

# Ingeniería en computación

Reporte de Proyecto Terminal

## **Administración de cuentas para cursos en un servidor Linux**

Alumno: David García Isario  
Matrícula: 204306905

Trimestre 09 - Invierno

14 de mayo de 2008

Director de proyecto: Dr. Francisco Javier Zaragoza Martínez  
No. Eco. 20197

# Índice

<u>INTRODUCCIÓN</u>	<u>3</u>
<u>EL PROBLEMA</u>	<u>4</u>
<u>DISEÑO DEL SISTEMA</u>	<u>6</u>
<u>FUNCIONAMIENTO GENERAL</u>	<u>7</u>
<u>MÓDULOS DEL SISTEMA</u>	<u>11</u>
<u>GUÍA DE USUARIO</u>	<u>18</u>
<u>MANUAL Y REQUERIMIENTOS DE INSTALACIÓN</u>	<u>22</u>
<u>SEGURIDAD</u>	<u>30</u>

# Introducción

Este documento tiene la finalidad de ser una guía para comprender de una manera más fácil el funcionamiento del sistema “Administración de cuentas de usuario” y así poder darle un mejor uso y mantenimiento.

Aquí se describe de manera detallada cada componente del sistema y en un futuro se puedan agregar nuevas funciones al sistema o porque no decirlo mejorar las ya existentes.

Sin embargo el enfoque para describe el sistema no es el del “COMO” si no mas bien el del “QUE”, es decir se explica paso a paso que hace el sistema para llegar al resultado deseado pero no se explica por decirlo de alguna manera el código de la aplicación.

Otra finalidad de esta guía es proveer la información necesaria para que cualquier persona pueda instalar el sistema. Sin embargo es lógico que las personas que tengan un conocimiento mínimo o nulo en el sistema operativo Linux serán más propensas a cometer errores o tener dudas sobre la instalación, por eso es recomendable que se estudie con anticipación las intrucciones más importantes del sistema operativo Linux si fuera necesario.

# El problema

La administración de cuentas de usuario es una parte esencial de la administración de sistemas dentro de una organización, particularmente dentro de la UAM la administración de cuentas de usuario es útil para algunos profesores los cuales cuentan con algún servidor en el cual asignan cuentas a sus alumnos para diferentes actividades relacionadas con las UEAs impartidas.

La administración de cuentas de usuario no es algo muy fácil de manejar especialmente cuando el número de usuarios es grande, este detalle puede hacer difícil la administración de las cuentas dentro del servidor y generar algunos problemas:

- Es necesario crear un convenio de nombres para los usuarios, específicamente en la UAM se requieren nombres de usuario que reflejen algún tipo de información para los profesores sobre el alumno y además que no sea muy difícil de aprender, por tanto la matrícula queda descartada ya que sólo es un número. En consecuencia hay que crear un convenio de nombres adecuado a las necesidades requeridas. He aquí algunas convenciones de nombres que otras organizaciones han utilizado:
  - Primer nombre (jorge, carlos, pedro, etc.)
  - Apellido (perez, obregon, ramirez, etc.)
  - Primera inicial, seguida del apellido (jperez, cobregon, pramirez, etc.)
  - Apellido, seguido del código del departamento (perez029, obregon454, ramirez191, etc.)
- Desde el punto de vista del sistema, el nombre de usuario (login) es la respuesta a la pregunta "¿quién es usted?". Como tal, los nombres de usuarios tienen un requerimiento principal — deben ser únicos. En otras palabras, cada usuario debe tener un nombre de usuario que sea diferente a todos los otros usuarios en ese sistema. Este hecho trae como consecuencia la generación de "colisiones". Una colisión ocurre cuando eventualmente existan dos individuos que, de acuerdo a la convención, obtendrían el mismo nombre de usuario.

Por lo tanto si el número de usuarios es grande y en base al convenio de nombres establecido se generan muchas colisiones, esta tarea puede llegar a ser difícil y laboriosa si se realiza de manera manual. Actualmente en la UAM algunos profesores realizan cada trimestre de manera manual la generación y verificación de login, password y la creación de las cuentas de usuario, he aquí la necesidad de automatizar el proceso y la razón de ser del sistema que se plantea en este documento.

- La generación de las contraseñas (passwords) es otro punto muy importante dentro de la administración de las cuentas para el manejo de la seguridad. Sin embargo estas pueden ser cambiadas posteriormente por los usuarios, entonces la pregunta es ¿qué tan robustas deben ser las contraseñas a crear?

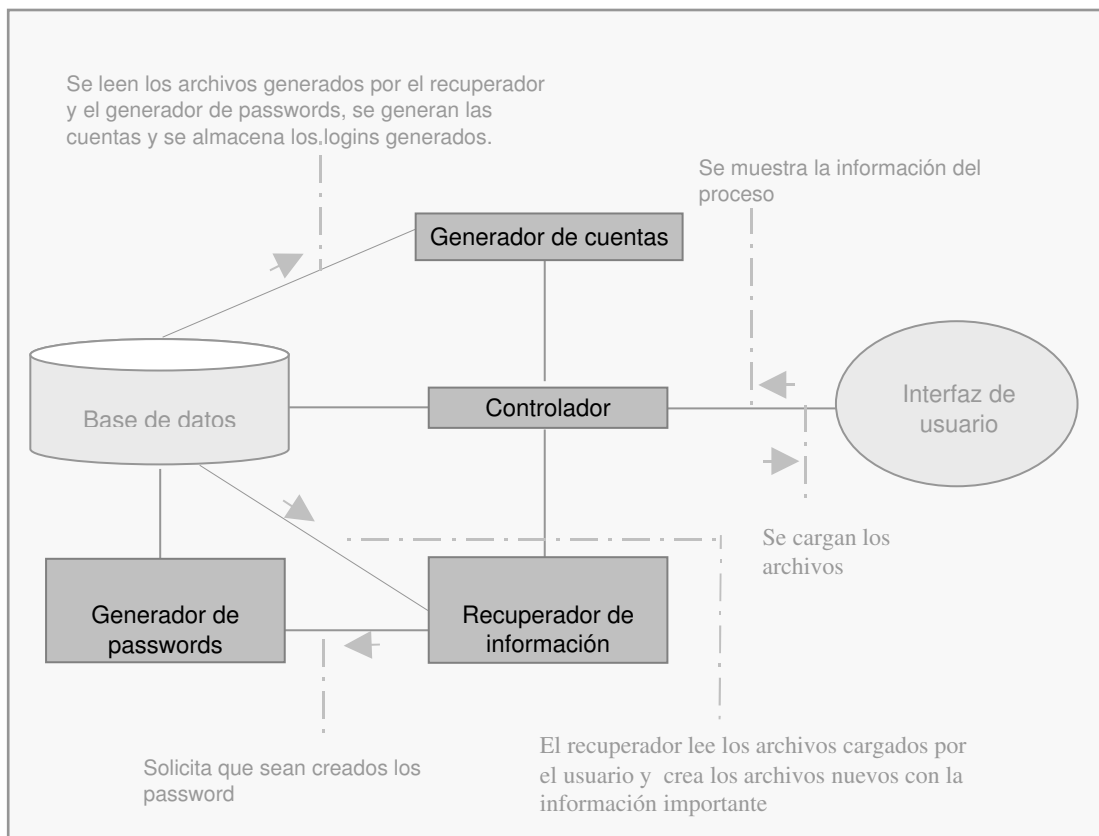
Particularmente en la UAM se requieren contraseñas que no incluyan las letras I, l, O (tengo que aclarar que la primera letra es la i mayúscula y la segunda letra es la L minúscula, este es un claro ejemplo de las confusiones que se generan con estos caracteres) y los números 1 y 0 ya que se ha comprobado que son motivo de confusión.

- Además de los problemas ya mencionados existen otros que tienen que ver con el tiempo de vida de dichas cuentas de usuario y su reutilización, es decir a los profesores les interesa que una cuenta de usuario que se le creó a cierto alumno para alguna materia X en un trimestre Y pueda ser utilizada por el mismo alumno en otras materias que impartirá el profesor en posteriores trimestres, por tanto se necesita saber si el alumno ya se le ha creado una cuenta de usuario, si es así generar un nuevo password para dicha cuenta pero la cuenta es la misma, si no es así crear la cuenta y generar el login y password. También se necesita eliminar las cuentas de usuario que ya no se utilizan por diferentes razones, por ejemplo, cuando un alumno concluye sus estudios o se da de baja.
- La información de cada usuario contenida en su cuenta es muy importante, por lo cual realizar eventualmente el respaldo (backup) de información de las cuentas en el servidor es una tarea no menos importante, que se complica conforme al número de cuentas existentes. Esta tarea se tiene que hacer debido a problemas que puedan surgir en el servidor que ocasionen la pérdida parcial o total de la información, o simplemente cuando el administrador desea eliminar ciertas cuentas se debe realizar el respaldo de la información que está contenida en éstas, por cualquier reclamo del usuario a futuro.

La administración de las cuentas como podemos ver, no es una actividad trivial especialmente porque está en juego mucha información de diferentes personas y por la cantidad de trabajo y tiempo que esto requiere, por tanto lo que se logró con este sistema es automatizar justamente todo el proceso que requiere la realización de dicha actividad desde la creación de login en base a la información contenida en un archivo, el password, la cuenta de usuario, el manejo de colisiones y el respaldo de información de las cuentas, en consecuencia el trabajo y la administración sobre las cuentas será mucho más fácil y sencillo de realizar.

# Diseño del sistema

En el diagrama siguiente se muestran los módulos que forman el sistema de administración de cuentas y algunos detalles de la comunicación entre ellos.



# Funcionamiento general

El proceso general que se sigue en el sistema es el siguiente:

- El profesor (usuario) ingresa a la ubicación del sistema mediante un navegador web, el servidor web solicitará su login y password que identifican que tiene permisos para entrar a dicha dirección. Si el profesor tiene los permisos entonces el servidor web le permite entrar a dicha dirección.
- A continuación se muestra al usuario la pantalla de login del sistema, nuevamente el profesor tendrá que ingresar un login y password para poder entrar directamente al sistema. El sistema encripta el login y password con md5 y verifica que sean correctos [6]. Si los datos son correctos se permite el ingreso al sistema, si no se comunica al profesor que sus datos no son correctos.
- Ya dentro del sistema el profesor carga los archivos en los cuales se encuentra la información de sus alumnos.
- El sistema valida los tipos de archivos que envió el profesor y si son correctos crea una carpeta dentro de la carpeta Transacciones del sistema, donde se guardará toda la información relacionada a esa transacción de creación de cuentas, por ejemplo: los archivos cargados al servidor por el usuario, los logins y passwords generados por el sistema, información de la carga etc. El nombre para esta carpeta será la cadena **CuentasDeUsuario** concatenándole al final el número de transacción que le corresponda.
- Posteriormente crea seis carpetas dentro de la carpeta anterior y un archivo de información de la transacción:
  1. **Loads**. En esta carpeta se guardan los archivos que se cargaron por el usuario al servidor. A los archivos se les dará el nombre de grupo1, grupo2, ... , grupoN.
  2. **Users**. En esta carpeta se guardarán los archivos generados del proceso de recuperación de la información de los alumnos correspondientes a los archivos cargados por el usuario. Estos serán utilizados para el proceso de generación de logins.

3. **Passwords.** En esta carpeta se guardarán los archivos generados del proceso de generación de passwords encriptados para su posterior utilización en el proceso de creación de cuentas.
  4. **PT.** En esta carpeta se guardarán los archivos generados del proceso de generación de passwords pero no están encriptados para su posterior utilización en el proceso impresión de cuentas creadas.
  5. **Logins.** En esta carpeta se guardarán los archivos generados del proceso de generación de logins para su posterior utilización en el proceso de creación de cuentas e impresión de cuentas.
  6. **Grupos.** En esta carpeta se guardarán los archivos generados del proceso de creación de cuentas para su posterior utilización en el proceso de creación de cuentas.
- El sistema comienza el proceso de recuperación de la información que se necesita para la creación de las cuentas utilizando la información contenida en cada archivo cargado por el usuario.
    1. Del archivo que se está procesando se obtiene el nombre de la UEA, grupo, el nombre y matrícula de cada uno de los alumnos.
    2. Se aplica un formato adecuado a la información de cada alumno y se guarda en un archivo de nombre **users<numero de archivo>** dentro de la carpeta Users. El formato de cada línea del archivo es:

```
A.Paterno:A.materno:Nombre:Matrícula
```

- El siguiente paso es crear los passwords para cada alumno dentro del archivo anterior y almacenarlos en un archivo llamado: **passwords<numero de archivo>** que se guardará en la carpeta Passwords. Estos passwords son encriptados mediante *crypt* o *md5* según la distribución de Linux que se ocupe [6]. Paralelamente se guardan en otro archivo los passwords no encriptados para así poder utilizarlos posteriormente en la impresión de información. El nombre de este archivo es **pt<numero de archivo>** y se guarda en la carpeta PT.

*La recuperación de información y la creación de passwords se realizarán para cada uno de los archivos cargados por el profesor.*



- Cuando ya se tiene los dos archivos anteriores (datos generales y passwords) correspondientes a cada archivo cargado se utiliza su contenido para comenzar el proceso de creación de las cuentas.

Antes de continuar con el paso siguiente es importante mencionar que el sistema se divide en dos partes: el proceso de generación de información necesaria para la creación de cuentas y el proceso de creación de cuentas, estos procesos se ejecutan en forma separada es decir, el primer proceso se realiza mediante una interfaz web y el segundo mediante un script en lenguaje shell de Linux, esto se debe a motivos de seguridad y se justifica por lo siguiente: si alguna persona con malas intenciones logra entrar a la primer parte del sistema mediante la interfaz web sólo podrá procesar la información para la creación de cuentas pero en ningún momento podrá crear las cuentas con dicha información, para que alguna persona pueda crear las cuentas tiene que ejecutar la otra parte del sistema dentro del servidor en una Terminal del sistema operativo y además tener permisos de súper usuario.

Con todo lo anterior podemos obtener un grado de seguridad mayor y evitarnos gran cantidad de problemas.

Como primer paso el sistema busca al alumno en base a su matrícula en el archivo **passwd** que es donde el Sistema Operativo (SO) guarda la información de las cuentas de usuario.

- Si el sistema encuentra al alumno sólo modifica su password y no le crea una cuenta nueva, además obtiene su login y lo guarda en un vector para después poder imprimirlo en el reporte final.
- Si el sistema no encuentra al alumno entonces le crea su login basándose en su nombre, verifica que éste no exista dentro del archivo passwd del SO y si no existe crea la cuenta. El proceso de creación y verificación del login se realizará tantas veces hasta que el login sea único.

*Este proceso se realiza para cada cuenta a crear.*

Los logins se almacenan en un archivo de nombre **logins<numero de archivo>** dentro de la carpeta Logins para su posterior utilización.

Cuando se termina de crear las cuentas, la información correspondiente a la carga se guarda en un archivo de nombre **grupo<numero de archivo>** en la carpeta de Grupos.

*La creación de cuentas se realizara para cada uno de los archivos cargados por el profesor.*

- Una vez creadas las cuentas el usuario puede solicitar ver la información correspondiente a este proceso. La información se presenta por grupo y con el formato siguiente:

Nombre del Alumno Matrícula Login Password
--

El usuario puede solicitar al sistema una impresión de esta información [5].

- El usuario también puede solicitar al sistema un respaldo de información de algún alumno dado o de un grupo de alumnos contenidos en un archivo [3].

# Módulos del sistema

El sistema consta de los siguientes módulos:

## Recuperación de la información

Esta parte del sistema se encarga de obtener la información importante de un archivo cargado por el usuario y aplicarle al resultado el formato adecuado para poder crear las cuentas.

La información que se obtiene es:

- Nombre de la UEA.
- Trimestre.
- Nombre y Matrícula de cada alumno.

Su funcionamiento se describe de la siguiente forma:

### **1. Obtener información relevante para la generación de las cuentas.**

Para lograr esto se busca dentro del archivo cargado por el usuario ciertas cadenas que indican a partir de donde se puede encontrar la información que es relevante para nuestro sistema.

- Para obtener el nombre de la UEA se busca primero la palabra UEA y a partir de esta posición se busca la primera ocurrencia de una cadena que sólo contenga caracteres de la A-Z y de a-z y que no contenga la subcadena UEA o GRUPO [3].
- Para obtener el trimestre se busca la cadena TRIMESTRE y a partir de su posición se busca la primera ocurrencia de una cadena de longitud 3 que contenga caracteres alfanuméricos [3].
- Para obtener el nombre y matrícula de todos los alumnos se busca la cadena ALUMNOS y a partir de su posición se busca la información de los alumnos. Debido a que el archivo puede estar en dos formatos diferentes el procedimiento varía de uno a otro y se describen a continuación [3].

## Formato doc

Una vez encontrada la cadena ALUMNOS se lee línea por línea el archivo para encontrar la primera ocurrencia de una cadena compuesta de 3 subcadenas diferentes separadas por el caracter tabulador y a partir de aquí se leen todas las cadenas con estas características [3].

## Formato html

Después de encontrar la cadena ALUMNOS se busca la primera ocurrencia de la cadena "<table", esto porque la información de los alumnos está organizada en una tabla html, a continuación se limpia la información relevante de los tags html [3].

## 2. Aplicar formato a la información

Una vez que se ha obtenido la información que nos interesa se aplica un formato al nombre del alumno y su matrícula, el cual consta de un caracter separador (para el funcionamiento del sistema se utilizó el caracter ":") entre cada apellido del alumno su nombre y matrícula. Por ejemplo: se tiene el nombre David García Isario y su matrícula 204306905, entonces la información que se inserta en el archivo es:



David:García:Isario:204306905

Para aplicar este tipo de formato se tiene que realizar un pequeño proceso que identifique de manera adecuada cada uno de los apellidos y el nombre del alumno, debido a que estas tres partes que componen el nombre en cualquiera de los dos formatos de donde se extrae la información vienen dados en una sola cadena y, en un principio parece no haber información que ayude a identificar que subcadenas forman cada uno de los apellidos y el nombre. Esta identificación no es del todo trivial, si consideramos la gran variedad de apellidos, nombres y del número de palabras de las cuales se componen. Por ejemplo se tiene el siguiente nombre:



De La Reguera Hernández María Elena

3 1 2

En el ejemplo anterior se puede ver claramente como ésta división de campos no se puede hacer de manera ligera, ya que el primer apellido consta de tres cadenas el segundo de una y el nombre de dos.

La clave para saber qué cadenas forman cada parte del nombre son las palabras que actúan como enlace, por ejemplo DE y LAS para el ejemplo de arriba. *Su posición y su ocurrencia* nos indicaran como debemos dividir adecuadamente.

Es decir estas palabras las conocemos, sabemos que actúan como enlace entre dos palabras o que describen a otra y que son comunes en la formación de los nombres, por lo tanto las podemos guardar y utilizar para saber si existen dentro de un nombre.

En principio partimos del hecho que cada nombre consta de tres partes: Paterno, Materno y Nombre, de esto podemos concluir que el caso trivial es que un nombre esté compuesto por tres subcadenas, consecuentemente sólo se tendría que sustituir su caracter separador " " por ":" para que el nombre estuviera listo y poder utilizarlo.

*Por lo anterior podemos iniciar verificando cuantas subcadenas tiene el nombre que vamos a procesar.*

Volviendo al ejemplo anterior "De La Reguera Hernández Maria Elena" tiene 6 subcadenas por tanto no es el caso trivial tenemos que aplicar algo más.

Si no tenemos el caso trivial *hay que buscar y contar las palabras que se conocen (DE, LA, LAS, DEL) .*

En nuestro ejemplo encontramos dos: DE y LA por lo tanto hay varias posibilidades:

Un apellido esta formado por cuatro palabras (incluyendo DE y LAS), el otro apellido esta formado por una y el nombre por una. Pero no es la única posibilidad hay más combinaciones. Es por esto que *la posición de estas palabras también es importante.*

A continuación se muestra el procedimiento que se implementó dentro del sistema para identificar cada apellido y el nombre:

Palabra anterior = "palabras\_desconocida"

Dividir el nombre en palabras

**Mientras** haya palabras

**Si** el número de palabras restantes por verificar es mayor a 1

**Si** la palabra actual no la conocemos y la palabra anterior no la conocemos

            Entonces la palabra anterior es el fin del apellido y la palabra actual el comienzo del otro apellido o nombre

**Si** la palabra actual no la conocemos y la palabra anterior si la conocemos

            Entonces la palabra actual pertenece al apellido actual y es la última

**Si** la palabra actual si la conocemos

            Entonces la palabra actual pertenece al apellido actual

**De lo contrario**

    La última palabra pertenece al nombre

Aplicando este algoritmo a nuestro ejemplo "**De La Reguera Hernández María Elena**"

*El primer paso queda:*

DE	LA	REGUERA	HERNADEZ	MARÍA	ELENA
----	----	---------	----------	-------	-------

*Dentro del ciclo:*

Todavía hay palabras, palabras restantes = 6 y la palabra actual = DE entonces entra en el tercer caso. Esta palabra pertenece al primer apellido.

Todavía hay palabras, palabras restantes = 5 y la palabra actual = LA entonces entra en el tercer caso. Esta palabra pertenece al primer apellido.

Todavía hay palabras, palabras restantes = 4 y la palabra actual = REGUERA entonces entra en el segundo caso. Esta palabra pertenece al primer apellido.

**Primer apellido = DE LA REGUERA**

Todavía hay palabras, palabras restantes = 3 y la palabra actual = HERNADEZ entonces entra en el primer caso. Esta palabra pertenece al segundo apellido.

**Segundo apellido=** HERNADEZ

Todavía hay palabras, palabras restantes = 2 y la palabra actual = MARIA entonces entra en el primer caso. Esta palabra pertenece al nombre.

**Nombre=** MARIA

Todavía hay palabras pero sólo hay una. Esta palabra pertenece al nombre.

**Nombre=** MARIA ELENA

Antes de guardar cada uno de los apellidos y nombre se realiza un proceso para quitar acentos, diéresis y cambiar la ñ por las n ya que no se puede crear cuentas con este tipo de caracteres [3].

Los archivos generados por esta parte del sistema corresponden a los archivos cargados por el usuario (es decir el número de archivos generados es igual al número de archivos cargados) y se guardan en la carpeta "Users" bajo el nombre de **users<numero de archivo>**.

La implementación completa del recuperador de información se puede encontrar en la clase Recuperador en lenguaje PHP.

## Generador de cuentas de usuario

Este módulo del sistema se encarga de generar las cuentas de usuario a partir del archivo "users.txt" generado por el recuperador de información y el archivo "passwords.txt" creado por el módulo generador de passwords. Su función se divide en varias tareas:

- Generar el login para el alumno.
- Validar que el login generado sea único y crear la cuenta.
- Generar el archivo de logins.
- Generar el archivo donde se guarda toda la información de cada cuenta creada.

### 1. Generar el login para el alumno

El módulo comienza verificando si el alumno tiene o no una cuenta dentro del servidor del profesor. Para realizar este paso el generador de cuentas busca el alumno por su Matrícula dentro del archivo "passwd" (archivo del sistema donde se guarda la información de las cuentas existentes), y dependiendo del resultado realiza lo siguiente:

- Si encuentra la Matrícula se actualiza solo su password, se obtiene su login correspondiente y este se guarda temporalmente en un vector.
- Si no la encuentra entonces se procede a generar su login. Este login se crea tomando la primera letra del nombre (si el nombre consta de más de una palabra entonces se toma la primer letra de cada palabra que forma al nombre) concatenado esta con el apellido paterno y concatenando al resultado la primera letra del apellido materno. Por ejemplo tenemos los alumnos:

David García Isario su login seria **dgarciai**.  
 Carlos Enrique Ortega Moreno su login seria **ceortegam**

## 2. Validar que el login generado sea único y crear la cuenta.

Una vez que se ha creado un login se verifica que este sea único, por lo cual se realiza una búsqueda del login dentro del archivo "passwd" del SO [2].

- Si la búsqueda es nula se guarda temporalmente el login dentro de un vector y se crea la cuenta con el login generado y el password correspondiente leído del archivo de passwords.
- Pero si la búsqueda es exitosa entonces el login que generamos ya existe y en consecuencia se agrega el numero uno al mismo login para hacerlo único. Ejemplo dgarciai1. Se vuelve a realizar la búsqueda con el login modificado y si el resultado es nulo se guarda el login de lo contrario se cambia el numero 1 por el 2. Estos pasos de búsqueda y modificación se realizarán mientras el login que generemos exista.

## 3. Generar el archivo de logins

Una vez creadas todas las cuentas se genera el archivo de logins a partir del vector que guarda temporalmente los logins generados y los que ya existían en el servidor.

## 4. Generar el archivo donde se guarda toda la información de cada cuenta creada.

Como último paso se genera un archivo .php donde se guarda toda la información importante de cada cuenta que se creó: nombre y apellidos del alumno al que se le creó la cuenta, matrícula, login y password. Este archivo servirá para poder imprimir la información.



## Generador de passwords

Esta parte del sistema se encarga de generar los passwords, encriptarlos y guárdalos en un archivo llamando **Passwords<numero de archivo>** dentro de la carpeta Passwords. También guarda los passwords sin encriptar en un archivo de nombre **pt<numero de archivo>** para después utilizarlos en la impresión de las cuentas.

Los passwords que se generan son aleatorios, formados por ocho caracteres alfanuméricos pero algunos de estos caracteres no son utilizados para formar los passwords porque entre ellos son muy parecidos y en ocasiones se confunden. Estos caracteres son: 1, l, I, O, 0.

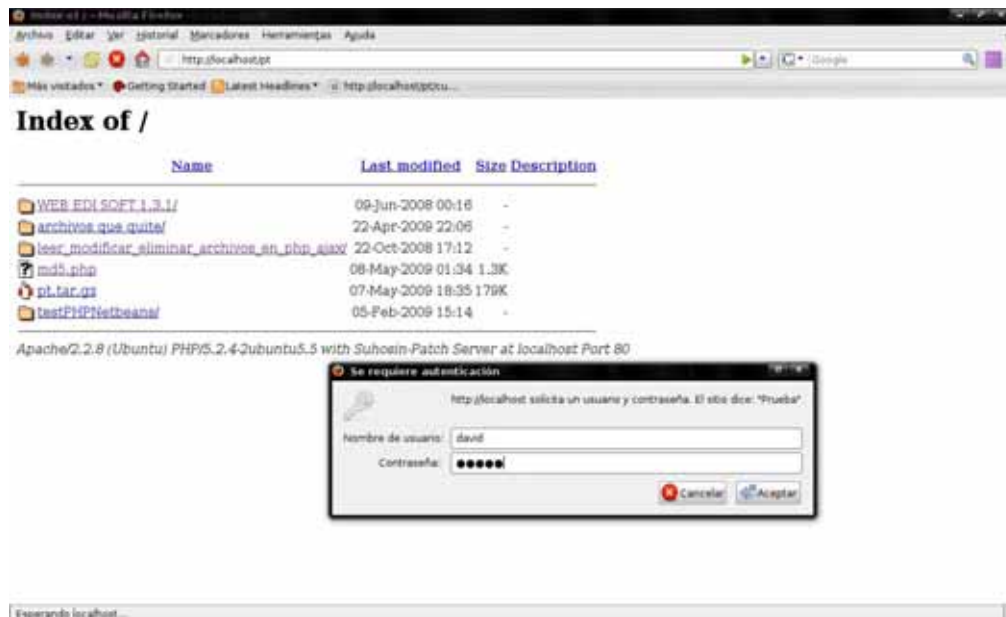
Los passwords son encriptados con md5 (algoritmo de reducción criptográfico de 128 bits) o crypt (método estándar de encriptación del Unix DES) según la distribución de Linux que se utilice [6]. Estos dos métodos de encriptamiento son de una sola vía, es decir una vez encriptados no se puede realizar el proceso inverso para desencriptar y obtener la cadena original.

Finalmente los passwords son almacenados en un archivo.

# Guía de usuario

Los pasos que el usuario debe seguir para utilizar el sistema de administración de cuentas son los siguientes:

1. Ingresar mediante un navegador Web a la dirección donde se haya instalado el sistema. Enseguida se abrirá un cuadro de dialogo que solicita login y password para poder ingresar a la carpeta donde se encuentra alojado el sistema, si usted tiene acceso a esta carpeta introduzca su login y password.



2. Enseguida se desplegará la pantalla siguiente, donde usted deberá ingresar su login y password para poder entrar al sistema.

The image shows a login form titled "Ingresar al sistema" (Log in to the system). The form is enclosed in a red border and contains the following elements:

- A header bar with the text "Ingresar al sistema".
- A label "Login" followed by a text input field.
- A label "Password" followed by a text input field.
- A button labeled "Enviar" (Send) located below the password field.

3. Cuando usted ya está dentro del sistema se muestra la pantalla de inicio siguiente.



Aquí usted puede realizar la carga de los archivos que contienen la información de los alumnos. Usted puede cargar uno o varios archivos a la vez.

Para cargar un archivo sólo debe dar clic en el botón examinar y buscar su archivo, si desea cargar otro archivo sólo dé un clic en el link "Subir otro archivo" enseguida le aparecerá otro botón "Examinar" nuevamente busque su otro archivo de interés. Este procedimiento se debe aplicar por cada archivo que usted quiera cargar al servidor.

Si por alguna razón usted ya no quiere cargar alguno de los archivos que ya seleccionó sólo debe dar clic en el link "Eliminar".

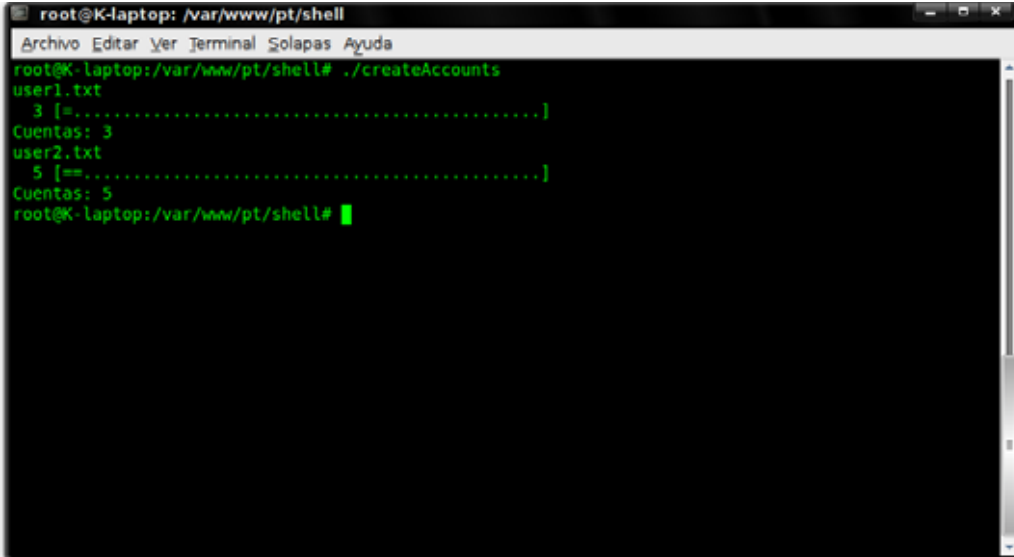
Una vez seleccionados los archivos de clic en el botón "Enviar" para cargar los archivos al servidor.

Es importante que usted al dar clic en el botón enviar no deje campos vacíos ya que no está permitido por el sistema y en consecuencia no podrá subir los archivos.

4. A continuación el sistema le mostrará la pantalla en la cual se muestra el estatus de la carga, número de archivos cargados y número de transacción etc.
5. Hasta aquí solo se ha procesado la información necesaria para crear las cuentas, ahora usted necesita ejecutar el script que creara las cuentas. Usted necesitara tener permisos de súper usuario para realizar este paso. Ejecute las siguientes instrucciones en una Terminal del sistema:

```
cd /ruta donde esta la carpeta del sistema/Pt/  
  
./createAccounts
```

Un ejemplo de este paso puede verse en la siguiente imagen:



```
root@K-laptop: /var/www/pt/shell  
Archivo Editar Ver Terminal Solapas Ayuda  
root@K-laptop:/var/www/pt/shell# ./createAccounts  
user1.txt  
3 [=====]  
Cuentas: 3  
user2.txt  
5 [=====]  
Cuentas: 5  
root@K-laptop:/var/www/pt/shell#
```

Con este paso usted ha creado las cuentas de cada archivo que cargó al servidor.

6. Vuelva a la pagina web del sistema y dé clic en la opción del menú “Cuentas creadas” para poder visualizar la información de las cuentas de usuario que se crearon correspondientes a cada archivo que usted cargó al servidor. En la imagen siguiente usted puede observar un ejemplo.

Administrador de cuentas

11:02:12 Lunes 11 de mayo de 2009

Cuentas creadas recargar

Grupo 1 Grupo 2 Grupo 3

Alumnos inscritos: 5

No.	Nombre	Matricula	Login	Password
1	MANZANARES SORIA SERGIO	204200272	jarroyog	xTUR8X74
2	BAUTISTA HERNANDEZ JORGE ALBERTO	201301215	jabautistah	Uu5YFP4x
3	BERNAL AVILA JORGE JAIME	204201113	jjbernal	HrGkotSz
4	BLANNO DE LA VEGA CESAR ALEJANDRO	204300810	cablannod	atwQpmPj
5	LEON CHORENO JUAN CARLOS	204302480	jcleonc	XrEFR4nu

Imprimir este grupo

Azcapotzalco @

7. Cuando usted desee salir del sistema sólo de clic en el botón “Terminar sesión” que se encuentra en la parte superior derecha de la pantalla del sistema. Es importante que usted al salir lo haga de este modo porque así destruye su sesión y no permite que otro usuario le robe la sesión para hacer mal uso del sistema.

# Manual y requerimientos de instalación.

## Requerimientos

En el servidor donde se va a instalar el sistema se deben tener previamente instalados los siguientes componentes:

### Un servidor web.

Debido a que toda la interfaz del sistema fue creada para la web es obligatorio contar con un servidor web dentro de nuestro servidor físico, el más usado y recomendable es Apache en su versión 2.x.x.

### El Zend 2 interprete de PHP 5.

Es necesario contar con esta versión del interprete porque gran parte del sistema esta escrito en lenguaje PHP y se utilizó programación orientada a objetos. Si se utilizara la versión de Zend 1 para PHP 4 el sistema simplemente no funcionará.

## Proceso de instalación.

1. El sistema completo está dentro de una carpeta, la cual debe ser copiada a la carpeta del Sistema Operativo /var/www/.
2. Se debe cambiar el propietario y grupo de la carpeta al usuario www-data [4]. Para realizar esto abra una terminal del sistema, cambie a súper usuario y ejecute las siguientes instrucciones:

```
cd /ruta donde esta la carpeta del sistema/Pt/  
chown www-data:www-data pt
```

Este paso servirá para poder tener acceso al sistema desde un navegador web y empezar a configurar la seguridad (para una mayor explicación vea seguridad).

3. Ahora se tendrá que editar el archivo "group" del sistema operativo. Abra una terminal, cambie a súper usuario y ejecute lo siguiente:

```
cd /ruta donde se encuentra el archivo group dentro de su sistema operativo/
```

Ahora busque dentro del archivo el grupo `www-data`, enseguida verifique si el usuario `www-data` y su usuario con el que ingresa al servidor ya se encuentran dentro de este grupo si no fuera así entonces agréguelos a este grupo de la siguiente manera [4]:

```
www-data:x:33:www-data,david #si no hubiera ningún usuario  
agregado a este grupo
```

```
www-data:x:33:algun_otro_usuario,www-data,david #si hubiera  
uno o más usuarios agregados ya al grupo
```

Como puede observar arriba se agregan los dos usuarios, cada usuario se separa por comas “,” y no se puede dejar algún tipo de espacios.

Ahora sólo falta asignar los permisos necesarios a la carpeta esto se realizara con la instrucción siguiente:

Cuando termine habrá bloqueado la entrada a la carpeta donde se guarda todo el sistema a otros usuarios que posean alguna cuenta dentro del servidor(para mayor información vea seguridad).

4. En este paso se protegerá la carpeta de la entrada de intrusos fuera del servidor mediante el internet. Por eso se tendrá que realizar lo siguiente:

Cambie el archivo “.htpasswd” [4] que se encuentra en la carpeta principal del sistema a una carpeta dentro de su directorio home de usuario. Abra el archivo, su contenido se muestra abajo :

```
david:HK4bEgiw5Hr.l
```

Lo anterior indica que el usuario `david` con la contraseña “`HK4bEgiw5Hr.l`” encriptada con `crypt` pueden entrar a la carpeta donde se encuentra el sistema.

Usted puede editar este archivo para agregar o borrar usuarios, abra una terminal, cambie a súper usuario y ejecute la siguiente instrucción:

```
htpasswd /ruta donde se encuentra el archivo .htpasswd/.htpasswd
```

Al ejecutar el comando anterior se solicitara un password y una confirmación de este, usted debe ingresar ambos para agregar el usuario al archivo.

Si queremos borrar un usuario tenemos que editar el archivo directamente con un editor de texto por ejemplo pico, la instrucción sería:

```
pico /ruta donde se encuentra el archivo .htpasswd/.htpasswd
```

Ahora sólo debe borrar la línea completa del usuario y guardar el archivo pulsado las teclas ctrl + o y confirmando pulsando la tecla enter.

Abra y edite el archivo “.htaccess” que se encuentra en la carpeta principal del sistema. Su contenido es el siguiente [4]:

```
AuthUserFile /ruta donde esta el archivo .htpasswd/.htpasswd
AuthGroupFile /dev/null
AuthName "Prueba"
AuthType Basic
Options All

<Limit GET POST PUT>
    require valid-user
</Limit>
```

Usted sólo tiene que cambiar el texto en negrita que indica la ruta donde usted coloco el archivo .htpasswd.

Una vez realizado el cambio guarde el archivo.

Ahora buscamos la carpeta de apache dentro del sistema operativo y dependiendo de la versión que se tenga agregamos al final del archivo “httpd.conf” o “apache2.conf” [4] las siguientes líneas:

```
<Directory "/var/www/Pt">
    AllowOverride All
</Directory>

<Files ~ "\.ini$">
    Order allow,deny
    Deny from all
</Files>
```

Con lo anterior se le indica a apache que queremos ocultar los archivos “.ini” y proteger la carpeta /var/www/Pt que es donde se encuentra el sistema, por lo tanto usted deberá cambiar esta dirección por la dirección donde usted coloque el sistema de cuentas.



Ahora se reinicia y se activa el servidor apache con los siguientes comandos [4]:

```
/etc/init.d/apache2 restart  
/etc/init.d/apache2 start
```

5. Si se realizara algún cambio de nombre o de ubicación de algún archivo del sistema se debe editar el archivo de configuración "configuration.ini" que se encuentra en la carpeta principal del sistema. En este archivo se encuentran los path de algunos archivos necesarios para el funcionamiento correcto del sistema, por lo tanto usted debe tener mucho cuidado al realizar la configuración de este archivo.

A continuación describiremos cada apartado del archivo de configuración:

Antes de ver a detalle el contenido del archivo de configuración tenga las siguientes consideraciones si desea cambiar alguno de sus valores:

- El nombre de las variable es fijo, es decir no puede cambiarse por ningún motivo, si así se hiciera el sistema dejaría de funcionar. Si por cuestiones de mantenimiento o alguna otra razón usted desea cambiar este valor tendrá que hacerlo también dentro del código del sistema.
- Los valores de cada variable no deben contener ningún tipo de caracter raro, espacio y deben conservar (si los hubiera) el caracter "/" que es necesario para la formación de los path.
- Los comentarios se realizan con el caracter ";" al principio de la línea que se quiera indicar como comentario.
- Entre el nombre de la variable, el caracter "=" y su valor no debe existir ningún espacio, tabulador o fin de línea debido a que el caracter "=" es el que define donde termina uno y donde empieza el otro. Ejemplo de lo que no se debe hacer:

```
Variable = Valor //hay espacios  
Variable= Valor //hay espacios  
Variable=      Valor //hay tabulador
```

## Estructura de archivo Configuration.ini

**pathNewfolder=../Transacciones/CuentasDeUsuario**

Esta variable contiene el nombre de la carpeta principal “Transacciones” donde se guarda cada transacción de creación de cuentas y para cada una se crea una carpeta llamada “CuentasDeUsuarioN” donde N es el número de transacción. El termino transacción se utiliza aquí para referirse al procesamiento y creación de cuentas en base a los archivos que usted sube en un solo lote al servidor.

La carpeta “CuentasDeUsuarioN” contendrá toda la información respectiva a la transacción de un lote de cuentas que fueron creadas.

Al nombre de la carpeta “CuentasDeUsuario” se añade un número al final correspondiente a la transacción que se haya ejecutado en orden cronológico. Por ejemplo al ejecutar la primera transacción de creación de cuentas el día de hoy, se creará la carpeta “CuentasDeUsuario1” y si ejecutamos una el día de mañana, se creará la carpeta “CuentasDeUsuario2” y así sucesivamente.

Usted puede modificar el nombre de cada una de las carpetas que hemos mencionado anteriormente teniendo en cuenta que los nombres que utilice no contengan espacios o caracteres raros y además que el formato que originalmente contiene la variable se respete:

../Nombre\_de\_carpeta/Nombre\_de\_carpeta

**folderfilesLoad=Loads/  
filesLoad=**

La variable folderfilesLoad contiene el nombre de la carpeta donde se guardarán los archivos que el usuario cargara al servidor respecto a una transacción.

La variable filesLoad contiene el nombre base para cada archivo cargado por el usuario al servidor. El nombre base es aquel que es común para cada archivo y su diferencia con el nombre verdadero es un numero que se agrega al final del nombre base. Por ejemplo si el usuario sube en una transacción de 3 archivos y el nombre base es “archivo” entonces el nombre verdadero que se asignara a cada archivo es archivo1, archivo2, archivo3. El valor original en el archivo de configuración es nulo es decir el nombre de cada archivo será solo un numero.

Si usted desea cambiar el valor de estas dos variables debe cumplir para ambas que el valor nuevo no contenga espacios o caracteres extraños además deben cumplir con el formato siguiente:

```
folderfilesLoad=Nombre_de_carpeta/  
filesLoad=Nombre_base_archivo
```

### **infoFileLoad=infoAccounts.txt**

Contiene el nombre del archivo donde se guarda la información correspondiente a la transacción. Por ejemplo número de archivos cargados por el usuario al servidor, número de alumnos por cada archivo, nombre de cada UEA por grupo etc.

### **passwordsFolder=Passwords/ nameFilePasswords=passwords**

La variable passwordsFolder contiene el nombre de la carpeta donde se guardarán los archivos que contiene los passwords encriptados de los usuarios generados por el sistema respecto a una transacción.

La variable nameFilePasswords contiene el nombre base para cada archivo donde se guardan los passwords encriptados generados por el sistema correspondiente a cada archivo cargado por el usuario.

### **usersFolder=Users/ nameUsersFile=user**

La variable usersFolder contiene el nombre de la carpeta donde se guardarán los archivos que contiene la información necesaria para la creación de las cuentas de los usuarios generados por el sistema respecto a una transacción.

La variable nameUsersFile contiene el nombre base para cada archivo donde se guarda la información necesaria para la creación de las cuentas por el sistema correspondientes a cada archivo cargado por el usuario.

El formato de este archivo es el siguiente:

A.paterno:A.materno:Nombre:Matrícula

### **loginsFolder=Logins/ nameFilelogins=login**

La variable loginsFolder contiene el nombre de la carpeta donde se guardarán los archivos que contiene los logins de los usuarios generados por el sistema respecto a una transacción.

La variable nameFilelogins contiene el nombre base para cada archivo donde se guardan los logins generados por el sistema correspondiente a cada archivo cargado por el usuario.

**groupsFolder=Grupos/  
nameGroup=grupo**

La variable groupsFolder contiene el nombre de la carpeta donde se guardarán los archivos que contiene la información final de la creación de las cuentas por el sistema respecto a una transacción.

La variable nameGroup contiene el nombre base para cada archivo donde se guarda la información final de la creación de las cuentas correspondientes a cada archivo cargado por el usuario.

Este archivo contiene la información siguiente:

- Número de alumno
- Nombre completo del alumno
- Matrícula
- Login
- Password(sin encriptar)

**sesionFile=../sesionFile.ini**

Esta variable contiene la ubicación del archivo sesionFile.ini el cual contiene información muy importante y necesaria para saber cuántas veces se han cargado archivos al servidor mediante el sistema y cuántas durante una sesión.

# Seguridad

En estos tiempos la administración de cuentas de usuario dentro de un servidor es muy importante ya que representa un factor vital en la buena integridad del funcionamiento del sistema. Si alguien llegara a apoderarse de una cuenta dentro del servidor que no le pertenece no quisiera imaginar lo que podría suceder. Por tanto para el sistema de administración de cuentas se gestionó la seguridad de manera cuidadosa, con el objetivo que este sistema sea una ayuda para el usuario y no un problema aun mayor por resolver.

Es importante mencionar que todo lo que se describe aquí se tiene que realizar como súper usuario y por tanto se deben de contar con los permisos necesarios de lo contrario la seguridad se verá seriamente comprometida y será mejor que no se utilice el sistema de administración de cuentas.

A continuación se describe los puntos de seguridad que se implementaron para que el sistema sea seguro y confiable:

## **Acceso restringido a la carpeta web donde se encuentra el sistema desde fuera del servidor.**

Ya que una parte del sistema está implementada para la plataforma web, el acceso desde fuera del servidor mediante el web a la carpeta donde se encuentra el sistema debía estar restringido y así evitar fuga de información de alguno de los archivos importantes del sistema.

Esto se realizó utilizando los archivos .htpasswd y .htaccess del Servidor Apache.

El archivo .htaccess sirve para indicar a el servidor apache que la carpeta web donde se encuentra necesita tener un acceso restringido [4].

El archivo .htpasswd sirve para indicar que los usuarios contenidos en él son los únicos que pueden acceder a la carpeta donde se encuentra el archivo .htaccess correspondiente [4].

Por lo tanto el primer paso es crear el archivo `.htaccess` y guardarlo en la carpeta donde se encuentra el sistema, el contenido de este es el siguiente:

```
AuthUserFile /home/david/.htpasswd
AuthGroupFile /dev/null
AuthName "Seguridad"
AuthType Basic
Options All
<Limit GET POST PUT>
    require valid-user
</Limit>
```

La primera línea es la que usted cambiará dependiendo del lugar donde se ubique dentro del servidor, el archivo `.htpasswd` correspondiente. Es esta línea la que liga los dos archivos y así indicamos la carpeta que queremos proteger y los usuarios que pueden entrar en ella.

Es importante mencionar que la ubicación del archivo `.htpasswd` debe ser fuera de la carpeta donde se publican las web locales o donde se encuentra el sistema, por ejemplo un buen lugar para ubicar el archivo `.htpasswd` es el directorio `home` del usuario. Esto se hace para evitar que este archivo sea más difícil de leer.

El segundo paso es crear el archivo `.htpasswd` y ubicarlo en la dirección que indica la primer línea del archivo `.htaccess`.

Para agregar un usuario a este archivo se tendrá que ejecutar la siguiente instrucción:

```
htpasswd /ruta donde se encuentra el archivo .htpasswd/.htpasswd
login de usuario
```

Después el sistema pedirá contraseña y confirmación de esta para el usuario. Una vez terminado lo anterior el usuario será agregado al archivo `.htaccess`.

La estructura del archivo .htpasswd es la siguiente:

```
login_de_usuario:password_encryptado_con_crypt
```

Si usted desea borrar algún usuario tendrá que editar el archivo ya que no existe instrucción alguna para hacer esto.

Ahora buscamos la carpeta de apache dentro del sistema operativo y dependiendo de la versión que se tenga agregamos al final del archivo "httpd.conf" o "apache2.conf" las siguientes líneas [4]:

```
<Directory "/var/www/Pt">  
    AllowOverride All  
</Directory>  
<Files ~ "\.ini$">  
    Order allow,deny  
    Deny from all  
</Files>
```

Usted debe cambiar el texto en negrita que indica la ruta absoluta de la carpeta donde se encuentra el sistema.

Además de la línea 4 a 7 indicamos que los archivos .ini sean ocultos a la vista de los usuarios de web.

Lo último es guardar los cambios al archivo.

Una vez realizado los pasos anteriores sólo falta reiniciar y volver a activar el servidor apache puesto que modificamos su archivo de configuración y necesitamos que cargue el archivo modificado para que se active el acceso restringido a la carpeta. Para esto ejecutamos las instrucciones siguientes:

```
/etc/init.d/apache2 restart  
/etc/init.d/apache2 start
```

Una vez realizados los pasos anteriores usted puede comprobar que el acceso a la carpeta ya está restringido. Para esto reinicie su navegador web y vaya a la dirección donde está alojado el sistema.

El resultado debe ser parecido al de la imagen de abajo, como se puede observar aparece un cuadro de dialogo que pide login y password para permitir el acceso a la carpeta, pero además si observa en la parte del árbol de carpetas no se muestra la carpeta a la cual quiere usted entrar en este caso la carpeta del sistema de cuentas.



## Acceso restringido a la carpeta web donde se encuentra el sistema desde adentro del servidor.

En el paso anterior condicionamos el acceso al sistema desde afuera del servidor, en este paso vamos a condicionar el acceso al sistema desde adentro, es decir vamos a restringir el acceso a los usuarios que tiene una cuenta dentro del servidor.

Este paso es muy importante ya que en la carpeta donde se publican las paginas web dentro del servidor (en Ubuntu /var/www/) cualquier usuario puede leer, crear o modificar cualquier tipo de archivo, por tal motivo sino se protege el sistema cualquier usuario podría al menos leer los archivos que genera el sistema, lo que significaría una grave vulnerabilidad para el servidor físico.

Es por eso que se deben seguir los siguientes pasos para la protección del sistema de usuarios internos dentro del servidor donde está alojado el sistema.



## **Cambiar el propietario y grupo de la carpeta donde se aloja el sistema por usuario www-data.**

Para realizar esto abra una terminal del sistema, cambie a súper usuario y ejecute las siguientes instrucciones:

```
cd /ruta donde esta la carpeta del sistema/pt/  
chown www-data:www-data pt
```

- La primer instrucción nos ubica en la carpeta del sistema.
- La segunda instrucción cambia el propietario y grupo de la carpeta observe que estos están separados por el caracter ":".

Estas instrucciones son ejecutadas para que la carpeta del sistema pueda ser visitada desde un navegador web y por ende se pueda utilizar el sistema. Si este paso no se realiza simplemente nadie podrá ocupar el sistema. Posteriormente se explicará como restringir el acceso a la carpeta y sólo aquellos con permisos puedan entrar.

## **Agregar al grupo www-data algunos usuarios.**

Este paso se realiza para restringir la entrada de la carpeta donde se guarda todo el sistema a otros usuarios que posean alguna cuenta dentro del servidor. Para poder hacer esto siga lo siguiente

Edite el archivo "group" del sistema operativo [1]. En este archivo se guardan los grupos existentes en el sistema operativo. Para realizar este paso debe tener permisos de súper usuario. Abra una terminal, cambie a súper usuario y ejecute la siguiente instrucción:

```
pico /etc/group
```

- Con la instrucción anterior se abre el archivo para editarlo.

Dentro del archivo identifique el grupo www-data, verifique que los usuarios www-data y su usuario no estén ya agregados, si ya estuvieran agregados cierre el archivo sin ningún cambio, si por el contrario no estuvieran agregados agregue los usuarios y guarde el archivo. Es importante recordar que cada usuario es separado por una coma.

Ahora solo falta asignar los permisos necesarios a la carpeta esto se realizará con la instrucción siguiente:

```
chmod 0770 carpeta_donde_esta_el_sistema
```

Con esto indicamos al sistema operativo que sólo el propietario y los usuarios que pertenecen a su grupo pueden entrar a la carpeta del sistema.

## Conclusiones

- La administración de cuentas de usuario no es una tarea fácil y además representa un punto muy importante dentro de un servidor por tal motivo es bueno contar con algún sistema que ayude a automatizar este proceso y así evitar pérdida de tiempo y errores humanos.
- Al realizar cualquier tipo de sistema que administre las cuentas de usuario, se tiene que poner especial atención en la seguridad que brinde dicho sistema al realizar su trabajo, ya que en lugar de ayudar a ahorrar tiempo y esfuerzo al administrador del servidor, por el contrario puede ayudar a terceras personas a poner en peligro la integridad del servidor.
- La combinación de Apache y Linux dio como resultado una plataforma de desarrollo muy segura y confiable para el desarrollo de este sistema, y además hay que tener en cuenta que es puramente software libre.

# Bibliografía

1. Dee-Ann Leblanc, "La Biblia de la administración de sistemas Linux", editorial Anaya multimedia.
2. Tom Adelstein, "Administración de sistemas Linux", editorial Anaya multimedia.
3. Mendel Cooper, "Advanced Bash-Scripting Guide", Versión 5.5.04.
4. Mohammend J. Kabir, "Apache Server 2 Bible", editorial Wiley.
5. Luke Welling, "Desarrollo Web con PHP y MySQL", editorial Anaya multimedia.
6. Bill von Hagen & Brian K. Jones, "Linux Server Hacks", editorial O'REILLY.
7. Richard Blum, "Linux Command Line and Shell Scripting", editorial Wiley.