

Instalación y configuración del sistema TocToc.

Jorge Hortelano Otero.

17 de febrero de 2006

Resumen

Estas páginas tiene como finalidad instruir al lector en la instalación y configuración del sistema TocToc. Así como presentar los requisitos necesarios en el sistema para el correcto funcionamiento del mismo.

1. Introducción.

El sistema TocToc es un ejemplo de *Captive Portal*. Un *Captive Portal* es¹un sistema que fuerza al cliente de una red a mostrar una página determinada (generalmente con propósito de identificación). Esto se realiza mediante al interceptación del tráfico HTTP, independientemente de la dirección seleccionada, hasta que el usuario tiene permiso de salir del portal. Se pueden observar el uso de los *Captive Portals* en muchos de los *Hotspots* que usa la tecnología WI-FI. También pueden ser usados para controlar los accesos en los medios guiados por cable.

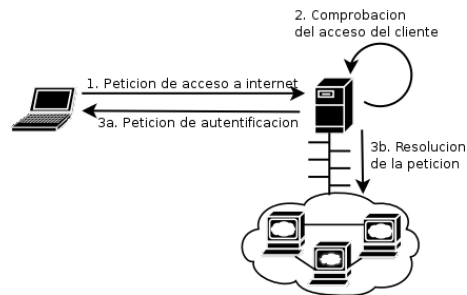


Figura 1: Estructura básica de un *Captive Portal*.

La red Wireless debe de estar completamente operativa como una red normal, ya sea en modo Ad-Hoc o mediante un punto de acceso. Esta red no debe tener ningún tipo de encriptación como WEP o WAP, ya que es el propio sistema el que se encarga de autentificar el usuario. Por ello, se debe de permitir primero el acceso libre a la red.

¹Según la Wikipedia: <http://wikipedia.org>

2. Requisitos.

Para el correcto funcionamiento del sistema TocToc, es necesario tener ya instalados los siguientes programas:

1. Un sistema de cifrado seguro como OpenSSL para dar seguridad a los clientes.
2. Servidor Apache 2.0 o superior. Dicho servidor deberá tener activado la conexión segura con SSL. Cómo se ha configurado la instalación del Apache para este proyecto se muestra de una forma detallada en el apéndice de este documento.
3. MySQL 4.1.10 o superior.
4. PHP 5.0 o superior. Este debe ser configurado para el uso del servidor Apache y MySQL. En el apéndice existe un apartado por si no estas familiarizado con la instalación de PHP.
5. Servidor DHCP propio.
6. El paquete *iproute*. Este paquete aparece en las distribuciones de Linux, por lo que consulte la distribución utilizada para saber como instalarlo.

3. Configuración del servidor Apache.

Aparte de la configuración normal del servidor Apache, hay que añadir unas modificaciones no convencionales para que el sistema TocToc funcione correctamente. Como configuración normal se entiende la generación de una carpeta ligada al mismo que contenga todas las páginas web del servidor, las opciones necesarias para que interprete los documentos PHP y la información relativa al administrador².

Como primera modificación extra, hay que descomentar la línea que aparece en `httpd.conf`: `ErrorDocument 404 /error/HTTP_NOT_FOUND.html.var` y modificarla por la siguiente: `ErrorDocument 404 /error/error.php`. Después de esto añadir los documentos `error.php` y `error_original.php`, que se incluye junto los archivos del sistema TocToc, en la carpeta `/Directorio_apache/errors/`. Esta página, simplemente redirecciona al cliente a la página de login del sistema. De esta forma el sistema se asegura que cuando capture un cliente que no este dado de alta, siempre se le muestre la página correspondiente y no salga un error. Conviene revisar esta corta página para observar que el link de redirección apunta a donde debe.

Una vez acabado esto, hay que realizar exactamente lo mismo para el error 403, de forma que impidamos el listado del contenido del directorio web y lo

²Todo esto se puede encontrar en la web oficial del Apache: <http://httpd.apache.org/docs/2.0/>. Existe un pequeño resumen de instalación al final de este texto, en el capítulo 12.

enviemos a la hoja de registro. Para desactivar la opción por defecto de listar contenidos y que aparezca el error 403 se debe comentar la línea *Options Indexes FollowSymLinks* que existe dentro de la declaración del directorio web. Hay que tener cuidado con estas opciones si se desea albergar múltiples páginas web en este servidor ya que los mensajes de error serán modificadas para todas.

Otro punto a tener en cuenta es que, debido a que el servidor Apache tiene que utilizar algunos comandos restringidos al root, se debe dar permisos al mismo mediante el fichero */etc/sudoers* añadiendo la siguiente línea: *apache ALL=NOPASSWD:/sbin/iptables, /sbin/tc, /bin/sh* que le permite ejecutar los comandos TC, IPTABLES y SH³ necesarios para el sistema.

Por último, recordar que el servidor Apache tiene un usuario en el sistema UNIX asociado, normalmente *apache* o *nobody*. Los scripts del sistema TocToc están preparados para que el usuario del servidor sea el denominado *apache*. Por ello, hay que modificar la línea *user* del archivo *httpd.conf* y colocar el usuario *apache*. Posiblemente haya que crear ese usuario en el sistema y dar, además, permisos de ejecución a los scripts del sistema para este usuario. Generalmente, el no hacer esto, no muestra ningún error al ejecutar el sistema, simplemente este se limita a ignorar los scripts para los cuales no tiene permisos, obteniendo unos resultados inesperados.

4. Copia y configuración de los ficheros necesarios.

4.1. Copia de los archivos.

Una vez esta todo funcionando en orden, es decir, se puede visualizar cualquier página web con nuestro servidor desde otro cliente, hay que colocar las propias páginas del sistema en sus sitio.

1. Mover la carpeta *web_TocToc* que contiene todos los ficheros PHP al directorio en el que el servidor Apache sirve las páginas web (llamémosle a partir de ahora “directorio *www*” o “*www*” simplemente).
2. Darle permisos de ejecución a los archivos para que PHP no tenga ningún problema con ellos. Eso significa que todos los archivos relacionados tienen que tener permisos de lectura para todos.
3. Guarda la carpeta *TocToc* que contiene los *scripts* de ejecución del sistema a tu equipo y dótale también de los permisos necesarios de ejecución. A lo largo del manual, nos referiremos a esta carpeta simplemente como *dir_scripts*.

³El administrador del sistema tiene que tener mucho cuidado si aloja otras páginas web para no darles a ellos permisos de lectura, escritura y ejecución en sus scripts y provocar así un agujero en su seguridad. Por ello, es aconsejable que esta máquina sea un servidor dedicado exclusivamente al sistema TocToc.

4.2. Configuración.

Los archivos que hay que configurar son los siguientes:

1. *www/include/colores.inc*.
2. *www/include/conexion.inc*.
3. *www/include/configuracion.inc*.
4. *www/include/TocToc.css*
5. *dir_scripts/TocToc.sh*.
6. */Directorio_apache/errors/error.php*

A continuación, se detalla la configuración y el uso de cada uno de ellos.

4.2.1. Archivo *colores.inc*.

Este archivo guarda la configuración básica de los colores de la página. Permite cambiar el aspecto de todo el sistema, junto con el archivo *css* y el *xsl*. Los nombres de las variables son autoexplicativos, y el color tiene que escribirse en formato RGB. La imagen que se encuentra aquí, corresponde con la imagen que aparecerá de fondo en cada página web.

4.2.2. Archivo *conexion.inc*

En este archivo se especifica, entre otras cosas, el *host* que contiene las bases de datos del sistema, el usuario que tiene acceso a las mismas y el *password* necesario para ello. Aquí tienes que poner los datos dependiendo de la instalación de MySQL en tu sistema.

Las dos constantes *BD_USUARIOS* y *BD* indican el nombre de las dos bases de datos del sistema. Por lo que no es necesario cambiarlos salvo que se haga lo mismo en las bases de datos MySQL.

4.2.3. Archivo *configuracion.inc*

Este archivo contiene la configuración específica de su sistema. Aquí se deben de introducir tanto algunos datos propios de su equipo, como otros datos que ya dependen de los resultados que se deseen que devuelva el sistema. A continuación se muestra el significado de las constantes aquí definidas:

1. *TOCTOC_FOLDER*: indica la carpeta que contiene todas las páginas web del fichero. Tiene que tener la estructura *www/TOCTOC_FOLDER/*.php*
2. *PAGINA_INTERNET*: a que página se direccionará el usuario cuando accede a Internet si previamente no ha elegido una.

3. *DIR_CLIENTES*: la carpeta que contiene los archivos que se utilizan para controlar los clientes. Tienen que ser la misma que la indicada en el script *TocToc.sh*.
4. *CITYTICKET_TOTAL*: aquí se indica el número de citytickets que quiere que maneje el sistema. La configuración de lo que se quiere que haga cada uno de ellos, viene definido en el vector *array_velocidad_conexion[cityticket][0 (bajada) - 1 (subida)]*. Por ejemplo: para la configuración *array_velocidad_conexion[2][0]=16* significa que para el Cityticket 1, su velocidad de bajada es de 16KB/s.
5. *LOG*: esta constante contiene el nombre del fichero de logs del sistema.
6. *TIEMPO_LOG*: el tiempo en segundos que se mantienen las entradas en el *log* antes de trasladarlas a la copia de seguridad del mismo. Por defecto es un mes.
7. *LOG_BAK*: el nombre en donde se guardarán las entradas del *log* una vez pasen el tiempo máximo que deben de quedar registradas.
8. *DIRECTORIO_LOG*: el directorio donde se almacenará el *log* del sistema.
9. *REFRESCO_ACCESO_WEB*: aquí se indica el tiempo que tardará en refrescarse la ventana de control de Internet que se le abrirá al cliente. Debe ser menor que el tiempo indicado en *TocToc.sh* o se corre el riesgo de que se desconecten los clientes sin motivo alguno.
10. *RED_WIRELESS*: el nombre que le da Linux al dispositivo de acceso inalámbrico. Generalmente suele ser o *eth1* o *wlan0*.
11. *RED_INTERNET*: el nombre del dispositivo que sale a Internet. Generalmente será *eth0*.
12. *IP_GATEWAY*: la IP que tiene el ordenador que actúa como Gateway (en donde está instalado el sistema TocToc). A esta IP se dirigirán los clientes para salir a Internet.
13. *PUERTOS_ACEPTADOS*: el rango de puertos que se reserva el sistema para poder añadir aquellos servidores permitidos a los que los clientes pueden salir sin pagar el acceso a Internet.
14. *TC*: la dirección del ejecutable del programa *tc* de Linux.
15. *IPT*: la dirección del ejecutable del *iptables* de Linux.
16. *TOCTOC_SCRIPT_PATH*: en donde se almacenan los scripts del servidor TocToc.
17. *TOCTOC_SCRIPT*: el nombre del script que activa el servidor TocToc.

18. *ROOT*: esta constante contiene el nombre del administrador del sistema que en principio no se mostrará en el mismo (es decir, será imposible de borrar, modificar y demás). Se puede usar esta cuenta para tener un administrador principal del sistema sin que le aparezca a los demás encargados del mantenimiento del mismo.
19. *PAGINA_PRINCIPAL*: la página a la que se dirigirá el sistema cuando un usuario acceda exitosamente en el sistema.
20. *CSS*: El archivo que contiene la hoja de estilos en cascada. Escribirlo mediante una variable permite probar con varios diseños cambiando esta variable.
21. *TITULO*: El título que aparecerá en la cabecera del navegador. Así se permite una fácil integración con otros sistemas.
22. *NOMBRE*: El nombre con el que se referenciará al sistema. El único valor de esto es mostrarlo en algunas páginas web.

4.2.4. Archivo *TocToc.css*.

Este archivo es una hoja de estilos en cascada que utiliza el sistema para darle el formato a los títulos de las páginas, los menús y demás. Se puede modificar a gusto del administrador sin que varíe en algo el comportamiento del sistema.

4.2.5. Archivo *TocToc.sh*.

Este es el script principal de TocToc. Al ejecutarlo prepara al equipo para recibir a los clientes según sus niveles de acceso. Es también el que impide que salgan a Internet sin permiso. Para configurarlo, hay que mirar las siguientes constantes:

1. *IPT*: Al igual que en el archivo *configuracion.inc* esta constante indica donde se encuentra el binario para ejecutar las *iptables*. En la distribución probada se encuentra en: */sbin/iptables*.
2. *TC*: Similar al anterior, pero para la ejecución del *tc*. Se encuentra en el equipo probado en: */sbin/tc*.
3. *ETLAN*: La red *ethernet* por la que se sale a Internet. Algo del tipo: *152.41.51.0/22*.
4. *WLAN*: La dirección de la red *wireless* por la que acceden los clientes. En este caso se ha elegido *192.168.1.0/24*.
5. *ETH*: El nombre que el sistema Linux da al dispositivo de red ethernet. Por ejemplo *eth0*.
6. *WIR*: El nombre del dispositivo wireless que le da el sistema operativo. Generalmente *eth1* o *wlan0*.

7. *PWLAN*: Esta constante contiene aquella parte de la dirección de red que es común entre todos los clientes. Sería el equivalente al aplicar la máscara a la red, pero que por razones de diseño, tenemos que indicar aquí explícitamente. En nuestro caso *192.168.1*.
8. *PUERTOS_ACEPTADOS*: Aquí se indican el rango de puertos reservados por el sistema para aquellas páginas web a las que se permita el acceso de los clientes aunque no tengan conexión a Internet permitido. Los que se incluyen en el archivo son *9400:9800*.
9. *IPS_ACEPTADAS*: Aquí se indica la IP de aquellos servidores a los que se quiera dar acceso sin que exista una conexión a Internet. Aquí se indica aquellos servidores que siempre estarán implícitos en el sistema y no se podrán modificar desde la consola de administración.
10. *DIR_PHP*: La ruta que permite al sistema encontrar todos los ficheros PHP. Algo como */www/TocToc*.
11. *DIR_CLIENTES*: Aquí se indica una carpeta en el que el servidor apache tendrá permisos para poder dejar todos los archivos de configuración que necesita para su correcto funcionamiento. Algo como */TocToc/clientes*.
12. *DIR_SCRIPTS*: En que directorio están los demás scripts que necesita ejecutar el servidor. Por ejemplo, */TocToc/scripts*.
13. *LOG*: Nombre del fichero en donde se guardan los *logs* del sistema. El nombre es indiferente mientras se mantenga en el sistema.
14. *DIR_LOG*: Allí donde se sitúa el archivo del *log*. Siguiendo las pautas anteriores, quedaría algo así como */TocToc/log*.
15. *TIEMPO*: El intervalo de tiempo que se espera el servidor para comprobar si el cliente ha caído o no. Se aconseja que sea unos segundos superior al tiempo indicado en la constante *REFRESCO_ACCESO_WEB* del archivo *configuracion.inc*.
16. *TOCTOC*: Esta variable indica el número final de la IP del ordenador en el que está alojado el servidor. Nótese que *PWLAN.TOCTOC* tienen que formar una dirección IP válida. En nuestro caso sería un *1*, que acabaría formando *192.168.1.1*.
17. *DHCPmin* y *DHCPmax*: Estas dos variables indican que rangos de direcciones del servidor DHCP se distribuyen entre los clientes que se conectan por red. *PWLAN.DHCPmin* y *PWLAN.DHCPmax* tienen que formar direcciones IP válidas.
18. *RATEUP* y *RATEDOWN*: Estas dos variables indican el ancho de banda máximo de subida y de bajada que utilizará el sistema para repartir entre los clientes. Lo aconsejable es que sea un 75 % más o menos del ancho de banda total.

4.2.6. Archivo */Directorio_apache/errors/error.php*

Este archivo deberá buscarse en la carpeta *error* del sistema TocToc si todavía no ha sido copiado a la carpeta de errores de Apache. Una vez realizado la copia, habrá que configurarlo revisando el valor de las siguientes constantes:

1. *IP_GATEWAY*: Aquí, al igual que en los archivos anteriores, se tendrá que indicar la dirección IP del dispositivo inalámbrico del ordenador en el que está instalado el sistema.
2. *RED_WIRELESS*: Escribir la dirección de la red Wireless para diferenciar los accesos al sistema como clientes TocToc o no. Es decir, aquellos equipos que entran por esta red son los que tiene que atrapar el sistema hasta que se dan de alta en el sistema.
3. *TOCTOC_FOLDER*: La carpeta en donde están todos los archivos PHP del sistema. Tiene que contener el mismo valor que *DIR_PHP* descrito en el apartado anterior.

5. Ejecución del sistema.

Una vez ha sido instalado y configurado todo lo anterior, la forma de que el sistema se ponga en funcionamiento es la siguiente.

1. Comprobar que los dispositivos de red del servidor están activos y funcionando correctamente.
2. Asegurarse que los servidores Apache y DHCP están en funcionamiento.
3. Poner en marcha el servidor TocToc mediante la ejecución del script *TocToc.sh*. Esto se hace de la siguiente forma: *“sh TocToc <start/stop><0/1>”* por ejemplo *“sh TocToc.sh start 0”*. El 0 indica que queremos que nos muestre por pantalla todos los mensajes, el 1 que los omita. Otra forma de poner en marcha el sistema es conectarse a *http://localhost/TocToc* e introducir un usuario que tenga privilegios de administrador en el sistema. Si es así, le aparecerá en el menú principal una opción de activación del sistema.
4. Ahora solo falta coger un cliente, conectarse a la red del sistema, y comprobar que todo funciona.

6. Detención del sistema.

También hay que saber como detener el sistema correctamente. Para detener el sistema, lo que hay que hacer es ejecutar el *script* descrito en la sección anterior de la siguiente forma: *“sh TocToc stop”*. Si no se detiene el sistema de esta forma, se queda un proceso activo en el sistema. Este proceso es el que vigila que

no se queden clientes activos si estos han caído de forma inesperada. Tener este proceso activo no perjudica a la computadora, pero cuando se vuelva a ejecutar el sistema TocToc, el tener dos procesos que eliminan clientes inactivos puede ser problemático para el correcto funcionamiento del mismo. Si esto ocurre, es posible que clientes que no están inactivos sean eliminados por el sistema debido al funcionamiento doble de este proceso.

7. Instalación básica de los programas.

7.1. Instalación del OpenSSL y generación de un certificado propio.

La instalación del sistema OpenSSL depende del sistema Linux elegido. Existen paquetes binarios relativos a cada distribución que pueden ser usados, facilitando la tarea de instalación. De otra forma, se tendrá que descargar el código de la web oficial <http://www.openssl.org/> y compilarlo para su funcionamiento:

1. Descomprimir el paquete descargado: `tar xvf openssl-0.9.X.tar.gz`.
2. Un simple `./configure && make && make install` debería bastar.

Una vez ha sido instalado, se debería pasar crear un certificado de seguridad. Para ello hay que seguir estos pasos:

1. Creación de la clave del servidor de 1024 bits: `openssl genrsa -out ./server.key 1024`.
2. Creación de la petición de certificado: `openssl req -new -key server.key -out server.csr`. Responder a las preguntas que se formulan. La contraseña hay que dejarla en blanco.
3. Generación de la clave privada: `openssl x509 -req -days 365 -set_serial N -in server.csr -signkey server.key -out server.crt`. En donde N es un número que se incrementa por el usuario por cada certificado creado.
4. Se pueden combinar ambos certificados en uno para mayor comodidad en su manejo: `cat server.key server.crt >combined.pem`.
5. También podemos comprobar que todo se ha generado correctamente de la siguiente forma: `openssl x509 -in combined.pem -noout -text`.
6. Si todo esta correcto, conviene dejar los certificados en su directorio histórico. Ese directorio es `/etc/ssl/certs`.

7.2. Instalación básica de Apache.

A continuación se presentan los pasos básicos para instalar el servidor Apache en un computador. No se aconseja usar los paquetes que vienen con las distribuciones ya que no se pueden configurar con las opciones necesarias antes de instalarlo.

1. Lo primero que hay que hacer es bajarse el paquete con el fuente del Apache de la web oficial. Para ello, hay que dirigirse a: `http://httpd.apache.org/download.cgi` y seleccionar la versión que se quiera bajar. El sistema TocToc a sido probado en la versión 2.0.55 de Apache.
2. Una vez se ha descargado el paquete, el siguiente paso es descomprimirlo. Para ello escribir en la línea de comandos: `tar xvf httpd-2_0_NN.tar.gz`
3. Entrar en la carpeta creada `httpd-2_0_NN` y configurar el Apache con las siguientes opciones activas: `./configure --prefix=/Apache --enable-so --enable-deflate --enable-ssl`.
4. Instalar el Apache: `make && make install`

Si no ha habido ningún error durante la instalación del programa, el servidor debe de estar listo para ser usado. Ahora solo falta editar el archivo `Directorio_apache/conf/httpd.conf` y introducir unos cambios básicos para que reconzca nuestros documentos.

1. Buscar la sección *Dynamic Shared Object (DSO) Support* y añadir la siguiente línea: `LoadModule php5_module modules/libphp5.so` que permite cargar PHP.
2. Buscar el string *Addtype* y añadir al final del listado de tipos insertados los documentos PHP de la forma: `AddType application/x-httpd-php .php .phtml` y `AddType application/x-httpd-php-source .phps`. Esto permite a Apache reconocer los documentos a partir de la terminación del archivo.
3. Buscar el string *Documentroot* y colocar aquí el directorio donde vayamos a albergar nuestras páginas web. Después hay que modificar unas líneas más abajo la sentencia `<Directory "/www/">` de forma que en *www* se indique el directorio seleccionado en *Documentroot*.
4. Añadir las páginas `index.php` a las páginas por defecto del servidor. Esto se consigue buscando la etiqueta *DirectoryIndex* y dejándola así: `DirectoryIndex index.html index.html.var index.php index.htm`. El orden es influyente, ya que indica al servidor la prioridad de las páginas a buscar.
5. Asegurarse que el usuario del servidor, es el usuario *apache*⁴. Si este usuario no existe en el sistema, añadirlo a los usuarios de Linux.

⁴Técnicamente, aquí puede ir cualquier usuario salvo el *root*. Sin embargo, los scripts que se reparten con el sistema TocToc, están configurados específicamente con este usuario.

6. Generar el archivo `/var/log/httpd/ssl_request_log`.
7. Para arrancar el apache en modo seguro utilizar: `“Directorio_apache/bin/apachectl startssl”`

7.3. Instalación de MySQL.

Para instalar MySQL, hay que dirigirse a `http://www.mysql.com/` y bajarse tanto los paquetes del servidor como del cliente. Existen paquetes para cada distribución de Linux, por lo que dependiendo de la que se tenga en la máquina, se deber seguir unas instrucciones u otras. En la página web está toda la información necesaria para ello. Por otra parte, existen dos aplicaciones que pueden resultar muy prácticas para gestionar las bases de datos. Sus nombres son: MySQL Query Browser y MySQL Administrator.

Una vez instalado el servidor, conviene añadir un password al administrador por razones de seguridad. Para ello, como root escribir: `“mysqladmin password el_password_que_quieras”`.

Ahora lo que hay que hacer es introducir los esquemas de la base de datos que usará el sistema TocToc. Para ello importa el archivo de bases de datos que se adjunta con el sistema. Para ello, puedes usar un programa como el mysql-admin o hacerlo desde la propia consola de la siguiente forma:

1. Genera la base de datos correspondiente (como root): `“mysqladmin create toctoc_database -p”`.
2. Incluirle los esquemas que vienen con el paquete: `“mysql toctoc_database <toctoc_database.schema -p”`.
3. Repetir los pasos 1 y 2 para la base de datos que gestiona los usuarios: `“mysqladmin create toctoc_users -p”` y `“mysql toctoc_users <toctoc_users.schema -p”`.

Después hay que hacer que la base de datos pertenezca al usuario `TocToc` para que el sistema pueda hacer las peticiones necesarias a la base de datos. Esto se hace en dos pasos. Primero entramos en mysql: `“mysql -u root -p”` para después cambiar los privilegios del usuario en la base de datos: `“grant all on TocToc_database.* to TocToc@localhost identified by 'el_password_que_queramos';”` `“grant all on TocToc_usuarios.* to TocToc@localhost identified by 'el_password_de_antes;”` y aplicamos lo hecho `“flush privileges”`.

Por último, hay que señalar que el esquema de la base de datos que se reparte con el resto de los ficheros, incluye ya un usuario generado con privilegios de administrador. El nombre es `root` y la contraseña de acceso es `toctocadmin`. Una vez testeado el sistema y comprobado que funciona, conviene cambiar lo antes posible la contraseña de este usuario. Este usuario es especial en tanto que no aparece en ningún sitio de administrador, por lo que conviene mantenerlo a salvo.

7.4. Instalación de PHP.

En esta sección se describe la instalación básica de PHP.

1. Configuración de la instalación de PHP para que use Apache y MySQL:
./configure --with-apxs2=/Apache/bin/apxs --with-mysql.
2. Compilación e instalación: *“make && make install”*
3. Configuración de las opciones de PHP. Lo más cómodo es trabajar con el archivo de ejemplo que trae el propio paquete. Para ello *“cp php.ini-dist /usr/local/lib/php.ini”*.
4. Realizar las siguientes modificaciones en el `php.ini` anterior. Primero: En la cadena `doc_root` poner la dirección de la carpeta donde se guardan las páginas web a mostrar. Indicar en la cadena `session.save_path` la carpeta en donde se desea que se guarden las sesiones web. Tiene que ser una carpeta con los permisos adecuados para que se puedan depositar los archivos de sesión, por ejemplo la carpeta *“/tmp”*.

7.5. Instalación del servidor DHCP.

Aquí se muestran los pasos y toda la información necesaria para configurar un servidor DHCP.

1. Bájese el paquete o fuente del servidor de tu distribuidora Linux, de su proveedor de paquetes habitual o de esta dirección: *ftp://ftp.phystech.com/pub/*
2. Si es un código fuente, compílelo siguiendo el procedimiento normal.
3. Generar el archivo */etc/dhcpd.conf* si este no existe. Generar la configuración del mismo dependiendo del tipo de red. Un ejemplo de como tiene que quedar esta red se encuentra en la siguiente dirección: *http://www.tldp.org/HOWTO/DHCP/x369.html*.
4. Añadir la siguiente entrada al encaminamiento: *“route add -host 255.255.255.255 dev eth0”* siendo `eth0` el dispositivo de red que permite el acceso a Internet.
5. Generar el archivo siguiente: *“touch /var/state/dhcp/dhcpd.leases”*.
6. Ejecutar el servidor: *“/usr/sbin/dhcpd”*.

7.6. Permisos de las carpetas.

Los permisos de las carpetas que contienen los archivos del sistema, tienen que dar acceso tanto al servidor Apache, como a aquel que tenga que ejecutar los scripts. Unos permisos inadecuados, puede resultar en que no funcione parte del sistema, o que este tenga un comportamiento inadecuado. Simplemente hay que asegurarse que todas las carpetas usadas en el proyecto tengan permiso de lectura y ejecución para el usuario *apache*. Hay que evitar que estos permisos afecten a cualquier otro usuario no autorizado.

En la carpeta donde se escriben los archivos de control de los clientes y en el de log, además tienen que tener los permisos de escritura adecuados. Requisitos.

Para el correcto funcionamiento del sistema TocToc, es necesario tener ya instalados los siguientes programas:

1. Un sistema de cifrado seguro como OpenSSL para dar seguridad a los clientes.
2. Servidor Apache 2.0 o superior. Dicho servidor deberá tener activado la conexión segura con SSL. Cómo se ha configurado la instalación del Apache para este proyecto se muestra de una forma detallada en el apéndice de este documento.
3. MySQL 4.1.10 o superior.
4. PHP 5.0 o superior. Este debe ser configurado para el uso del servidor Apache y MySQL. En el apéndice existe un apartado por si no estas familiarizado con la instalación de PHP.
5. Servidor DHCP propio.
6. El paquete *iproute*. Este paquete aparece en las distribuciones de Linux, por lo que consulte la distribución utilizada para saber como instalarlo.

8. Configuración del servidor Apache.

Aparte de la configuración normal del servidor Apache, hay que añadir unas modificaciones no convencionales para que el sistema TocToc funcione correctamente. Como configuración normal se entiende la generación de una carpeta ligada al mismo que contenga todas las páginas web del servidor, las opciones necesarias para que interprete los documentos PHP y la información relativa al administrador⁵.

Como primera modificación extra, hay que descomentar la línea que aparece en `httpd.conf`: `"ErrorDocument 404 /error/HTTP_NOT_FOUND.html.var"` y modificarla por la siguiente: `"ErrorDocument 404" /error/error.php`. Después de esto añadir los documentos `"error.php"` y `"error_original.php"`, que se incluye junto los archivos del sistema TocToc, en la carpeta `"/Directorio_apache/errors/"`. Esta página, simplemente redirecciona al cliente a la página de login del sistema. De esta forma el sistema se asegura que cuando capture un cliente que no este dado de alta, siempre se le muestre la página correspondiente y no salga un error. Conviene revisar esta corta página para observar que el enlace de redirección apunta a donde debe.

Una vez acabado esto, hay que realizar exactamente lo mismo para el error 403, de forma que impidamos el listado del contenido del directorio web y lo

⁵Todo esto se puede encontrar en la web oficial del Apache: <http://httpd.apache.org/docs/2.0/>. Existe un pequeño resumen de instalación al final de este texto, en el capítulo 12.

enviemos a la hoja de registro. Para desactivar la opción por defecto de listar contenidos y que aparezca el error 403 se debe comentar la línea *Options Indexes FollowSymLinks* que existe dentro de la declaración del directorio web. Hay que tener cuidado con estas opciones si se desea albergar múltiples páginas web en este servidor ya que los mensajes de error serán modificadas para todas.

Otro punto a tener en cuenta es que, debido a que el servidor Apache tiene que utilizar algunos comandos restringidos al root, se debe dar permisos al mismo mediante el fichero */etc/sudoers* añadiendo la siguiente línea: *apache ALL=NOPASSWD:/sbin/iptables, /sbin/tc, /bin/sh* que le permite ejecutar los comandos TC, IPTABLES y SH⁶ necesarios para el sistema.

Por último, recordar que el servidor Apache tiene un usuario en el sistema UNIX asociado, normalmente *apache* o *nobody*. Los scripts del sistema TocToc están preparados para que el usuario del servidor sea el denominado *apache*. Por ello, hay que modificar la línea *user* del archivo *httpd.conf* y colocar el usuario *apache*. Posiblemente haya que crear ese usuario en el sistema y dar, además, permisos de ejecución a los scripts del sistema para este usuario. Generalmente, el no hacer esto, no muestra ningún error al ejecutar el sistema, simplemente este se limita a ignorar los scripts para los cuales no tiene permisos, obteniendo unos resultados inesperados.

9. Copia y configuración de los ficheros necesarios.

9.1. Copia de los archivos.

Una vez esta todo funcionando en orden, es decir, se puede visualizar cualquier página web con nuestro servidor desde otro cliente, hay que colocar las propias páginas del sistema en sus sitio.

1. Mover la carpeta *web_TocToc* que contiene todos los ficheros PHP al directorio en el que el servidor Apache sirve las páginas web (denominado a partir de ahora “directorio *www*” o “*www*” simplemente).
2. Darle permisos de ejecución a los archivos para que PHP no tenga ningún problema con ellos. Eso significa que todos los archivos relacionados tienen que tener permisos de lectura para todos.
3. Guarda la carpeta *TocToc* que contiene los scripts de ejecución del sistema a tu equipo y dótale también de los permisos necesarios de ejecución. A lo largo del manual, nos referiremos a esta carpeta simplemente como *dir_scripts*.

⁶El administrador del sistema tiene que tener mucho cuidado si aloja otras páginas web para no darles a ellos permisos de lectura, escrita y ejecución en sus scripts y provocar así un agujero en su seguridad. Por ello, es aconsejable que esta máquina sea un servidor dedicado exclusivamente al sistema TocToc.

9.2. Configuración.

Los archivos que hay que configurar son los siguientes:

1. *www/include/colores.inc.*
2. *www/include/conexion.inc.*
3. *www/include/configuracion.inc.*
4. *www/include/TocToc.css*
5. *dir_scripts/TocToc.sh.*
6. */Directorio_apache/errors/error.php*

A continuación, se detalla la configuración y el uso de cada uno de ellos.

Archivo *colores.inc.* Este archivo guarda la configuración básica de los colores de la página. Permite cambiar el aspecto de todo el sistema, junto con el archivo *css* y el *xsl*. Los nombres de las variables son autoexplicativos, y el color tiene que escribirse en formato RGB. La imagen que se encuentra aquí, corresponde con la imagen que aparecerá de fondo en cada página web.

Archivo *conexion.inc* En este archivo se especifica, entre otras cosas, el *host* que contiene las bases de datos del sistema, el usuario que tiene acceso a las mismas y el *password* necesario para ello. Aquí tienes que poner los datos dependiendo de la instalación de MySQL en tu sistema.

Las dos constantes *BD_USUARIOS* y *BD* indican el nombre de las dos bases de datos del sistema. Por lo que no es necesario cambiarlos salvo que se haga lo mismo en las bases de datos MySQL.

Archivo *configuracion.inc* Este archivo contiene la configuración específica de su sistema. Aquí se deben de introducir tanto algunos datos propios de su equipo, como otros datos que ya dependen de los resultados que se deseen que devuelva el sistema. A continuación se muestra el significado de las constantes aquí definidas:

1. *TOCTOC_FOLDER*: indica la carpeta que contiene todas las páginas web del fichero. Tiene que tener la estructura *www/TOCTOC_FOLDER/*.php*
2. *PAGINA_INTERNET*: a que página se direccionará el usuario cuando accede a Internet si previamente no ha elegido una.
3. *DIR_CLIENTES*: la carpeta que contiene los archivos que se utilizan para controlar los clientes. Tienen que ser la misma que la indicada en el script *TocToc.sh.*

4. *CITYTICKET_TOTAL*: aquí se indica el número de *citytickets* que quiere que maneje el sistema. La configuración de lo que se quiere que haga cada uno de ellos, viene definido en el vector *array_velocidad_conexion[cityticket][0 (bajada) - 1 (subida)]*. Por ejemplo: para la configuración *array_velocidad_conexion[2][0]=16* significa que para el Cityticket 1, su velocidad de bajada es de 16KB/s.
5. *LOG*: esta constante contiene el nombre del fichero de logs del sistema.
6. *TIEMPO_LOG*: el tiempo en segundos que se mantienen las entradas en el *log* antes de trasladarlas a la copia de seguridad del mismo. Por defecto es un mes.
7. *LOG_BAK*: el nombre en donde se guardarán las entradas del *log* una vez pasen el tiempo máximo que deben de quedar registradas.
8. *DIRECTORIO_LOG*: el directorio donde se almacenará el *log* del sistema.
9. *REFRESCO_ACCESO_WEB*: aquí se indica el tiempo que tardará en refrescarse la ventana de control de Internet que se le abrirá al cliente. Debe ser menor que el tiempo indicado en *TocToc.sh* o se corre el riesgo de que se desconecten los clientes sin motivo alguno.
10. *RED_WIRELESS*: el nombre que le da Linux al dispositivo de acceso inalámbrico. Generalmente suele ser *eth1* o *wlan0*.
11. *RED_INTERNET*: el nombre del dispositivo que sale a Internet. Generalmente será *eth0*.
12. *IP_GATEWAY*: la IP que tiene el ordenador que actúa como Gateway (en donde está instalado el sistema TocToc). A esta IP se dirigirán los clientes para salir a Internet.
13. *PUERTOS_ACEPTADOS*: el rango de puertos que se reserva el sistema para poder añadir aquellos servidores permitidos a los que los clientes pueden salir sin pagar el acceso a Internet.
14. *TC*: la dirección del ejecutable del programa *tc* de Linux.
15. *IPT*: la dirección del ejecutable del *iptables* de Linux.
16. *TOCTOC_SCRIPT_PATH*: en donde se almacenan los scripts del servidor TocToc.
17. *TOCTOC_SCRIPT*: el nombre del script que activa el servidor TocToc.
18. *ROOT*: esta constante contiene el nombre del administrador del sistema que en principio no se mostrará en el mismo (es decir, será imposible de borrar, modificar y demás). Se puede usar esta cuenta para tener un administrador principal del sistema sin que le aparezca a los demás encargados del mantenimiento del mismo.

19. *PAGINA_PRINCIPAL*: la página a la que se dirigirá el sistema cuando un usuario acceda exitosamente en el sistema.
20. *CSS*: El archivo que contiene la hoja de estilos en cascada. Escribirlo mediante una variable permite probar con varios diseños cambiando esta variable.
21. *TITULO*: El título que aparecerá en la cabecera del navegador. Así se permite una fácil integración con otros sistemas.
22. *NOMBRE*: El nombre con el que se referenciará al sistema. El único valor de esto es mostrarlo en algunas páginas web.

Archivo *TocToc.css*. Este archivo es una hoja de estilos en cascada que utiliza el sistema para darle el formato a los títulos de las páginas, los menús y demás. Se puede modificar a gusto del administrador sin que varíe en algo el comportamiento del sistema.

Archivo *TocToc.sh*. Este es el script principal de TocToc. Al ejecutarlo prepara al equipo para recibir a los clientes según sus niveles de acceso. Es también el que impide que salgan a Internet sin permiso. Para configurarlo, hay que mirar las siguientes constantes:

1. *IPT*: Al igual que en el archivo *configuracion.inc* esta constante indica donde se encuentra el binario para ejecutar las *iptables*. En la distribución probada se encuentra en: */sbin/iptables*.
2. *TC*: Similar al anterior, pero para la ejecución del *tc*. Se encuentra en el equipo probado en: */sbin/tc*.
3. *ETLAN*: La red *ethernet* por la que se sale a Internet. Algo del tipo: *152.41.51.0/22*.
4. *WLAN*: La dirección de la red *wireless* por la que acceden los clientes. En este caso se ha elegido *192.168.1.0/24*.
5. *ETH*: El nombre que el sistema Linux da al dispositivo de red *ethernet*. Por ejemplo *eth0*.
6. *WIR*: El nombre del dispositivo wireless que le da el sistema operativo. Generalmente *eth1* o *wlan0*.
7. *PWLAN*: Esta constante contiene aquella parte de la dirección de red que es común entre todos los clientes. Sería el equivalente al aplicar la máscara a la red, pero que por razones de diseño, tenemos que indicar aquí explícitamente. En nuestro caso *192.168.1*.

8. *PUERTOS_ACEPTADOS*: Aquí se indican el rango de puertos reservados por el sistema para aquellas páginas web a las que se permita el acceso de los clientes aunque no tengan conexión a Internet permitido. Los que se incluyen en el archivo son *9400:9800*.
9. *IPS_ACEPTADAS*: Aquí se indica la IP de aquellos servidores a los que se quiera dar acceso sin que exista una conexión a Internet. Aquí se indica aquellos servidores que siempre estarán implícitos en el sistema y no se podrán modificar desde la consola de administración.
10. *DIR_PHP*: La ruta que permite al sistema encontrar todos los ficheros PHP. Algo como */www/TocToc*.
11. *DIR_CLIENTES*: Aquí se indica una carpeta en el que el servidor apache tendrá permisos para poder dejar todos los archivos de configuración que necesita para su correcto funcionamiento. Algo como */TocToc/clientes*.
12. *DIR_SCRIPTS*: En que directorio están los demás scripts que necesita ejecutar el servidor. Por ejemplo, */TocToc/scripts*.
13. *LOG*: Nombre del fichero en donde se guardan los *logs* del sistema. El nombre es indiferente mientras se mantenga en el sistema.
14. *DIR_LOG*: Allí donde se sitúa el archivo del *log*. Siguiendo las pautas anteriores, quedaría algo así como */TocToc/log*.
15. *TIEMPO*: El intervalo de tiempo que se espera el servidor para comprobar si el cliente ha caído o no. Se aconseja que sea unos segundos superior al tiempo indicado en la constante *REFRESCO_ACCESO_WEB* del archivo *configuracion.inc*.
16. *TOCTOC*: Esta variable indica el número final de la IP del ordenador en el que está alojado el servidor. Nótese que *PWLAN.TOCTOC* tienen que formar una dirección IP válida. En nuestro caso sería un *1*, que acabaría formando *192.168.1.1*.
17. *DHCPmin* y *DHCPmax*: Estas dos variables indican que rangos de direcciones del servidor DHCP se distribuyen entre los clientes que se conectan por red. *PWLAN.DHCPmin* y *PWLAN.DHCPmax* tienen que formar direcciones IP válidas.
18. *RATEUP* y *RATEDOWN*: Estas dos variables indican el ancho de banda máximo de subida y de bajada que utilizará el sistema para repartir entre los clientes. Lo aconsejable es que sea un 75 % más o menos del ancho de banda total.

Archivo /Directorio_apache/errors/error.php Este archivo deberá buscarse en la carpeta *error* del sistema TocToc si todavía no ha sido copiado a la carpeta de errores de Apache. Una vez realizado la copia, habrá que configurarlo revisando el valor de las siguientes constantes:

1. *IP_GATEWAY*: Aquí, al igual que en los archivos anteriores, se tendrá que indicar la dirección IP del dispositivo inalámbrico del ordenador en el que está instalado el sistema.
2. *RED_WIRELESS*: Escribir la dirección de la red Wireless para diferenciar los accesos al sistema como clientes TocToc o no. Es decir, aquellos equipos que entran por esta red son los que tiene que atrapar el sistema hasta que se dan de alta en el sistema.
3. *TOCTOC_FOLDER*: La carpeta en donde están todos los archivos PHP del sistema. Tiene que contener el mismo valor que *DIR_PHP* descrito en el apartado anterior.

10. Ejecución del sistema.

Una vez ha sido instalado y configurado todo lo anterior, la forma de que el sistema se ponga en funcionamiento es la siguiente.

1. Comprobar que los dispositivos de red del servidor están activos y funcionando correctamente.
2. Asegurarse que los servidores Apache y DHCP están en funcionamiento.
3. Poner en marcha el servidor TocToc mediante la ejecución del script *TocToc.sh*. Esto se hace de la siguiente forma: “*sh TocToc <start/stop><0/1>*” por ejemplo “*sh TocToc.sh start 0*”. El 0 indica que queremos que nos muestre por pantalla todos los mensajes, el 1 que los omita. Otra forma de poner en marcha el sistema es conectarse a *http://localhost/TocToc* e introducir un usuario que tenga privilegios de administrador en el sistema. Si es así, le aparecerá en el menú principal una opción de activación del sistema.
4. Ahora solo falta coger un cliente, conectarse a la red del sistema, y comprobar que todo funciona.

11. Detención del sistema.

También hay que saber como detener el sistema correctamente. Para detener el sistema, lo que hay que hacer es ejecutar el *script* descrito en la sección anterior de la siguiente forma: “*sh TocToc stop*”. Si no se detiene el sistema de esta forma, se queda un proceso activo en el sistema. Este proceso es el que vigila que no se queden clientes activos si estos han caído de forma inesperada. Tener este

proceso activo no perjudica a la computadora, pero cuando se vuelva a ejecutar el sistema TocToc, el tener dos procesos que eliminan clientes inactivos puede ser problemático para el correcto funcionamiento del mismo. Si esto ocurre, es posible que clientes que no están inactivos sean eliminados por el sistema debido al funcionamiento doble de este proceso.

12. Instalación básica de los programas.

12.1. Instalación del OpenSSL y generación de un certificado propio.

La instalación del sistema OpenSSL depende del sistema Linux elegido. Existen paquetes binarios relativos a cada distribución que pueden ser usados, facilitando la tarea de instalación. De otra forma, se tendrá que descargar el código de la web oficial <http://www.openssl.org/> y compilarlo para su funcionamiento:

1. Descomprimir el paquete descargado: `tar xvf openssl-0.9.X.tar.gz`.
2. Un simple `./configure && make && make install` debería bastar.

Una vez ha sido instalado, se debería pasar crear un certificado de seguridad. Para ello hay que seguir estos pasos:

1. Creación de la clave del servidor de 1024 bits: `openssl genrsa -out ./server.key 1024`.
2. Creación de la petición de certificado: `openssl req -new -key server.key -out server.csr`. Responder a las preguntas que se formulan. La contraseña hay que dejarla en blanco.
3. Generación de la clave privada: `openssl x509 -req -days 365 -set_serial N -in server.csr -signkey server.key -out server.crt`. En donde N es un número que se incrementa por el usuario por cada certificado creado.
4. Se pueden combinar ambos certificados en uno para mayor comodidad en su manejo: `cat server.key server.crt >combined.pem`.
5. También podemos comprobar que todo se ha generado correctamente de la siguiente forma: `openssl x509 -in combined.pem -noout -text`.
6. Si todo esta correcto, conviene dejar los certificados en su directorio histórico. Ese directorio es `/etc/ssl/certs`.

12.2. Instalación básica de Apache.

A continuación se presentan los pasos básicos para instalar el servidor Apache en un computador. No se aconseja usar los paquetes que vienen con las distribuciones ya que no se pueden configurar con las opciones necesarias antes de instalarlo.

1. Lo primero que hay que hacer es bajarse el paquete con el fuente del Apache de la web oficial. Para ello, hay que dirigirse a: "`http://httpd.apache.org/download.cgi`" y seleccionar la versión que se quiera bajar. El sistema TocToc a sido probado en la versión 2.0.55 de Apache.
2. Una vez se ha descargado el paquete, el siguiente paso es descomprimirlo. Para ello escribir en la línea de comandos: "`tar zxvf httpd-2_0_NN.tar.gz`"
3. Entrar en la carpeta creada `httpd-2_0_NN` y configurar el Apache con las siguientes opciones activas: "`./configure --prefix=/Apache --enable-so --enable-deflate --enable-ssl`".
4. Instalar el Apache: `make && make install`

Si no ha habido ningún error durante la instalación del programa, el servidor debe de estar listo para ser usado. Ahora solo falta editar el archivo "`Directorio_apache/conf/httpd.conf`" y introducir unos cambios básicos para que reconozca nuestros documentos.

1. Buscar la sección *Dynamic Shared Object (DSO) Support* y añadir la siguiente línea: "`LoadModule php5_module modules/libphp5.so`" que permite cargar PHP.
2. Buscar la cadena *Addtype* y añadir al final del listado de tipos insertados los documentos PHP de la forma: "`AddType application/x-httpd-php .php .phtml`" y "`AddType application/x-httpd-php-source .phps`". Esto permite a Apache reconocer los documentos a partir de la terminación del archivo.
3. Buscar el string *Documentroot* y colocar aquí el directorio donde vayamos a albergar nuestras páginas web. Después hay que modificar unas líneas más abajo la sentencia `<Directory "/www/">` de forma que en *www* se indique el directorio seleccionado en *Documentroot*.
4. Añadir las páginas `index.php` a las páginas por defecto del servidor. Esto se consigue buscando la etiqueta *DirectoryIndex* y dejándola así: "`DirectoryIndex index.html index.html.var index.php index.htm`". El orden es influyente, ya que indica al servidor la prioridad de las páginas a buscar.
5. Asegurarse que el usuario del servidor, es el usuario *apache*⁷. Si este usuario no existe en el sistema, añadirlo a los usuarios de Linux.
6. Generar el archivo `/var/log/httpd/ssl_request_log`.
7. Para arrancar el apache en modo seguro utilizar: "`Directorio_apache/bin/apachectl startssl`"

⁷Técnicamente, aquí puede ir cualquier usuario salvo el *root*. Sin embargo, los scripts que se reparten con el sistema TocToc, están configurados específicamente con este usuario.

12.3. Instalación de MySQL.

Para instalar MySQL, hay que dirigirse a <http://www.mysql.com/> y bajarse tanto los paquetes del servidor como del cliente. Existen paquetes para cada distribución de Linux, por lo que dependiendo de la que se tenga en la máquina, se deber seguir unas instrucciones u otras. En la página web está toda la información necesaria para ello. Por otra parte, existen dos aplicaciones que pueden resultar muy prácticas para gestionar las bases de datos. Sus nombres son: MySQL Query Browser y MySQL Administrator.

Una vez instalado el servidor, conviene añadir un password al administrador por razones de seguridad. Para ello, como root escribir: *“mysqladmin password el_password_que_quieras”*.

Ahora lo que hay que hacer es introducir los esquemas de la base de datos que usará el sistema TocToc. Para ello importa el archivo de bases de datos que se adjunta con el sistema. Para ello, puedes usar un programa como el mysql-admin o hacerlo desde la propia consola de la siguiente forma:

1. Genera la base de datos correspondiente (como root): *“mysqladmin create toctoc_database -p”*.
2. Incluirle los esquemas que vienen con el paquete: *“mysql toctoc_database <toctoc_database.schema -p”*.
3. Repetir los pasos 1 y 2 para la base de datos que gestiona los usuarios: *“mysqladmin create toctoc_users -p”* y *“mysql toctoc_users <toctoc_users.schema -p”*.

Después hay que hacer que la base de datos pertenezca al usuario *TocToc* para que el sistema pueda hacer las peticiones necesarias a la base de datos. Esto se hace en dos pasos. Primero entramos en mysql: *“mysql -u root -p”* para después cambiar los privilegios del usuario en la base de datos: *“grant all on TocToc_database.* to TocToc@localhost identified by 'el_password_que_queramos’;”* *“grant all on TocToc_usuarios.* to TocToc@localhost identified by 'el_password_de_antes’;”* y aplicamos lo hecho *“flush privileges”*.

Por último, hay que señalar que el esquema de la base de datos que se reparte con el resto de los ficheros, incluye ya un usuario generado con privilegios de administrador. El nombre es *root* y la contraseña de acceso es *toctocadmin*. Una vez testeado el sistema y comprobado que funciona, conviene cambiar lo antes posible la contraseña de este usuario. Este usuario es especial en tanto que no aparece en ningún sitio de administrador, por lo que conviene mantenerlo a salvo.

12.4. Instalación de PHP.

En esta sección se describe la instalación básica de PHP.

1. Configuración de la instalación de PHP para que use Apache y MySQL: *“./configure --with-apxs2=/Apache/bin/apxs --with-mysql”*.

2. Compilación e instalación: *"make && make install"*
3. Configuración de las opciones de PHP. Lo más cómodo es trabajar con el archivo de ejemplo que trae el propio paquete. Para ello *"cp php.ini-dist /usr/local/lib/php.ini"*.
4. Realizar las siguientes modificaciones en el `php.ini` anterior. Primero: En la cadena `doc_root` poner la dirección de la carpetas donde se guardan las páginas web a mostrar. Indicar en la cadena `session.save_path` la carpeta en donde se desea que se guarden las sesiones web. Tiene que ser una carpeta con los permisos adecuados para que se puedan depositar los archivos de sesión, por ejemplo la carpeta *"/tmp"*.

12.5. Instalación del servidor DHCP.

Aquí se muestran los pasos y toda la información necesaria para configurar un servidor DHCP.

1. Bájese el paquete o fuente del servidor de tu distribuidora Linux, de su proveedor de paquetes habitual o de esta dirección: *ftp://ftp.phystech.com/pub/*
2. Si es un código fuente, compílelo siguiendo el procedimiento normal.
3. Generar el archivo */etc/dhcpd.conf* si este no existe. Generar la configuración del mismo dependiendo del tipo de red. Un ejemplo de como tiene que quedar esta red se encuentra en la siguiente dirección: *http://www.tldp.org/-HOWTO/DHCP/x369.html*.
4. Añadir la siguiente entrada al encaminamiento: *"route add -host 255.255.-255.255 dev eth0"* siendo `eth0` el dispositivo de red que permite el acceso a Internet.
5. Generar el archivo siguiente: *"touch /var/state/dhcp/dhcpd.leases"*.
6. Ejecutar el servidor: *"/usr/sbin/dhcpd"*.

12.6. Permisos de las carpetas.

Los permisos de las carpetas que contienen los archivos del sistema, tienen que dar acceso tanto al servidor Apache, como a aquel que tenga que ejecutar los scripts. Unos permisos inadecuados, puede resultar en que no funcione parte del sistema, o que este tenga un comportamiento inadecuado. Simplemente hay que asegurarse que todas las carpetas usadas en el proyecto tengan permiso de lectura y ejecución para el usuario *apache*. Hay que evitar que estos permisos afecten a cualquier otro usuario no autorizado.

En la carpeta donde se escriben los archivos de control de los clientes y en el de log, además tienen que tener los permisos de escritura adecuados.