesario determinar lo que se hizo, por quién y las consecuencias.

AUDITORÍA

Independientemente de que todos estos aspectos de seguridad son importantes, cada organización definirá los que son funcionales para ella. Esto se debe a que las diferentes organizaciones manejan distintas prioridades en materia de seguridad, con base en sus políticas. Por ejemplo: en un banco las preocupaciones más importantes de seguridad son la integridad y la auditoría, mientras que la confidencialidad y la disponibilidad son las siguientes en la escala. En cambio, en una universidad, la integridad y la disponibilidad pueden ser los elementosprimordiales.

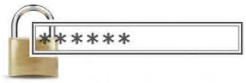
Por ello la formulación de políticas requieren una cuidadosa reflexión, planificación y toma de decisiones basadas en políticas acertadas y en el análisis de riesgos. Por otra parte, es preciso que el administrador comprenda a fondo el funcionamiento del ambiente de trabajo, el sistema y los usuarios,para proveer de un nivel de confianza a los consumidores del sistema.

Existe cierta normatividad en cuestiones de seguridad en cómputo, la revisión y utilización de éstas queda a libre decisión de cada organización. De entre las más conocidas se encuentran:

	▪ La **Organización Internacional para la Estandarización (ISO)**, en mayo del año 1999, publicó la norma ISO 17799 como un conjunto integral de controles que abarcan las mejores prácticas para la administración de seguridad de información.
	▪El **Instituto de Gobierno de TI** provee buenas prácticas para los procesos de TI, definidos en cuatro dominios, que incluyen aproximadamente 220 controles clasificados bajo 34 objetivos de alto nivel, a través de su publicación *Control Objetives for Information and Relates Techology (COBIT)*.
	▪ El **Departamento de Defensa de los Estados Unidos** realizó un documento conocido como el **Libro Naranja (Orange Book)**, en donde estable los requerimientos que debe cumplir un sistema para poder ser calificado como confiable.

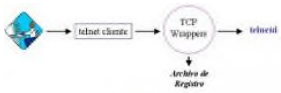
12.3. Herramientas básicas de seguridad

Unix ofrece tres formas básicas de prevención a problemas de seguridad:

Contraseñas 	Se han diseñado para evitar que usuarios no autorizados puedan obtener acceso a cualquier sistema, incluso a través de canales autorizados.
Seguridad de red 	Una variedad de mecanismos de seguridad de red diseñados para **prevenir conexiones no autorizadas**.
Permisos de archivos 	Los **permisos a los archivos** están diseñados para permitir que sólo los usuarios designados tengan el acceso a los distintos comandos, archivos, programas y recursos del sistema.

Para contrarrestar los problemas de seguridad, existen varios programas con un fin específico, a continuación se enlistan en la tabla:

Herramienta	Función
SECURE SHELL 	El objetivo fue sustituir al comando `telnet, ftp` y “`comandos r`”, por ser inseguros, ya que la contraseña se transmite sin cifrar. Secure Shell cifra todo el tráfico (incluyendo contraseñas) para evitar el espionaje, robo de conexiones y otros ataques.

NCARP	Monitorea la integridad y confiabilidad en los archivos, permisos de sistema y busca vulnerabilidades. Puede interferir con los programas `Cracklib` y `Tripwire`.
TCP-WRAPPER 	Monitorea las conexiones que realiza a nuestro sistema. Cuando se presenta una conexión, la analiza, verifica archivos de configuración y registros de bitácora y, cuando determina que sí es una conexión válida, ejecuta el servicio.
TIGER SCRIPTS 	Se utiliza para realizar auditorías de seguridad y detección de intrusos. Se considera la siguiente generación de `COPS`.
CRACKLIB 	Determina las contraseñas que sean fáciles de adivinar.
PGP 	Es un sistema de dominio público que utiliza una combinación de sistemas de llave pública y privada para proporcionar privacidad y autentificación de mensajes.
TRIPWIRE 	Sistema que utiliza algoritmos tipo *Message Digest* para proporcionar verificación de integridad de ciertos archivos.

S/KEY 	Sistema que utiliza el algoritmo de *Message Digest* para evitar la transferencia de contraseñas por medio de la red.
NMAP 	Realiza una exploración de la red. Determina qué hosts están disponibles en la red, qué servicios (nombre del programa y la versión), y sistemas operativos (versiones) se están ejecutando.
KERBEROS 	Sistema de autentificación de redes inseguras que se basa en el modelo de distribución de llaves.

Otros Mecanismos

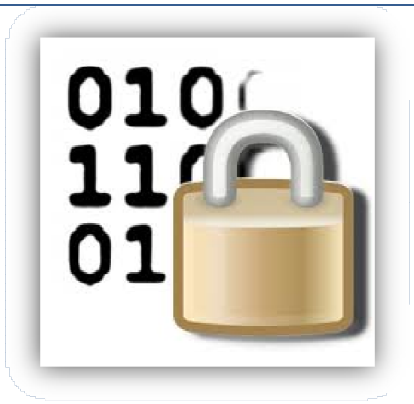
Firewalls

Es un sistema que aísla una red del exterior, controlando muy estrictamente el paso de usuarios e información hacia y desde la red. Se compone de dos partes:

1.*Compuerta (gate),* que se encarga de pasar la información entre las dos redes (externa e interna), y

2. *Ruteador (choke), s*olamente deja pasar paquetes de información que vengan de la compuerta o que estén destinados a ella.

Contraseñas



El primer control de seguridad del sistema es la autentificación de usuarios. Es importante que las contraseñas sean elegidas con las siguientes características:

1. Mínimo de 8 caracteres de longitud.
2. Se tecleen de forma rápida.
3. Posean dígitos y/o signos de puntación.
4. Fáciles de recordar.
5. Letras minúsculas y mayúsculas.

Evitar emplear:

6. Palabras que estén en el diccionario.
7. Palabras que solo las escribimos al revés.
8. Secuencia de caracteres o repetición de los mismos (1223, 5555. abcde)
9. Información personal (fecha de nacimiento, número de casa)

La mayoría de intromisiones se debe a contraseñas débiles, es decir se pueden adivinar fácilmente.

Aunque la mayor parte de la responsabilidad acerca de las contraseñas recae sobre los usuarios, el administrador puede tomar ciertas medidas para limitar la vulnerabilidad, como son:

→ Asignación de contraseñas.

- El administrador asignará la contraseña y no le permitirá que los usuarios la modifiquen.

→ Utilizar programas que adivinen la contraseña.

→ Expiración de contraseñas.

- Consiste en que el sistema automáticamente le pida a un usuario que cambie su contraseña después de cierto tiempo.

→ Utilizar un algoritmo de cifrado diferente.

→ Usar nombres de usuarios que no sean obvios.

→ Educar a los usuarios.

Copias de seguridad

La realización de copias de seguridad de la información que se encuentra en el equipo contrarresta algunos problemas de seguridad del sistema, debido a que permite restaurar el sistema a su estado anterior o volver a crearlo en un nuevo hardware si el equipo se dañó. También ofrecen protección contra la pérdida de datos y deterioros del sistema de archivos.

Sin embargo, si alguien roba los datos de su sistema, éste continuará trabajando de forma habitual ya que no sufrió alteración o destrucción de algún componente, pero ya no se podrá confiar en el sistema, en este caso que las copias de seguridad son irrelevantes.

Bibliografía básica del tema 12

Frisch, A. (2002).*Essential System Administration*. (3rd ed.) O'Reilly Media : Sebastopol, CA.

Garfinkel, E. et al. (2003). *Practical Unix and Internet*. (3rd ed.) O'Reilly Media: Sebastopol, CA.

Bibliografía complementaria

Tanenbaum, A. (2003) *Sistemas Operativos Modernos*. (2ª ed.). México:Pearson Education.

Sitios de Internet

IT GovernanceInstitute. (2007). "COBIT 4.1" IT Governance Institute, EUA. Disponible en línea para su descarga:http://www.isaca.org/Knowledge-Center/COBIT/Pages/Overview.aspx, consultado el 10/01/12.

Diccionario Usuario Casero. UNAM-CERT, Dirección General de Cómputo y de Tecnologías de la Información y Comunicación. Disponible en línea: http://www.seguridad.unam.mx/usuario-casero/diccionario, consultado el 10/01/12.

Activiades de aprendizaje

A.12.1. Elabora un mapa conceptual en el que especifiques los elementos que el administrador de sistemas debe tomar en cuenta para crear políticas de seguridad adecuadas para la organización y el sistema del que es responsable.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. Define con tus propias palabras, el términode seguridad informática.
2. Explica cuáles son las amenazas de seguridad en los sistemas Unix más comunes.
3. ¿Cuál es el beneficio de contar con copias de seguridad en el caso de que la unidad de almacenamiento se haya dañado?
4. ¿Cuál es el objetivo de implementar medidas de seguridad en los sistemas?
5. ¿Cuáles son los elementos básicos, que debe tomar en cuenta un administrador de sistemas, para prevenir los problemas de seguridad en los equipos y sistemas que maneja?
6. Explica el funcionamiento del *firewall*.
7. ¿Qué medidas de seguridad puede implementar el administrador de sistema para aminorar las vulnerabilidades en el sistema?
8. ¿Qué herramientas de seguridad se usan para monitorear la red y cuáles son sus ventajas?

Examen de autoevaluación

( ) 1. Aspectos de seguridad que hace referencia a que los servicios y recursos no se degraden y se encuentren accesibles en el momento en que se requieran.

( ) 2. La esencia de este elemento es regular el acceso al sistema.

( ) 3. Sistema de autentificación de redes inseguras basadas en un modelo de distribución de llaves.

( ) 4. Los componentes del firewall son:

( ) 5. Aspectos de seguridad que tiene la función de proteger toda la información del sistema de posibles daños o alteraciones:

( ) 6. Uno de los objetivos principales de implementar medidas de seguridad en un sistema, es propiciar que los datos tengan:

( ) 7. Implementando esta medida de seguridad, es posible restaurar el sistema a su estado anterior o recuperar información que se haya perdido.

( ) 8. Su principal función en un sistema es asegurar la autentificación de los usuarios que quieran acceder a la información o a realizar modificaciones en el sistema o equipo.

A Consistencia
B Copia de seguridad
C Control
D Confidencialidad
E Disponibilidad
F El administrador
G Ruteador y gate
H Contraseña
I Políticas de seguridad
J PGP
K Kerberos
L Tripware y S/Key

( ) 9. Su formulación requiere de una profunda reflexión, planificación y análisis, y desde los altos niveles de la organización hasta los menores, deben acatar estos lineamientos. ( ) 10. Sistema de dominio público que usa una llave pública y una privada para proporcionar privacidad y autentificación de mensajes. ( ) 11. El primer control de seguridad en los sistemas Unix es: ( ) 12. Utilizan el algoritmo *Message Digest* para proporcionar integridad en archivos y evitar la transferencia de contraseñas a través de la red.	

TEMA 13. INSTALACIÓN DE SOFTWARE

Objetivo particular

El alumno identificará los mecanismos para agregar o remover programas de un sistema operativo Unix.

Temario detallado (4 horas)

13.1. Instalación de sistema
13.2. Instalación de aplicaciones

Introducción

Es esencial que el sistema operativo Unix y los programas que lo componen se encuentren disponibles cuando sean requeridos por los usuarios. Por ello, el administrador de sistemas debe dedicarle tiempo a esta actividad.

La instalación de los sistemas operativos y de programas es un proceso mediante el cual son trasladados los programas a un sistema de cómputo y en algunos casos deben ser configurados para ser utilizados. Estos programas tienen un periodo de caducidad, en el cual funcionan acordecon los principios para los que fueron programados,pero cuando el rendimiento de este software disminuye, se requiere de su inmediata actualización, o bien, su completa sustitución. Con base en ello, el administrador debe estar al pendiente del desempeño del sistema operativo y de las aplicaciones que lo conforman, para evitar atrasos en las tareas o que surjan problemas de funcionamiento o seguridad en el mismo sistema.

13.1. Instalación de sistema

Es fundamental comprender el proceso de instalación del sistema operativo Unix. Cada versión de Unix cuenta con un método de instalación particular, donde se especifican ciertos valores como son: *particiones, nombre del equipo, datos de red, programas que se instalarán.* Cada proceso de instalación del sistema operativo cuenta con una interfaz única, ésta puede presentarse en forma gráfica o en modo texto.

Consideraciones de instalación

Para instalar cualquier sistema operativo Unix,se deben tomar en cuenta los requerimientos de software, así como el soporte de hardware necesario para realizarlo. Por ello,seidentificarán los dispositivos que posee el equipo a instalar, como son: tarjeta *madre, características del procesador, modelo y tamaño de memoria, tipo y tamaño de disco(s)/particiones, características del monitor y tarjeta gráfica, ratón, controladores de DVD/CD y tarjetas de red.*

Una consideración importante a revisar del hardware es que éste funcione correctamente con la versión del sistema operativo Unix que se utilizará. Recordemos que el kernel funciona como un asignador de recursos para cualquier proceso de usuario que necesite hacer uso de los recursos del sistema.

Preparación de la instalación

Algunas cuestiones previas al proceso de instalación son las siguientes:

a) Medios en red y medios locales

Existen dos maneras de instalación principalmente, con medios locales y medios en red.

- Los medios locales son: DVD*/CD-ROM, Disco duro, Dispositivos USB.*
- Los medios en red son: *NFS, FTP, HTTP*.

b) Visión general de la instalación

Actualmente los programas de instalación permiten realizar la instalación de una manera muy sencilla y flexible, aunque no es una regla general, la ejecución de los programas se puede presentar en modo texto o gráfico.

c) Conceptos básicos requeridos

Los conceptos básicos importantes y necesarios para la instalación son:

- Particiones de disco
  - Identificar dónde comienza y termina cada partición.
  - Verificar si la partición es "activa".
  - Identificar el tipo de la(s) partición(es).
  - Identificar partición(es) extendida.

- Dispositivos

Reconocer los elementos que conforman al sistema de cómputo en el ambiente operativo que se utilizará.

- Swap

El área de intercambio es un espacio en disco reservado para ser usado como memoria virtual. Este espacio generalmente es una partición, el tamaño de esta partición dependerá de factores como:

- Cantidad de memoria (RAM)

- Cantidad de usuarios
- Servicios que preste el sistema
- Carga del sistema

- Cuenta root

La cuenta de root es identificada por el sistema como “superusuario”,es decir, posee todos los privilegios de “Administrador del Sistema”.

- Ambiente gráfico (X Window)

Interfaz de usuario de modo gráfico, que permite correr aplicaciones con elementos gráficos, como botones, menús, barras de desplazamiento, etc. Sobre esta interfaz corren los "manejadores de ventanas" y los "ambientes de escritorio".

Proceso de instalación

Como se ha mencionado, cada versión de sistema operativo Unix cuenta con un instalador creado por el fabricante del sistema operativo. Para generalizar el proceso se presenta el siguiente esquema:


VERIFICACIÓN DE LUGAR FÍSICO
• Condiciones electricas.
• Condiciones ambientales.
• Seguridad física.
COMENZAMOS LA INSTALACIÓN
• Utilizando el medio elegido y con los datos anteriores.
REINICIAR EL EQUIPO
• Si la instalación fue exitosa , el equipo se iniciará con el sistema operativo instalado.
OBJETIVO DEL SERVIDOR
• Determinar servicios, programas y usuarios.
DATOS DE CONFIGURACIÓN
• Nombre del equipo.
• Datos de red:
• dirección
• máscara
• ruteador
• servidor de nombres
• etc.
• Esquema de particiones y tipos de sistemas de archivos.
• Sistema de autentificación y contraseña(s).
• Usuarios y máquinas que van a tener acceso a los distintos servicios.
PERSONALIZAR EL SISTEMA
• Detener servicios no necesarios o inseguros.
• Configurar el sistema.
• Establecer mecanismos de seguridad:
• Firewall.
• Herramientas.
• Instalar programas requeridos.
• Agregar usuarios.
ELECCIÓN DE PROGRAMAS
• Programas seguros.
• Programas gratuitos o con licencia.
REVISIÓN DE HARDWARE Y DOCUMENTACIÓN
• El hardware no presente fallas.
• El sistema soporte los dipositivos.
MÁQUINA EN PRODUCCÓN
• Monitorear que el equipo realice la operacion para la cual fue creado.
• Tareas de administración.

13.2. Instalación de aplicaciones

Durante la operación de un servidor se necesitará incorporar nuevos programas, actualizar los existentes o remover los que han quedado en desuso. Para realizar esto, debemos distinguir qué tipo de programa se va instalar, si éste es compatible con el sistema o no, es decir, qué organización o persona ha creado la aplicación, y si la configuración del sistema actual cumple con los requisitos de instalación.

La distribución de aplicaciones se presenta en dos formas, como se muestra en la siguiente tabla:

TIPO DE APLICACIÓN	DESCRIPCIÓN	VENTAJA	DESVENTAJA
Binarias	También denominado paquete, es una colección de archivos y directorios en un formato definido como binario. Cada versión de Unix proporciona un conjunto de utilidades para instalar, quitar y verificar un paquete.	Más fáciles de instalar.	No tenemos un control real de cómo fueron compiladas. Proporcionan opciones por omisión.

Código fuente	Es el conjunto de archivos que almacenan los programas que conforman una aplicación, para que ésta pueda ser utilizada, deberá ser compilada para generar los programas ejecutables.	Podemos configurar de acuerdo con el sistema. Tenemos control sobre rutas y opciones.	Más difíciles de instalar.

Si el programa que se va instalar está dispuesto en forma binaria, se utilizará un manejador de paquetes, una herramienta con la funcionalidad de automatizar el proceso de instalación, actualización, configuración y eliminación de paquetes de software o programas. Cada sistema operativo basado en Unix posee su propio sistema de manejo de paquetes. En la siguiente tabla se proporcionan algunas utilerías para el manejo de éstos.

HERRAMIENTA	DESCRIPCIÓN	SISTEMA OPERATIVO
PKG	`pkgadd –a package_file` Instala un programa, donde package_file es programa a instalar.	Solaris
	`pkgrm package_file` Elimina el programa especificado.	
installp	`installp –ac package_file` Agrega y/o actualiza el software especificado.	AIX
	`installp –u package_file` Borra el programa indicado.	

RPM	`rpm -i package_file` Instala un programa, donde package_file es programa a instalar.	Linux Red Hat Enterprise Fedora,SUSE/openSUSE, Mandriva, CentOS, etc.
	`rpm -e package_file` Elimina el programa especificado.	
Yum	`yum install paquete` Instala la última versión del paquete indicado.	Linux Red Hat Enterprise Fedora,SUSE/openSUSE, CentOS, etc.
	`yum remove programa` Remueve el paquete indicado. **Nota**: *Es un administrador de paquetes que automáticamente determina y resuelve las dependencias a través de la red.*	

Tardist	`inst> keep all`
	`inst> i package_file`
	`inst> go`
	Es una interfaz, con `i` seguida del nombre de programa se instala.
		Irix

	`inst>r package_name`
	`inst> go`
	Borra del sistema el programa indicado.	

Apt	`apt-get install package_file` Instala lo(s) programa(s) señalados.	Linux Debian, Mint, Ubuntu, Kubuntu, etc.
	`apt-get remove package_file` Suprime el software indicado. **Nota**: *Es un administrador de paquetes que automáticamente determina y resuelve las dependencias.*	

Normalmente, cuando se instala algún programa desde el código fuente, los archivos se presentan de la siguiente forma:

Archivos empaquetados en formato	Forma de desempaquetar
Tar (.tar)	`tar xvf archivo.tar`
Cpio (.cpio)	`cpio -i < archivo.cpio`

Archivos comprimidos en formato	Forma de descomprimir
gzip (.gz, .tgz)	`gunzip archivo.gz`
compress (.Z)	`uncompress archivo.Z`
pack (.z)	`unpack archivo.z`
zip (.zip)	`unzip archivo.zip`
bzip(.bz2)	`bunzip archivo.bz2`

También existen archivos y comandos para instalación de aplicaciones.

ARCHIVO	**DESCRIPCIÓN**
README	Una vez que se ha desempaquetado el software, debemos leer el archivo `readme`. En él encontraremos información general y, en algunos casos, el procedimiento para la instalación del software.
INSTALL	Provee información relacionada con el proceso de compilación e instalación del software, así como requerimientos y configuración.
MAKEFILE	Archivo de texto que contiene reglas que indican a *make* qué construir y cómo construirlo. Cada regla contiene: Objetivo (target). Lista de dependencias. Comandos.

PROGRAMA	DESCRIPCIÓN
Configure	Script que sirve para verificar automáticamente la configuración del equipo donde se llevará a cabo la compilación de un software, permite crear de manera automática un makefile, en algunos casos ejecuta el comando make y permite realizar la instalación del software.
Make	Herramienta que permite ejecutar los comandos requeridos para compilar e instalar un software. Construye una base de datos de información sobre dependencias entre archivos y bibliotecas, permitiendo verificar automáticamente la disponibilidad de archivos necesarios para la compilación.
make install	Comando que permite copiar los archivos necesarios para el funcionamiento del software en las rutas adecuadas, establece sus permisos y, si resulta posible, sus propietarios y grupos. Puede crear directorios destino si éstos no existen.

Consideraciones al instalar aplicaciones

- Obtener los programas de software de los sitios oficiales y/o autorizados.
- Cuando sea posible, adquirir los archivos fuente (src) y compilar las aplicaciones, es decir, no utilizar distribuciones binarias.
- No compilar como superusuario, utilizar una cuenta no privilegiada.
- Cuando sea posible, no utilizar la cuenta root para la instalación de aplicaciones.

Bibliografía básica del tema 13

Nemeth, Evi; Snyder, Garth; Hein, Trent R. (2007). *Linux administration handbook.*(2nd ed.)Stoughton, Massachusetts: Pearson Education. [Vista previa]

Solaris System Engineers (2011).*Oracle® Solaris 10 8/11 Installation Guide:Basic Installations.* EUA:Oracle Corporation andits affiliates.Disponible en línea: http://docs.oracle.com/cd/E23823_01/html/E23799/eyprf.html

Bibliografía complementaria

IBM International (2001) *IBM Certification Study Guide -pSeries AIX System Administration.* EUA:IBM Corporation.

Solaris System Engineers (2009) *Solaris10 System Administration Essentials.* EUA: Prentice Hall.

Sitios de Intenert

Dasguspta T. et al (2002) “IBM Certification Study Guide - AIX 5L Installation and System Recover”. IBM Redbooks. IBM Corporation, disponible en línea: http://www.redbooks.ibm.com/redbooks/SG246183.html, consultado el 10/01/12.

Jorba Esteve, J. (2010). “Administración Local. Administración de sistemas GNU/Linux”. Universidad Abierta de Cataluña / *OpenCourseWare.* Disponible en línea:

http://materials.cv.uoc.edu/continguts/PID_00157329/web/main/materias/PID_00157328-5.pdf, consultado el 10/01/12.

Stallman, Richard. (1997). “El derecho a leer”, disponible en línea: http://www.gnu.org/philosophy/right-to-read.es.html, consultado el 10/01/12.

Actividades de aprendizaje

A.13.1. Elabora un cuadro sinóptico donde establezcas los elementos principales del procedimiento de instalación de un sistema, tomando como baselos siguientes datos:

- *Sistema operativo*: Linux “CentOS”
- *Lugar*: aula del curso,
- *Programas y Servicios*: web, bases de datos (msql), programas básicos del sistema,
- *Nombre del servidor*: practica.fca.unam.mx,
- *Dirección numérica*: 192.168.33.55,
- *Servidor de nombres*: dns.fca.unam.mx, usuarios: alumno1, curso1, profesor.

A.13.2. Elabora un mapa mental donde especifiques las consideraciones que el administrador de sistemas debe tomar en cuenta al instalar un programa en un sistema Unix.

Cuestionario de autoevaluación

Contesta el siguiente cuestionario.

1. ¿En qué consiste el proceso de instalación de sistemas operativos y de programas?
2. Explica cuáles son las consideraciones que debe tomar en cuenta el administrador de sistemas, para instalar un sistema Unix.
3. Indica cuáles son las etapas previas a la instalación de un sistema operativo.
4. ¿Cuáles son los factores que influyen en la definición del tamaño del área de intercambio?
5. ¿Qué son las aplicaciones de tipo binario?
6. ¿Por qué debe actualizarse el sistema operativo y los programas?
7. Explica cuáles son las ventajas de utilizar código fuente en la instalación de una aplicación.
8. ¿Cuáles son las consideraciones que se deben tomar en cuenta para instalar aplicaciones?
9. ¿Qué finalidad tiene el archivo README?
10. ¿Qué tipo de conocimientos deberá poseer la persona que llevará a cabo una instalación?

Examen de autoevaluación

Elige la respuesta correcta para las siguientes preguntas.

1. Los requerimientos a tomar en cuenta para el proceso de instalación de un sistema, son los:
   a) de soporte de hardware
   b) de software y hardware
   c) financieros del equipo
   d) de soporte a la licencias del sistema

2. Forma parte del proceso de instalación de un sistema:
   a) Datos de configuración del sistema a instalar
   b) Información de la configuración de los servidores presentes en la red
   c) Datos de los servidores presentes en la red
   d) Estudio de factibilidad técnica de los administradores

3. Es el conjunto de archivos que almacenan los programas que conforman una aplicación:
   a) Sistema operativo
   b) Herramientas de instalación
   c) Programas binarios
   d) Código fuente

4. Son los medios locales que permiten instalar un sistema operativo o una aplicación:
   a) DVD/CD-ROM, HTTP, Dispositivos USB
   b) NFS, DVD/CD-ROM, Disco duro, Dispositivos USB
   c) DVD/CD-ROM, Disco duro, Dispositivos USB
   d) DVD/CD-ROM, FTP, Disco duro, Dispositivos

5. Una desventaja de utilizar un programa tipo binario para instalar una aplicación es que:
   a) Se tiene control real sobre las opciones de compilación
   b) Son los difíciles de instalar
   c) Las opciones de compilación son sencillas
   d) No se tiene un control real de cómo fueron compiladas

6. La instrucción que permite ejecutar los comandos requeridos para compilar e instalar un software es:
   a) make
   b) configure
   c) readme
   d) código fuente

7. Es el manejador de paquetes del sistema operativo AIX:
   a) make
   b) Apt
   c) Installp
   d) yum

8. El siguiente archivo unix.tar.gz, se encuentra en formato:
   a) comprimido con gzip
   b) agrupado con tar y comprimido con compress
   c) agrupado con cpio y comprimido con gzip
   d) agrupado con tar y comprimido con gzip

9. Al momento de instalar una aplicación, es importante considerar:
   a) obtener los programas de software de algún sitio de internet comercial.
   b) obtener los programas de software de los sitios oficiales y/o autorizados.
   c) emplear solo medios locales para instalar las distribuciones de software con interfaz de modo gráfico.
   d) compilar las aplicaciones con la cuenta de root antes de instalarlas.

10. Una ventaja de instalar una aplicación desde el código fuente es que:
   a) se configuran de acuerdo con el sistema.
   b) no se pueden configurar.
   c) son fáciles de instalar.
   d) las opciones de compilación son sencillas.

Bibliografía básica

Bayuk, Jennifer. (1996).*Security through process management*. En: National Information Systems Security, Conference Proceedings, octubre 22-25. Baltimore. [Versión electrónica], disponible en línea: http://csrc.nist.gov/nissc/1996/papers/NISSC96/paper015/bayuk.pdf

Burguess, M. (2000). *Principles of Network and System Administration.*Chichester, Sussex:John Wiley & Sons.

Flores, Y. (2006). *Introducción a UNIX.*Plan de Becarios en Supercómputo, DGSCA, Universidad Nacional Autónoma de México.

_____ (2006). *Tópicos selectos de Administración de Supercómputo*. Plan de Becarios en Cómputo de Alto Rendimiento, DGSCA, Universidad Nacional Autónoma de México.

Flores, Y.; Caballero, R. (2006). *Técnicas básicas de Administración del Sistema Operativo UNIX*. Plan de Becarios en Supercómputo, DGSCA, Universidad Nacional Autónoma de México

Frisch, A. (2002). *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media.

Frisch, Æleen. (2002) *Essential System Administration*. (3rd ed.) Sebastopol, CA: O'Reilly Media. [Vista previa]

Garfinkel, Simson; Spafford, Gene & Shwartz, Alan. (2003). *Practical Unix and Internet Security*. (3rd ed.) O'Reilly Media: Sebastopol, CA. [Vista previa]

Loukides, Michael Kosta; Loukides, Mike. (1990). *System Performance Tuning*.Sebastopol, CA: O'Reilly Media

Maxwell, S. (2002).*UNIX System Administration. A Beginner's Guide*. EUA: McGraw-Hill. [Vista previa]

Musumeci, Gian-Paolo D.; Loukides, Mike. (2002). *System Performance Tuning*. (2nd ed.) Sebastopol, CA: O'Reilly Media. [vista previa]

Nemeth, Evi; Snyder, Garth; Hein, Trent R. (2007). *Linux administration handbook.* (2nd ed.) Stoughton, Massachusetts: Pearson Education. [Vista previa]

Nemeth, E., Hein, Trent R., Snyder, G. & Whaley, B. (2010). *UNIX and Linux system administration handbook*. (4th ed.) Boston, MA: Prentice Hall. [Vista previa]

Powers S; Peek J; O'Reilly T and Loukides M. (2003). *Unix Power Tools*(3rd ed.)Sebastopol, CA: O'Reilly and Associates.

Sánchez, S. (1999). *UNIX y Linux guía práctica*. México: Alfaomega/Ra-Ma.

Sarwar, S. A., Koretsky, R., Sarwar, S. M. (2002). *El libro de UNIX*. Madrid: Addison Wesley.

Solaris System Engineers (2011).*Oracle® Solaris 10 8/11 Installation Guide:Basic Installations.* EUA:Oracle Corporation andits affiliates.Disponible en línea: http://docs.oracle.com/cd/E23823_01/html/E23799/eyprf.html

Bibliografía complementaria

Alcalde, Eduardo; García, Miguel y Peñuelas,Salvador. (1988). *Informática básica*. México: McGraw-Hill.

Burguess, M. (2000). *Principles of Network and System Administration.*Chichester, Sussex: John Wiley & Sons.

Dijker, B. (Ed.) (1996) *A Guide to Developing Computing Policy Documents*, Short Topics in System Administration #2, The System Administrators Guild, USENIX Association.

HP. (2008). *Guía del administrador de sistemas HP-UX: Tareas de administración rutinarias. HP-UX 11i versión 3*, disponible en línea: http://bizsupport2.austin.hp.com/bc/docs/support/SupportManual/c01976185/c01976185.pdf, consultado el 10/01/12

IBM International (2001) *IBM Certification Study Guide -pSeries AIX System Administration.* EUA: IBM Corporation.

Kernighan, B. y Pike, R. (1987). *El entorno de programación UNIX*. México: Prentice Hall.

Pons, Nicolás. (2009). “Permisos de acceso a usuarios”. *LINUX, Principios básicos del uso del sistema*, Barcelona, Eni, cap. 7, pp. 145-150.

Disponible en línea: http://www.editions-eni.fr/_Download/1f6a0d7b-226d-43fb-a909-ba7354beb8f2/Linux_(Extracto-del-Libro).pdf, consultado el 29/07/11.

Pruett, C., Strickland, K. Vetter, S. (2001). *IBM eServer Certification Study Guide - pSeries AIX System Administration.*Disponible en línea:http://www.redbooks.ibm.com/abstracts/sg246191.html?Open , consultado el 10/01/12.

Sánchez, S. (1999).*UNIX y Linux guía práctica*. México: Alfaomega/Ra-Ma.

Shah, S., Soyinka, W. (2007) *Manual de Administración de LINUX.*(4ª ed.) México: McGraw Hill.

Silberschatz, A., Galvin, P., Gagne, G. (2006) *Fundamentos de Sistemas Operativos*. (7ª ed.). Madrid: McGraw-Hill/Interamericana.

Smith, R. W. (2009). *LPIC-1 Linux Professional Institute Certification. Study Guide.* (2ª ed.) Indianápolis, IN: Wiley Publishing, Inc

Solaris System Engineers (2009) *Solaris10 System Administration Essentials.* EUA: Prentice Hall.

Stallings, W. (1997). *Sistemas operativos*. (2ª ed.) Madrid: Prentice Hall.

Strang, J., Mui, L., O´Reilly, T. (1992). *Termcap & Terminfo*. Sebastopol, CA: O'Reilly & Associates, Inc. [Vista previa]

Tanenbaum, Andrew S. (2003). *Redes de computadoras*. (4ª ed.) México: Pearson Educación. [Vista previa]

_____ (1998).*Sistemas Operativos. Diseño e implementación*. (2ª ed.), México: Prentice Hall.

_____ (2003) *Sistemas Operativos Modernos*. (2ª ed.). México:Pearson Education

Zamboni, D. (1993). *Notas de utilerías de UNIX*. Plan de Becarios en Supercómputo, DGSCA, Universidad Nacional Autónoma de México.

RESPUESTAS A LOS EXÁMENES DE AUTOEVALUACIÓN

ADMINISTRACIÓN EN UNIX

Tema 1	
1.	a
2.	b
3.	a
4.	b
5.	a
6.	c
7.	d
8.	c
9.	d
10.	d

Tema2	
1.	a
2.	b
3.	a
4.	a
5.	d
6.	c
7.	c
8.	b
9.	c

Tema3	
1.	c
2.	a
3.	d
4.	c
5.	a
6.	c
7.	d
8.	c
9.	d
10.	c

Tema4	
1.	b
2.	c
3.	b
4.	b
5.	d
6.	b
7.	b
8.	b
9.	c
10.	c

Tema5	
1.	d
2.	d
3.	c
4.	a
5.	b
6.	d
7.	b
8.	d
9.	c
10.	b

Tema6	
1.	c
2.	c
3.	a
4.	b
5.	a
6.	c
7.	c
8.	d
9.	c
10.	c

Tema7	
1	c
2	a
3	a
4	a
5	a
6	d
7	b
8	a
9	c
10	c

Tema8	
1	d
2	c
3	a
4	b
5	c
6	b
7	d
8	a
9	c
10	b

Tema9	
1	b
2	b
3	d
4	d
5	c
6	b
7	a
8	a
9	b
10	d

Tema10	
1	d
2	c
3	a
4	c
5	c
6	a
7	c
8	b
9	a
10	c

Tema11	
1	a
2	a
3	a
4	d
5	a
6	a
7	a
8	c
9	a
10	a

Tema12	
1	e
2	c
3	k
4	g
5	a
6	d
7	b
8	h
9	i
10	j
11	f

Tema13	
1	b
2	a
3	c
4	c
5	d
6	a
7	c
8	d
9	b
10	a