

WIKIUNIX

MIKIŌNIX

RESUMEN

NOELIA SALES MONTES

Tutorial en formato wiki sobre sistemas operativos Unix con plataforma de prueba

Noelia Sales Montes, Manuel Palomo Duarte

*C/ Tamarindos, 4-6. 11007. Cádiz. Tlfn: 956 07 76 54. Correo: noelia.salesmontes@alum.uca.es
Escuela Superior de Ingeniería. C/ Chile, 1. 11002 - Cádiz*

Extracto:

El objetivo principal del proyecto es elaborar un sistema informático que ayude en el aprendizaje de los sistemas operativos UNIX a nivel de administración. Para ello se ha realizado un sistema con tres partes fundamentales: un sistema wiki que facilite la interconexión y el desarrollo colaborativo de documentación, un sistema GNU/Linux virtual que permita realizar los ejercicios y una serie de programas que funcionen sobre dicho sistema para comprobar si los ejercicios están realizados correctamente. El sistema wiki incluye una separación de conceptos por niveles de dificultad, así como diversas etiquetas que ayudan a estructurar el contenido. El sistema virtual se ha cuidado que ocupe el menor espacio posible, para que fuera posible su ejecución en máquinas con una cantidad limitada de recursos. Además, en dicho sistema actualizan por Internet los programas de comprobación de ejercicios, que están definidos en función de existencia y consulta de ficheros a partir de información en el wiki.

Palabras clave: UNIX, administración, wiki, virtualización, aprendizaje.

Índice

1. Alcance	5
2. Visión general	6
3. Desarrollo	6
3.1. Contenido teórico	7
3.2. Soporte de comprobación	10
4. Conclusiones	11
4.1. El trabajo realizado	11
4.2. El futuro del proyecto	12
5. Publicaciones	12

Objetivos

El objetivo principal del proyecto es elaborar un sistema informático que ayude en el aprendizaje de los sistemas operativos UNIX a nivel de administración. Además se cuidará la facilidad para ampliar y mejorar su contenido.

El tutorial debe cumplir los siguientes requisitos:

1. El contenido teórico principal debe estar estructurado en niveles de complejidad, de forma clara, concisa e interrelacionada.
2. El soporte en el que se publique dicho contenido debe permitir y promover su ampliación y mejora colaborativa, para evitar quede obsoleto.
3. Debe disponer de ejercicios y de un sistema de control de la resolución correcta de éstos cuando sea posible. De modo que el propio usuario pueda verificar si ha comprendido los conceptos teóricos y sabe aplicarlos de forma práctica.
4. El producto final debe estar disponible en Internet para toda persona interesada en el tema.

La estructuración de la temática que debe cubrir el contenido teórico principal es la siguiente:

Nivel de principiante:

- Características
- Clasificación
- Estructura
- Conceptos básicos

Nivel de usuario iniciado:

- Órdenes básicas
- Sistema de ficheros

Nivel de usuario habitual:

- Editor VI
- Órdenes avanzadas
- Instalación de software

Nivel de administración:

- Arranque y parada del sistema
- Administración de usuarios y grupos
- Administración del sistema de ficheros
- Instalación de un servidor FTP con vsftpd
- Introducción a la seguridad en un sistema UNIX

El desarrollo del contenido teórico debe realizarse en un lenguaje sencillo, claro y conciso suficiente para permitir que un estudiante universitario de cualquier Ingeniería Informática pueda seguir sus contenidos sin problema. Aunque este es el usuario principal, también está dirigido a cualquier persona que quiera adentrarse en este área de conocimiento, ya que se parte de un nivel prácticamente cero hasta el nivel de administración.

Por otro lado, el soporte debe permitir una gestión simple y una navegación cómoda por el contenido. Todo lo relacionado con la posible colaboración de otros usuarios debe ser lo más sencilla posible, sin necesidad de herramientas o aplicaciones que supongan un impedimento en lugar de una ayuda.

Contexto: Breve historia de los sistemas UNIX

Los Laboratorios Bell fueron los responsables del desarrollo de UNIX como un proyecto de investigación privado dirigido por un pequeño grupo de personas que empezó en 1969.¹ El objetivo del grupo fue diseñar un sistema operativo que cumpliera los siguientes requisitos:

- Que fuera simple y elegante.
- Que estuviera escrito en un lenguaje de alto nivel.
- Que permitiera reutilizar el código.

UNIX tenía una relativamente pequeña parte de su código escrita en ensamblador (la que se encargaba de gestionar el hardware, más conocida como *kernel*) y el resto del código escrito en C.

A medida que el desarrollo avanzaba se realizaban pequeñas modificaciones al código del *kernel* (dependiendo de la plataforma) y se desarrollaron muchas utilidades en C. A través de esta evolución el *kernel* y el software asociado se extendieron hasta que un sistema operativo completo se desarrolló “sobre” el *kernel*.

Como evolución de UNIX aparecieron multitud de sistemas derivados, que se basaban en las mismas características (puede visualizarse un esquema en la Figura 1).

¹Para conocer más detalles acerca de estos sistemas accede a la [página correspondiente](#) en el wiki y revisa sus referencias.

2. Visión general

Es necesario aclarar que el contenido teórico del proyecto podía incluir muchísimos temas. Por ello se ha decidido cumplir las necesidades de la asignatura Administración de Sistemas Operativos impartida en la titulación de Ingeniería Técnica en Informática de Sistemas en la Universidad de Cádiz mediante entrevistas mantenidas con su responsable y tutor de este Proyecto Fin de Carrera.

En cuanto a la estructura de esta Memoria del Proyecto Fin de Carrera, tras este capítulo donde se presentan los objetivos y la visión en general del proyecto, se expone el desarrollo del calendario mediante un diagrama de Gantt describiendo con detalle las distintas etapas de desarrollo del proyecto.

El capítulo siguiente contiene la descripción general del proyecto, la perspectiva del producto, las funciones que cumple, los requisitos y las características que deben de cumplir los usuarios para un correcto aprovechamiento del producto. Las restricciones generales, dependencias, requisitos y planificaciones para futuras versiones son parte del contenido de esta sección del proyecto. Se incluye una explicación detallada de las herramientas elegidas para la elaboración del proyecto y el porqué de la toma de dichas decisiones.

Seguidamente describimos el desarrollo del proyecto, donde se especifica la metodología de desarrollo que hemos seguido, así como los distintos requisitos del sistema para pasar a su posterior análisis. Una vez realizado el proceso de análisis diseñamos el sistema a partir la información proporcionada para presentar en el apartado de codificación algunos aspectos relevantes de las distintas implementaciones que hemos llevado a cabo en este proyecto. Como parte del desarrollo, también se abordan las pruebas y el mantenimiento. Se ha de tener en cuenta que cada uno de estos apartados se abarca desde dos puntos de vista distintos: por un lado desde la elaboración del contenido teórico y por otro desde la del soporte para pruebas.

Finalmente, se presentan las conclusiones generales obtenidas una vez realizado el proyecto para pasar inmediatamente a los manuales de usuario y de instalación.

Además se presentan las referencias bibliográficas donde se incluyen las fuentes consultadas para la elaboración de este proyecto, un resumen que engloba las generalidades fundamentales del producto y de la metodología de desarrollo del proyecto, una guía de utilización (manual de usuario), una guía de instalación, un compendio del software utilizado y otro de los lenguajes de programación y finalmente la licencia completa del documento.

Por último, se adjunta un volumen con el contenido de referencia del tutorial para complementar esta Memoria del Proyecto Fin de Carrera. Así puede ser consultado para contrastar las afirmaciones que se han justificado en este proyecto.

3. Desarrollo

Para el desarrollo del proyecto hemos seguido dos metodologías, debido a que las dos partes que componen el proyecto difieren entre sí, aunque se complementan una a la otra y no pueden separarse del todo.

3.1. Contenido teórico

Primero se definen los niveles de complejidad como cuatro categorías de menor a mayor nivel de conocimientos:

- **Principiante:**

Se refiere a aquellos usuarios que no poseen conocimientos acerca de los sistemas UNIX.

- **Usuario iniciado:**

Se refiere a los usuarios que hayan utilizado en alguna ocasión estos sistemas, aunque tan sólo desde el punto de vista del usuario básico de escritorio. Necesitan conocer más a fondo la funcionalidad básica y la forma de trabajar de un usuario común.

- **Usuario habitual:**

Este tipo de usuario ya conoce los sistemas UNIX y puede trabajar de forma normal con ellos, aunque necesita de determinados consejos o prácticas para aprovecharlos.

- **Administrador:**

El administrador ya no es un usuario común al que le basta con poder trabajar en su propio equipo. Necesita conocimientos mucho más avanzados para poder controlar todo lo que ocurre en los sistemas que administre. Así podrá prevenir, evitar y solucionar los problemas que puedan surgir.

En cada uno de esos niveles se encuentran encuadrados determinados artículos principales, los cuales se detallan a continuación, clasificados en función de dichos niveles.

Dentro del NIVEL PRINCIPIANTE se encuentran los artículos:

1. **Características:**

Introducción a los sistemas operativos que se encuendran bajo la familia UNIX, especificando sus características.

2. **Clasificación:**

Distinción entre las subfamilias de sistemas operativos que derivan del sistema inicial UNIX.

3. **Estructura:**

Descripción de los componentes básicos de la un sistema operativo de la familia Unix: kernel, shell e interfaz.

4. **Conceptos básicos:**

Definición del vocabulario básico y las palabras que se suelen emplear en el ámbito de los sistemas Unix.

Dentro del NIVEL USUARIO INICIADO se encuentran los artículos:

1. **Órdenes básicas:**

Explicación a fondo de las órdenes más básicas que se deben conocer en un sistema Unix por su utilización común.

2. Sistema de archivos:

Información de los directorios base que conforman el sistema de archivos de los sistemas UNIX.

Dentro del NIVEL USUARIO HABITUAL se encuentran los artículos:

1. VI:

Referencia breve a la utilización del editor VI, que está presente en (casi) cualquier sistema UNIX por defecto.

2. Órdenes avanzadas:

Compendio de las órdenes de un sistema UNIX, con su modo de utilización y un enlace al manual del sistema.

3. Instalación de software:

Guía explicativa acerca de cómo instalar software.

Dentro del NIVEL ADMINISTRADOR se encuentran los artículos:

1. Arranque y parada del sistema:

Explicaciones de los niveles de la forma de comportarse de estos sistemas en su inicio y su apagado, así como los niveles y las formas en que puede hacerse.

2. Usuarios y grupos:

Administración y control de los usuarios y los grupos dentro de un sistema UNIX.

3. Administración del sistema de archivos:

Dedicada a un nivel más complejo de los sistemas de archivos y a la forma de trabajar de un administrador de sistemas UNIX respecto a ellos.

4. Instalación de un servidor FTP con vsftpd:

Proceso de instalación de un servidor FTP

5. Seguridad:

Introducción a algunas herramientas útiles que ayudan a que los sistemas UNIX sean más seguro ante ataques y/o eventualidades.

Al trabajar en esta parte del proyecto fue necesario solventar una serie de problemas, los cuales se detallan a continuación:

Adaptar *MediaWiki* a nuestro proyecto El motor *MediaWiki* es completamente versátil y muy potente, pero cuando empezamos a trabajar con él nos dimos cuenta que su estado base no era suficiente, dado que este es un wiki bastante peculiar.

Entonces tuvimos que ampliar las extensiones que se encontraban en el wiki y desarrollar una serie de plantillas para facilitar el trabajo, ya no a los desarrolladores actuales sino a los que puedan colaborar en el futuro.

Se trabajó especialmente en la navegabilidad del wiki, para facilitar el acceso a los visitantes, dado que el contenido real es bastante extenso y cualquiera puede “perderse” fácilmente.

Contenido libre Al empezar a recabar información nos preguntamos: ¿De dónde saco el contenido? Lógicamente la autora sabía que debería desarrollar ella misma gran parte del temario, pero siempre es conveniente proporcionar referencias que sustenten nuestra documentación.

Investigando un poco no fue difícil encontrar fuentes de conocimiento libre muy bien estructuradas y detalladas.

Las modificaciones extensas y repetitivas A la hora de mantener un wiki se percibe lo monótonas y repetitivas que pueden llegar a ser ciertas tareas. De ahí que fuera muy útil trabajar con *pywikipedia* para desarrollar una serie de *bots* que nos ayuden a realizar dichas tareas.

Su codificación es muy simple, sobre todo comparada con el tiempo que ahorran al no tener que trabajar manualmente.

3.2. Soporte de comprobación

El soporte de comprobación se basa en dos componentes:

- La máquina virtual donde se realizarán los ejercicios que hace las veces de entorno cerrado.
- Los “scripts” de comprobación que verificarán si los ejercicios se han desarrollado correctamente.

Los artículos para los cuales se desarrollarán ejercicios con sus respectivos *scripts* de control son:

- **ÓRDENES BÁSICAS (NIVEL USUARIO INICIADO):**

Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes *ls*, *cp*, *cd*, *mv*, *rm*, entre otras.

Debe quedar claro el trabajo básico en la terminal, sobre todo en lo referente a creación, movimiento, eliminación y modificación de archivos y directorios.

- **SISTEMA DE ARCHIVOS (NIVEL USUARIO INICIADO):**

Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes *chmod*, *chgrp*, *chown*, entre otras.

Debe quedar clara la relación del sistema de archivos con los permisos de todo su contenido. El usuario debe aprender a modificar según le convenga dichos permisos.

- **ÓRDENES AVANZADAS (NIVEL USUARIO HABITUAL):**

Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes *tar*, *find*, *grep*.

Debe quedar clara la utilidad de estas órdenes más complejas. Precisamente su sintaxis más extensa, las hacen mucho más versátiles a la hora de poder sacarles más partido.

- **ARRANQUE Y PARADA DEL SISTEMA (NIVEL ADMINISTRACIÓN):**

Se desea que con este grupo de ejercicios el usuario asimile el comportamiento del sistema al iniciarse y cerrarse.

Debe quedar clara la utilidad de los distintos niveles de arranque y cómo se manejan y modifican sus características.

- **USUARIOS Y GRUPOS (NIVEL ADMINISTRACIÓN):** Se desea que con este grupo de ejercicios el usuario aprenda a manejar las órdenes *useradd*, *userdel*, *usermod*,...
Debe quedar claro cómo se manejan los usuarios y los grupos, sobre todo a la hora de administrar los permisos de cada uno y de controlar su nivel de modificación.

En esta parte del proyecto, nos encontramos con diversas complicaciones antes de conseguir llegar al estado actual de los componentes:

La máquina virtual y Ubuntu A la hora de instalar la máquina virtual, la primera idea fue instalar la distribución *Ubuntu* y comprimir la imagen con la utilidad de exportación de *VirtualBox*. Sin embargo, el tamaño del archivo ya comprimido era demasiado grande como para que fuera viable su descarga.

Además el entorno no era lo suficientemente ligero como para permitir su utilización a casi cualquier usuario con un equipo decente.

Por ello, tuvimos que buscar otras alternativas, entre las cuales encontramos una distribución hermana de *Ubuntu*: *Xubuntu*. Era la mejor elección para no tener que hacer demasiadas modificaciones a los ejercicios.

Las primeras versiones de los *scripts* En un principio se realizaba un *script* específico por cada conjunto de ejercicios. Para la desarrolladora actual no había problema ninguno, puesto que conocía la forma de codificar. Pero este método requería que todo aquel que quiera desarrollar ejercicios sepa programar en un determinado lenguaje de programación.

Esto no es útil si lo que deseamos es que la comunidad se abra a casi cualquier persona: un determinado usuario puede saber cómo corregir un ejercicio de sistemas UNIX, pero no tiene porqué saber programarlo.

De ahí que se tomara otra decisión importante: desarrollar un único *script*, el cual sirva para realizar diferentes tipos de comprobaciones en función del contenido de unos ficheros de comprobación.

Estos ficheros son simples de rellenar, ya que están escritos en modo texto, no contienen caracteres extraños y su función viene explicada en el manual de usuario.

Actualización automática del sistema de comprobación El sistema de comprobación no es estático: se basa en una serie de directorios y archivos que se encuentran en una forja y van cambiando con el tiempo, en función de la ampliación de los ejercicios.

Entonces era necesario desarrollar otro sistema para que la máquina virtual actualizara automáticamente estos componentes sin involucrar al usuario.

La solución fue incluir un *script* en las órdenes del sistema y llamarla al inicio de la sesión del usuario implicado. Así cada vez que el usuario entre en el sistema (siempre y cuando tenga conexión a internet) tendrá el sistema de comprobación actualizado.

4. Conclusiones

4.1. El trabajo realizado

Terminado ya el proyecto *Tutorial en formato wiki sobre Sistemas Operativos UNIX con soporte para pruebas* se han cubierto los siguientes objetivos propuestos:

- Hemos conseguido un sistema editable por un grupo de trabajo de manera sencilla con una interfaz amigable.
- El contenido de dicho sistema está bien estructurado y clasificado en niveles de dificultad.
- Se ha construido un sistema donde realizar ejercicios y donde comprobar su resolución de forma automática.
- El sistema es accesible a cualquiera que desee acceder a él a través de la dirección:

<http://osl.uca.es/wikiunix>

Ha sido un trabajo duro e intenso que será liberado de forma oficial bajo licencia libre una vez presentado el proyecto ante el tribunal para su evaluación. Este hito supondrá la culminación de esta primera etapa del proyecto, a partir de la cual podrá ser leído y trabajado por todo aquel que lo considere interesante. La siguiente etapa será de revisión, mejora y ampliación del contenido y contará con una participación externa abierta (espero que abundante).

4.2. El futuro del proyecto

El producto resultado de este PFC es un producto evolutivo. Por su naturaleza está ideado para que la comunidad logre una evolución propia de su utilización. La posibilidad de que los usuarios modifiquen, completen y agreguen temas es una característica intrínseca al proyecto por lo que las futuras versiones parten del interés de los usuarios.

Sería bastante interesante crear otras máquinas virtuales. En este proyecto se ha utilizado una máquina con la distribución *Xubuntu*, que tiene como entorno de escritorio XCFE. En un futuro podría incluirse otra distribución que utilice KDE por ejemplo, que es más actual y extendido.

En este sentido también sería muy didáctico ampliar la batería de ejercicios y artículos en torno a los entornos gráficos de escritorio, puesto que actualmente los ejercicios se centran en la terminal.

Por otro lado, sería conveniente seguir ampliando el mantenimiento del wiki con nuevos bots. Sobre todo pensando en que la wiki va a ser accesible a cualquiera, habrá que desarrollar algunos bots de control, para que ayuden a los administradores a controlar las nuevas entradas.

5. Publicaciones

- Tecnologías Wiki y Conocimiento Abierto en la Universidad.
Manuel Palomo, Inmaculada Medina, Emilio J. Rodríguez, Noelia Sales.
Actas de la V Conferencia Internacional de Software Libre, páginas 16-19. Cáceres, Diciembre 2009. I.S.B.N.: 978-84-692-8739-2
- Tecnologías wiki en la docencia de Ingeniería Informática
Manuel Palomo Duarte, Emilio J. Rodríguez Posada, Inmaculada Medina Bulo y Noelia Sales Montes
XVI Jornadas de Enseñanza Universitaria de la Informática (JENUi2010)
Enviado a la espera de aceptación.

Referencias

- [1] *Mark G. Sobell*, Manual práctico de Linux.
Ed. Anaya. 2005.
I.S.B.N.: 978-84-415-2350-0
- [2] *Noah Gift, Jeremy M. Jones*, Python para administración de sistemas Unix y Linux.
Ed. Anaya, O'Reilly. 2008.
I.S.B.N.: 978-84-415-2540-5
- [3] *Matthias Kalle Dalheimer, Matt Welsh*, Guía de referencia y aprendizaje Linux
Ed. O'Reilly. 2006.
I.S.B.N.: 84-415-2031-3
- [4] *Eric S. Raymond*, The cathedral and the bazaar
Ed. O'Reilly. 2001.
I.S.B.N.: 978-0-596-00108-7
- [5] *Alex Martelli, Anna Martelli Ravenscroft, David Ascher*, Python Cookbook. Recipes from the Python Community.
Ed. O'Reilly. 2005.
I.S.B.N.: 978-0-596-00797-3